Check Point
SOFTWARE TECHNOLOGIES LTD.

15 January 2020

NEXT GENERATION SECURITY GATEWAY

R80.40

Administration Guide

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 R80.40

For more about this release, see the R80.40 home page.

Latest Version of this Document

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
15 January 2020	First release of this document

Table of Contents

Glossary

A

Administrator
A user with permissions to manage Check Point security products and the network environment.

API
In computer programming, an application programming interface (API) is a set of subroutine definitions, protocols, and tools for building application software. In general terms, it is a set of clearly defined methods of communication between various software components.

Appliance
A physical computer manufactured and distributed by Check Point.

B

Bond
A virtual interface that contains (enslaves) two or more physical interfaces for redundancy and load sharing. The physical interfaces share one IP address and one MAC address. See "Link Aggregation".

Bonding
See "Link Aggregation".

Bridge Mode
A Security Gateway or Virtual System that works as a Layer 2 bridge device for easy deployment in an existing topology.

C

CA
Certificate Authority. Issues certificates to gateways, users, or computers, to identify itself to connecting entities with Distinguished Name, public key, and sometimes IP address. After certificate validation, entities can send encrypted data using the public keys in the certificates.

Certificate
An electronic document that uses a digital signature to bind a cryptographic public key to a specific identity. The identity can be an individual, organization, or software entity. The certificate is used to authenticate one identity to another.

Cluster
Two or more Security Gateways that work together in a redundant configuration - High Availability, or Load Sharing.

Cluster Member
A Security Gateway that is part of a cluster.

CoreXL
A 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 Firewall Instance
Also CoreXL FW 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.

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. For details, see sk92449.

D

DAIP Gateway
A Dynamically Assigned IP (DAIP) Security Gateway is a Security Gateway where the IP address of the external interface is assigned dynamically by the ISP.

Data Type
A classification of data. The Firewall classifies incoming and outgoing traffic according to Data Types, and enforces the Policy accordingly.

Database
The Check Point database includes all objects, including network objects, users, services, servers, and protection profiles.

Distributed Deployment
The Check Point Security Gateway and Security Management Server products are deployed on different computers.

Domain
A network or a collection of networks related to an entity, such as a company, business unit or geographical location.

Domain Log Server
A Log Server for a specified Domain. It stores and processes logs from Security Gateways that are managed by the corresponding Domain Management Server. Acronym: DLS.

Domain Management Server
A virtual Security Management Server that manages Security Gateways for one Domain, as part of a Multi-Domain Security Management environment. Acronym: DMS.

E

Expert Mode
The name of the full command line shell that gives full system root permissions in the Check Point Gaia operating system.

External Network
Computers and networks that are outside of the protected network.

External Users
Users defined on external servers. External users are not defined in the Security Management Server database or on an LDAP server. External user profiles tell the system how to identify and authenticate externally defined users.

F

Firewall
The software and hardware that protects a computer network by analyzing the incoming and outgoing network traffic (packets).

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 restrictive shell (role-based administration controls the number of commands available in the shell).

Gaia Portal
Web interface for Check Point Gaia operating system.

H

Hotfix
A piece of software installed on top of the current software in order to fix some wrong or undesired behavior.

I

ICA
Internal Certificate Authority. A component on Check Point Management Server that issues certificates for authentication.

Internal Network
Computers and resources protected by the Firewall and accessed by authenticated users.

IPv4
Internet Protocol Version 4 (see RFC 791). A 32-bit number - 4 sets of numbers, each set can be from 0 - 255. For example, 192.168.2.1.

IPv6
Internet Protocol Version 6 (see RFC 2460 and RFC 3513). 128-bit number - 8 sets of hexadecimal numbers, each set can be from 0 - ffff. For example, FEDC:BA98:7654:3210:FEDC:BA98:7654:3210.

J

Jumbo Hotfix Accumulator
Collection of hotfixes combined into a single package. Acronyms: JHA, JHF.

L

Link Aggregation
Various methods of combining (aggregating) multiple network connections in parallel to increase throughput beyond what a single connection could sustain, and to provide redundancy in case one of the links should fail.

Log
A record of an action that is done by a Software Blade.

Log Server
A dedicated Check Point computer that runs Check Point software to store and process logs in Security Management Server or Multi-Domain Security Management environment.

M

Management High Availability
Deployment and configuration mode of two Check Point Management Servers, in which they automatically synchronize the management databases with each other. In this mode, one Management Server is Active, and the other is Standby. Acronyms: Management HA, MGMT HA.

Management Interface
Interface on Gaia computer, through which users connect to Portal or CLI. Interface on a Gaia Security Gateway or Cluster member, through which Management Server connects to the Security Gateway or Cluster member.

Management Server
A Check Point Security Management Server or a Multi-Domain Server.

Multi-Domain Log Server
A computer that runs Check Point software to store and process logs in 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 Security Management
A centralized management solution for large-scale, distributed environments with many different Domain networks.

Multi-Domain Server
A computer that runs Check Point software to host virtual Security Management Servers called Domain Management Servers. Acronym: MDS.

N

Network Object
Logical representation of every part of corporate topology (physical machine, software component, IP Address range, service, and so on).

O

Open Server
A physical computer manufactured and distributed by a company, other than Check Point.

P

Primary Multi-Domain Server
The Multi-Domain Server in Management High Availability that you install as Primary.

R

Rule
A set of traffic parameters and other conditions in a Rule Base that cause specified actions to be taken for a communication session.

Rule Base
Also Rulebase. All rules configured in a given Security Policy.

S

Secondary Multi-Domain Server
The Multi-Domain Server in Management High Availability that you install as Secondary.

SecureXL
Check Point product that accelerates IPv4 and IPv6 traffic. Installed on Security Gateways for significant performance improvements.

Security Gateway
A computer that runs Check Point software to inspect traffic and enforces Security Policies for connected network resources.

Security Management Server
A computer that runs Check Point software to manage the objects and policies in Check Point environment.

Security Policy
A 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.

Single Sign-On
A property of access control of multiple related, yet independent, software systems. With this property, a user logs in with a single ID and password to gain access to a connected system or systems without using different usernames or passwords, or in some configurations seamlessly sign on at each system. This is typically accomplished using the Lightweight Directory Access Protocol (LDAP) and stored LDAP databases on (directory) servers. Acronym: SSO.

SmartConsole
A Check Point GUI application used to manage Security Policies, monitor products and events, install updates, provision new devices and appliances, and manage a multi-domain environment and each domain.

SmartDashboard
A legacy Check Point GUI client used to create and manage the security settings in R77.30 and lower versions.

Software Blade
A software blade is a security solution based on specific business needs. Each blade is independent, modular and centrally managed. To extend security, additional blades can be quickly added.

SSO
See "Single Sign-On".

Standalone
A Check Point computer, on which both the Security Gateway and Security Management Server products are installed and configured.

T

Traffic
Flow of data between network devices.

U

Users
Personnel authorized to use network resources and applications.

V

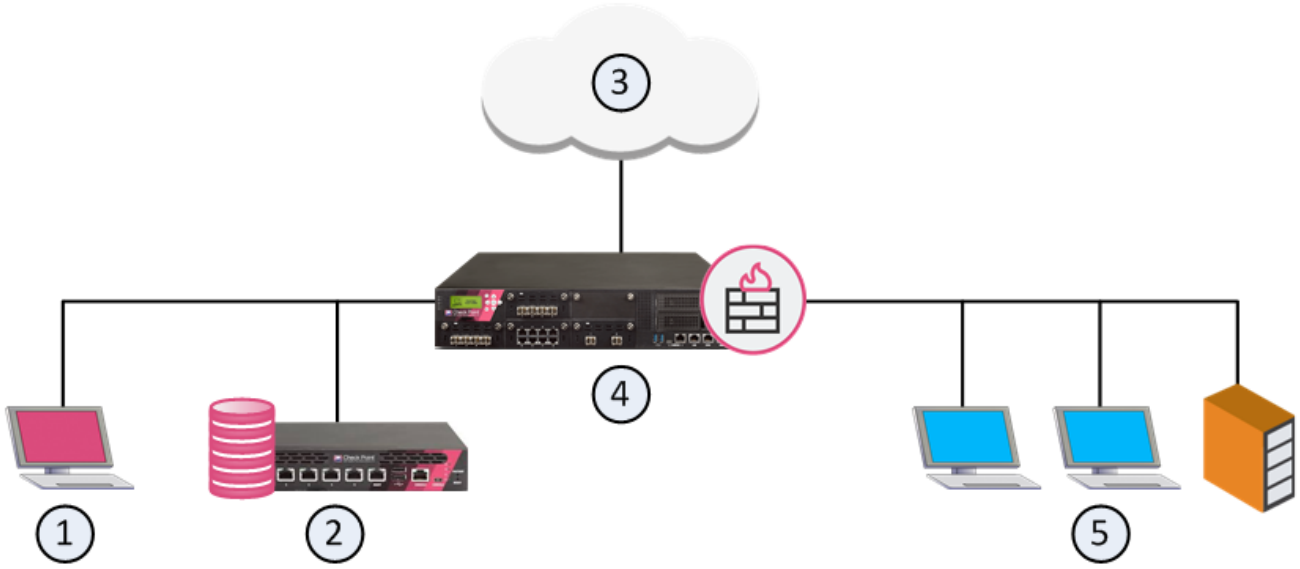
VLAN
Virtual Local Area Network. Open servers or appliances connected to a virtual network, which are not physically connected to the same network.

VLAN Trunk
A connection between two switches that contains multiple VLANs.

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.

Check Point Next Generation Security Gateway Solution



Item	Description
1	SmartConsole
2	Security Management Server
3	Internet and external networks
4	Security Gateway (or Cluster)
5	Internal network

These are the primary components of a Check Point Firewall solution:

- **Security Gateway** (or **Cluster**) - The engine that enforces the organization's security policy, is an entry point to the LAN, and is managed by the Security Management Server.
- **Security Management Server**- The application that manages, stores, and distributes the security policy to Security Gateways.
- **SmartConsole** - A Check Point GUI application used to manage security policies, monitor products and events, install updates, provision new devices and appliances, and manage a multi-domain environment.

Notes:

- For information about Cluster, see the *R80.40 ClusterXL Administration Guide*.
- For information about Security Management Server and SmartConsole, see the *R80.40 Security Management Administration Guide*.

Security Policy

In This Section:

Security Policy is a collection of rules and settings that control network traffic and enforce organization guidelines for data protection and access to resources with packet inspection.

Check Point solution provides several types of Security Policies.

Access Control Policy

Description

Access Control Policy consists of these parts:

- **Access Control Rule Base**

  For more information, see the *R80.40 Security Management Administration Guide*.

  In addition, see sk120964 - ATRG: Unified Policy.

  Contains unified simple and granular rules to control access from specified sources to specified destinations over specified protocols.

  If you enable Identity Awareness Software Blade on your Security Gateways, you can also use Access Role objects as the source and destination in a rule. This lets you easily make rules for individuals or different groups of users.

Rule structure:

No	Name	Source	Destination	VPN	Services & Applications	Action	Time	Track	Install On

#	Your Rule Name	Specific Source objects	Specific Destination objects	Specific or All VPN Communities	Specific or All Service objects
					Specific or All Application objects
						`Accept`
						or
						`Drop`
						or
						`Reject`
						or
						`User Auth`
						or
						`Client Auth`
							`Any`
							or
							Specific Time object
								`Log`
								(with `Accounting`)
								or
								`Alert`
								or
								`None`
									`Policy Targets`

- **NAT Rule Base**

  For more information, see the *R80.40 Security Management Administration Guide*.

  Contains automatic and manual rules for Network Address Translation (NAT).

  Rule structure:

No	Original Source	Original Destination	Original Services	Translated Source	Translated Destination	Translated Services	Install On	Comments
Automatic Generated Rules								
NAT Rules for X (Y-Z)								
#	Specific Source objects	Specific Destination objects	Specific or All Service objects	`=Original` or Specific object	`=Original` or Specific object	`=Original` or Specific object	`Policy Targets` or Specific Security Gateway and Cluster objects	Your Comment

- **Desktop Rule Base**

  For more information, see the SmartDashboard Help (press F1).

Prerequisites:

1. In the Security Gateway (Cluster) object, enable the **IPsec VPN** and the **Policy Server** Software Blades.
2. In the Policy Package, enable the **Desktop Security**.

This policy is installed on the Security Management Server. Remote Access Clients download this policy when a VPN Site update is performed. Once downloaded, this policy determines access control on the Remote Access Client machines.

The Desktop Policy consists of two Rule Bases:

- **Inbound Rules** - Control connections directed at the client machine
- **Outbound Rules** - Control connections initiated by the client machine

Rule structure:

No	Source	Desktop	Service	Action	Track	Comment

#	`Any`
	or
	Specific Source objects
		`All Users@Any`
		or
		Specific User Group objects
			`Any`
			or
			Specific Service objects
				`Accept`
				or
				`Block`
				or
				`Encrypt`
					`None`
					or
					`Log`
					or
					`Alert`
						Your Comment

Threat Prevention Policy

Description

For more information, see the *R80.40 Threat Prevention Administration Guide*.

Determines how the system inspects connections for bots and viruses. The primary component of the policy is the Rule Base. The rules use the Malware database and network objects.

If you enable Identity Awareness Software Blade on your Security Gateways, you can also use Access Role objects as the scope in a rule. This lets you easily make rules for individuals or different groups of users.

Rule structure:

No	Name	Protected Scope	Source	Destination	Protection/ Site/ File/ Blade	Services	Action	Track	Install On	Comments
#	Your Rule Name	Specific objects	Specific Source objects	Specific Destination objects	N/A (or your specific objects in an exception rule)	Any or Specific Service objects	Basic or Optimized or Strict or *Your Profile*	None or Log or Alert In addition: Packet Capture Forensics	Policy Targets or Specific Security Gateway and Cluster objects	Your Comment

HTTPS Inspection Policy

Description

For more information, see the *R80.40 Security Management Administration Guide*.

Lets you inspect the HTTP / HTTPS traffic on these Software Blades:

- Anti-Bot
- Anti-Virus
- Application Control
- Content Awareness (Data Awareness)
- Data Loss Prevention
- IPS
- Threat Emulation
- URL Filtering

Security Gateways cannot inspect HTTPS traffic because it is encrypted. You can enable the HTTPS Inspection feature to let the Security Gateways create new SSL connections with the external site or server. The Security Gateways are then able to decrypt and inspect HTTPS traffic that uses the new SSL connections.

Rule structure:

No	Name	Source	Destination	Services	Category/ Custom Application	Action	Track	Blade	Install On	Certificate	Comment
#	Your Rule Name	Any	APPI_global_obj_Internet or Specific Destination objects	TLS default services or Specific Service objects	Any or Specific objects	Inspect or Bypass	None or Log or Alert	All or Specific Blade	Policy TLS Targets or Specific Security Gateway and Cluster objects	Outbound Certificate or Your Certificate for Inbound Inspection	Your Comment

Data Loss Prevention Policy

Description

For more information, see the *R80.40 Data Loss Prevention Administration Guide*.

Prevents unintentional data leaks by catching protected data before it leaves your organization.

Rule structure:

Flag	Name	Data	Source	Destination	Protocol	Exceptions	Action	Track	Severity	Install On	Time	Category	Comment
Category Name(Y-Z)													
No Flag or Follow Up or Improve Accuracy	Your Rule Name	Specific Data Type	My Organization or Specific Source objects	Outside My Org or Specific Destination objects	Any or E-mail or FTP or HTTP	Shows: none or The number of exceptions added for this rule (double-click this cell)	Detect or Inform User or Ask User or Prevent or Watermark	Email or Log or Alert and how to store an incident	Low or Medium or High or Critical	DLP Blades	Any	None or Specific Category	Your Comment

Geo Policy

Description

For more information, see the SmartConsole Online Help (press F1).

Rule structure:

Country	Direction	Action	Track	Comments
Specific Country object	From and To Country or From Country or To Country	Accept or Drop	None or Log or Alert	Your Comment

Mobile Access Policy

Description

For more information, see the *R80.40 Mobile Access Administration Guide*.

Controls which user groups have access to which applications, when connecting through a Mobile Access Security Gateway.

Rule structure:

No	Users	Applications	Install On	Comment
#	`All Users` or Specific User objects	`Any` or Specific Custom Application objects	`Any` or Specific Security Gateway objects	Your Comment

Firewall Software Blade

This is the main Software Blade that enforces the Access Control and NAT policies on Security Gateways and Cluster Members

IPsec VPN Software Blade

This Software Blade lets the Security Gateways and Cluster Members encrypt and decrypt traffic to and from other gateways and clients.

For more information, see:

- *R80.40 Site to Site VPN Administration Guide*
- sk104760 - ATRG: VPN Core (requires **Advanced** access to Check Point Support Center)
- sk108600 - VPN Site-to-Site with 3rd party (requires **Advanced** access to Check Point Support Center)

Policy Server Software Blade

This Software Blade lets you configure a **Desktop Security** Policy for Remote Access Clients.

This policy controls how the Firewall Software Blade on Remote Access Clients inspects the traffic.

For more information, see:

- *"Security Policy" on page 23* > Section *Access Control Policy* > Section *Desktop Rule Base*
- *R80.40 Remote Access VPN Administration Guide*

Remote Access VPN

If employees remotely access sensitive information from different locations and devices, system administrators must make sure that this access does not become a security vulnerability.

- Check Point's Remote Access VPN solutions let you create a VPN tunnel between a remote user and the internal network.

  For more information, see the *R80.40 Remote Access VPN Administration Guide*.

- The Mobile Access Software Blade extends the functionality of Remote Access solutions to include many clients and deployments.

  For more information, see the *R80.40 Mobile Access Administration Guide*.

Threat Prevention

To challenge today's malware landscape, Check Point's comprehensive Threat Prevention solution offers a multi-layered, pre- and post-infection defense approach and a consolidated platform that enables enterprise security to detect and block modern malware.

For more information, see the *R80.40 Threat Prevention Administration Guide*.

These Software Blades provide Threat Prevention:

- *"Anti-Bot Software Blade" on page 34*
- *"Anti-Virus Software Blade" on page 35*
- *"Threat Extraction Software Blade" on page 36*
- *"Threat Emulation Software Blade" on page 37*
- *"IPS Software Blade" on page 39*

Anti-Bot Software Blade

This Software Blade discovers infections by correlating multiple detection methods:

- Performs post-infection detection of bots on hosts.
- Prevents bot damages by blocking bot C&C (Command and Control) communications.
- Is continuously updated from ThreatCloud, a collaborative network to fight cybercrime.

For more information, see:

- *R80.40 Threat Prevention Administration Guide*
- sk92264 - ATRG: Anti-Bot and Anti-Virus (requires **Advanced** access to Check Point Support Center)

In addition, see *"UserCheck" on page 47*.

Anti-Virus Software Blade

This Software Blade:

- Performs pre-infection detection and blocking of malware at the Security Gateway (by correlating multiple detection engines before users are affected).
- Is continuously updated from ThreatCloud.

For more information, see:

- *R80.40 Threat Prevention Administration Guide*
- sk92264 - ATRG: Anti-Bot and Anti-Virus (requires **Advanced** access to Check Point Support Center)

In addition, see *"UserCheck" on page 47*.

Threat Extraction Software Blade

Part of the SandBlast suite.

This Software Blade:

- Provides protection against incoming malicious content.
- Removes exploitable content, including active content and embedded objects, reconstructs files to eliminate potential threats, and promptly delivers sanitized content to users to maintain business flow.

  To remove possible threats, creates a safe copy of the file, while the inspects the original file for potential threats.

For more information, see:

- *R80.40 Threat Prevention Administration Guide*
- sk114807 - ATRG: Threat Extraction

In addition, see *"UserCheck" on page 47*.

Threat Emulation Software Blade

Part of the SandBlast suite.

This Software Blade quickly inspects files and runs them in a virtual sandbox to discover malicious behavior.

Discovered malware is prevented from entering the network.

The emulation service reports and automatically shares the newly identified threat information with other customers.

For more information, see:

- *R80.40 Threat Prevention Administration Guide*
- sk114806 - ATRG: Threat Emulation (requires **Advanced** access to Check Point Support Center)

In addition, see *"UserCheck" on page 47*.

Mail Transfer Agent (MTA)

This feature is required by the Threat Emulation Software Blade to inspect SMTP traffic.

For more information, see:

- *R80.40 Threat Prevention Administration Guide*
- sk109699 - ATRG: Mail Transfer Agent (MTA) (requires **Advanced** access to Check Point Support Center)

IPS Software Blade

This Software Blade:

- Delivers complete and proactive intrusion prevention.
- Delivers thousands of signatures, behavioral and preemptive protections.
- Gives another layer of security on top of Check Point Firewall technology.
- Protects both clients and servers, and lets you control the network usage of certain applications.

The hybrid detection engine provides multiple defense layers, which allows it excellent detection and prevention capabilities of known threats and in many cases future attacks as well. It also allows unparalleled deployment and configuration flexibility and excellent performance.

For more information, see:

- *R80.40 Threat Prevention Administration Guide*
- sk95193 - ATRG: IPS (requires **Advanced** access to Check Point Support Center)

Identity Awareness Software Blade

Traditionally, firewalls use IP addresses to monitor traffic and are unaware of the user and computer identities behind those IP addresses. Identity Awareness removes this notion of anonymity since it maps users and computer identities. This lets you enforce access and audit data based on identity.

Identity Awareness is an easy to deploy and scalable solution. It is applicable for both Active Directory and non-Active Directory based networks, as well as for employees and guest users.

Identity Awareness uses the Source and Destination IP addresses of network traffic to identity users and computers. You can use these elements as matching criteria in the Source and Destination fields of your policy rules:

- The identity of users or user groups
- The identity of computers or computer groups

Identity Awareness lets you define policy rules for specified users, who send traffic from specified computers or from any computer. Likewise, you can create policy rules for any user on specified computers.

Identity Awareness gets identities from the configured identity sources.

For more information, see:

- *R80.40 Identity Awareness Administration Guide*
- sk86441 - ATRG: Identity Awareness

Content Awareness Software Blade

This Software Blade provides data visibility and enforcement in unified Access Control Policy.

You can set the direction of the data in the Access Control Policy to one of these:

- **Download Traffic** - Into the organization
- **Upload Traffic** - Out of the organization
- **Any Direction**

You can set Data Types in the Access Control Policy to one of these:

- **Content Types** - Classified by analyzing the file content (for example: PCI - credit card numbers, International Bank Account Numbers - IBAN)
- **File Types** - Classified by analyzing the file ID (for example: Viewer File - PDF, Executable file, Presentation file)

You can select one of these services:

- `CheckPointExchangeAgent`
- `ftp`
- `http`
- `https`
- `HTTP_proxy`
- `HTTPS_proxy`
- `smtp`
- `Squid_NTLM`

For more information, see the:

- *R80.40 Security Management Administration Guide*
- SmartConsole Online Help
- sk119715 - ATRG: Content Awareness (CTNT) (requires **Advanced** access to Check Point Support Center)

Note - Content Awareness and Data Loss Prevention (see *"Data Loss Prevention Software Blade" on page 45*) both use Data Types in the Access Control Policy. However, they have different features and capabilities. They work independently, and the Security Gateway enforces them separately.

Mobile Access Software Blade

Check Point Mobile Remote Access VPN Software Blade is the safe and easy solution to connect to corporate applications over the internet with your mobile device or PC. The solution provides enterprise-grade remote access with both Layer 3 VPN and SSL VPN. It gives you simple, safe and secure connectivity to your email, calendar, contacts and corporate applications. At the same time, it protects networks and endpoint computers from threats.

The Mobile Access Portal lets mobile and remote workers connect easily and securely to critical resources over the internet.

Check Point Mobile Apps enables secure encrypted communication from unmanaged smartphones and tablets to your corporate resources.

For more information, see:

- *R80.40 Mobile Access Administration Guide*
- sk104577 - ATRG: Mobile Access Blade

Application Control Software Blade

This Software Blade detects or blocks traffic for applications:

- **Granular Application Control**: Identifies, allows, or blocks thousands of applications. This provides protection against the increasing threat vectors and malware introduced by internet applications.
- **Largest application library with AppWiki**: Comprehensive application control that uses the industry's largest application library. It scans for and detects more than 4,500 applications and more than 100,000 Web 2.0 widgets. Check Point database is updated frequently with worldwide Apps and Widgets.

For more information, see:

- *R80.40 Security Management Administration Guide*
- sk112249 - Best Practices - Application Control
- sk73220 - ATRG: Application Control (requires **Advanced** access to Check Point Support Center)

In addition, see *"UserCheck" on page 47*.

URL Filtering Software Blade

This Software Blade lets you control access to web sites and applications based on their categorization.

For more information, see:

- *R80.40 Security Management Administration Guide*
- sk92743 - ATRG: URL Filtering (requires **Advanced** access to Check Point Support Center)

In addition, see *"UserCheck" on page 47*.

Data Loss Prevention Software Blade

This Software Blade prevents unintentional data leaks by catching protected data before it leaves your organization.

This Software Blade identifies, monitors, and protects data transfer through deep content inspection and analysis of transaction parameters (such as source, destination, data object, and protocol), with a centralized management framework. In short, DLP detects and prevents the unauthorized transmission of confidential information.

Note - Data Loss Prevention is also known as Data Leak Prevention, Information Leak Detection and Prevention, Information Leak Prevention, Content Monitoring and Filtering, and Extrusion Prevention.

For more information, see the:

- *R80.40 Data Loss Prevention Administration Guide*
- SmartConsole Online Help.
- sk73660 - ATRG: Data Loss Prevention (DLP) (requires **Advanced** access to Check Point Support Center)

Note - Data Loss Prevention and Content Awareness (see *"Content Awareness Software Blade" on page 41*) both use Data Types in the Access Control Policy. However, they have different features and capabilities. They work independently, and the Security Gateway enforces them separately.

In addition, see *"UserCheck" on page 47*.

Anti-Spam & Email Security Software Blade

This Software Blade enforces Anti-Spam:

- **Based on content fingerprint** - Identifies spam by analyzing known and emerging distribution patterns. By avoiding a search for keywords and phrases that might classify a legitimate email as spam and instead focusing on other message characteristics, this solution offers a high spam detection rate with a low number of false positives.
- **Based on IP Reputation** - Blocks known spammers.
- **Based on user defined IP addresses and Sender / Domains** - Blocks senders identified by either name, domain, or IP address.

You can configure:

- Directional scanning for SMTP traffic
- Directional scanning for POP3 traffic
- Network exceptions
- List of allowed email senders

For more information, see:

- *R80.40 Threat Prevention Administration Guide*
- SmartDashboard built-in help

UserCheck

This feature gives users a warning when there is a potential risk of data loss or security violation.

This helps users to prevent security incidents and to learn about the organizational security policy.

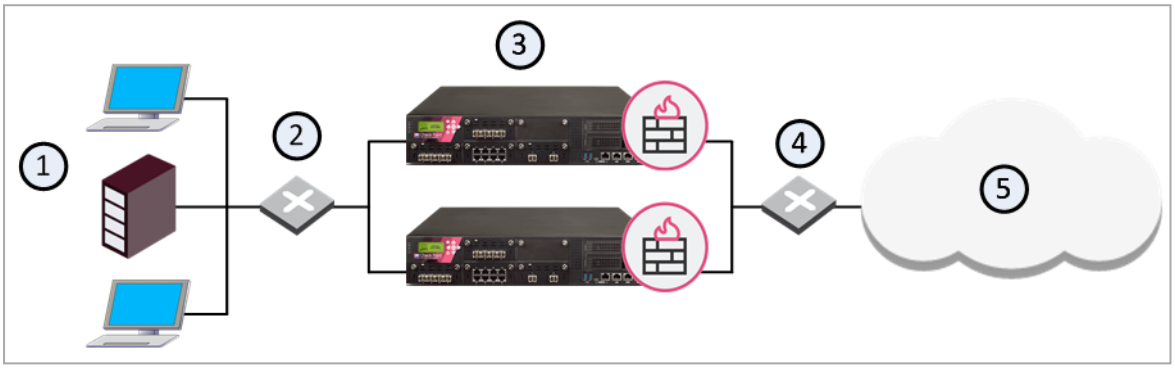
This feature is required by these Software Blades:

- *"Threat Emulation Software Blade" on page 37*)
- *"Threat Extraction Software Blade" on page 36*
- *"Anti-Bot Software Blade" on page 34*
- *"Anti-Virus Software Blade" on page 35*
- *"Data Loss Prevention Software Blade" on page 45*
- *"Application Control Software Blade" on page 43*
- *"URL Filtering Software Blade" on page 44*

ClusterXL Software Blade

ClusterXL is a Check Point software-based cluster solution for Security Gateway redundancy and Load Sharing. A ClusterXL Security Cluster contains identical Check Point Security Gateways.

- A High Availability Security Cluster ensures Security Gateway and VPN connection redundancy by providing transparent failover to a backup Security Gateway in the event of failure.
- A Load Sharing Security Cluster provides reliability and also increases performance, as all members are active.



Item	Description
1	Internal network
2	Switch for internal network
3	Security Gateways with ClusterXL Software Blade
4	Switch for external networks
5	Internet

For more information, see the *R80.40 ClusterXL Administration Guide*.

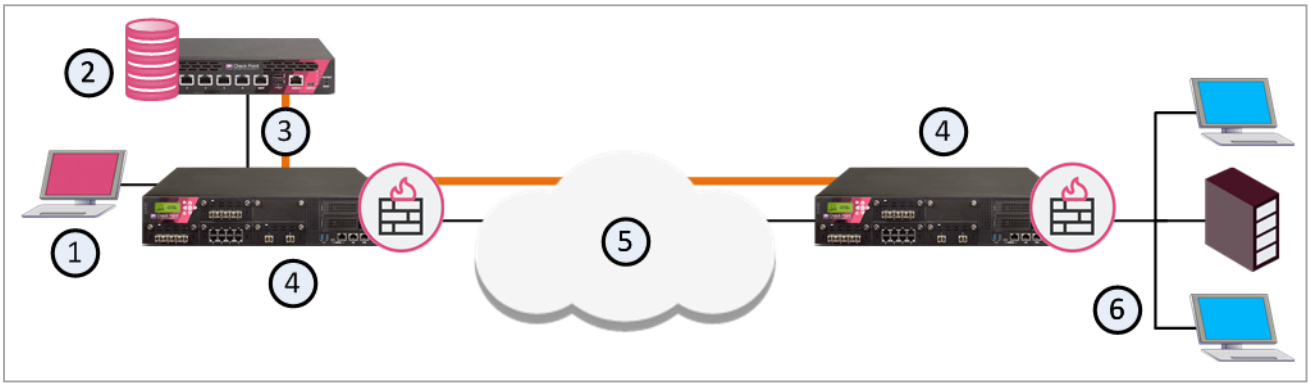
QoS Software Blade

QoS is a policy based bandwidth management solution that lets you:

- Prioritize business-critical traffic, such as ERP, database and Web services traffic, over lower priority traffic.
- Guarantee bandwidth and control latency for streaming applications, such as Voice over IP (VoIP) and video conferencing.
- Give guaranteed or priority access to specified employees, even if they are remotely accessing network resources.

You deploy QoS with the Security Gateway.

QoS is enabled for both encrypted and unencrypted traffic.



Item	Description
1	SmartConsole
2	Security Management Server
3	QoS Policy
4	Security Gateway with QoS Software Blade
5	Internet
6	Internal network

QoS leverages the industry's most advanced traffic inspection and bandwidth control technologies. Check Point patented Stateful Inspection technology captures and dynamically updates detailed state information on all network traffic. This state information is used to classify traffic by service or application. After traffic has been classified, QoS applies an innovative, hierarchical, Weighted Fair Queuing (WFQ) algorithm to accurately control bandwidth allocation.

For more information, see the *R80.40 QoS Administration Guide*.

VSX

Virtual System eXtension product runs several virtual firewalls on the same hardware.

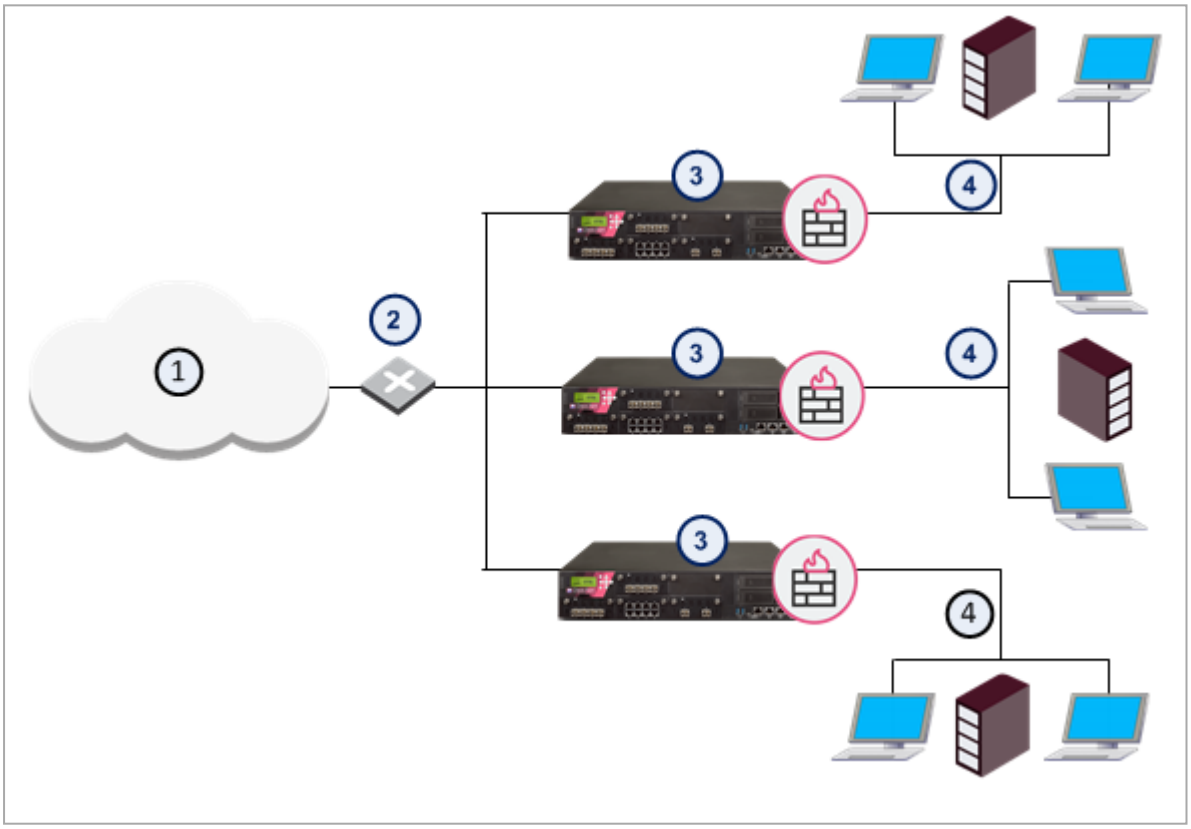
Each **Virtual System** works as a Security Gateway, typically protecting a specified network. When packets arrive at the VSX Gateway, it sends traffic to the Virtual System protecting the destination network. The Virtual System inspects all traffic and allows or rejects it according to rules defined in the security policy.

In order to better understand how virtual networks work, it is important to compare physical network environments with their virtual (VSX) counterparts. While physical networks consist of many hardware components, VSX virtual networks reside on a single configurable VSX Gateway or cluster that defines and protects multiple independent networks, together with their virtual components.

Example Physical Network Topology

In a typical deployment with multiple Security Gateways, each protects a separate network.

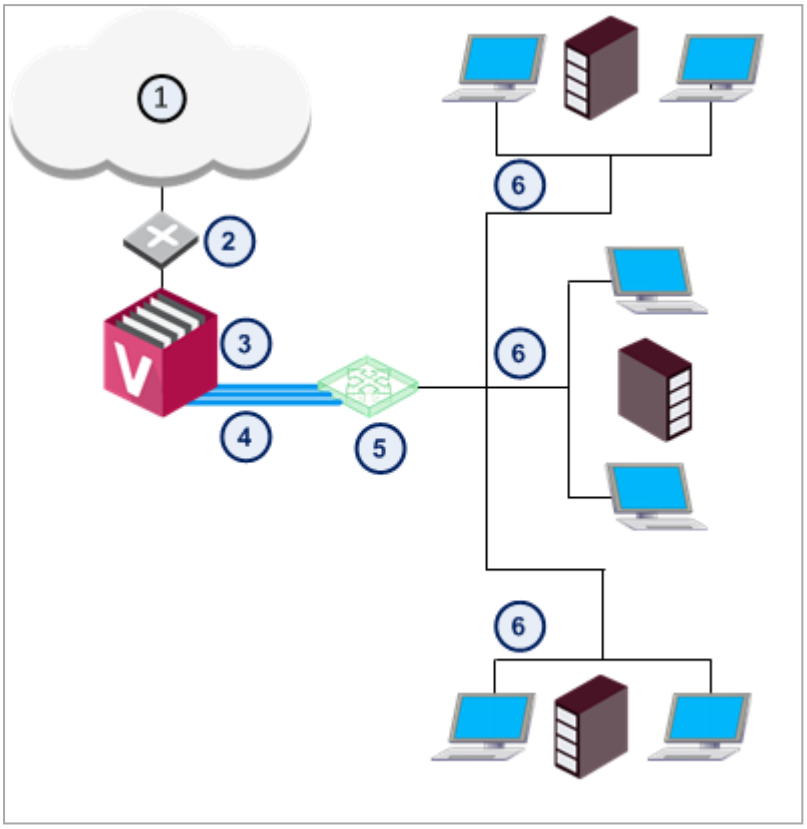
Each physical Security Gateway has interfaces to the perimeter router and to the network it protects.



Item	Description
1	Internet
2	Router
3	Security Gateways
4	Network

Example VSX Virtual Network Topology

Deploy one VSX Gateway with four Virtual Systems to protect multiple networks.



Item	Description
1	Internet
2	Router
3	VSX Gateway. Each Virtual System in a VSX environment is a Security Gateway, with the same security and networking functionality as a physical gateway. Each handles packet traffic to and from the one network it protects.
4	Warp Links. Virtual interfaces and network cables connect the Virtual Systems and the Virtual Switch.
5	Virtual Switch. Connects all the Virtual Systems to the Internet router.
6	Networks

For more information, see the *R80.40 VSX Administration Guide*.

SecureXL

This feature accelerates traffic that passes through Security Gateway.

For more information, see:

- *R80.40 Performance Tuning Administration Guide*
- sk153832 - ATRG: SecureXL for R80.20 and above (requires **Advanced** access to Check Point Support Center)
- sk98348 - Best Practices - Security Gateway Performance

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.

For more information. see:

- *R80.40 Performance Tuning Administration Guide*
- sk98737 - ATRG: CoreXL (requires **Advanced** access to Check Point Support Center)
- sk98348 - Best Practices - Security Gateway Performance

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 lets you configure 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).

For more information, see:

- *R80.40 Performance Tuning Administration Guide*
- sk98348 - Best Practices - Security Gateway Performance

ICAP

The **Internet Content Adaptation Protocol (ICAP)** is a lightweight HTTP-like protocol (request and response protocol), which is used to extend transparent proxy servers. This frees up resources and standardizes the way in which new features are implemented. ICAP is usually used to implement virus scanning and content filters in transparent HTTP proxy caches.

The ICAP allows ICAP Clients to pass HTTP / HTTPS messages to ICAP Servers for content adaptation. The ICAP Server executes its transformation service on these HTTP / HTTPS messages and sends responses to the ICAP Client, usually with modified HTTP / HTTPS messages. The adapted HTTP / HTTPS messages can be HTTP / HTTPS requests, or HTTP / HTTPS responses.

You can configure Check Point Security Gateway as:

- ICAP Client - To send the HTTP / HTTPS messages to ICAP Servers for content adaptation.
- ICAP Server - To perform content adaptation in the HTTP / HTTPS messages received from ICAP Clients.
- Both ICAP Client and ICAP Server at the same time.

Check Point Security Gateway configured for ICAP can work with third party ICAP devices without changing the network topology.

For more information, see the *R80.40 Threat Prevention Administration Guide*.

HTTPS Inspection

Lets you inspect the HTTP / HTTPS traffic on these Software Blades:

- Anti-Bot
- Anti-Virus
- Application Control
- Content Awareness (Data Awareness)
- Data Loss Prevention
- IPS
- Threat Emulation
- URL Filtering

Security Gateways cannot inspect HTTPS traffic because it is encrypted. You can enable the HTTPS Inspection feature to let the Security Gateways create new SSL connections with the external site or server. The Security Gateways are then able to decrypt and inspect HTTPS traffic that uses the new SSL connections.

For more information, see:

- *R80.40 Threat Prevention Administration Guide*
- sk108202 - Best Practices - HTTPS Inspection
- sk65123 - HTTPS Inspection FAQ

HTTP/HTTPS Proxy

You can configure the Security Gateway to act as an HTTP/HTTPS Proxy on your network.

In such configuration, the Security Gateway becomes an intermediary between hosts that communicate with each other through the Security Gateway. It does not allow a direct connection between these hosts.

Each successful connection creates two different connections:

- One connection between the client in the organization and the proxy (Security Gateway).
- One connection between the proxy (Security Gateway) and the actual destination.

These proxy modes are supported:

- **Transparent** - All HTTP traffic on specified ports and interfaces is intercepted and processed by the Proxy code in the Security Gateway. No configuration is required on the clients.
- **Non Transparent** - All HTTP/HTTPS traffic on specified ports and interfaces is intercepted and processed by the Proxy code in the Security Gateway. Configuration of the proxy address and port is required on client machines.

For more information, see:

- SmartDashboard built-in help
- sk110013 - How to configure Check Point Security Gateway as HTTP/HTTPS Proxy (requires **Advanced** access to Check Point Support Center)
- sk92482 - Performance impact from enabling HTTP/HTTPS Proxy functionality (requires **Advanced** access to Check Point Support Center)

Hardware Security Module (HSM)

Why Use an HSM?

Hardware Security Module (HSM) is a device that is used to store cryptographic keys.

HSM adds an extra layer of security to the network. HSM is designed to provide dedicated cryptographic functionality.

When Check Point Security Gateway uses an HSM Server, the HSM Server holds these objects for outbound HTTPS Inspection:

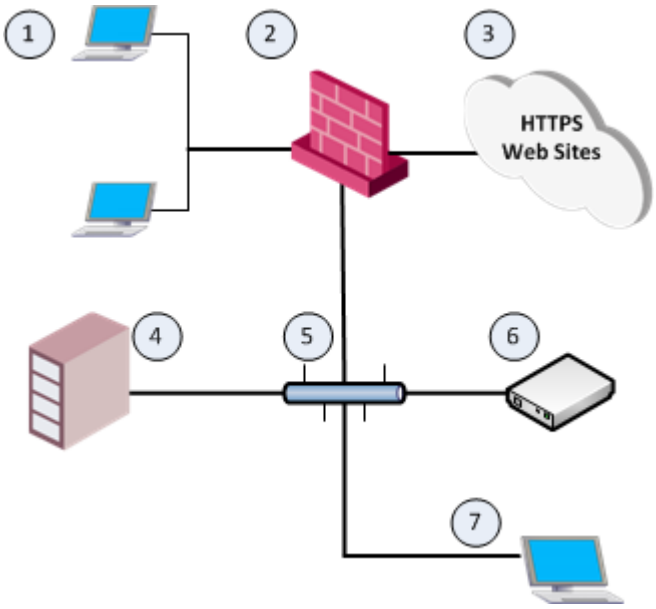
1. The Certificate Authority (CA) certificate (certificate buffer + key pair).

   The administrator creates the CA certificate and key pair before configuring the Security Gateway to work with an HSM Server.

2. Two to three key pairs for fake certificates.

   These keys are created during the initialization of the HTTPS Inspection daemon on the Security Gateway with 1024-bit, 2048-bit, or 4096-bit length.

The Check Point Environment with an HSM Server



Item	Description
1	Internal computers that connect to HTTPS web sites through the Check Point Security Gateway.
2	Check Point Security Gateway with HTTPS Inspection enabled.
3	HTTPS web sites on the Internet.
4	Check Point Security Management Server that manages the Check Point Security Gateway.
5	Interconnecting Network.
6	HSM Server that stores and serves the SSL keys and certificates to the Check Point Security Gateway.
7	HSM Client workstation used to create a Certificate Authority (CA) certificate on the HSM Server.

Note - Check Point Security Gateway uses the HSM Server only for outbound HTTPS Inspection.

Configuring Your HSM Environment

In This Section:

This section describes how to configure the Check Point environment to work with the **Gemalto Luna SP SafeNet HSM**.

The SafeNet Cryptographic Engine enables the SafeNet Network HSM functionality by providing:

- Secure cryptographic storage.
- Cryptographic acceleration.
- Administrative access control.
- Policy management.
- Detection of modifications done to the data.

Workflow

Use this workflow to configure your Check Point Gateway to work with the HSM Appliance Server:

"Step 1: Extract the Gemalto Help Package" on the next page

"Step 2: Configure the Gemalto HSM Appliance Server to Work with Check Point Gateway" on page 62

"Step 3: Configure the Gemalto HSM Client Workstation" on page 64

"Step 4: Create the CA Certificate on the Gemalto HSM Appliance Server" on page 65

"Step 5: Configure the Check Point Security Gateway to Work with the Gemalto HSM Appliance Server" on page 67

Step 1: Extract the Gemalto Help Package

You must use the Gemalto configuration documents to configure the Gemalto HSM environment.

Procedure

Step	Description
1	Use a Window-based computer.
2	Download this package: Gemalto SafeNet HSM Help package  **Note** - Software Subscription or Active Support plan is required to download this package.
3	Extract the Gemalto HSM Help package to some folder.
4	Open the extracted Gemalto HSM Help folder.
5	Double-click the **START_HERE.html** file. The *Gemalto SafeNet Network HSM 6.2.2 Product Documentation* opens.

Step 2: Configure the Gemalto HSM Appliance Server to Work with Check Point Gateway

Use the Gemalto Help documents to install and configure the HSM Appliance Server.

Procedure

Step	Description
1	Install the HSM Appliance. From the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to *Installation Guide > SafeNet Network HSM Hardware Installation*.
2	Perform the initial configuration of the HSM Appliance and the HSM Appliance Server. From the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to *Configuration Guide >* and follow from *[Step 1]* to *[Step 6]*.
3	Run the `sysconf recenCert` command in LunaSH to generate a new HSM Appliance Server certificate (`server.pem`). From the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to *Configuration Guide > [Step 7] Create a trusted link and register Client and Appliance with each other*.

Step	Description

4	Complete the configuration of your HSM Appliance Server to work with Check Point Security Gateway.
	
	Run these commands in LunaSH:
	
	■ Set the applicable partition to be active and auto-activated:
	
	`lunash:> partition showPolicies -partition <Partition Name>`
	
	`lunash:> partition changePolicy -partition <Partition Name> -policy 22 -value 1`
	
	`lunash:> partition changePolicy -partition <Partition Name> -policy 23 -value 1`
	
	`lunash:> partition showPolicies -partition <Partition Name>`
	
	  **Note** - If you do not set the partition to stay auto-activated, the partition does not stay activated when the machine is shut down for more than two hours.
	
	■ Disable the client source IP address validation by NTLS upon an NTLA client connection:
	
	`lunash:> ntls ipcheck disable`
	
	  **Note** - This will allow HSM Appliance Server to accept traffic from Check Point Cluster members that is hidden behind Cluster VIP address, and from Check Point Security Gateways hidden behind NAT.

Step 3: Configure the Gemalto HSM Client Workstation

You use the Gemalto HSM Client Workstation to create a CA Certificate on the Gemalto HSM Appliance Server.

Check Point Gateway uses this CA Certificate for HTTPS Inspection when storing and accessing SSL keys on the Gemalto HSM Appliance Server.

Procedure

Step	Description
1	Install a Windows-based or Linux-based computer to use as an HSM Client workstation.
2	Download and install this software package on the HSM Client workstation computer: **SafeNet HSM Client for Workstation**  **Note** - Software Subscription or Active Support plan is required to download this package. From *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to the *Installation Guide > SafeNet HSM Client Software Installation*.
3	Establish a Trust Link between the HSM Client workstation and the HSM Appliance Server. On the HSM Client workstation, run in LunaCM: `lunacm:> clientconfig deploy ...` From the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to *Configuration Guide > [Step 7] Create a trusted link and register Client and Appliance with each other*.  **Note** - The configuration will *not* work on Linux OS with *glibc* version lower than 2.7 (for example: Red Hat 5 or lower, Gaia R77.20 or lower). In such case, follow the instructions in **Step 5** > *"Establish a Trust Link between the Check Point Security Gateway and the Gemalto HSM Appliance Server." on page 68*.

Step 4: Create the CA Certificate on the Gemalto HSM Appliance Server

Procedure

Step	Description
1	On the HSM Client workstation computer, open a command prompt or a terminal window.

2	Use the "`cmu generatekeypair`" command to create a key pair.
	Example:
	`# cd /usr/safenet/lunaclient/bin`
	`# ./cmu generatekeypair -modulusBits=2048 -publicExponent=65537 -labelPublic="CAPublicKeyPairLabel" -labelPrivate="CAPrivateKeyPairLabel" -sign=T -verify=T`

3	When prompted, enter a password: *Example:* `Enter a password for the token in slot 0: <Password for the partition on HSM Appliance Server that you configured in Step 2>`
4	Select the RSA mechanism by entering the corresponding number: `[1] PKCS [2] FIPS 186-3 Only Primes [3] FIPS 186-3 Auxiliary Primes`

5	Run the `cmu list` command to view the handles of the key pair you created.
	Example:
	`Enter password for token in slot 0 : <Password for the partition on HSM Appliance Server that you configured in Step 2>`
	`handle=17 label=CAPrivateKeyPairLabel`
	`handle=18 label=CAPublicKeyPairLabel`

6	Use the handle numbers from the previous Step 5 to create the CA certificate. *Example:* `# ./cmu selfsigncertificate -privatehandle=17 -CN="www.myhsm.cp" -sha1WithRSA -startDate 20170720 -endDate 20190101 -serialNum=123456789abcdef`

Step	Description

7	Run the `cmu list` command to view the handles of the CA certificate you created.
	Example:
	`Please enter password for token in slot 0 : <Password for the partition on HSM Appliance Server that you configured in Step 2>`
	`handle=13 label=www.myhsm.cp`
	`handle=17 label=CAPrivateKeyPairLabel`
	`handle=18 label=CAPublicKeyPairLabel`
	  **Note** - You will use the numbers of these three handles on Check Point Security Gateway in the `$FWDIR/conf/hsm_configuration.C` file.

Step 5: Configure the Check Point Security Gateway to Work with the Gemalto HSM Appliance Server

Important - If Check Point Cluster environment, perform this procedure on each Cluster Member.

Workflow:

1. Install the Gemalto HSM Simplified Client software packages on the Check Point Security Gateway.

Procedure

Step	Description
1	Download this software package: Gemalto SafeNet HSM Simplified Client for Check Point Gateway  **Note** - Software Subscription or Active Support plan is required to download this package.
2	Copy the software package to the Check Point Security Gateway to some directory.
3	Connect to the command line on the Check Point Security Gateway.
4	Log in to the Expert mode.
5	Go to the directory with the packages: `# cd /<Path>/<To>/<Directory>`
6	Extract the packages: `# tar -xvf <Name of Package>.tar`

7	Install these packages:
	`# rpm -Uvh configurator-6.2.2-4.i386.rpm`
	`# rpm -Uvh libcryptoki-6.2.2-4.i386.rpm`
	`# rpm -Uvh vtl-6.2.2-4.i386.rpm`

2. Establish a Trust Link between the Check Point Security Gateway and the Gemalto HSM Appliance Server.

   a. On the Check Point Security Gateway, perform these steps

   Instructions

   i. Connect to the command line.

   ii. Log in to the Expert mode.

   iii. Go to the SafeNet HSM Simplified Client installation directory:

   ```
   # cd /usr/safenet/lunaclient/bin/
   ```

   iv. Import the HSM Appliance Server certificate, **server.pem**, from the HSM Appliance to the Security Gateway:

   Important - The period at the end is part of the syntax.

   ```
   # scp admin@<IP Address of HSM Appliance>:server.pem .
   ```

   v. Register the HSM Appliance Server certificate, **server.pem**, with the Check Point Security Gateway:

   ```
   # ./vtl addServer -n <IP Address of HSM Appliance> -c
   server.pem
   ```

   vi. Create a certificate and private key for the Check Point Security Gateway:

   ```
   # ./vtl createCert -n <IP Address of CP Gateway>
   ```

   Notes:

   - Use the IP address of the interface that connects to the HSM Appliance.

     In a Check Point cluster, use the IP address of the cluster member, and not the Cluster Virtual IP address.

   - The private key file is created and written to:

     `/usr/safenet/lunaclient/cert/client/<IP Address of CP Gateway>Key.pem`

   - The certificate file is created and written to:

     `/usr/safenet/lunaclient/cert/client/<IP Address of CP Gateway>.pem`

   vii. Copy the Check Point Security Gateway certificate file that you created to the HSM Appliance

   Important - The colon at the end is part of the syntax.

   ```
   # scp <IP Address of CP Gateway>.pem admin@<IP Address
   of HSM Appliance>:
   ```

b. On the HSM Appliance, in LunaSH, perform these steps

Instructions

i. Register the Check Point Security Gateway certificate with the HSM Appliance Server:

```
lunash:> client register -client <Desired Name of HSM Client> -ip <IP Address of CP Gateway>
```

ii. Restart the Network Trust Link service:

```
lunash:> service restart ntls
```

iii. Confirm the Check Point Security Gateway registration:

```
lunash:> client list
```

iv. Assign the Check Point Security Gateway to the applicable partition:

```
lunash:> client assignPartition -client <Configured Name of HSM Client> -partition <Partition Name>
```

v. Examine the partition access:

```
lunash:> client show -client <Configured Name of HSM Client>
```

c. On the Check Point Security Gateway, perform this step

Instructions

Examine the partition access:

```
# ./vtl verify
```

Notes:

- For more information, see *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*.

  For information about establishing a Trust Link, go to *Appliance Administration Guide > Configuration without One-step NTLS > [Step 7] Create a Network Trust Link Between the Client and the Appliance*.

- If you need to establish a new Trust Link, you have to delete the current Trust Link (see *"Deleting a Trust Link with the HSM Appliance Server" on page 73*).

3. Configure HTTPS Inspection on the Check Point Security Gateway to work with the Gemalto HSM Appliance Server.

Important Notes

- *Before* you configure the HTTPS Inspection on the Security Gateway to work with the Gemalto HSM Appliance Server, you must enable and configure HTTPS Inspection on the Check Point Security Gateway, install the applicable Access Control Policy, and confirm that HTTPS Inspection works correctly *without* the Gemalto HSM Appliance Server.

  See the *R80.40 Security Management Administration Guide*.

- After any change in the `$FWDIR/conf/hsm_configuration.C` file on the Check Point Security Gateway, you must fetch or install the Access Control Policy on the Security Gateway.

- If the Gemalto HSM Appliance Server is not available when you fetch or install policy on the Check Point Security Gateway, the HTTPS Inspection is not able to inspect the outbound HTTPS traffic.

  As a result, internal computers are not able to access HTTPS web sites.

  To resolve this, make sure that the Gemalto HSM Appliance Server is up and running, there is physical connectivity between the Check Point Security Gateway and the Gemalto HSM Appliance, the Trust Link is established with the Gemalto HSM Appliance Server, and then fetch or install the policy on the Security Gateway.

  In addition, see *"Disabling Communication from the Check Point Gateway to the Gemalto HSM Appliance Server" on page 72*.

Procedure

a. Connect to the command line on the Security Gateway.

b. Log in to the Expert mode.

c. Edit the configuration file `$FWDIR/conf/hsm_configuration.C`:

```
# vi $FWDIR/conf/hsm_configuration.C
```

d. Based on the output of the "`cmu list`" command from **Step 4**, add details of the CA certificate from the HSM Appliance Server to this configuration file.

Example:

```
(
 :enabled ("yes") # "yes" / "no"
 :CA_cert_public_key_handle (18)
 :CA_cert_private_key_handle (17)
 :CA_cert_buffer_handle (13)
 :token_id ("<Password for the partition on HSM Appliance
Server that you configured in Step 2>")
)
```

e. On the Security Gateway, fetch the local policy:

```
# fw fetch local
```

f. Confirm that HTTPS Inspection is activated successfully on outbound traffic.

g. From an internal computer, connect to any HTTPS web site.

h. On the internal computer, in the web browser, you should receive the signed CA certificate from the HSM Appliance Server.

Additional Actions for a Gemalto HSM Appliance Server

Disabling Communication from the Check Point Gateway to the Gemalto HSM Appliance Server

You can disable communication from the Check Point Gateway to an HSM Appliance.

For example, when the HSM Appliance is under maintenance.

Step	Description
1	Connect to the command line on the Check Point Security Gateway.
2	Log in to the Expert mode.
3	Back up the current configuration file `$FWDIR/conf/hsm_configuration.C`: `cp -v $FWDIR/conf/hsm_configuration.C{,_BKP}`
4	Edit the current configuration file `$FWDIR/conf/hsm_configuration.C`: `# vi $FWDIR/conf/hsm_configuration.C`
5	Set the value of the "`:enabled`" attribute to "`no`": `:enabled ("no")`
6	Save the changes in the file and exit the editor.
7	Fetch the local policy: `# fw fetch local`

Deleting a Trust Link with the HSM Appliance Server

If you need to establish new Trust Link between a Check Point Gateway and an HSM Appliance Server, you must delete the current Trust Link.

For example, when you replace or reconfigure a Check Point Gateway, or an HSM Appliance Server.

Step	Description

1	Delete the current Trust Link on the Check Point Gateway:
	1. Connect to the command line.
	2. Log in to the Expert mode.
	3. Go to the SafeNet HSM Simplified Client installation directory:
	`# cd /usr/safenet/lunaclient/bin/`
	4. Delete the old Trust Link:
	`# ./vtl deleteServer -n <IP Address of HSM Appliance>`

2	Delete the current Trust Link on the HSM Appliance:
	1. Connect to the HSM Appliance over SSH.
	2. Examine the list of configured HSM Clients:
	`lunash:> client list`
	3. Delete the Check Point HSM Client:
	`lunash:> client delete -client <Name of HSM Client>`

Note - For more information, see *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*.

Configuring a Second Interface on a Gemalto HSM Appliance for NTLS

Step	Description
1	Connect to the HSM Appliance over SSH.
2	Examine all the configured interfaces: `lunash:> network show`
3	Add a new interface: `lunash:> network interface -device <Name of Interface> -ip <IP Address> -netmask <NetMask> [-gateway <IP Address>]`
4	Enable Network Trust Link Service (NTLS) on all the interfaces.

Note - For more information, see *Gemalto SafeNet Network HSM 6.2.2 Product Documentation > LunaSH Command Reference Guide > LunaSH Commands*.

Monitoring HTTPS Inspection when Working with an HSM Server

When HTTPS Inspection daemon **wstlsd** initializes on Check Point Security Gateway, it checks whether this Security Gateway is configured to with the Gemalto HSM Appliance Server.

- You can see the applicable logs in **SmartConsole > Logs & Monitor > Logs** tab.

  See *"Monitoring HTTPS Inspection with HSM in SmartConsole Logs" on page 76*.

- You can query the HTTPS Inspection on the Security Gateway or Cluster Members over SNMP.

  See *"Monitoring HTTPS Inspection with HSM over SNMP" on page 80*.

- You can run the "`cpstat https_inspection`" command on the Security Gateway or Cluster Members.

  See *"Monitoring HTTPS Inspection with HSM in CLI" on page 90*.

Note - To see detailed information about **wstlsd** initialization, follow sk105559: How to debug WSTLSD daemon.

Monitoring HTTPS Inspection with HSM in SmartConsole Logs

To see the HTTPS Inspection logs about the Gemalto HSM Appliance Server in **SmartConsole**:

Step	Description
1	From the left navigation panel, click **Logs & Monitor > Logs**. Click **Logs & Monitor > Logs** tab.
2	At the top, click the **Logs** tab.
3	In the search field, enter: `type:Control`
4	Double-click on the applicable log.
5	In the log, refer to the **More** section.

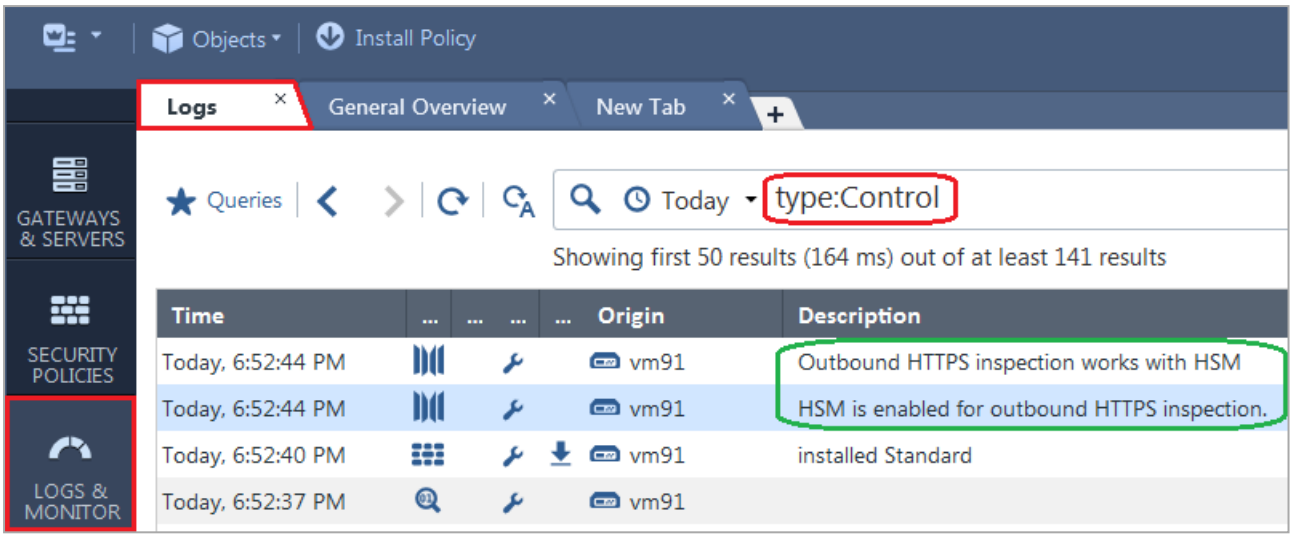
Possible logs are:

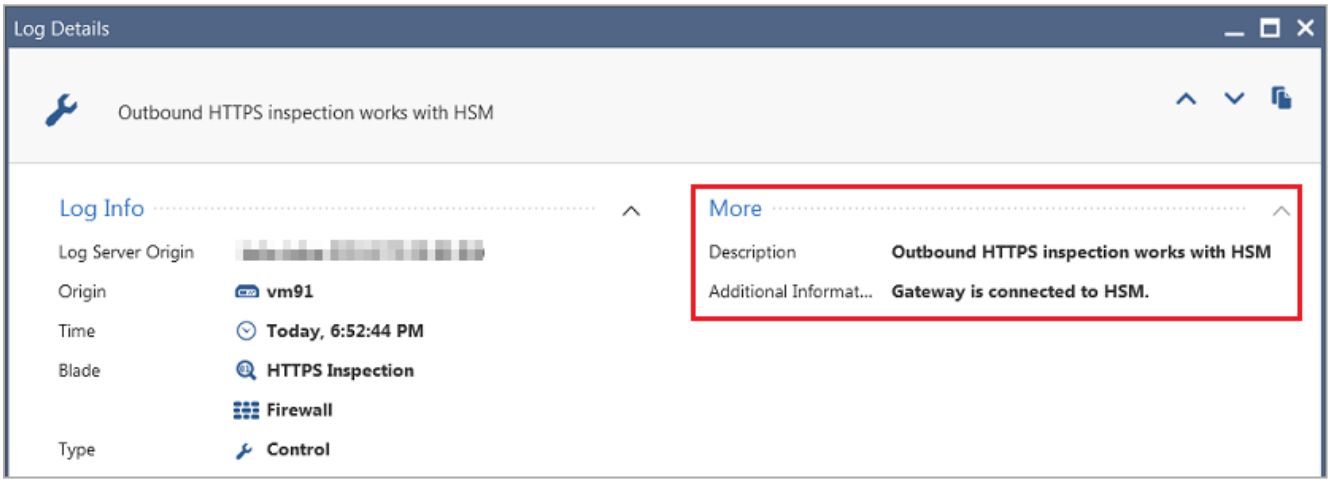
Log Description	Log Additional Information	Explanation
`HSM is enabled for outbound HTTPS inspection`		The value of the `:enabled()` attribute is set to `"yes"` in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway.

Log Description	Log Additional Information	Explanation
`HSM is disabled for outbound HTTPS inspection`		One of these: ■ The value of the "`:enabled()`" attribute is set to "`no`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway. ■ Gemalto HSM Simplified Client software packages are not installed on the Security Gateway. ■ The `$FWDIR/conf/hsm_configuration.C` file does not exist on the Security Gateway. ■ The `:enabled()` attribute is corrupted in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway.  **Important** - In the above cases, outbound HTTPS Inspection works without Gemalto HSM Appliance Server.
`Outbound HTTPS inspection works with HSM`	`Gateway is connected to HSM`	All these conditions were met: 1. The value of the "`:enabled()`" attribute is set to "`yes`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway. 2. Security Gateway was able to connect to the HSM Appliance Server.

Log Description	Log Additional Information	Explanation
`Outbound HTTPS inspection is off due to HSM error`	One of these strings: ■ `HSM configuration file is corrupted` ■ `Loading HSM library failed` ■ `There is no trust or no connectivity with HSM server` ■ `Login to HSM partition failed` ■ `Error importing CA certificate from HSM server` ■ `Error generating key pair on HSM server`	See the **Log Additional Information** column.

Example:


Objects
Install Policy
Logs
General Overview
New Tab
GATEWAYS & SERVERS
SECURITY POLICIES
LOGS & MONITOR
Queries
Today
type:Control
Showing first 50 results (164 ms) out of at least 141 results
Time
Origin
Description
Today, 6:52:44 PM
vm91
Outbound HTTPS inspection works with HSM
Today, 6:52:44 PM
vm91
HSM is enabled for outbound HTTPS inspection.
Today, 6:52:40 PM
vm91
installed Standard
Today, 6:52:37 PM
vm91


Log Details
Outbound HTTPS inspection works with HSM
Log Info
Log Server Origin
Origin
vm91
Time
Today, 6:52:44 PM
Blade
HTTPS Inspection
Firewall
Type
Control
More
Description
Outbound HTTPS inspection works with HSM
Additional Informat...
Gateway is connected to HSM.

Monitoring HTTPS Inspection with HSM over SNMP

You can query the HTTPS Inspection status and the status of connection to the Gemalto HSM Appliance Server on the Security Gateway over SNMP:

- Full OID is:

```
.iso.org.dod.internet.private.enterprises.checkpoint.products.httpsInspection
```

- Numerical OID is:

```
.1.3.6.1.4.1.2620.1.54
```

"HTTPS Inspection status"

To get the **HTTPS Inspection status**, query this SNMP object:

SNMP OID	Returned strings	Explanation
`httpsInspectionStatus` .1.3.6.1.4.1.2620.1.54.1	`On`	HTTPS Inspection feature is configured on the Security Gateway.
	`Off`	HTTPS Inspection feature is not configured on the Security Gateway.

"HTTPS Inspection status description"

To get the **HTTPS Inspection status description**, query this SNMP object:

SNMP OID	Returned strings	Explanation
`httpsInspectionStatusDescription` .1.3.6.1.4.1.2620.1.54.2	`HTTPS Inspection is on`	HTTPS Inspection feature is configured on the Security Gateway.
	`HTTPS Inspection is off`	HTTPS Inspection feature is not configured on the Security Gateway.

"HSM configuration status"

To get the **HSM configuration status**, query this SNMP object:

SNMP OID	Returned strings	Explanation
`hsmStatus.hsmEnabled` .1.3.6.1.4.1.2620.1.54.3.1	`Enabled`	The value of the "`:enabled()`" attribute is set to "`yes`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway.
	`Disabled`	One of these: ■ The value of the "`:enabled()`" attribute is set to "`no`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway. ■ Gemalto HSM Simplified Client software packages are not installed on the Security Gateway. ■ The `$FWDIR/conf/hsm_configuration.C` file does not exist on the Security Gateway. ■ The "`:enabled()`" attribute is corrupted in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway.  **Important** - In the above cases, outbound HTTPS Inspection works without Gemalto HSM Appliance Server, and SSL keys are stored on the Security Gateway.

"HSM configuration status description"

To get the **HSM configuration status description**, query this SNMP object:

SNMP OID	Returned strings	Explanation
`hsmStatus.hsmEnabledDescription` .1.3.6.1.4.1.2620.1.54.3.2	`HSM is enabled for HTTPS inspection`	The value of the "`:enabled()`" attribute is set to "`yes`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway.
	`HSM is disabled for HTTPS inspection`	One of these: ■ Gemalto HSM Simplified Client software packages are not installed on the Security Gateway. ■ The `$FWDIR/conf/hsm_configuration.C` file does not exist on the Security Gateway. ■ HTTPS Inspection daemon **wstlsd** was not able to read the value of the "`:enabled()`" attribute in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway. ■ The "`:enabled()`" attribute is corrupted in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway.  **Important** - In the above cases, outbound HTTPS Inspection works without Gemalto HSM Appliance Server, and SSL keys are stored on the Security Gateway.

"HSM partition access status"

To get the **HSM partition access status**, query this SNMP object:

SNMP OID	Returned strings	Explanation
`hsmStatus.hsmPartitionAccess` .1.3.6.1.4.1.2620.1.54.3.3	`N/A`	Security Gateway could not check access its partition on HSM Appliance Server. Most probably, because HSM configuration is disabled on the Security Gateway.
	`Accessible`	Security Gateway was able to access its partition on HSM Appliance Server.
	`Not Accessible`	Security Gateway was not able to access its partition on HSM Appliance Server due to an error.

"HSM partition access status description"

To get the **HSM partition access status description**, query this SNMP object:

SNMP OID	Returned strings	Explanation
`hsmStatus.hsmPartitionAccessDescription` .1.3.6.1.4.1.2620.1.54.3.4	`HSM partition access cannot be checked`	Security Gateway could not check access its partition on HSM Appliance Server. Most probably, because HSM configuration is disabled on the Security Gateway.
	`Gateway can access HSM partition for HTTPS inspection`	Security Gateway was able to access its partition on HSM Appliance Server.
	`Gateway cannot access HSM partition for HTTPS inspection: <error>`	Security Gateway was not able to access its partition on HSM Appliance Server due to an error. Possible error messages are: ■ `HSM configuration file is corrupted` ■ `Loading HSM library failed` ■ `There is no trust or no connectivity with HSM server` ■ `Login to HSM partition failed`

"Outbound HTTPS Inspection status"

To get the **Outbound HTTPS Inspection status**, query this SNMP object:

SNMP OID	Returned strings	Explanation
`hsmStatus.outboundStatus` .1.3.6.1.4.1.2620.1.54.3.5	`HSM on`	All these conditions were met: 1. The value of the "`:enabled()`" attribute is set to "`yes`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway. 2. Security Gateway was able to connect to the HSM Appliance Server.
	`HSM off`	One of these: ■ The value of the "`:enabled()`" attribute is set to "`no`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway. ■ Gemalto HSM Simplified Client software packages are not installed on the Security Gateway. ■ The `$FWDIR/conf/hsm_configuration.C` file does not exist on the Security Gateway. ■ The "`:enabled()`" attribute is corrupted in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway.  **Important** - In the above cases, outbound HTTPS Inspection works without Gemalto HSM Appliance Server, and SSL keys are stored on the Security Gateway.

SNMP OID	Returned strings	Explanation
	`HSM error`	All these conditions were met: 1. The value of the "`:enabled()`" attribute is set to "`yes`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway. 2. An error occurred. **Important** - In this case, outbound HTTPS Inspection will not work, and HTTPS traffic will not pass.

Note - The conditions for the returned strings are calculated on the Security Gateway during the start of the HTTPS Inspection daemon `wstlsd`, or during policy installation. For example, you can get "`hsmStatus.hsmEnabled = HSM enabled`" and "`hsmStatus.outboundStatus = HSM off`", because when the `wstlsd` daemon started, or during last policy installation, the HSM configuration was disabled.

"Outbound HTTPS Inspection status description"

To get the **Outbound HTTPS Inspection status description**, query this SNMP object:

SNMP OID	Returned strings	Explanation
`hsmStatus.outboundStatusDescription` .1.3.6.1.4.1.2620.1.54.3.6	`Outbound HTTPS inspection works with HSM`	All these conditions were met: 1. The value of the "`:enabled()`" attribute is set to "`yes`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway. 2. Security Gateway was able to connect to the HSM Appliance Server.
	`Outbound HTTPS inspection works without HSM`	The value of the "`:enabled()`" attribute is set to "`no`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway, or this file does not exist.

SNMP OID	Returned strings	Explanation
	`Outbound HTTPS inspection is off due to HSM error: <error>`	All these conditions were met: 1. The value of the `":enabled()"` attribute is set to `"yes"` in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway. 2. An error occurred.  **Important** - In this case, outbound HTTPS Inspection will not work, and HTTPS traffic will not pass. Possible error messages are: ■ `HSM configuration file is corrupted` ■ `Loading HSM library failed` ■ `There is no trust or no connectivity with HSM server` ■ `Login to HSM partition failed` ■ `Error importing CA certificate from HSM server` ■ `Error generating key pair on HSM server`

Note - The conditions for the returned strings are calculated on the Security Gateway during the start of the HTTPS Inspection daemon `wstlsd`, or during policy installation. For example, you can get "`hsmStatus.hsmEnabledDescription = HSM is enabled for HTTPS inspection`" and "`hsmStatus.outboundStatusDescription = Outbound HTTPS inspection works without HSM`", because when the `wstlsd` daemon started, or during last policy installation, the HSM configuration was disabled.

Examples

```
# snmpwalk -m $CPDIR/lib/snmp/chkpnt.mib -On -v 2c -c public localhost 1.3.6.1.4.1.2620.1.54

.1.3.6.1.4.1.2620.1.54.1.0 = STRING: On
.1.3.6.1.4.1.2620.1.54.2.0 = STRING: HTTPS Inspection is on
.1.3.6.1.4.1.2620.1.54.3.1.0 = STRING: Enabled
.1.3.6.1.4.1.2620.1.54.3.2.0 = STRING: HSM is enabled for HTTPS inspection
.1.3.6.1.4.1.2620.1.54.3.3.0 = STRING: Accessible
.1.3.6.1.4.1.2620.1.54.3.4.0 = STRING: Gateway can access HSM partition for HTTPS inspection
.1.3.6.1.4.1.2620.1.54.3.5.0 = STRING: HSM on
.1.3.6.1.4.1.2620.1.54.3.6.0 = STRING: Outbound HTTPS inspection works with HSM
```

```
# snmpwalk -m $CPDIR/lib/snmp/chkpnt.mib -Oa -v 2c -c public localhost 1.3.6.1.4.1.2620.1.54

CHECKPOINT-MIB::httpsInspectionStatus.0 = STRING: On
CHECKPOINT-MIB::httpsInspectionStatusDescription.0 = STRING: HTTPS Inspection is on
CHECKPOINT-MIB::hsmEnabled.0 = STRING: Enabled
CHECKPOINT-MIB::hsmEnabledDescription.0 = STRING: HSM is enabled for HTTPS inspection
CHECKPOINT-MIB::hsmPartitionAccess.0 = STRING: Accessible
CHECKPOINT-MIB::hsmPartitionAccessDescription.0 = STRING: Gateway can access HSM partition for HTTPS
inspection
CHECKPOINT-MIB::outboundStatus.0 = STRING: HSM on
CHECKPOINT-MIB::outboundStatusDescription.0 = STRING: Outbound HTTPS inspection works with HSM
```

For more information about SNMP on Gaia OS, see the *R80.40 Gaia Administration Guide* > Chapter *System Management* > Section *SNMP*.

Monitoring HTTPS Inspection with HSM in CLI

Run the "**cpstat https_inspection**" command on the Security Gateway to see the HTTPS Inspection status and the status of connection to the Gemalto HSM Appliance Server.

Syntax

```
cpstat -h
```

```
cpstat https_inspection -f {default | hsm_status | all}
```

For more information about this command, see the *R80.40 CLI Reference Guide* > Chapter *Security Gateway Commands* > Section *cpstat*.

Example outputs

```
[Expert@GW:0]# cpstat https_inspection -f default

HTTPS inspection status (On/Off):    On
HTTPS inspection status description: HTTPS Inspection is on

[Expert@GW:0]#
```

```
[Expert@GW:0]# cpstat https_inspection -f hsm_status

HSM enabled (Enabled/Disabled):                  Enabled
HSM enabled description:                         HSM is enabled for HTTPS inspection
HSM partition access (Accessible/Not Accessible): Accessible
HSM partition access description:                 Gateway can access to HSM partition for HTTPS
inspection
Outbound status (HSM on/HSM off/HSM error):      HSM on
Outbound status description:                     Outbound HTTPS inspection works with HSM

[Expert@GW:0]#
```

```
[Expert@GW:0]# cpstat https_inspection -f all

HTTPS inspection status (On/Off):                On
HTTPS inspection status description:             HTTPS Inspection is on
HSM enabled (Enabled/Disabled):                  Enabled
HSM enabled description:                         HSM is enabled for HTTPS inspection
HSM partition access (Accessible/Not Accessible): Accessible
HSM partition access description:                 Gateway can access to HSM partition for HTTPS
inspection
Outbound status (HSM on/HSM off/HSM error):      HSM on
Outbound status description:                     Outbound HTTPS inspection works with HSM

[Expert@GW:0]#
```

Explanation about the "HTTPS Inspection status"

Item	Possible returned strings	Explanation
`HTTPS inspection status (On/Off)`	`On`	HTTPS Inspection feature is configured on the Security Gateway.
	`Off`	HTTPS Inspection feature is not configured on the Security Gateway.

Explanation about the "HTTPS Inspection status description"

Item	Possible returned strings	Explanation
`HTTPS inspection status description`	`HTTPS Inspection is on`	HTTPS Inspection feature is configured on the Security Gateway.
	`HTTPS Inspection is off`	HTTPS Inspection feature is not configured on the Security Gateway.

Explanation about the "HSM configuration status"

Item	Possible returned strings	Explanation
`HSM enabled (Enabled/Disabled)`	`Enabled`	The value of the "`:enabled()`" attribute is set to "`yes`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway.
	`Disabled`	One of these: ■ The value of the "`:enabled()`" attribute is set to "`no`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway. ■ Gemalto HSM Simplified Client software packages are not installed on the Security Gateway. ■ The `$FWDIR/conf/hsm_configuration.C` file does not exist on the Security Gateway. ■ The "`:enabled()`" attribute is corrupted in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway.  **Important** - In the above cases, outbound HTTPS Inspection works without Gemalto HSM Appliance Server, and SSL keys are stored on the Security Gateway.

Explanation about the "HSM configuration status description"

Item	Possible returned strings	Explanation
`HSM enabled description`	`HSM is enabled for HTTPS inspection`	The value of the "`:enabled()`" attribute is set to "`no`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway.
	`HSM is disabled for HTTPS inspection`	One of these: ■ The value of the "`:enabled()`" attribute is set to "`no`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway. ■ Gemalto HSM Simplified Client software packages are not installed on the Security Gateway. ■ The `$FWDIR/conf/hsm_configuration.C` file does not exist on the Security Gateway. ■ The "`:enabled()`" attribute is corrupted in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway.  **Important** - In the above cases, outbound HTTPS Inspection works without Gemalto HSM Appliance Server, and SSL keys are stored on the Security Gateway.

Explanation about the "HSM partition access status"

Item	Possible returned strings	Explanation
`HSM partition access (Accessible/Not Accessible)`	`N/A`	Security Gateway could not check access its partition on HSM Appliance Server.
	`Accessible`	Security Gateway was able to access its partition on HSM Appliance Server.
	`Not Accessible`	Security Gateway was not able to access its partition on HSM Appliance Server due to an error.  **Important** - In the above case, outbound HTTPS Inspection will not work, and HTTPS traffic will not pass.

Explanation about the "HSM partition access status description"

Item	Possible returned strings	Explanation
`HSM partition access description`	`HSM partition access cannot be checked`	Security Gateway could not check access its partition on HSM Appliance Server. Most probable, because HSM configuration is disabled on the Security Gateway.
	`Gateway can access HSM partition for HTTPS inspection`	Security Gateway was able to access its partition on HSM Appliance Server.
	`Gateway cannot access HSM partition for HTTPS inspection: <error>`	Security Gateway was not able to access its partition on HSM Appliance Server due to an error. All these conditions were met: 1. The value of the "`:enabled()`" attribute is set to "`yes`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway. 2. An error occurred. **Important** - In the above case, outbound HTTPS Inspection will not work, and HTTPS traffic will not pass. Possible error messages are: ■ `HSM configuration file is corrupted` ■ `Loading HSM library failed` ■ `There is no trust or no connectivity with HSM server` ■ `Login to HSM partition failed`

Explanation about the "Outbound HTTPS Inspection status"

Item	Possible returned strings	Explanation
`Outbound status (HSM on/HSM off/HSM error)`	`HSM on`	All these conditions were met: 1. The value of the "`:enabled()`" attribute is set to "`yes`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway. 2. Security Gateway was able to connect to the HSM Appliance Server.
	`HSM off`	One of these: ■ The value of the "`:enabled()`" attribute is set to "`no`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway. ■ Gemalto HSM Simplified Client software packages are not installed on the Security Gateway. ■ The `$FWDIR/conf/hsm_configuration.C` file does not exist on the Security Gateway. ■ The "`:enabled()`" attribute is corrupted in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway.  **Important** - In the above cases, outbound HTTPS Inspection works without Gemalto HSM Appliance Server, and SSL keys are stored on the Security Gateway.
	`HSM error`	All these conditions were met: 1. The value of the "`:enabled()`" attribute is set to "`yes`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway. 2. An error occurred.  **Important** - In the above case, outbound HTTPS Inspection will not work, and HTTPS traffic will not pass.

 **Note** - The conditions for the returned strings are calculated on the Security Gateway during the start of the HTTPS Inspection daemon `wstlsd`, or during policy installation. For example, you can get "`HSM enabled (Enabled/Disabled) = Enabled`" and "`Outbound status (HSM on/HSM off/HSM error) = HSM off`", because when the `wstlsd` daemon started, or during last policy installation, the HSM configuration was disabled.

Explanation about the "Outbound HTTPS Inspection status description"

Item	Possible returned strings	Explanation
`Outbound status description`	`Outbound HTTPS inspection works with HSM`	All these conditions were met: 1. The value of the "`:enabled()`" attribute is set to "`yes`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway. 2. Security Gateway was able to connect to the HSM Appliance Server.
	`Outbound HTTPS inspection works without HSM`	The value of the "`:enabled()`" attribute is set to "`no`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway, or this file does not exist.
	`Outbound HTTPS inspection is off due to HSM error: <error>`	All these conditions were met: 1. The value of the "`:enabled()`" attribute is set to "`yes`" in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway. 2. An error occurred. **Important** - In the above case, outbound HTTPS Inspection will not work, and HTTPS traffic will not pass. Possible error messages are: ■ `HSM configuration file is corrupted` ■ `Loading HSM library failed` ■ `There is no trust or no connectivity with HSM server` ■ `Login to HSM partition failed` ■ `Error importing CA certificate from HSM server` ■ `Error generating key pair on HSM server`

Note - The conditions for the returned strings are calculated on the Security Gateway during the start of the HTTPS Inspection daemon `wstlsd`, or during policy installation. For example, you can get "`HSM enabled (Enabled/Disabled) = Enabled`" and "`Outbound status description = Outbound HTTPS inspection works without HSM`", because when the `wstlsd` daemon started, or during last policy installation, the HSM configuration was disabled.

ISP Redundancy on a Security Gateway

In This Section:

Note - For information about ISP Redundancy on a Cluster, see the *R80.40 ClusterXL Administration Guide*.

Introduction

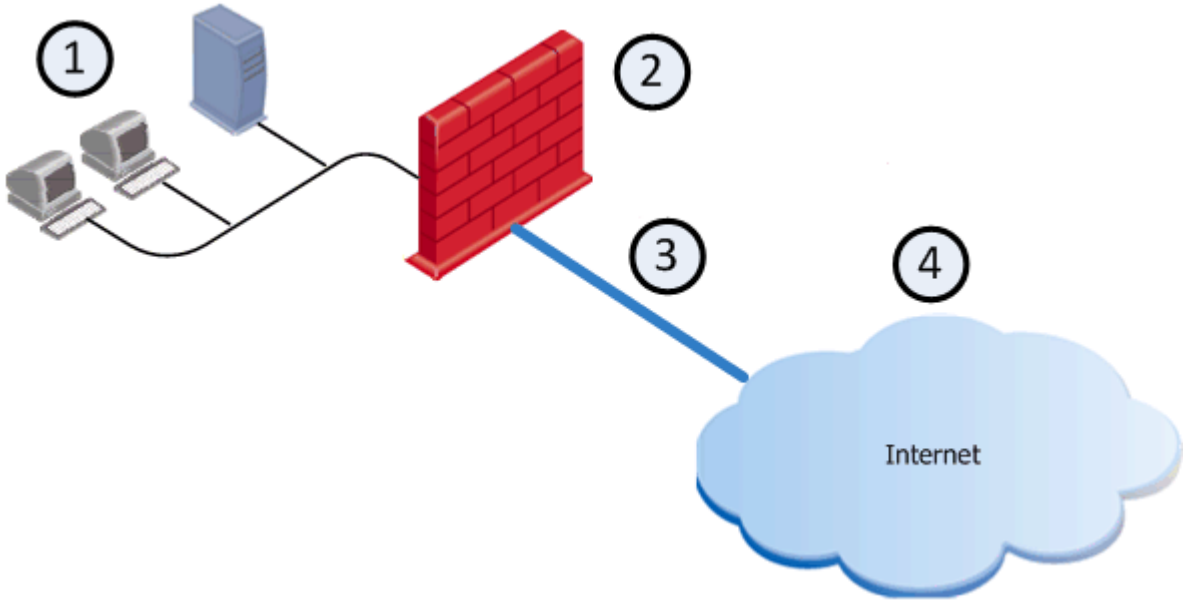
ISP Redundancy connects a Security Gateway to the Internet through redundant Internet Service Provider (ISP) links.

ISP Redundancy monitors the ISP links and chooses the best current link.

Notes:

- R80.40 supports two ISPs.
- ISP Redundancy is intended to traffic that originates on your internal networks and goes to the Internet.

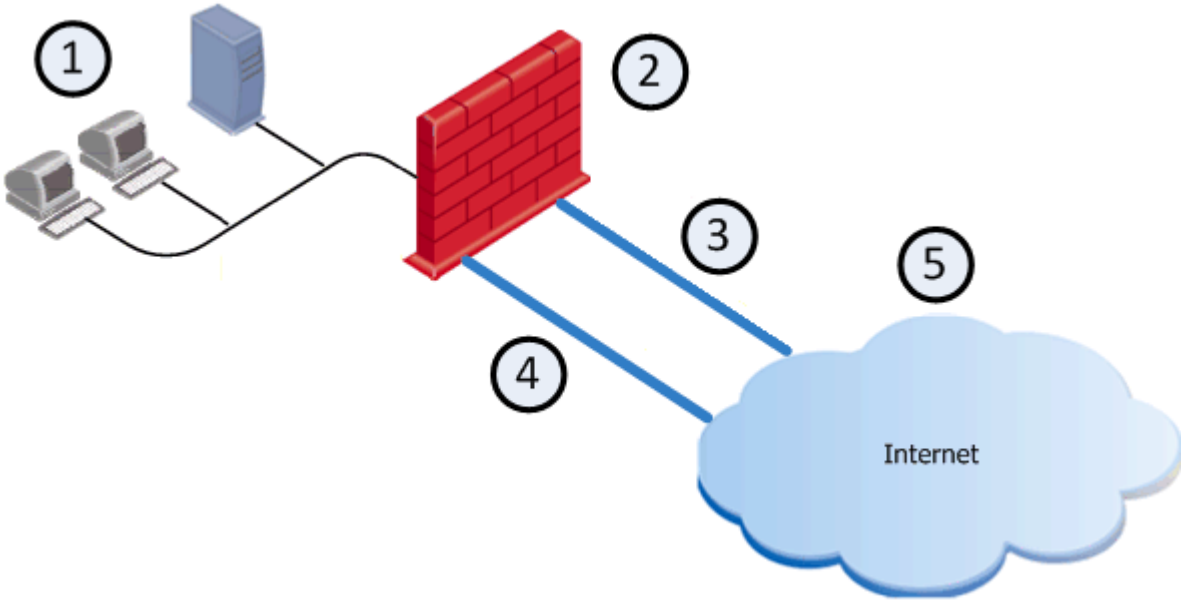
Example of a typical deployment with a single ISP link



Item	Description
1	Internal network
2	Security Gateway
3	ISP
4	Internet

Example of a typical deployment with two dedicated physical interfaces for two ISP links

Best Practice - We recommend this deployment, because it is simpler than deployment with one dedicated physical interface.



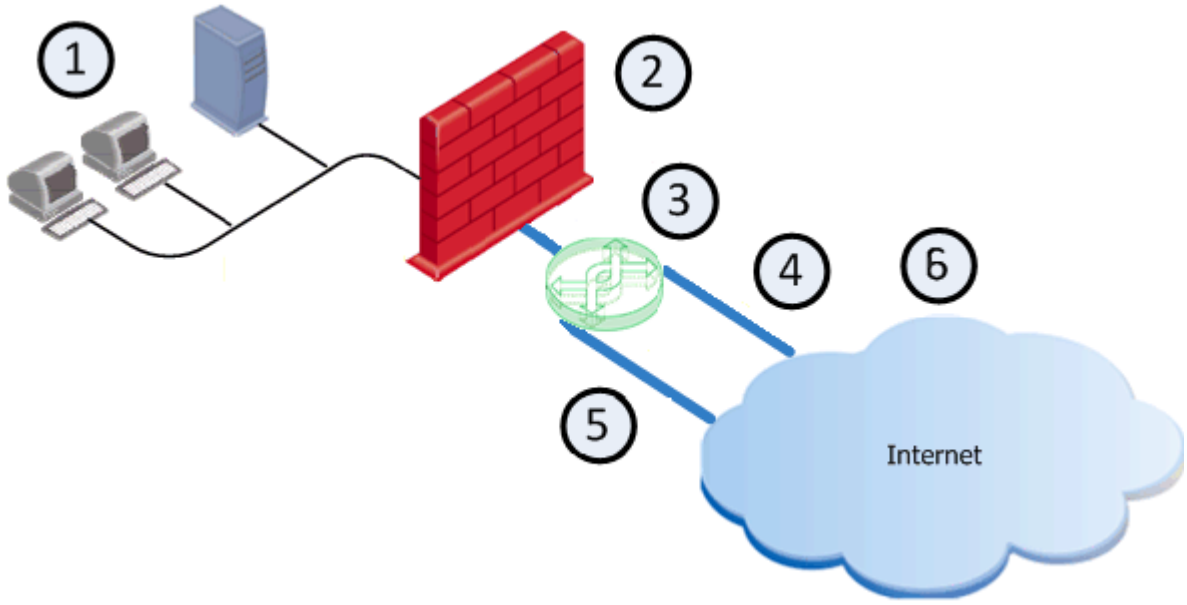
Item	Description
1	Internal network
2	Security Gateway
3	ISP A
4	ISP B
5	Internet

Example of a typical deployment with one dedicated physical interface for two ISP links

If only one external interface is available on the Security Gateway, you can configure two subnets on the same external interface.

(See the *R80.40 Gaia Administration Guide* > Chapter *Network Management* > Section *Network Interfaces* > Section *Aliases*.)

Both ISP links are then connected to the same Security Gateway interface, but to different next hop routers, usually through a switch.



Item	Description
1	Internal network
2	Security Gateway
3	Switch
4	ISP A
5	ISP B
6	Internet

ISP Redundancy Modes

ISP Redundancy configuration modes control the behavior of outgoing connections from internal clients to the Internet:

Mode	Description
Load Sharing	Uses the two links to distribute load of connections. Connections coming in are alternated. You can configure best relative loads for the links (set a faster link to handle more load). New connections are randomly assigned to a link. If one link fails, the other link takes the load. In this mode, incoming connections can reach the application servers through either ISP link because the Security Gateway can answer DNS requests for the IP address of internal servers with IP addresses from both ISPs by alternating their order.
Primary/Backup	Uses one link for connections. It switches to the Backup link if the Primary link fails. When the Primary link is restored, new connections are assigned to it. Existing connections continue on the Backup link until they are complete. In this mode, incoming connections (from the Internet to application servers in the DMZ or internal networks) also benefit, because the Security Gateway returns packets using the same ISP Link, through which the connection was initiated.

Best Practice:

- If both ISPs are basically the same, use the Load Sharing mode to ensure that you are making the best use of both ISPs.
- You may prefer to use one of your two ISPs that is more cost-effective in terms of price and reliability. In that case, use Primary/Backup mode and set the more cost-effective ISP as the Primary ISP link.

Outgoing Connections

- In ISP Redundancy **Load Sharing** mode, outgoing traffic that exits the Security Gateway on its way to the Internet is distributed between the ISP Links. You can set a relative weight for how much you want each of the ISP Links to be used.

  For example, if one link is faster, it can be configured to route more traffic across that ISP link than the other.

- In ISP Redundancy **Primary/Backup** mode, outgoing traffic uses an active primary link.

  Hide NAT is used to change the source address of outgoing packets to the address of the interface, through which the packet leaves the Security Gateway. This allows return packets to be automatically routed through the same ISP link, because their destination address is the address of the correct link. Hide NAT is configured by the administrator.

Incoming Connections

For external users to make incoming connections, the administrator must give each application server two routable IP addresses, one for each ISP. The administrator must also configure Static NAT to translate the routable addresses to the real server address.

If the servers handle different services (for example, HTTP and FTP), you can use NAT to employ only two routable IP addresses for all the publicly available servers.

External clients use one of the two addresses. In order to connect, the clients must be able to resolve the DNS name of the server to the correct IP address.

Note - In the following example, the subnets **172.16.0.0/24** and **192.168.0.0/24** represent public routable addresses.

In the following example, the Web server **www.example.com** is assigned an IP address from each ISP:

- 192.168.1.2 from ISP A
- 172.16.2.2 from ISP B

If the **ISP Link A** is down, then IP address **192.168.1.2** becomes unavailable, and the clients must be able to resolve the URL **www.example.com** to the IP address **172.16.2.2**.

An incoming connection is established, based on this example, in the following sequence:

1. When an external client on the Internet contacts **www.example.com**, the client sends a DNS query for the IP address of this URL.

   The DNS query reaches the Security Gateway. The Security Gateway has a built-in mini-DNS server that can be configured to intercept DNS queries (of Type A) for servers in its domain.
2. A DNS query arriving at an interface that belongs to one of the ISP links, is intercepted by the Security Gateway.
3. If the Security Gateway recognizes the name of the host, it sends one of the following replies:
   - In ISP Redundancy **Primary/Backup** mode, the Security Gateway replies only with the IP addresses associated with the Primary ISP link, as long as the Primary ISP link is active.
   - In ISP Redundancy **Load Sharing** mode, the Security Gateway replies with two IP addresses, alternating their order.
4. If the Security Gateway is unable to handle DNS requests (for example, it may not recognize the host name), it passes the DNS query to its original destination or the DNS server of the domain **example.com**.
5. When the external client receives the reply to its DNS query, it opens a connection. Once the packets reach the Security Gateway, the Security Gateway uses Static NAT to translate the destination IP address **192.168.1.2** or **172.16.2.2** to the real server IP address **10.0.0.2**.
6. The Security Gateway routes the reply packets from the server to the client through the same ISP link that was used to initiate the connection.

Configuring ISP Redundancy on a Security Gateway

1. Connect with SmartConsole to the Security Management Server or Domain Management Server that manages this Security Gateway.
2. From the left navigation panel, click **Gateways & Servers**.
3. Open the Security Gateway object.
4. Click **Other** > **ISP Redundancy**.
5. Select **Support ISP Redundancy**.
6. Select the redundancy mode - **Load Sharing** or **Primary/Backup**.
7. Configure the ISP Links.

Procedure

Make sure you have the ISP data - the speed of the link and next hop IP address.

Automatic vs Manual configuration:

- If the Security Gateway object has two interfaces with Topology "**External**" in the **Network Management** page, you can configure the ISP links automatically.

Configuring ISP links automatically

a. Click **Other** > **ISP Redundancy**.

b. Click **Set initial configuration**.

The ISP Links are added automatically.

c. For **Primary/Backup** mode, make sure the Primary interface is first in the list. Use the arrows on the right to change the order.

d. Click **OK**.

- If the Security Gateway object only one interface with Topology "**External**" in the **Network Management** page, you must configure the ISP links manually.

  ### Configuring ISP links manually

  a. Click **Other** > **ISP Redundancy**.

  b. In the **IPS Links** section, click **Add**.

     The **ISP Link** window opens.

  c. Click the **General** tab.

  d. In the **Name** field, enter a name of this link (desired text).

     The name you enter here is used in the ISP Redundancy commands (see *"Controlling ISP Redundancy from CLI" on page 110*).

  e. Select the **Interface** of the Security Gateway for this ISP link.

     - If the Security Gateway object has two interfaces with Topology "**External**" in the **Network Management** page, set each ISP link to a different interface.

       If one of the ISP links is the connection to a backup ISP, configure the ISP Redundancy Script (see *"Controlling ISP Redundancy from CLI" on page 110*).

     - If the Security Gateway object only one interface with Topology "**External**" in the **Network Management** page, set each ISP link to connect to this interface.

  f. Configure the **Next Hop IP Address**.

     - If the Security Gateway object has two interfaces with Topology "**External**" in the **Network Management** page, leave this field empty and click **Get from routing table**. The next hop is the default gateway.

     - If the Security Gateway object only one interface with Topology "**External**" in the **Network Management** page, set each ISP link to a different next hop router.

  g. For ISP Redundancy in Load Sharing mode, enter the **Weight** value.

     For equal traffic distribution between the two IPS link, enter **50** in each ISP link.

     If one ISP link is faster, increase this value and decrease it for the other ISP link, so that the sum of these two values is always equal 100.

  h. Click the **Advanced** tab.

  i. Define hosts to be monitored, to make sure the link is working.

     Add the applicable objects to the **Selected hosts** section.

  j. Click **OK**.

8. Configure the Security Gateway to be the DNS server.

Procedure

The Security Gateway, or a DNS server behind it, must respond to DNS queries.

It resolves IP addresses of servers in the DMZ (or another internal network).

Get a public IP address from each ISP.

If public IP addresses are not available, register the domain to make the DNS server accessible from the Internet.

The Security Gateway intercepts DNS queries "Type A" for the web servers in its domain that come from external hosts.

- If the Security Gateway recognizes the external host, it replies:
  - In ISP Redundancy **Load Sharing** mode, the Security Gateway replies with two IP addresses, alternating their order.
  - In ISP Redundancy **Primary/Backup** mode, the Security Gateway replies with the IP addresses of the active ISP link.
- If the Security Gateway does not recognize the host, it passes the DNS query on to the original destination, or to the domain DNS server.

To enable DNS server:

a. Click **Other** > **ISP Redundancy**.

b. Select **Enable DNS Proxy**.

c. Click **Configure**.

d. Add your DMZ or Web servers. Give each server two public IP addresses - one from each ISP.

e. In the **DNS TTL**, enter a number of seconds.

This sets a Time To Live for each DNS reply.

DNS servers in the Internet cannot cache your DNS data in the reply for longer than the TTL.

f. Click **OK**.

g. Configure Static NAT to translate the public IP addresses to the real server's IP address.

External clients use one of the two IP addresses.

Note - If the servers use different services (for example, HTTP and FTP), you can use NAT for only two public IP addresses.

h. Define an Access Control Policy rule:

Name	Source	Destination	VPN	Services & Applications	Action	Track	Install On
DNS Proxy	Applicable sources	Applicable DNS Servers	Any	domain_udp	Accept	None	Policy Targets

To register the domain and get IP addresses:

a. Register your domain with the two ISP.

b. Tell the ISP the two IP addresses of the DNS server that respond to DNS queries for the domain.

c. For each server in the DMZ, get two public IP addresses, one from each ISP.

d. In SmartConsole, click **Menu** > **Global properties**.

e. From the left tree, click **NAT - Network Address Translation**.

f. In the **Manual NAT rules** section, select **Translate destination on client side**.

g. Click **OK**.

9. Configure the Access Control Policy for ISP Redundancy.

Procedure

The Access Control Policy must allow connections through the ISP links, with Automatic Hide NAT on network objects that start outgoing connections.

a. In the properties of the object for an internal network, select **NAT** > **Add Automatic Address Translation Rules**.

b. Select **Hide behind the gateway**.

c. Click **OK**.

d. Define rules for publicly reachable servers (Web servers, DNS servers, DMZ servers).

- If you have one public IP address from each ISP for the Security Gateway, define Static NAT.

  Allow specific services for specific servers.

  For example, make NAT rules, so that incoming HTTP connections from the two ISPs reach a Web server, and DNS traffic from the ISP reach the DNS server.

  Example: Manual Static Rules for a Web Server and a DNS Server

Original Source	Original Destination	Original Services	Translated Source	Translated Destination	Translated Services	Install On	Comment
Any	Host object with IP address of Web Server	http	= Original	S 50.50.50.2	= Original	Policy Targets	Incoming Web - ISP A
Any	Host object with IP address of Web Server	http	= Original	S 60.60.60.2	= Original	Policy Targets	Incoming Web - ISP B
Any	Host object with IP address of DNS Server	domain_udp	= Original	S 50.50.50.3	= Original	Policy Targets	Incoming DNS - ISP A
Any	Host object with IP address of DNS Server	domain_udp	= Original	S 60.60.60.3	= Original	Policy Targets	Incoming DNS - ISP B

- If you have a public IP address from each ISP for each publicly reachable server (in addition to the Security Gateway), define NAT rules:

  i. Give each server a private IP address.

  ii. Use the public IP addresses in the **Original Destination**.

  iii. Use the private IP address in the **Translated Destination**.

  iv. Select **Any** as the **Original Service**.

Note - If you use Manual NAT, then automatic ARP does not work for the IP addresses behind NAT. You need to configure the `local.arp` file as described in sk30197.

10. Install the Access Control Policy on this Security Gateway object.

ISP Redundancy and VPN

Note - ISP Redundancy settings override the **VPN Link Selection** settings.

When ISP Redundancy is enabled, VPN encrypted connections survive a failure of an ISP link.

The settings in the ISP Redundancy page override settings in the **IPsec VPN > Link Selection** page.

Configuring ISP Redundancy for VPN with a Check Point peer

Step	Instructions
1	Connect with SmartConsole to the Security Management Server or Domain Management Server that manages this Security Gateway.
2	From the left navigation panel, click **Gateways & Servers**.
3	Open the Security Gateway object.
4	In the left navigation tree, go to **Other** > **ISP Redundancy**.
5	Select **Apply settings to VPN traffic**.
6	In the left navigation tree, go to **IPsec VPN** > **Link Selection**.
7	Make sure that **Use ongoing probing. Link redundancy mode** shows the mode of the ISP Redundancy: **High Availability** (for Primary/Backup) or **Load Sharing**. The **VPN Link Selection** now only probes the ISP configured in ISP Redundancy.

Configuring ISP Redundancy for VPN with a third-party peer

If the VPN peer is *not* a Check Point Security Gateway, the VPN may fail, or the third-party device may continue to encrypt traffic to a failed ISP link.

- Make sure the third-party VPN peer recognizes encrypted traffic from the secondary ISP link as coming from the Check Point cluster.
- Change the configuration of ISP Redundancy to *not* use these Check Point technologies:
  - **Use Probing** - Makes sure that **Link Selection** uses another option.
  - The options **Load Sharing**, **Service Based Link Selection**, and **Route based probing** works only on Check Point Security Gateways and Clusters.

    If used, the Security Gateway or Cluster Members use one link to connect to the third-party VPN peer.

    The link with the highest prefix length and lowest metric is used.

Controlling ISP Redundancy from CLI

You can control the ISP Redundancy behavior from CLI.

Force ISP Link State

Use the "`fw isp_link`" command to force the ISP link state to Up or Down.

Use this to test installation and deployment, or to force the Security Gateway to recognize the true link state if it cannot (the ISP link is down but the gateway sees it as up).

- You can run this command on the Security Gateway:

```
fw isp_link <Name of ISP Link in SmartConsole> {up | down}
```

- You can run this command on the Security Management Server:

```
fw isp_link <Name of Security Gateway Object> <Name of ISP Link in
SmartConsole> {up | down}
```

For more information, see the *R80.40 CLI Reference Guide* > Chapter *Security Gateway Commands* - Section *fw* - Section *fw isp_link*.

The ISP Redundancy Script

When the Security Gateway starts, or an ISP link state changes, the `$FWDIR/bin/cpisp_update` script runs on the Security Gateway.

This script changes the default route of the Security Gateway.

For example, you can force the Security Gateway to change the state of a dialup interface to match that state of its ISP link.

Edit this script to enable a dialup connection for one of the ISP links.

To configure a dialup connection:

1. In the script on the Security Gateway, enter the command to change the dialup interface state:
   - If the ISP link goes down:

     ```
     fw isp_link <Name of ISP Link in SmartConsole> down
     ```

   - If the ISP link goes up:

     ```
     fw isp_link <Name of ISP Link in SmartConsole> up
     ```

2. If you use PPPoE or PPTP xDSL modems, in the PPPoE or PPTP configuration, the **Use Peer as Default Gateway** option must not be selected.

Mirror and Decrypt

The Mirror and Decrypt feature performs these actions on your Security Gateway, or Cluster:

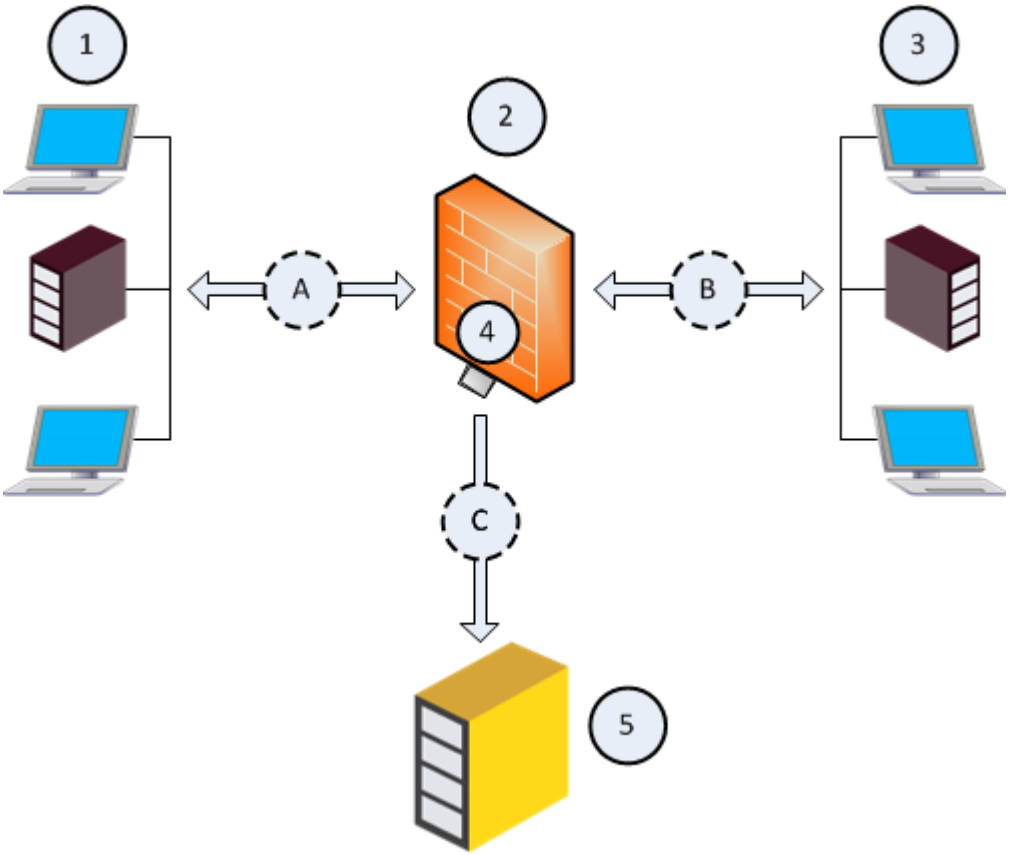
Action	Description
Only mirror of all traffic	Your Security Gateway or Cluster clones all traffic (including HTTPS without decryption) that passes through it, and sends it out of the designated physical interface.
Mirror and Decrypt of HTTPS traffic	Your Security Gateway or Cluster clones all HTTPS traffic that passes through it, decrypts it, and sends it in clear-text out of the designated physical interface.  **Note** - If you wish to decrypt the HTTPS traffic, you must enable and configure the HTTPS Inspection on your Security Gateway, or Cluster.

You can add a third-party Recorder or Packet-Broker in your environment and forward to it the traffic that passes through your Security Gateway, or Cluster.

This Recorder or Packet-Broker must work in monitor (promiscuous) mode to accept the decrypted and mirrored traffic from your Security Gateway, or Cluster.

Security Gateway, or Cluster works only with *one* Recorder, which is directly connected to a designated physical network interface (NIC) on the Check Point Gateway, or Cluster Members.

Example Topology and Traffic Flow:



Item	Description
1	First network that sends and receives traffic through the Security Gateway (2).
2	Security Gateway, through which networks (1) and (3) send and receive their traffic.
3	Second network that sends and receives traffic through the Security Gateway (2).
4	Designated physical interface on the Security Gateway (2).
5	Recorder, or Packet-Broker that works in a monitor (promiscuous) mode.
A	Traffic flow between the first network (1) and the Security Gateway (2).
B	Traffic flow between the second network (3) and the Security Gateway (2).
C	Flow of the decrypted and mirrored traffic from the Security Gateway(2) to the Recorder, or Packet-Broker (5).

Source MAC address of the decrypted and mirrored packets

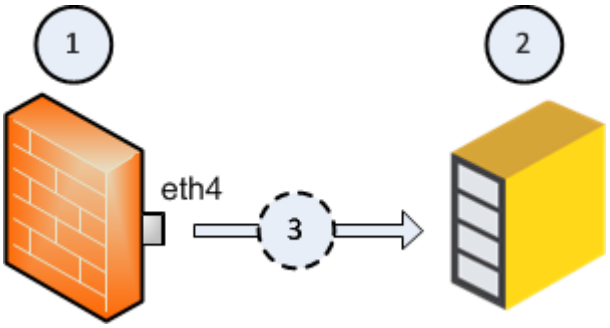
Traffic	Source MAC address of the decrypted and mirrored packets the Security Gateway and Cluster Members send
Mirror only of all traffic	MAC address of the designated physical interface.
Mirror and Decrypt of HTTPS traffic	00:00:00:00:00:00:

Mirror and Decrypt Requirements

Item	Description
1	Designated network interface for Mirror and Decrypt: a. Select a designated physical interface on your Security Gateway, or *each* cluster member.  **Important:** ■ On cluster members, you must select an interface with the *same name* (for example, `eth3` on each cluster member). ■ Select an interface with the largest available throughput (for example, 10G, 40G), because this interface passes the combined traffic from all other interfaces. b. Assign a dummy IP address to the designated interface.  **Important** - This IP address cannot collide with other IP addresses used in your environment. This IP address cannot belong to subnets used in your environment. Make sure to configure the correct subnet mask. After you enable traffic mirroring on this interface in SmartConsole, all other traffic that is routed to this interface is dropped. c. On cluster members, you must configure this designated physical interface in the `$FWDIR/conf/discntd.if` file.  **Note** - This prevents the interfaces that are not used from sending Cluster Control Protocol (CCP) packets that can overwhelm the Mirror and Decrypt recorders.
2	Maximum Transmission Unit (MTU) on the Mirror and Decrypt designated physical interface: ■ MTU value has to be 1500 (default), or at least the maximum MTU value from other interfaces on the Security Gateway.
3	HTTPS Inspection for decrypting the HTTPS traffic: ■ You must enable the HTTPS Inspection in SmartConsole in the object of the Security Gateway, Cluster, or Virtual System. ■ You must configure the HTTPS Inspection Rule Base.
4	Access Rules for traffic you wish to Mirror and Decrypt: ■ You must create special rules in the Access Control Policy for the traffic you wish to mirror and decrypt.

Configuring Mirror and Decrypt in Gateway mode

Example topology:



Item	Description
1	Security Gateway, through which your networks send and receive their traffic.
2	Recorder, or Packet-Broker that works in a monitor (promiscuous) mode.
3	Flow of the decrypted and mirrored traffic from the Security Gateway (**1**) to the Recorder, or Packet-Broker (**2**).
eth4	Designated physical interface on the Security Gateway (**1**).

Workflow for configuring Mirror and Decrypt in Gateway mode:

Step	Description
1	Read and follow the *"Mirror and Decrypt Requirements" on page 114*.
2	Prepare the Security Gateway, or *each* cluster member. See *"Preparing the Security Gateway or each Cluster Member" on page 116*.
3	Configure the Mirror and Decrypt in the Security Gateway, or Cluster object in SmartConsole. See *"Configuring Mirror and Decrypt in SmartConsole for Gateway Mode" on page 118*.

Preparing the Security Gateway or each Cluster Member

Step	Description
1	Select a designated physical interface for Mirror and Decrypt on the Security Gateway, or *each* cluster member.  **Important** - On cluster members, you must select an interface with the *same name* (for example, `eth3` on each cluster member).
2	Configure a dummy IP address on this designated physical interface.  **Important** - This IP address cannot collide with other IP addresses used in your environment. This IP address cannot belong to subnets used in your environment. Make sure to configure the correct subnet mask. After you enable traffic mirroring on this interface in SmartConsole, all other traffic that is routed to this interface is dropped. For instructions about configuring an IP address on a physical interface, see the *R80.40 Gaia Administration Guide* - Chapter *Network Management* - Section *Network Interfaces* - Section *Physical Interfaces*.
3	Configure the required Maximum Transmission Unit (MTU) on this designated physical interface. MTU has to be the default 1500, or at least the maximal MTU value from other interfaces on the Security Gateway. For instructions about configuring an MTU on a physical interface, see the *R80.40 Gaia Administration Guide* - Chapter *Network Management* - Section *Network Interfaces* - Section *Physical Interfaces*.

Step	Description

4	 **Important** - On cluster members, you must configure this designated physical interface in the `$FWDIR/conf/discntd.if` file on *each* Cluster Member.
	
	a. Connect to the command line on each Cluster Member.
	b. Log in to the Expert mode.
	c. Create the `$FWDIR/conf/discntd.if` file:
	`touch $FWDIR/conf/discntd.if`
	d. Edit the `$FWDIR/conf/discntd.if` file in the Vi editor:
	`vi $FWDIR/conf/discntd.if`
	e. Write the name of the designated physical interface. After the interface name, you must press Enter.
	Note - Comments are not allowed in this file.
	f. Save the changes in the file and exit the editor.
	
	  **Note** - To apply the configuration from the file and make it persistent, install an Access Control Policy on the cluster object. You install the Access Control Policy later, after the required configuration steps in the SmartConsole.

Configuring Mirror and Decrypt in SmartConsole for Gateway Mode

Workflow for Security Gateway, or Cluster in Gateway mode:

1. Enable the HTTPS Inspection in the object of your Security Gateway, or Cluster (for decrypting the HTTPS traffic).

   Procedure

Step	Description
a	Connect with SmartConsole to the Management Server.
b	From the left navigation panel, click **Gateways & Servers**.
c	Open the object of the Security Gateway, or Cluster.
d	From the navigation tree, click **HTTPS Inspection**.
e	View and export the certificate.
f	Check **Enable HTTPS Inspection**.
g	Click **OK**.

2. Configure the HTTPS Inspection Rule Base (for decrypting the HTTPS traffic).

   Procedure

Step	Description
a	From the left navigation panel, click **Security Policies**.
b	From the left tree, click **HTTPS Inspection**.
d	Configure the HTTPS Inspection Rule Base. See *R80.40 Security Management Administration Guide*. For more settings, in the **HTTPS Tools** section, click **Additional Settings**.
e	Publish the SmartConsole session.

3. Activate the Mirror and Decrypt in the object of your Security Gateway, or Cluster.

Procedure

Step	Description
a	From the left navigation panel, click **Gateways & Servers**.
b	Open the object of the Security Gateway, or Cluster.
c	From the navigation tree of the gateway object, click **Network Management**.
d	From the top toolbar, click **Get Interfaces Without Topology**.
e	Make sure the interface designated for Mirror and Decrypt is listed with the dummy IP address.
f	Select the interface designated for Mirror and Decrypt and click **Edit**.
g	From the navigation tree, click **General**.
h	In the **General** section: In the **Network Type** field, select **Private**.  **Note** - This field shows only in **Cluster** objects.
i	In the **Topology** section: Click **Modify**. The **Topology Settings** window opens.
j	In the **Leads To** section: i. Select **Override**. ii. Select **This Network (Internal)**. iii. Select **Network defined by the interface IP and Net Mask**.
k	In the **Security Zone** section: i. Select **User defined**. ii. Do not check the **Specify Security Zone**.
l	In the **Anti-Spoofing** section: Clear the **Perform Anti-Spoofing based on interface topology**.
m	Click **OK** to save the changes and close the **Topology Settings** window.
n	From the navigation tree of the Security Gateway, or Cluster object: Click the **[+]** near the **Other** and click **Mirror and Decrypt**.

Step	Description
o	Check **Mirror gateway traffic to interface**. The **Mirror and Decrypt - User Disclaimer** window opens. i. Read the text carefully. ii. Check **I agree to the terms and conditions**. iii. Click **OK** to accept and close the disclaimer.
p	In the **Mirror gateway traffic to interface** field, select the designated physical interface.
q	Click **OK** to save the changes and close the Security Gateway, or Cluster properties window.

4. Configure the Mirror and Decrypt rules in the Access Control Policy for the traffic you wish to mirror and decrypt.

Procedure

Best Practice - We recommend you to configure a new separate Access Control Layer to contain Mirror and Decrypt rules. Alternatively, you can configure the Mirror and Decrypt rules in the regular Rule Base.

Important - When you configure the Mirror and Decrypt rules, these limitations apply:

- In the Mirror and Decrypt rules, you must *not* select Content criteria, such as Application, URL Filtering, Service matched by IP Protocol, Content Awareness.
- Above the Mirror and Decrypt rules, you must *not* configure other rules that contain Content criteria, such as Application, URL Filtering, Service matched by IP Protocol, Content Awareness.
- You must configure rules that contain an excluded source or an excluded destination above the Mirror and Decrypt rules.

  The **Name** column of these rules cannot contain these strings: **<M&D>**, **<M&d>**, **<m&D>**, or **<m&d>**.

The procedure below describes how to configure the Mirror and Decrypt rules in a separate Access Control Layer:

Step	Description
a	In SmartConsole, from the left navigation panel, click **Security Policies**.
b	Create a new Access Control Layer in the Access Control Policy.
c	In SmartConsole top left corner, click **Menu > Manage policies and layers**.

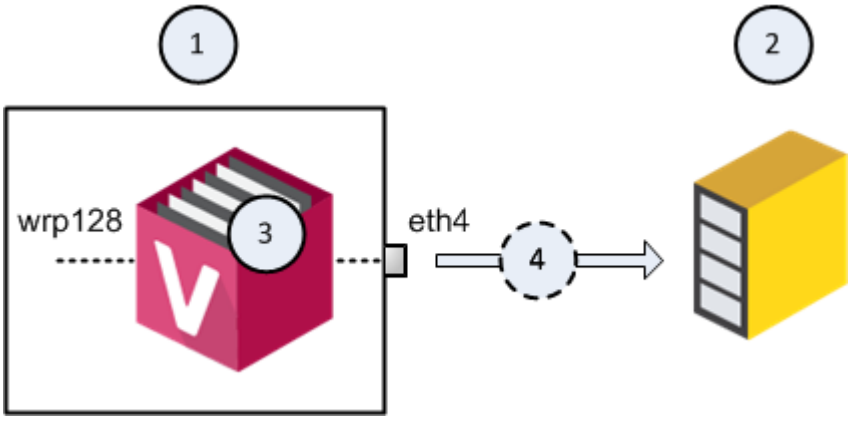
Step	Description
d	Select the existing policy and click **Edit** (the pencil icon). Alternatively, create a new policy.
e	From the navigation tree of the **Policy** window, click **General**.
f	In the **Policy Types** section, make sure you select **only** the **Access Control**.
g	In **Access Control** section, click on the **+** (plus) icon. A pop up window opens.
h	In the top right corner of this pop up window, click **New Layer**. The **Layer Editor** window opens.
i	From the navigation tree of the **Layer Editor** window, click **General**.
j	In the **Blades** section, make sure you select **only** the **Firewall**.
k	On other pages of the **Layer Editor** window, configure additional applicable settings. Click **OK**.
l	In the **Access Control** section, you see the **Network** Layer and the new **Access Control** Layer.
m	Click **OK** to save the changes and close the **Policy** window.
n	In SmartConsole, at the top, click the tab of the applicable policy.
o	In the **Access Control** section, click the new Access Control Layer. In the default rule, you must change the `Action` column from `Drop` to `Accept` to *not* affect the policy enforcement: ■ **Name** - Your text  **Important** - You *cannot* use these strings: **<M&D>**, **<M&d>**, **<m&D>**, or **<m&d>** ■ **Source** - ***Any** ■ **Destination** - ***Any** ■ **VPN** - ***Any** ■ **Services & Applications** - ***Any** ■ **Action** - Must contain **Accept** ■ **Track** - **None** ■ **Install On** - ***Policy Targets**

Step	Description
p	Above the existing Cleanup rule, add the applicable rules for the traffic you wish to Mirror and Decrypt. You must configure the Mirror and Decrypt rules as follows: ■ **Name** - Must contain one of these strings (the angle brackets **<>** are mandatory): • **<M&D>** • **<M&d>** • **<m&D>** • **<m&d>** ■ **Source** - Select the applicable objects ■ **Destination** - Select the applicable objects ■ **VPN** - Must leave the default `*Any` ■ **Services & Applications** - Select the applicable services (to decrypt the HTTPS traffic, select the applicable HTTP, HTTPS, or Proxy services) ■ **Action** - Must contain **Accept** ■ **Track** - Select the applicable option (**None**, **Log**, or **Alert**) ■ **Install On** - Must contain one of these objects: • ***Policy Targets** (this is the default) • The Security Gateway, or Cluster object, whose version is **R80.20** or higher
	 **Important:** ■ In the Mirror and Decrypt rules, you must *not* select Content criteria, such as Application, URL Filtering, Service matched by IP Protocol, Content Awareness. ■ Above the Mirror and Decrypt rules in this Ordered Layer, you must *not* configure other rules that contain Content criteria, such as Application, URL Filtering, Service matched by IP Protocol, Content Awareness. ■ You must configure rules that contain an excluded source or an excluded destination above the Mirror and Decrypt rules. The **Name** column of these rules cannot contain these strings: **<M&D>**, **<M&d>**, **<m&D>**, or **<m&d>**.
q	Publish the SmartConsole session.
r	Install the Access Control Policy.

Step	Description
s	If in a Mirror and Decrypt rule you set the **Track** to **Log**, then you can filter the logs for this rule by the **Access Rule Name**, which contains the configured string: **<M&D>**, **<M&d>**, **<m&D>**, or **<m&d>**.

Configuring Mirror and Decrypt in VSX mode

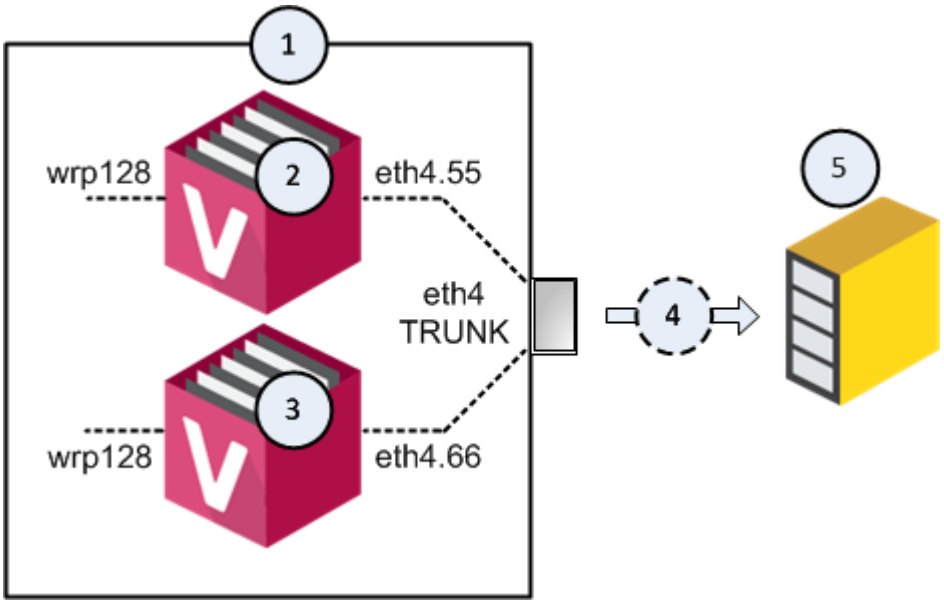
Example topology for one Virtual System:



Item	Description
1	VSX Gateway.
2	Recorder, or Packet-Broker that works in a monitor (promiscuous) mode.
3	Virtual System, through which your networks send and receive their traffic.
4	Flow of the decrypted and mirrored traffic from the VSX Gateway (**1**) to the Recorder, or Packet-Broker (**2**).
eth4	Designated physical interface on the VSX Gateway (**1**). Virtual System (**3**) connects directly to this physical interface.
wrp128	One of the virtual interfaces on the Virtual System (**3**).

Example topology for several Virtual Systems:

Note - This topology requires you to configure a VLAN Trunk on the Recorder or Packet-Broker. The VLAN Trunk on the Recorder or Packet-Broker must accept all VLAN IDs that you configure in the objects of the applicable Virtual Systems in SmartConsole.



Item	Description
1	VSX Gateway.
2	First Virtual System, through which your networks send and receive their traffic.
3	Second Virtual System, through which your networks send and receive their traffic.
4	Flow of the decrypted and mirrored traffic from the VSX Gateway (**1**) to the Recorder, or Packet-Broker (**5**).
5	Recorder, or Packet-Broker.
eth4	Designated physical interface on the VSX Gateway (**1**). This interface is configured as VLAN Trunk in the VSX Gateway object in SmartConsole. Virtual Systems (**2** and **3**) connect to this VLAN Trunk interface with VLAN interfaces.
eth4.55	VLAN interface on the first Virtual System (**2**).
eth4.66	VLAN interface on the second Virtual System (**3**).
wrp128	One of the virtual interfaces on the Virtual Systems (**2** and **3**).

Important - It is not supported to change the designated physical interface with the "`vsx_util change_interfaces`" command. For information about this command, see the *R80.40 VSX Administration Guide*.

Workflow for configuring Mirror and Decrypt in VSX mode:

Step	Description
1	Read and follow the *"Mirror and Decrypt Requirements" on page 114*.
2	Prepare the VSX Gateway, or *each* VSX cluster member. See *"Preparing the VSX Gateway or each VSX Cluster Member" on page 127*.
3	Configure the Mirror and Decrypt in the Virtual System object in SmartConsole. See: ■ *"Configuring Mirror and Decrypt in SmartConsole for One Virtual System" on page 129*. ■ *"Configuring Mirror and Decrypt in SmartConsole for Several Virtual Systems" on page 135*.

Preparing the VSX Gateway or each VSX Cluster Member

Item	Description
1	Select a designated physical interface for Mirror and Decrypt on the VSX Gateway, or *each* VSX cluster member.  **Important** - On VSX cluster members, you must select an interface with the *same name* (for example, `eth3` on each VSX cluster member).
2	Do *not* configure an IP address on this designated physical interface.
3	Configure the required Maximum Transmission Unit (MTU) on this designated physical interface. MTU has to be the default 1500, or at least the maximal MTU value from other interfaces on the VSX Gateway, or VSX cluster member. For instructions about configuring an MTU on a physical interface, see *R80.40 Gaia Administration Guide* - Chapter *Network Management* - Section *Network Interfaces* - Section *Physical Interfaces*.

Item	Description

4	 **Important** - In VSX cluster, you must configure this designated physical interface in the `$FWDIR/conf/discntd.if` file on *each* VSX Cluster Member.
	
	a. Connect to the command line.
	b. Log in to the Expert mode.
	c. Go to the context of the Virtual System 0:
	`vsenv 0`
	Output shows:
	`Context is set to Virtual Device <Name of VSX Gateway> (ID 0).`
	d. Create the `$FWDIR/conf/discntd.if` file:
	`touch $FWDIR/conf/discntd.if`
	e. Edit the `$FWDIR/conf/discntd.if` file in the Vi editor:
	`vi $FWDIR/conf/discntd.if`
	f. Write the name of the designated physical interface. After the interface name, you must press Enter.
	Note - Comments are not allowed in this file.
	g. Save the changes in the file and exit the Vi editor.
	
	  **Note** - To apply the configuration from the file and make it persistent, install an Access Control Policy on the VSX cluster object. You install the Access Control Policy later, after the required configuration steps in the SmartConsole.

Configuring Mirror and Decrypt in SmartConsole for One Virtual System

Workflow for one Virtual System:

1. Enable the HTTPS Inspection in the object of the Virtual System (for decrypting the HTTPS traffic).

   Procedure

Step	Description
a	Connect with SmartConsole to the Management Server.
b	From the left navigation panel, click **Gateways & Servers**.
c	Open the Virtual System object.
d	From the navigation tree, click **HTTPS Inspection**.
e	View and export the certificate.
f	Check **Enable HTTPS Inspection**.
g	Click **OK**.

2. Configure the HTTPS Inspection Rule Base (for decrypting the HTTPS traffic).

   Procedure

Step	Description
a	From the left navigation panel, click **Security Policies**.
b	From the left tree, click **HTTPS Inspection**.
d	Configure the HTTPS Inspection Rule Base. See *R80.40 Security Management Administration Guide*. For more settings, in the **HTTPS Tools** section, click **Additional Settings**.
e	Publish the SmartConsole session.

3. Add the designated physical interface in the object of the Virtual System.

   Procedure

Step	Description
a	In SmartConsole, open the Virtual System object.

Step	Description
b	From the navigation tree, click **Topology**.
c	From the top toolbar, click **New > Regular**.
d	On the **General** tab: i. In the **Interface** field, select the designated physical interface. ii. In the **IPv4 Configuration** section: ■ In the **IP Address** field, enter a dummy IP address. ■ In the **Net Mask** field, enter the applicable net mask.  **Important** - This IP address cannot collide with other IP addresses used in your environment. This IP address cannot belong to subnets used in your environment. Make sure to configure the correct subnet mask. After you enable traffic mirroring on this interface in SmartConsole, all other traffic that is routed to this interface is dropped. iii. Do not check the **Propagate route to adjacent Virtual Devices (IPv4)**. iv. In the **MTU** field, enter the applicable MTU. See *"Mirror and Decrypt Requirements" on page 114*. v. In the **Security Zone** field, leave the default **None**. vi. Click **OK**.

4. Activate the Mirror and Decrypt in the object of the Virtual System.

Procedure

Step	Description
a	From the left navigation panel, click **Gateways & Servers**.
b	Open the Virtual System object.
c	From the navigation tree of the gateway object, click the **[+]** near the **Other** and click **Mirror and Decrypt**.

Step	Description
d	Check **Mirror gateway traffic to interface**. The **Mirror and Decrypt - User Disclaimer** window opens. i. Read the text carefully. ii. Check **I agree to the terms and conditions**. iii. Click **OK** to accept and close the disclaimer.
e	In the **Mirror gateway traffic to interface** field, select the designated physical interface.
f	Click **OK** to save the changes and close the Virtual System properties window.

5. Configure the Mirror and Decrypt rules in the Access Control Policy for the traffic you wish to mirror and decrypt.

Procedure

Best Practice - We recommend you to configure a new separate Access Control Layer to contain Mirror and Decrypt rules. Alternatively, you can configure the Mirror and Decrypt rules in the regular Rule Base.

Important - When you configure the Mirror and Decrypt rules, these limitations apply:

- In the Mirror and Decrypt rules, you must *not* select Content criteria, such as Application, URL Filtering, Service matched by IP Protocol, Content Awareness.
- Above the Mirror and Decrypt rules, you must *not* configure other rules that contain Content criteria, such as Application, URL Filtering, Service matched by IP Protocol, Content Awareness.
- You must configure rules that contain an excluded source or an excluded destination above the Mirror and Decrypt rules.

  The **Name** column of these rules cannot contain these strings: **<M&D>**, **<M&d>**, **<m&D>**, or **<m&d>**.

The procedure below describes how to configure the Mirror and Decrypt rules in a separate Access Control Layer:

Step	Description
a	In SmartConsole, from the left navigation panel, click **Security Policies**.
b	Create a new Access Control Layer in the Access Control Policy.
c	In SmartConsole top left corner, click **Menu > Manage policies and layers**.

Step	Description
d	Select the existing policy and click **Edit** (the pencil icon). Alternatively, create a new policy.
e	From the navigation tree of the **Policy** window, click **General**.
f	In the **Policy Types** section, make sure you select **only** the **Access Control**.
g	In **Access Control** section, click on the **+** (plus) icon. A pop up window opens.
h	In the top right corner of this pop up window, click **New Layer**. The **Layer Editor** window opens.
i	From the navigation tree of the **Layer Editor** window, click **General**.
j	In the **Blades** section, make sure you select **only** the **Firewall**.
k	On other pages of the **Layer Editor** window, configure additional applicable settings. Click **OK**.
l	In the **Access Control** section, you see the **Network** Layer and the new **Access Control** Layer.
m	Click **OK** to save the changes and close the **Policy** window.
n	In SmartConsole, at the top, click the tab of the applicable policy.
o	In the **Access Control** section, click the new Access Control Layer. In the default rule, you must change the `Action` column from `Drop` to `Accept` to *not* affect the policy enforcement: ■ **Name** - Your text  **Important** - You *cannot* use these strings: **<M&D>**, **<M&d>**, **<m&D>**, or **<m&d>** ■ **Source** - ***Any** ■ **Destination** - ***Any** ■ **VPN** - ***Any** ■ **Services & Applications** - ***Any** ■ **Action** - Must contain **Accept** ■ **Track** - **None** ■ **Install On** - ***Policy Targets**

Step	Description
p	Above the existing Cleanup rule, add the applicable rules for the traffic you wish to Mirror and Decrypt. You must configure the Mirror and Decrypt rules as follows: ■ **Name** - Must contain one of these strings (the angle brackets **<>** are mandatory): • **<M&D>** • **<M&d>** • **<m&D>** • **<m&d>** ■ **Source** - Select the applicable objects ■ **Destination** - Select the applicable objects ■ **VPN** - Must leave the default `*Any` ■ **Services & Applications** - Select the applicable services (to decrypt the HTTPS traffic, select the applicable HTTP, HTTPS, or Proxy services) ■ **Action** - Must contain **Accept** ■ **Track** - Select the applicable option (**None**, **Log**, or **Alert**) ■ **Install On** - Must contain one of these objects: • ***Policy Targets** (this is the default) • The Security Gateway, or Cluster object, whose version is **R80.20** or higher
	 **Important:** ■ In the Mirror and Decrypt rules, you must *not* select Content criteria, such as Application, URL Filtering, Service matched by IP Protocol, Content Awareness. ■ Above the Mirror and Decrypt rules in this Ordered Layer, you must *not* configure other rules that contain Content criteria, such as Application, URL Filtering, Service matched by IP Protocol, Content Awareness. ■ You must configure rules that contain an excluded source or an excluded destination above the Mirror and Decrypt rules. The **Name** column of these rules cannot contain these strings: **<M&D>**, **<M&d>**, **<m&D>**, or **<m&d>**.
q	Publish the SmartConsole session.
r	Install the Access Control Policy.

Step	Description
s	If in a Mirror and Decrypt rule you set the **Track** to **Log**, then you can filter the logs for this rule by the **Access Rule Name**, which contains the configured string: **<M&D>**, **<M&d>**, **<m&D>**, or **<m&d>**.

Configuring Mirror and Decrypt in SmartConsole for Several Virtual Systems

Workflow for several Virtual Systems:

1. Enable the HTTPS Inspection in the objects of applicable Virtual Systems (for decrypting the HTTPS traffic).

   Procedure

Step	Description
a	Connect with SmartConsole to the Management Server.
b	From the left navigation panel, click **Gateways & Servers**.
c	Open the Virtual System object.
d	From the navigation tree, click **HTTPS Inspection**.
e	View and export the certificate.
f	Check **Enable HTTPS Inspection**.
g	Click **OK**.

2. Configure the HTTPS Inspection Rule Base (for decrypting the HTTPS traffic).

   Procedure

Step	Description
a	From the left navigation panel, click **Security Policies**.
b	From the left tree, click **HTTPS Inspection**.
d	Configure the HTTPS Inspection Rule Base. See *R80.40 Security Management Administration Guide*. For more settings, in the **HTTPS Tools** section, click **Additional Settings**.
e	Publish the SmartConsole session.

3. Define the designated physical interface as VLAN Trunk in the object of the VSX Gateway, or VSX Cluster.

Procedure

Note - If the Recorder or Packet-Broker connects to the VSX Gateway, or VSX Cluster members through a Switch, configure a VLAN Trunk on the applicable Switch port. The VLAN Trunk port on the Switch must accept all VLAN IDs that you configure in the applicable Virtual Systems.

Step	Description
1	In SmartConsole, open the object of the VSX Gateway, or VSX Cluster.
2	From the navigation tree, click **Physical Interfaces**.
3	Check the box **VLAN Trunk** near the designated physical interface.
4	Click **OK**.

4. Add the designated physical interface in the object of each applicable Virtual System.

Procedure

Step	Description
a	In SmartConsole, open the Virtual System object.
b	From the navigation tree, click **Topology**.
c	From the top toolbar, click **New > Regular**.

Step	Description
d	On the **General** tab: i. In the **Interface** field, select the designated physical interface. ii. In the **IPv4 Configuration** section: ■ In the **IP Address** field, enter a dummy IP address. ■ In the **Net Mask** field, enter the applicable net mask.  **Important** - This IP address cannot collide with other IP addresses used in your environment. This IP address cannot belong to subnets used in your environment. Make sure to configure the correct subnet mask. After you enable traffic mirroring on this interface in SmartConsole, all other traffic that is routed to this interface is dropped. iii. Do not check the **Propagate route to adjacent Virtual Devices (IPv4)**. iv. In the **MTU** field, enter the applicable MTU. See *"Mirror and Decrypt Requirements" on page 114*. v. In the **Security Zone** field, leave the default **None**. vi. Click **OK**.

5. Activate the Mirror and Decrypt in the object of each applicable Virtual System.

Procedure

Step	Description
a	From the left navigation panel, click **Gateways & Servers**.
b	Open the Virtual System object.
c	From the navigation tree of the gateway object, click the **[+]** near the **Other** and click **Mirror and Decrypt**.
d	Check **Mirror gateway traffic to interface**. The **Mirror and Decrypt - User Disclaimer** window opens. i. Read the text carefully. ii. Check **I agree to the terms and conditions**. iii. Click **OK** to accept and close the disclaimer.
e	In the **Mirror gateway traffic to interface** field, select the designated physical interface.

Step	Description
f	Click **OK** to save the changes and close the Virtual System properties window.

6. Configure the Mirror and Decrypt rules in the Access Control Policy for the traffic you wish to mirror and decrypt.

Procedure

Best Practice - We recommend you to configure a new separate Access Control Layer to contain Mirror and Decrypt rules. Alternatively, you can configure the Mirror and Decrypt rules in the regular Rule Base.

Important - When you configure the Mirror and Decrypt rules, these limitations apply:

- In the Mirror and Decrypt rules, you must *not* select Content criteria, such as Application, URL Filtering, Service matched by IP Protocol, Content Awareness.
- Above the Mirror and Decrypt rules, you must *not* configure other rules that contain Content criteria, such as Application, URL Filtering, Service matched by IP Protocol, Content Awareness.
- You must configure rules that contain an excluded source or an excluded destination above the Mirror and Decrypt rules.

  The **Name** column of these rules cannot contain these strings: **<M&D>**, **<M&d>**, **<m&D>**, or **<m&d>**.

The procedure below describes how to configure the Mirror and Decrypt rules in a separate Access Control Layer:

Step	Description
a	In SmartConsole, from the left navigation panel, click **Security Policies**.
b	Create a new Access Control Layer in the Access Control Policy.
c	In SmartConsole top left corner, click **Menu > Manage policies and layers**.
d	Select the existing policy and click **Edit** (the pencil icon). Alternatively, create a new policy.
e	From the navigation tree of the **Policy** window, click **General**.
f	In the **Policy Types** section, make sure you select **only** the **Access Control**.
g	In **Access Control** section, click on the **+** (plus) icon. A pop up window opens.
h	In the top right corner of this pop up window, click **New Layer**. The **Layer Editor** window opens.

Step	Description
i	From the navigation tree of the **Layer Editor** window, click **General**.
j	In the **Blades** section, make sure you select **only** the **Firewall**.
k	On other pages of the **Layer Editor** window, configure additional applicable settings. Click **OK**.
l	In the **Access Control** section, you see the **Network** Layer and the new **Access Control** Layer.
m	Click **OK** to save the changes and close the **Policy** window.
n	In SmartConsole, at the top, click the tab of the applicable policy.
o	In the **Access Control** section, click the new Access Control Layer. In the default rule, you must change the `Action` column from `Drop` to `Accept` to *not* affect the policy enforcement: ■ **Name** - Your text  **Important** - You *cannot* use these strings: **<M&D>**, **<M&d>**, **<m&D>**, or **<m&d>** ■ **Source** - ***Any** ■ **Destination** - ***Any** ■ **VPN** - ***Any** ■ **Services & Applications** - ***Any** ■ **Action** - Must contain **Accept** ■ **Track** - **None** ■ **Install On** - ***Policy Targets**

Step	Description
p	Above the existing Cleanup rule, add the applicable rules for the traffic you wish to Mirror and Decrypt. You must configure the Mirror and Decrypt rules as follows: ■ **Name** - Must contain one of these strings (the angle brackets **<>** are mandatory): • **<M&D>** • **<M&d>** • **<m&D>** • **<m&d>** ■ **Source** - Select the applicable objects ■ **Destination** - Select the applicable objects ■ **VPN** - Must leave the default `*Any` ■ **Services & Applications** - Select the applicable services (to decrypt the HTTPS traffic, select the applicable HTTP, HTTPS, or Proxy services) ■ **Action** - Must contain **Accept** ■ **Track** - Select the applicable option (**None**, **Log**, or **Alert**) ■ **Install On** - Must contain one of these objects: • ***Policy Targets** (this is the default) • The Security Gateway, or Cluster object, whose version is **R80.20** or higher
	 **Important:** ■ In the Mirror and Decrypt rules, you must *not* select Content criteria, such as Application, URL Filtering, Service matched by IP Protocol, Content Awareness. ■ Above the Mirror and Decrypt rules in this Ordered Layer, you must *not* configure other rules that contain Content criteria, such as Application, URL Filtering, Service matched by IP Protocol, Content Awareness. ■ You must configure rules that contain an excluded source or an excluded destination above the Mirror and Decrypt rules. The **Name** column of these rules cannot contain these strings: **<M&D>**, **<M&d>**, **<m&D>**, or **<m&d>**.
q	Publish the SmartConsole session.
r	Install the Access Control Policy.

Step	Description
s	If in a Mirror and Decrypt rule you set the **Track** to **Log**, then you can filter the logs for this rule by the **Access Rule Name**, which contains the configured string: **<M&D>**, **<M&d>**, **<m&D>**, or **<m&d>**.

Mirror and Decrypt Logs

To Mirror and Decrypt the traffic, you create special rules in the Access Control Policy.

The Mirror and Decrypt feature adds the applicable information to the regular Security Gateway logs.

To see the Mirror and Decrypt logs in SmartConsole:

Item	Description
1	Connect with SmartConsole to the Management Server.
2	From the left navigation panel, click **Logs & Monitor > Logs**.
3	In the search field, enter: `type:Control`
4	Double-click on the log and refer to the **More** section.

The Mirror and Decrypt logs show this information in the **More** section > **Mirror and Decrypt** field:

Action	Description
Mirror only	Security Gateway only mirrored the traffic.
Decrypt and mirror	Security Gateway decrypted and mirrored the HTTP / HTTPS traffic Note - This can be the case even for a clear-text HTTP connection, because the HTTPS Inspection inspects it first (example is all connections that use proxy 8080).
Partial mirroring (HTTPS inspection Bypass)	Security Gateway started to decrypt the traffic, but stopped later due to a Bypass rule (for example, a rule with a Category). Therefore, the mirrored connection is not complete.

ConnectControl - Server Load Balancing

ConnectControl is a Check Point solution for balancing the traffic that passes through Check Point Security Gateway or Cluster towards servers behind the Check Point Security Gateway or Cluster.

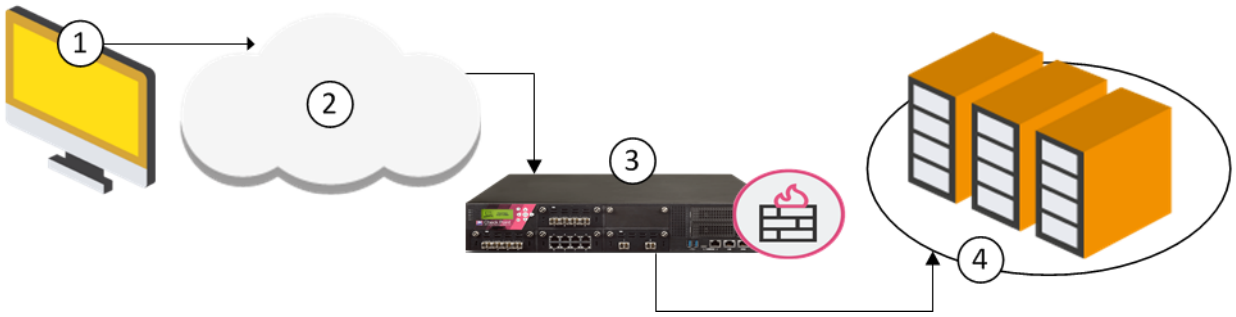
ConnectControl does not consume more memory or CPU processing power on Security Gateway or Cluster Members.

ConnectControl Packet Flow

Load-balanced servers are represented by one Virtual IP address.

In SmartConsole, you define a *Logical Server* object that represents a group of physical servers.

The Logical Server takes service requests for the load-balanced application and directs the requests to the applicable physical server.



When a client requests access to an application that is load balanced by ConnectControl, the request goes through the Security Gateway or Cluster.

Item	Description
1	**Client request** - A client starts a connection with the logical IP address of the application server (the address assigned to the Logical server).
2	**Internet** - The service request goes through the Internet.
3	**Security Gateway** - The service request arrives at the destination public IP address of the Logical Server, which is on the Security Gateway. The request is matched to the Logical Server rule in the Rule Base. The gateway directs the request to the internal IP address of the Logical Server group.
4	**Logical Server** - ConnectControl determines which server in the Logical Server group is best for the request, based on the selected load-balancing method.

Note - Make sure that rules that allow traffic for services to ConnectControl Logical Servers and that server groups are before Access Control Policy rules that allow traffic for those services.

Configuring ConnectControl

This procedure explains the steps to set up ConnectControl in your environment.

Procedure

1. In the SmartConsole, click **Objects** menu > **Object Explorer** (or press **Ctrl+E**).
2. Define a **Host** object for each of the servers that will be load-balanced.

   In the **Object Explorer**, from the toolbar, click **New** > **Host**.
3. Define a **Network Group** object to contain all **Host** objects for each of the servers that will be load-balanced.

   ### Instructions

   In the **Object Explorer**, from the toolbar, click **New** > **Network Group**.

   a. Name the group (for example, `HTTP_Server_Group`).

   b. Add the **Host** objects for each of the servers.

   Best Practice - We recommend adding no more than 29 objects.

4. Define the **Logical Server** object.

   ### Instructions

   a. In the **Object Explorer**, from the toolbar, click **New** > **Network Object** > **More** > **Logical Server**.

   b. In the **New Logical Server** window, enter a name for the ConnectControl Logical Server.

   c. Enter a Virtual IP address.

   Make sure the IP address is a public IP address.

   All traffic to be load-balanced, must be directed through the cluster.

   ### Note for a cluster environment

   If the assigned IP address is on the same subnet as a Cluster Virtual IP address, you also need to configure a Manual ARP proxy entry for this IP address.

   i. Click **Menu >Global properties > NAT - Network Address Translation**.

   ii. Select **Merge manual proxy ARP configuration**.

   iii. Click **OK**.

   iv. Configure the `$FWDIR/conf/local.arp` file as described in sk30197.

   v. Install the Access Control Policy on this cluster object.

d. Select the **Server type**.

Logical Server Types

When you create the Logical server object, configure the server type as **HTTP** or **Other**. This distinction is important. ConnectControl handles the connection to the client differently for each server type.

- The **HTTP** server type uses HTTP redirection.

  This type supports offsite HTTP servers and form-based applications, but only works with the HTTP protocol. An HTTP Logical server makes sure that all HTTP-connection sessions are directed to one server, which is a requirement for many Web applications. ConnectControl finds the correct physical server, behind the firewall or offsite, based on the selected load-balancing method. The session connections continue to go to that one server.

- The **Other** server type uses NAT (address translation) to send traffic to the grouped servers.

  This Logical server supports all protocols (including HTTP) and gives the most effectively balanced load. It requires servers to be NATed by the gateway. ConnectControl mediates each service request and then selects the server to get that request. It uses NAT to change the destination IP address of the incoming packet. If a return connection is opened, the connection is automatically established between the server and the client. The server's source address in the packet is translated to the IP address of the Logical server. On the packet's return, the firewall translates the packet's original address to the IP address of the Logical server.

e. Select the **Server group**.

Select the **Server Group** object that you defined earlier (or define a new **Server Group** object).

The members of the group must be hosts, Security Gateways, or OSE devices.

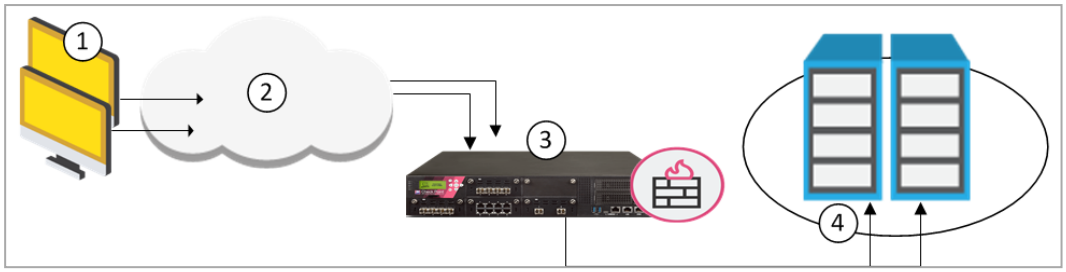
f. Select **Use persistent server mode** that fits your environment.

Persistency

This setting maintains a client's connection to the server that ConnectControl first selected.

- **Persistency by server** is useful for HTTP applications, such as forms, in a load-balanced environment with multiple Web servers. ConnectControl directs an HTTP client to one server for all requests. This allows clients to fill forms without the data loss that occurs if different servers take the requests.

- **Persistency by service** is useful if you are load balancing multiple services in your server group. For example, in a redundant environment of two servers, each running HTTP and FTP, ConnectControl directs traffic from one client to the server of the correct service. This prevents heavy load on one server, which can happen with **Persistency by server**.



Item	Description
1	Multiple client requests for HTTP and FTP.
2	Internet.
3	Security Gateway. The service requests arrive at the destination public IP address of the Logical Server, which is on the Security Gateway. The gateway directs the requests to the internal IP address of the Logical Server group.
4	Logical Server group with two servers, each with FTP and HTTP services. ConnectControl balances the load between the servers.

g. Select a **Balance method** that fits your environment.

Load Balancing Methods

ConnectControl distributes network traffic to load-balanced servers according to one of these predefined balancing methods:

Method	Description
Random	The Security Gateway directs service requests to servers at random. This method is a good choice when all the load-balanced servers have similar RAM and CPU and are located on the same segment.
Server load	The Security Gateway determines which server is best equipped to handle the new connection.
Round Robin	The Security Gateway directs service requests to the next server in the sequence. This method is a good choice when all the load balanced servers have similar RAM and CPU and are on the same segment.
Round Trip	Not supported.
Domain	Not supported.

h. Click **OK**.

5. Close the Object Explorer window.
6. From the left navigation panel, click **Security Policies** and click **Access Control**.
7. Add the Load Balancing rule to the Access Control Policy Rule Base:

Source	Destination	Services & Applications	Action
`*Any`	*Logical Server object*	*Load-balanced Services*	`Accept` or `User Auth` or `Client Auth`

8. For applications that use HTTP redirection, add a rule to allow the Network Group object (that contains load-balanced server objects) to communicate directly with the clients:

Source	Destination	Services & Applications	Action
`*Any`	*Network Group object*	`http`	`Accept`

9. Configure global settings for ConnectControl.

Instructions

a. At the top, click **Menu** > **Global properties**.

b. From the left tree, click **ConnectControl**.

c. Configure the settings that fit your environment:

- **Server Availability**

  This configures how ConnectControl finds available servers.

  - The **Server availability check interval** control the number of seconds between pings from the Security Gateway or Cluster to the load-balanced servers.
  - The **Server check retries** controls the number of attempts to contact a non-responsive server after ConnectControl stops directing connections to it.

- **Server Persistency**

  If you enabled **Persistency by server**, you can set a timeout for a client to use one server. If a server becomes unavailable, ConnectControl directs new connections to a new, available server. This bypasses the persistency and optimizes load balancing.

- **Server Load Balancing**

  Not supported.

d. Click **OK**.

10. Install the Access Control Policy on this Security Gateway or Cluster object.

Monitoring Software Blade

This Software Blade enables administrator to monitor these counters in real-time:

- System counters (CPU usage, Used Virtual Memory, Free Disk Space, and so on)
- Traffic connections
- Traffic throughput

To see System and Traffic counters in SmartConsole:

1. From the left navigation panel, click **Gateways & Servers**.
2. In the top pane, select the Security Gateway (or Cluster) object.
3. In the bottom pane, click the **Summary** tab and click the **Device & License Information** link at the bottom.
4. From the left tree, click **System Counters** and **Traffic**.
5. For a cluster object, from the top drop-down menu, select the Cluster Member.

To see User and VPN Tunnel counters in SmartView Monitor:

1. From the left navigation panel, click **Logs & Monitor > Logs**.
2. At the bottom, click the **Tunnel & User Monitoring** link.

For more information, see:

- *R80.40 Logging and Monitoring Administration Guide*
- *R77 SmartView Monitor Administration Guide*

Cloud Security

Check Point cloud security protects assets in the cloud from the most sophisticated threats with dynamic scalability, intelligent provisioning and consistent control across physical and virtual networks.

For more information, see:

- *R80.40 CloudGuard Controller Administration Guide*
- https://www.checkpoint.com/products/

Advanced Routing

Gaia OS supports:

- Dynamic Routing protocols - OSPF, BGP, and RIP.
- Dynamic Multicast Routing - PIM Sparse Mode (SM), PIM Dense Mode (DM), PIM Source-Specific Multicast (SSM), and IGMP.
- Different routing options.

You can configure these routing protocols and options in Gaia Portal and Gaia Clish.

For more information, see the *R80.40 Gaia Advanced Routing Administration Guide*.

SNMP

SNMP, as implemented on Check Point platforms, enables an SNMP manager to monitor the device using `GetRequest`, `GetNextRequest`, `GetBulkRequest`, and a select number of traps.

The Check Point implementation also supports using `SetRequest` to change these attributes: `sysContact`, `sysLocation`, and `sysName`. You must configure read-write permissions for set operations to work.

Check Point Gaia supports SNMP v1, v2, and v3.

For more information, see the *R80.40 Gaia Administration Guide* > Chapter *System Management* > Section *SNMP*.

Deploying a Single Security Gateway in Monitor Mode

Introduction to Monitor Mode

You can configure Monitor Mode on a single Check Point Security Gateway's interface.

This lets the Check Point Security Gateway listen to traffic from a Mirror Port or Span Port on a connected switch.

Use the Monitor Mode to analyze network traffic without changing the production environment.

The mirror port on a switch duplicates the network traffic and sends it to the Security Gateway with an interface configured in Monitor Mode to record the activity logs.

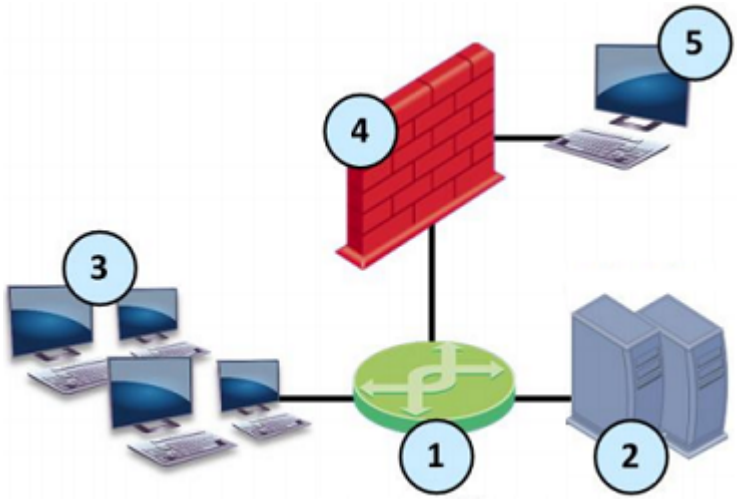
You can use the Monitor Mode:

- To monitor the use of applications as a permanent part of your deployment
- To evaluate the capabilities of the Software Blades:
  - The Security Gateway neither enforces any security policy, nor performs any active operations (prevent / drop / reject) on the interface in the Monitor Mode.
  - The Security Gateway terminates and does not forward all packets that arrive at the interface in the Monitor Mode.
  - The Security Gateway does not send any traffic through the interface in the Monitor Mode.

Benefits of the Monitor Mode include:

- There is no risk to your production environment.
- It requires minimal set-up configuration.
- It does not require TAP equipment, which is expensive.

Example Topology for Monitor Mode



Item	Description
1	Switch with a mirror or SPAN port that duplicates all incoming and outgoing packets. The Security Gateway connects to a mirror or SPAN port on the switch.
2	Servers.
3	Clients.
4	Security Gateway with an interface in Monitor Mode.
5	Security Management Server that manages the Security Gateway.

For More About Monitor Mode

See the *R80.40 Installation and Upgrade Guide* > Chapter *Special Scenarios for Security Gateways* > Section *Deploying a Security Gateway in Monitor Mode*.

Deploying a Single Security Gateway or ClusterXL in Bridge Mode

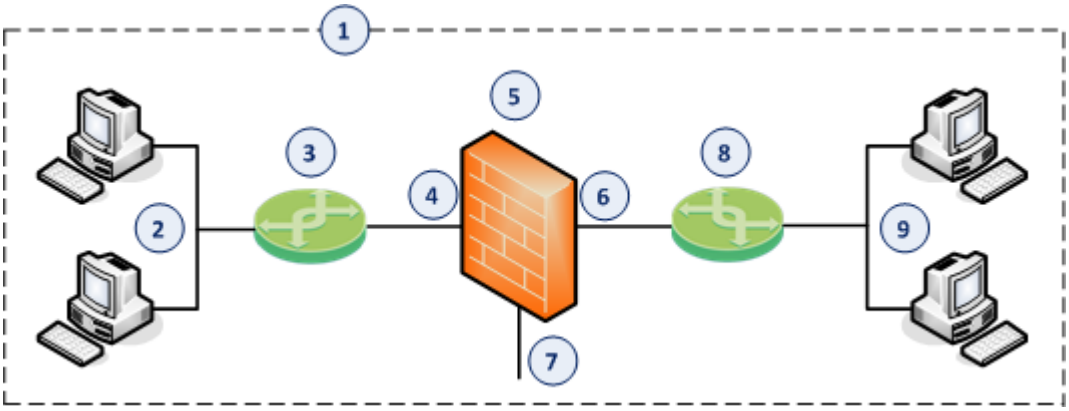
Introduction to Bridge Mode

If you cannot divide the existing network into several networks with different IP addresses, you can install a Check Point Security Gateway (or a ClusterXL) in the Bridge Mode.

A Security Gateway (or ClusterXL) in Bridge Mode is invisible to Layer 3 traffic.

When traffic arrives at one of the bridge slave interfaces, the Security Gateway (or Cluster Members) inspects it and passes it to the second bridge slave interface.

Example Topology for a single Security Gateway in Bridge Mode



Item	Description
1	Network, which an administrator needs to divide into two Layer 2 segments. The Security Gateway in Bridge Mode connects between these segments.
2	First network segment.
3	Switch that connects the first network segment to one bridged slave interface **(4)** on the Security Gateway in Bridge Mode.
4	One bridged slave interface (for example, `eth1`) on the Security Gateway in Bridge Mode.
5	Security Gateway in Bridge Mode.
6	Another bridged slave interface (for example, `eth2`) on the Security Gateway in Bridge Mode.
7	Dedicated Gaia Management Interface (for example, `eth0`) on the Security Gateway.
8	Switch that connects the second network segment to the other bridged slave interface **(6)** on the Security Gateway in Bridge Mode.
9	Second network segment.

For More About Bridge Mode

See the *R80.40 Installation and Upgrade Guide* > Chapter *Special Scenarios for Security Gateways* > Section *Deploying a Security Gateway or a ClusterXL in Bridge Mode*.

Security Before Firewall Activation

To protect the Security Gateway and network, Check Point Security Gateway has baseline security:

Baseline Security	Name of Policy	Description
Boot Security	`defaultfilter`	Security during boot process.
Initial Policy	`InitialPolicy`	Security before a policy is installed for the first time, or when Security Gateway failed to load the policy.

Important - If you disable the boot security or unload the currently installed policy, you leave your Security Gateway, or a Cluster Member without protection.

Best Practice - Before you disable the boot security, we recommend to disconnect your Security Gateway, or a Cluster Member from the network completely.

For additional information, see these commands:

Command	Description
`$CPDIR/bin/cpstat -f policy fw`	Shows the currently installed policy. See *"cpstat" on page 208*.
`$FWDIR/bin/control_bootsec {-r | -R}`	Disables the boot security. See *"control_bootsec" on page 173*.
`$FWDIR/bin/control_bootsec [-g | -G]`	Enables the boot security. See *"control_bootsec" on page 173*.
`$FWDIR/bin/comp_init_policy [-u | -U]`	Deletes the local state policy. See *"comp_init_policy" on page 170*.
`$FWDIR/bin/comp_init_policy [-g | -G]`	Creates the local state Initial Policy. See *"comp_init_policy" on page 170*.
`$FWDIR/bin/fw unloadlocal`	Unloads the currently installed policy. See *"fw unloadlocal" on page 391*.

Boot Security

The Boot Security protects the Security Gateway and its networks, during the boot:

- Disables the IP Forwarding in Linux OS kernel
- Loads the Default Filter Policy

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

The Default Filter Policy

The Default Filter Policy (`defaultfilter`) protects the Security Gateway from the time it boots up until it installs the user-defined Security Policy.

Boot Security disables IP Forwarding and loads the Default Filter Policy.

There are three Default Filters templates on the Security Gateway:

Default Filter Mode	Default Filter Policy File	Description
Boot Filter	`$FWDIR/lib/defaultfilter.boot`	This filter: ■ Drops all incoming packets that have the same source IP addresses as the IP addresses assigned to the Security Gateway interfaces ■ Allows all outbound packets from the Security Gateway
Drop Filter	`$FWDIR/lib/defaultfilter.drop`	This filter drops all inbound *and* outbound packets on the Security Gateway.  **Best Practice** - If the boot process requires that the Security Gateway communicate with other hosts, do *not* use the *Drop Filter*.

Default Filter Mode	Default Filter Policy File	Description
Filter for Dynamically Assigned Gateways (DAG)	`$FWDIR/lib/defaultfilter.dag`	This filter for Security Gateways with Dynamically Assigned IP address: ■ Allows all DHCP Requests ■ Allows all DHCP Replies ■ Uses Boot Filter: a. Drops all incoming packets that have the same source IP addresses as the IP addresses assigned to the Security Gateway interfaces b. Allows all outbound packets from the Security Gateway

Selecting the Default Filter Policy

Step	Description
1	Make sure to configure and install a Security Policy on the Security Gateway.
2	Connect to the command line on the Security Gateway.
3	Log in to the Expert mode.
4	Back up the current Default Filter Policy file: `cp -v $FWDIR/conf/defaultfilter.pf{,_BKP}`

Step	Description

5	Create a new Default Filter Policy file.
	■ To create a new Boot Filter, run:
	`cp -v $FWDIR/lib/defaultfilter.boot $FWDIR/conf/defaultfilter.pf`
	■ To create a new Drop Filter, run:
	`cp -v $FWDIR/lib/defaultfilter.drop $FWDIR/conf/defaultfilter.pf`
	■ To create a new DAG Filter, run:
	`cp -v $FWDIR/lib/defaultfilter.dag $FWDIR/conf/defaultfilter.pf`

6	Compile the new Default Filter file:
	`fw defaultgen`
	■ The new complied Default Filter file for IPv4 traffic is:
	`$FWDIR/state/default.bin`
	■ The new complied Default Filter file for IPv6 traffic is:
	`$FWDIR/state/default.bin6`

7	Get the path of the Default Filter Policy file:
	`$FWDIR/boot/fwboot bootconf get_def`
	Example:
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot bootconf get_def`
	`/etc/fw.boot/default.bin`
	`[Expert@MyGW:0]#`

8	Copy new complied Default Filter file to the path of the Default Filter Policy file.
	■ For IPv4 traffic, run:
	`cp -v $FWDIR/state/default.bin /etc/fw.boot/default.bin`
	■ For IPv6 traffic, run:
	`cp -v $FWDIR/state/default.bin6 /etc/fw.boot/default.bin6`

Step	Description
9	Make sure to connect to the Security Gateway over a serial console.  **Important** - If the new Default Filter Policy fails and blocks all access through the network interfaces, you can unload that Default Filter Policy and install the working policy.
10	Reboot the Security Gateway.

Defining a Custom Default Filter

Administrators with Check Point INSPECT language knowledge can define customized Default Filters.

 **Important** - Make sure your customized Default Filter policy does not interfere with the Security Gateway boot process.

Step	Description
1	Make sure to configure and install a Security Policy on the Security Gateway.
2	Connect to the command line on the Security Gateway.
3	Log in to the Expert mode.
4	Back up the current Default Filter Policy file: `cp -v $FWDIR/conf/defaultfilter.pf{,_BKP}`

5	Create a new Default Filter Policy file.
	
	■ To use the Boot Filter as a template, run:
	
	`cp -v $FWDIR/lib/defaultfilter.boot $FWDIR/conf/defaultfilter.pf`
	
	■ To use the Drop Filter as a template, run:
	
	`cp -v $FWDIR/lib/defaultfilter.drop $FWDIR/conf/defaultfilter.pf`
	
	■ To use the DAG Filter as a template, run:
	
	`cp -v $FWDIR/lib/defaultfilter.dag $FWDIR/conf/defaultfilter.pf`

Step	Description
6	Edit the new Default Filter Policy file to include the applicable INSPECT code.  **Important** - Your customized Default Filter must not use these functions: ■ `Logging` ■ `Authentication` ■ `Encryption` ■ `Content Security`

7	Compile the new Default Filter file:
	`fw defaultgen`
	■ The new complied Default Filter file for IPv4 traffic is:
	`$FWDIR/state/default.bin`
	■ The new complied Default Filter file for IPv6 traffic is:
	`$FWDIR/state/default.bin6`

8	Get the path of the Default Filter Policy file:
	`$FWDIR/boot/fwboot bootconf get_def`
	Example:
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot bootconf get_def`
	`/etc/fw.boot/default.bin`
	`[Expert@MyGW:0]#`

9	Copy new complied Default Filter file to the path of the Default Filter Policy file.
	■ For IPv4 traffic, run:
	`cp -v $FWDIR/state/default.bin /etc/fw.boot/default.bin`
	■ For IPv6 traffic, run:
	`cp -v $FWDIR/state/default.bin6 /etc/fw.boot/default.bin6`

Step	Description
10	Make sure to connect to the Security Gateway over a serial console.  **Important** - If the new Default Filter Policy fails and blocks all access through the network interfaces, you can unload that Default Filter Policy and install the working policy.
11	Reboot the Security Gateway.

Using the Default Filter Policy for Maintenance

It is sometimes necessary to stop the Security Gateway for maintenance. It is not always practical to disconnect the Security Gateway from the network (for example, if the Security Gateway is on a remote site).

To stop the Security Gateway for maintenance and maintain security, you can run:

Command	Description
`cpstop -fwflag -default`	■ Shuts down Check Point processes ■ Loads the Default Filter policy (`defaultfilter`)
`cpstop -fwflag -proc`	■ Shuts down Check Point processes ■ Keeps the currently loaded kernel policy ■ Maintains the Connections table, so that after you run the `cpstart` command, you do not experience dropped packets because they are "out of state"  **Note** - Only security rules that do not use user space processes continue to work.

The Initial Policy

Until the Security Gateway administrator installs the Security Policy on the Security Gateway for the first time, security is enforced by an Initial Policy.

The Initial Policy operates by adding the predefined implied rules to the Default Filter policy.

These implied rules forbid most communication, yet allow the communication needed for the installation of the Security Policy.

The Initial Policy also protects the Security Gateway during Check Point product upgrades, when a SIC certificate is reset on the Security Gateway, or in the case of a Check Point product license expiration.

Note - During a Check Point upgrade, a SIC certificate reset, or license expiration, the Initial Policy overwrites the user-defined policy.

The sequence of actions during boot of the Security Gateway until a Security Policy is loaded for the first time:

Step	Description
1	The Security Gateway boots up.
2	The Security Gateway disables IP Forwarding and loads the Default Filter policy.
3	The Security Gateway configures the interfaces.
4	The Security Gateway services start.
5	The Security Gateway fetches the Initial Policy from the local directory.
6	Administrator installs the user-defined Security Policy from the Management Server.

The Security Gateway enforces the Initial Policy until administrator installs a user-defined policy.

In subsequent boots, the Security Gateway loads the user-defined policy immediately after the Default Filter policy.

There are different Initial Policies for Standalone and distributed setups:

- In a Standalone configuration, where the Security Management Server and the Security Gateway are on the same computer, the Initial Policy allows CPMI management communication only.

  This permits SmartConsole clients to connect to the Security Management Server.

- In a distributed configuration, where the Security Management Server is on one computer and the Security Gateway is on a different computer, the Initial Policy:

  - Allows the **cpd** and **fwd** daemons to communicate for SIC (to establish trust) and for Policy installation.
  - Does not allow CPMI connections through the Security Gateway.

    The SmartConsole is not be able to connect to the Security Management Server, if the SmartConsole must access the Security Management Server through a Security Gateway with the Initial Policy.

Troubleshooting: Cannot Complete Reboot

In some configurations, the Default Filter policy prevents the Security Gateway from completing the reboot after installation.

Firstly, look at the Default Filter. Does the Default Filter allow traffic required by the boot procedures?

Secondly, if the boot process cannot finish successfully, remove the Default Filter:

Step	Description
1	Connect to the Security Gateway over serial console.
2	Reboot the Security Gateway.
3	During boot, press any key to enter the Boot Menu.
4	Select the **Start in maintenance mode**.
5	Enter the Expert mode password.

6	Set the Default Filter to not load again:
	a. Go to the `$FWDIR` directory:
	`cd /opt/CPsuite-<VERSION>/fw1/`
	b. Set the Default Filter to not load again:
	`./fwboot bootconf set_def`

7	In the `$FWDIR/boot/boot.conf` file, examine the value of the "`DEFAULT_FILTER_PATH`":
	a. Go to the `$FWDIR` directory:
	`cd /opt/CPsuite-<VERSION>/fw1/`
	b. examine the value of the "`DEFAULT_FILTER_PATH`":
	`grep DEFAULT_FILTER_PATH boot/boot.conf`

8	Reboot the Security Gateway.

Command Line Reference

See the *R80.40 CLI Reference Guide*.

Below is a limited list of applicable commands.

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:

Character	Description

TAB	Shows the available nested subcommands:
	`main command`
	`→ nested subcommand 1`
	`→ → nested subsubcommand 1-1`
	`→ → nested subsubcommand 1-2`
	`→ nested subcommand 2`
	Example:
	`cpwd_admin`
	`config`
	`-a <options>`
	`-d <options>`
	`-p`
	`-r`
	`del <options>`
	Meaning, you can run only **one** of these commands:
	■ This command: `cpwd_admin config -a <options>`
	■ Or this command: `cpwd_admin config -d <options>`
	■ Or this command: `cpwd_admin config -p`
	■ Or this command: `cpwd_admin config -r`
	■ Or this command: `cpwd_admin del <options>`

Curly brackets or braces **{ }**	Enclose a list of available commands or parameters, separated by the vertical bar **|**. User can enter only one of the available commands or parameters.

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.

comp_init_policy

Description

Generates, loads, or removes the Initial Policy on a Security Gateway, or a Cluster Member.

Until the Security Gateway or cluster administrator installs the user-defined Security Policy on the Security Gateway or Cluster Members for the first time, security is enforced by an Initial Policy.

The Initial Policy operates by adding "implied rules" to the Default Filter.

These rules forbid most of the communication, but allow the communication needed for the installation of the Security Policy.

The Initial Policy also protects a Security Gateway or Cluster Members in these cases:

- During Check Point product upgrades
- When a SIC certificate is reset on the Security Gateway or Cluster Member
- When Check Point product license expires

The Initial Policy is enforced until a policy is installed, and is never loaded again. In subsequent boots, the regular policy is loaded immediately after the Default Filter.

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

Notes:

- You must run this command from the Expert mode.
- The Initial Policy overwrites the user-defined policy.
- Output of the "`cpstat -f policy fw`" command (see *"cpstat" on page 208*) shows the name of this policy as "`InitialPolicy`".
- Security Gateway, or Cluster Member stores the installed Access Control Policy in these directories:
  - `$FWDIR/state/__tmp/FW1/`
  - `$FWDIR/state/local/FW1/`
  - `$FWDIR/state/<Name of Cluster Object>/FW1/`
- Refer to these related commands:
  - *"control_bootsec" on page 173*
  - *"fwboot bootconf" on page 402*
  - *"fw defaultgen" on page 288*
  - *"fwboot default" on page 415*

Syntax

```
[Expert@HostName:0]# $FWDIR/bin/comp_init_policy [-u | -U]
[Expert@HostName:0]# $FWDIR/bin/comp_init_policy [-g | -G]
```

Parameters

Parameter	Description
No Parameters	The command runs with the last used parameter.
`-u` `-U`	Performs these steps: 1. Removes an attribute **:InitialPolicySafe (true)** from the "**: (FW1**" section the Check Point Registry file (`$CPDIR/registry/HKLM_registry.data`). 2. Removes the policy files from the `$FWDIR/state/local/FW1/` directory.

`-g`
`-G`
	Performs these steps:
	1. Removes an attribute **:InitialPolicySafe (true)** from the "**: (FW1**" section in the Check Point Registry file (`$CPDIR/registry/HKLM_registry.data`).
	2. Generates the Initial Policy in the `$FWDIR/state/local/FW1/` directory.
	You can use this parameter, if there is no Initial Policy generated.
	If Initial Policy was already generated, make sure that after removing the Initial Policy, you delete the `$FWDIR/state/local/FW1/` directory on the Security Gateway, or Cluster Member.
	This parameter generates the Initial Policy and ensures that Security Gateway loads it the next time it fetches a policy (at "`cpstart`", at next boot, or with the "`fw fetch localhost`" command).
	The "`comp_init_policy -g`" command only works, if currently there is no policy installed on the Security Gateway, or Cluster Member.
	If you run one of these pairs of the commands, the original policy is still loaded:
	■ `comp_init_policy -g`
	`fw fetch localhost`
	■ `comp_init_policy -g`
	`cpstart`
	■ `comp_init_policy -g`
	`reboot`

Example

```
[Expert@GW:0]# cd $FWDIR/state/local/FW1/
[Expert@GW:0]#

[Expert@GW:0]# pwd
/opt/CPsuite-R80.40/fw1/state/local/FW1
[Expert@GW:0]#

[Expert@GW:0]# ls -l
total 7744
-rw-r--r-- 1 admin root   20166 Jun 13 16:34 install_policy_report.txt
-rw-r--r-- 1 admin root      55 Jun 13 16:34 install_policy_report_timing.txt
-rw-r--r-- 1 admin root   37355 Jun 13 16:34 local.Sandbox-persistence.xml
... output was cut for brevity ...
-rw-r--r-- 1 admin root    2278 Jun 13 16:34 local.vsx_cluster_netobj
-rw-r--r-- 1 admin root    5172 Jun 13 16:34 local.{939922F7-DF98-4988-B776-B70B9B8340F3}
-rw-r--r-- 1 admin root   10328 Jun 13 16:34 local.{B9D14722-3936-4B33-814B-F87EA4062BEB}
-rw-r----- 1 admin root   14743 Jun 13 16:34 manifest.C
-rw-r--r-- 1 admin root    7381 Jun 13 16:34 policy.info
-rw-r--r-- 1 admin root    2736 Jun 13 16:34 policy.map
-rw-r--r-- 1 admin root      51 Jun 13 16:34 sig.map
[Expert@GW:0]#

[Expert@GW:0]# comp_init_policy -u
erasing local state..
[Expert@GW:0]#

[Expert@GW:0]# ls -l
total 0
[Expert@GW:0]#

[Expert@GW:0]# comp_init_policy -g
initial_module:
Compiled OK.
initial_module:
Compiled OK.
[Expert@GW:0]#

[Expert@GW:0]# ls -l
total 56
-rw-rw---- 1 admin root    8 Jul 19 19:51 local.ctlver
-rw-rw---- 1 admin root 4514 Jul 19 19:51 local.fc
-rw-rw---- 1 admin root 4721 Jul 19 19:51 local.fc6
-rw-rw---- 1 admin root  235 Jul 19 19:51 local.ft
-rw-rw---- 1 admin root  317 Jul 19 19:51 local.ft6
-rw-rw---- 1 admin root  135 Jul 19 19:51 local.fwrl.conf
-rw-rw---- 1 admin root   14 Jul 19 19:51 local.ifs
-rw-rw---- 1 admin root  833 Jul 19 19:51 local.inspect.lf
-rw-rw---- 1 admin root  243 Jul 19 19:51 local.lg
-rw-rw---- 1 admin root  243 Jul 19 19:51 local.lg6
-rw-rw---- 1 admin root    0 Jul 19 19:51 local.magic
-rw-rw---- 1 admin root    3 Jul 19 19:51 local.set
-rw-rw---- 1 admin root   51 Jul 19 19:51 sig.map
[Expert@GW:0]#
```

control_bootsec

Description

Controls the boot security - loading of both the Default Filter policy (`defaultfilter`) and the Initial Policy (`InitialPolicy`) during boot on a Security Gateway, or a Cluster Member.

Warning - If you disable the boot security, you leave your Security Gateway, or a Cluster Member without any protection during the boot. Before you disable the boot security, we recommend to disconnect your Security Gateway, or a Cluster Member from the network completely.

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

Notes:

- You must run this command from the Expert mode.
- The changes made with this command survive reboot.
- Refer to these related commands:
  - *"comp_init_policy" on page 170*
  - *"fwboot bootconf" on page 402*
  - *"fw defaultgen" on page 288*
  - *"fwboot default" on page 415*

Syntax

```
[Expert@GW:0]# $FWDIR/bin/control_bootsec [-g | -G]
[Expert@GW:0]# $FWDIR/bin/control_bootsec {-r | -R}
```

Parameters

Parameter	Description

No Parameter
`-g`
`-G`
	Enables the boot security:
	1. Executes the "`$FWDIR/boot/fwboot bootconf set_def $FWDIR/boot/default.bin`" command that updates the path to the Default Filter policy in the `$FWDIR/boot/boot.conf` file to point to the correct policy file (`DEFAULT_FILTER_PATH /etc/fw.boot/default.bin`).
	2. Executes the "`$FWDIR/bin/comp_init_policy -g`" command that:
	a. Removes the attribute "**:InitialPolicySafe (true)**" from the section "**: (FW1**" in the Check Point Registry (the `$CPDIR/registry/HKLM_registry.data` file).
	b. Generates the Initial Policy files in the `$FWDIR/state/local/FW1/` directory.

`-r` `-R`	Disables the boot security: 1. Executes the "`$FWDIR/boot/fwboot bootconf set_def`" command that updates the path to the Default Filter policy in the `$FWDIR/boot/boot.conf` file to point nowhere (`DEFAULT_FILTER_PATH 0`). 2. Executes the "`$FWDIR/bin/comp_init_policy -u`" command that: a. Adds the attribute "**:InitialPolicySafe (true)**" to the section "**: (FW1**" in the Check Point Registry (the `$CPDIR/registry/HKLM_registry.data` file). b. Deletes all files in the `$FWDIR/state/local/FW1/` directory.

Example 1 - Disabling the boot security

```
[Expert@GW:0]# cd $FWDIR/state/local/FW1/
[Expert@GW:0]#

[Expert@GW:0]# pwd
/opt/CPsuite-R80.40/fw1/state/local/FW1
[Expert@GW:0]#

[Expert@GW:0]# ls -l
total 7736
-rw-rw---- 1 admin root   11085 Jul 19 20:16 install_policy_report.txt
-rw-rw---- 1 admin root      56 Jul 19 20:16 install_policy_report_timing.txt
-rw-rw---- 1 admin root   37355 Jul 19 20:16 local.Sandbox-persistence.xml
-rw-rw---- 1 admin root       3 Jul 19 20:16 local.ad_query_profiles
... ... ...
-rw-r----- 1 admin root   14743 Jul 19 20:16 manifest.C
-rw-rw---- 1 admin root    7381 Jul 19 20:16 policy.info
-rw-rw---- 1 admin root    2736 Jul 19 20:16 policy.map
-rw-rw---- 1 admin root      51 Jul 19 20:16 sig.map
[Expert@GW:0]#

[Expert@GW:0]# $FWDIR/bin/control_bootsec -r
Disabling boot security
FW-1 will not load a default filter on boot
[Expert@GW:0]#

[Expert@GW:0]# cat $FWDIR/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@GW:0]#

[Expert@GW:0]# grep InitialPolicySafe $CPDIR/registry/HKLM_registry.data
                                      :InitialPolicySafe (true)
[Expert@GW:0]#

[Expert@GW:0]# ls -l
total 0
[Expert@GW:0]#
```

Example 2 - Enabling the boot security

```
[Expert@GW:0]# cd $FWDIR/state/local/FW1/
[Expert@GW:0]#

[Expert@GW:0]# pwd
/opt/CPsuite-R80.40/fw1/state/local/FW1
[Expert@GW:0]#

[Expert@GW:0]# control_bootsec -g
Enabling boot security
[Expert@GW:0]#

[Expert@GW:0]# cat $FWDIR/boot/boot.conf
CTL_IPFORWARDING        1
DEFAULT_FILTER_PATH     /opt/CPsuite-R80.40/fw1/boot/default.bin
KERN_INSTANCE_NUM       3
COREXL_INSTALLED        1
KERN6_INSTANCE_NUM      2
IPV6_INSTALLED  0
CORE_OVERRIDE   4
[Expert@GW:0]#

[Expert@GW:0]# grep InitialPolicySafe $CPDIR/registry/HKLM_registry.data
[Expert@GW:0]#

[Expert@GW:0]# ls -l
total 56
-rw-rw---- 1 admin root    8 Jul 19 20:22 local.ctlver
-rw-rw---- 1 admin root 4514 Jul 19 20:22 local.fc
-rw-rw---- 1 admin root 4721 Jul 19 20:22 local.fc6
-rw-rw---- 1 admin root  235 Jul 19 20:22 local.ft
-rw-rw---- 1 admin root  317 Jul 19 20:22 local.ft6
-rw-rw---- 1 admin root  135 Jul 19 20:22 local.fwrl.conf
-rw-rw---- 1 admin root   14 Jul 19 20:22 local.ifs
-rw-rw---- 1 admin root  833 Jul 19 20:22 local.inspect.lf
-rw-rw---- 1 admin root  243 Jul 19 20:22 local.lg
-rw-rw---- 1 admin root  243 Jul 19 20:22 local.lg6
-rw-rw---- 1 admin root    0 Jul 19 20:22 local.magic
-rw-rw---- 1 admin root    3 Jul 19 20:22 local.set
-rw-rw---- 1 admin root   51 Jul 19 20:22 sig.map
[Expert@GW:0]#
```

cp_conf

Description

Configures or reconfigures a Check Point product installation.

Note - The available options for each Check Point computer depend on the configuration and installed products.

Syntax on a Security Gateway

```
cp_conf
      -h
      adv_routing <options>
      auto <options>
      corexl <options>
      fullha <options>
      ha <options>
      intfs <options>
      lic <options>
      sic <options>
      snmp <options>
```

Parameters

Parameter	Description
`-h`	Shows the entire built-in usage.
`adv_routing <options>`	Enables or disables the Advanced Routing feature on this Security Gateway. **Important** - Do *not* use these outdated commands. To configure Advanced Routing, see the *R80.40 Gaia Advanced Routing Administration Guide*.
`auto <options>`	Shows and configures the automatic start of Check Point products during boot. See *"cp_conf auto" on page 179*.
`corexl <options>`	Enables or disables CoreXL on this Security Gateway. See *"cp_conf corexl" on page 180*.

Parameter	Description
`fullha <options>`	Manages Full High Availability Cluster. See *"cp_conf fullha" on page 182*.
`ha <options>`	Enables or disables cluster membership on this Security Gateway. See *"cp_conf ha" on page 183*.
`intfs <options>`	Sets the topology of interfaces on a Security Gateway, which you manage with SmartProvisioning. See *"cp_conf intfs" on page 184*.
`lic <options>`	Manages Check Point licenses. See *"cp_conf lic" on page 185*.
`sic <options>`	Manages SIC on this Security Gateway. See *"cp_conf sic" on page 187*.
`snmp <options>`	Do *not* use these outdated commands. To configure SNMP, see the *R80.40 Gaia Administration Guide* - Chapter *System Management* - Section *SNMP*.

cp_conf auto

Description

Shows and controls which of Check Point products start automatically during boot.

Note - This command corresponds to the option **Automatic start of Check Point Products** in the *"cpconfig" on page 189* menu.

Syntax

```
cp_conf auto
      -h
      {enable | disable} <Product1> <Product2> ...
      get all
```

Parameters

Parameter	Description
`-h`	Shows the applicable built-in usage.
`{enable | disable} <Product1> <Product2> ...`	Controls whether the installed Check Point products start automatically during boot. This command is for Check Point use only.
`get all`	Shows which of these Check Point products start automatically during boot: ■ Check Point Security Gateway ■ QoS (former FloodGate-1) ■ SmartEvent Suite

Example from a Security Gateway

```
[Expert@GW:0] cp_conf auto get all
The Check Point Security Gateway will start automatically at boot time.

QoS will start automatically at boot time.

SmartEvent Suite is not installed.

[Expert@GW:0]#
```

cp_conf corexl

Description

Enables or disables CoreXL.

For more information, see the *R80.40 Performance Tuning Administration Guide*.

Important:

- This command is for Check Point use only.

  To configure CoreXL, use the **Check Point CoreXL** option in the *cpconfig* 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 406*.

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]#
```

cp_conf fullha

Description

Manages the state of the Full High Availability Cluster:

- Enables the Full High Availability Cluster
- Disables the Full High Availability Cluster
- Deletes the Full High Availability peer
- Shows the Full High Availability state

Important - To configure a Full High Availability cluster, follow the *R80.40 Installation and Upgrade Guide*.

Syntax

```
cp_conf fullha
      enable
      del_peer
      disable
      state
```

Parameters

Parameter	Description
`enable`	Enables the Full High Availability on this computer.
`del_peer`	Deletes the Full High Availability peer from the configuration.
`disable`	Disables the Full High Availability on this computer.
`state`	Shows the Full High Availability state on this computer.

Example

```
[Expert@Cluster_Member:0]# cp_conf fullha state
FullHA is currently enabled
[Expert@Cluster_Member:0]#
```

cp_conf ha

Description

Enables or disables cluster membership on this Security Gateway.

Important - This command is for Check Point use only. To configure cluster membership, you must use the *"cpconfig" on page 189* command.

For more information, see the *R80.40 ClusterXL Administration Guide*.

Syntax

```
cp_conf ha {enable | disable} [norestart]
```

Parameters

Parameter	Description
`enable`	Enables cluster membership on this Security Gateway. This command is equivalent to the option **Enable cluster membership for this gateway** in the *"cpconfig" on page 189* menu.
`disable`	Disables cluster membership on this Security Gateway. This command is equivalent to the option **Disable cluster membership for this gateway** in the *"cpconfig" on page 189* menu.
`norestart`	**Optional:** Specifies to apply the configuration change without the restart of Check Point services. The new configuration takes effect only after reboot.

Example 1 - Enable the cluster membership without restart of Check Point services

```
[Expert@MyGW:0]# cp_conf ha enable norestart

Cluster membership for this gateway was enabled successfully
Important: This change will take effect after reboot.

[Expert@MyGW:0]#
```

Example 2 - Disable the cluster membership without restart of Check Point services

```
[Expert@MyGW:0]# cp_conf ha disable norestart
cpwd_admin:
Process CPHAMCSET process has been already terminated

Cluster membership for this gateway was disabled successfully
Important: This change will take effect after reboot.

[Expert@MyGW:0]#
```

cp_conf intfs

Description

Sets the topology of interfaces on a Security Gateway, which you manage with SmartProvisioning.

For more information, see the *R80.40 SmartProvisioning Administration Guide*.

Syntax

```
cp_conf intfs
      get
      set
            auxiliary <Name of Interface>
            DMZ <Name of Interface>
            external <Name of Interface>
            internal <Name of Interface>
```

Parameters

Parameter	Description
`get`	Shows the list of configured interfaces.
`set`	Configures the topology of the specified interface: ■ `auxiliary` ■ `DMZ` ■ `external` ■ `internal`

cp_conf lic

Description

Shows, adds and deletes Check Point licenses.

Note - This command corresponds to the option **Licenses and contracts** in the *"cpconfig" on page 189* menu.

Syntax

```
cp_conf lic
      -h
      add -f <Full Path to License File>
      add -m <Host> <Date> <Signature Key> <SKU/Features>
      del <Signature Key>
      get [-x]
```

Parameters

Parameter	Description
`-h`	Shows the applicable built-in usage.
`add -f <Full Path to License File>`	Adds a license from the specified Check Point license file. You get this license file in the Check Point User Center.
`add -m <Host> <Date> <Signature Key> <SKU/Features>`	Adds the license manually. You get these license details in the Check Point User Center.
`del <Signature Key>`	Delete the license based on its signature. This is the same command as the *"cplic del" on page 199*.
`get [-x]`	Shows the local installed licenses. If you specify the "`-x`" parameter, output also shows the signature key for every installed license. This is the same command as the *"cplic print" on page 200*.

Example 1 - Adding the license from the file

```
[Expert@HostName:0]#  cp_conf lic add -f ~/License.lic
License was installed successfully.
[Expert@HostName:0]#

[Expert@HostName:0]# cp_conf lic get
Host             Expiration   Signature                                 Features
192.168.3.28     25Aug2019    xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx   CPMP-XXX
[Expert@HostName:0]#
```

Example 2 - Adding the license manually

```
[Expert@MyHostName:0]# cp_conf lic add -m MyHostName 25Aug2019 xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx CPMP-
XXX
License was successfully installed
[Expert@MyHostName:0]#

[Expert@MyHostName:0]# cp_conf lic get
Host             Expiration   Signature                                 Features
192.168.3.28     25Aug2019    xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx   CPMP-XXX
[Expert@MyHostName:0]#
```

cp_conf sic

Description

Manages SIC on the Security Gateway.

For additional information, see sk65764: How to reset SIC.

Note - This command corresponds to the option **Secure Internal Communication** in the *"cpconfig" on page 189* menu.

Syntax

```
cp_conf
      -h
      sic
            cert_pull <Management Server> <DAIP GW object>
            init <Activation Key> [norestart]
            state
```

Parameters

Parameter	Description
`-h`	Shows the built-in usage.
`cert_pull <Management Server> <DAIP GW object>`	For DAIP Security Gateways, pulls a SIC certificate from the specified Management Server for the specified DAIP Security Gateway: ■ *<Management Server>* - IPv4 address or HostName of the Security Management Server or Domain Management Server ■ *<DAIP GW object>* - Name of the DAIP Security Gateway object as configured in SmartConsole
`init <Activation Key> [norestart]`	Resets the one-time SIC activation key. The optional parameter "`norestart`" specifies not to restart Check Point services.
`state`	Shows the current state of the SIC Trust.

Example

```
[Expert@MyGW:0]# cp_conf sic state

Trust State: Trust established

[Expert@MyGW:0]#
```

cpconfig

Description

This command starts the Check Point Configuration Tool.

This tool lets you configure specific settings for the installed Check Point products.

Important - In 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 *R80.40 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 *R80.40 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: ■ *R80.40 Installation and Upgrade Guide*. ■ *R80.40 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: ■ *R80.40 Installation and Upgrade Guide*. ■ *R80.40 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 *R80.40 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 *R80.40 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: ■ *R80.40 Installation and Upgrade Guide*. ■ *R80.40 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: ■ *R80.40 Installation and Upgrade Guide*. ■ *R80.40 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 the *R80.40 Performance Tuning Administration Guide*.

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) :
```

cpinfo

Description

A utility that collects diagnostics data on your Check Point computer at the time of execution.

It is mandatory to collect these data when you contact Check Point Support about an issue on your Check Point computer.

For more information, see sk92739.

cplic

Description

The `cplic` command lets you manage Check Point licenses.

You can run this command in Gaia Clish or in the Expert Mode.

License Management is divided into three types of commands:

Licensing Commands	Applies To	Description
Local licensing commands	Management Servers, Security Gateways and Cluster Members	You execute these commands locally on the Check Point computers.
Remote licensing commands	Management Servers only	You execute these commands on the Security Management Server or Domain Management Server. These changes affect the managed Security Gateways and Cluster Members.
License Repository commands	Management Servers only	You execute these commands on the Security Management Server or Domain Management Server. These changes affect the licenses stored in the local license repository.

For more about managing licenses, see the *R80.40 Security Management Administration Guide*.

Syntax for Local Licensing on a Security Gateway or Cluster Member

```
cplic [-d]
      {-h | -help}
      check <options>
      contract <options>
      del <options>
      print <options>
      put <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.
`{-h | -help}`	Shows the applicable built-in usage.
`check <options>`	Confirms that the license includes the feature on the local Security Gateway or Security Management Server. See *"cplic check" on page 195*.
`contract <options>`	Manages (deletes and installs) the Check Point Service Contract on the local Check Point computer. See *"cplic contract" on page 197*.
`del <options>`	Deletes a Check Point license on a host, including unwanted evaluation, expired, and other licenses. See *"cplic del" on page 199*.
`print <options>`	Prints details of the installed Check Point licenses on the local Check Point computer. See *"cplic print" on page 200*.
`put <options>`	Installs and attaches licenses on a Check Point computer. See *"cplic put" on page 202*.

cplic check

Description

Confirms that the license includes the feature on the local Security Gateway or Management Server. See sk66245.

Syntax

```
cplic check {-h | -help}
```

```
cplic [-d] check [-p <Product>] [-v <Version>] [{-c | -count}] [-t
<Date>] [{-r | -routers}] [{-S | -SRusers}] <Feature>
```

Parameters

Parameter	Description
`{-h | -help}`	Shows the applicable built-in usage.
`-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.
`-p <Product>`	Product, for which license information is requested. Some examples of products: ■ `fw1` - FireWall-1 infrastructure on Security Gateway / Cluster Member (all blades), or Management Server (all blades) ■ `mgmt` - Multi-Domain Server infrastructure ■ `services` - Entitlement for various services ■ `cvpn` - Mobile Access ■ `etm` - QoS (FloodGate-1) ■ `eps` - Endpoint Software Blades on Management Server
`-v <Version>`	Product version, for which license information is requested.
`{-c | -count}`	Outputs the number of licenses connected to this feature.

Parameter	Description
`-t <Date>`	Checks license status on future date. Use the format **ddmmyyyy**. A feature can be valid on a given date on one license, but invalid on another.
`{-r | -routers}`	Checks how many routers are allowed. The *<Feature>* option is not needed.
`{-S | -SRusers}`	Checks how many SecuRemote users are allowed.
<Feature>	Feature, for which license information is requested.

Example from a Security Gateway

```
[Expert@GW]# cplic print -p
Host Expiration Primitive-Features
W.X.Y.Z 23Mar2016 ::CK-XXXXXXXXXXX fw1:6.0:swb fw1:6.0:abot fw1:6.0:ips fw1:6.0:appi fw1:6.0:aspm fw1:6.0:av1000 fw1:6.0:urlf
fw1:6.0:av fw1:6.0:vsx5 fw1:6.0:cpls fw1:6.0:cluster-u fw1:6.0:mpu fw1:6.0:sxl_vpn fw1:6.0:sxl_fw fw1:6.0:sxl_ppk fw1:6.0:connect
fw1:6.0:pam etm:6.0:fgcountunl etm:6.0:fg etm:6.0:tclog etm:6.0:fgvpn fw1:6.0:identity cvpn:6.0:ccvunl cvpn:6.0:cvpnunlimited
fw1:6.0:des fw1:6.0:strong fw1:6.0:encryption cvpn:6.0:cvpn fw1:6.0:dlp evnt:6.0:smrt_evnt fw1:6.0:ipsa fw1:6.0:spcps fw1:6.0:pam
fw1:6.0:enchostsunlimit fw1:6.0:aes fw1:6.0:rdp fw1:6.0:isakmp fw1:6.0:xlate fw1:6.0:auth fw1:6.0:content fw1:6.0:sync fw1:6.0:fm
fw1:6.0:blades fw1:6.0:sr5000 fw1:6.0:hostsunlimit fw1:6.0:mc_all_8 fw1:6.0:multicore
[Expert@GW]#
```

Example from a Cluster Member

```
[Expert@GW]# cplic check cluster-u
cplic check 'cluster-u': license valid
[Expert@GW]#
[Expert@GW]# cplic check -c cluster-u
cplic check 'cluster-u': 9 licenses
[Expert@GW]#
```

cplic contract

Description

Deletes the Check Point Service Contract on the local Check Point computer.

Installs the Check Point Service Contract on the local Check Point computer.

Note

- For more information about Service Contract files, see sk33089: What is a Service Contract File?
- If you install a Service Contract on a managed Security Gateway / Cluster Member, you must update the license repository on the applicable Management Server - either with the "`cplic get`" command, or in SmartUpdate.

Syntax

```
cplic contract -h
```

```
cplic [-d] contract
      del
            -h
            <Service Contract ID>
      put
            -h
            [{-o | -overwrite}] <Service Contract File>
```

Parameters

Parameter	Description
`{-h | -help}`	Shows the applicable built-in usage.
`-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.
`del`	Deletes the Service Contract from the `$CPDIR/conf/cp.contract` file on the local Check Point computer.
`put`	Merges the Service Contract to the `$CPDIR/conf/cp.contract` file on the local Check Point computer.
<Service Contract ID>	ID of the Service Contract.
`{-o | -overwrite}`	Specifies to overwrite the current Service Contract.
<Service Contract File>	Path to and the name of the Service Contract file. First, you must download the Service Contract file from your Check Point User Center account.

cplic del

Description

Deletes a Check Point license on a host, including unwanted evaluation, expired, and other licenses.

This command can delete a license on both local computer, and on remote managed computers.

Syntax

```
cplic del {-h | -help}
cplic [-d] del [-F <Output File>] <Signature> <Object Name>
```

Parameters

Parameter	Description
`{-h | -help}`	Shows the applicable built-in usage.
`-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.
`-F <Output File>`	Saves the command output to the specified file.
`<Signature>`	The signature string within the license. To see the license signature string, run the *"cplic print" on page 200* command.
`<Object Name>`	The name of the Security Gateway / Cluster Member object as defined in SmartConsole.

cplic print

Description

Prints details of the installed Check Point licenses on the local Check Point computer.

Note - On a Security Gateway / Cluster Member, this command prints all installed licenses (both Local and Central).

Syntax

```
cplic print {-h | -help}
```

```
cplic [-d] print[{-n | -noheader}] [-x] [{-t | -type}] [-F <Output File>] [{-p | -preatures}] [-D]
```

Parameters

Parameter	Description
`{-h | -help}`	Shows the applicable built-in usage.
`-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.
`{-n | -noheader}`	Prints licenses with no header.
`-x`	Prints licenses with their signature.
`{-t | -type]`	Prints licenses showing their type: Central or Local.
`-F <Output File>`	Saves the command output to the specified file.
`{-p | -preatures}`	Prints licenses resolved to primitive features.
`-D`	On a Multi-Domain Server, prints only Domain licenses.

Example 1

```
[Expert@HostName:0]# cplic print
Host             Expiration   Features
192.168.3.28     25Aug2019    CPMP-XXX  CK-XXXXXXXXXXX
[Expert@HostName:0]#
```

Example 2

```
[Expert@HostName:0]# cplic print -x
Host             Expiration   Signature                          Features
192.168.3.28     25Aug2019    xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx   CPMP-XXX  CK-XXXXXXXXXXX
[Expert@HostName:0]#
```

cplic put

Description

Installs one or more Local licenses on a Check Point computer.

Note - You get the license details in the Check Point User Center.

Syntax

```
cplic put {-h | -help}
```

```
cplic [-d] put [{-o | -overwrite}] [{-c | -check-only}] [{-s | -select}] [-F <Output File>] [{-P | -Pre-boot}] [{-k | -kernel-only}] -l <License File> [<Host>] [<Expiration Date>] [<Signature>] [<SKU/Features>]
```

Parameters

Parameter	Description
`{-h | -help}`	Shows the applicable built-in usage.
`-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.
`{-o | -overwrite}`	On a Security Gateway / Cluster Member, this command erases only the local licenses, but not central licenses that are installed remotely.
`{-c | -check-only}`	Verifies the license. Checks if the IP of the license matches the Check Point computer and if the signature is valid.
`{-s | -select}`	Selects only the local license whose IP address matches the IP address of the Check Point computer.
`-F <Output File>`	Saves the command output to the specified file.
`{-P | -Pre-boot}`	Use this option after you have upgraded and before you reboot the Check Point computer. Use of this option will prevent certain error messages.

Parameter	Description
`{-K | -kernel-only}`	Pushes the current valid licenses to the kernel. For use by Check Point Support only.
`-l <License File>`	Name of the file that contains the license.
<Host>	Hostname or IP address of the Security Gateway / Cluster Member for a local license. Hostname or IP address of the Security Management Server / Domain Management Server for a central license.
<Expiration Date>	The license expiration date.
<Signature>	The signature string within the license. Case sensitive. The hyphens are optional.
<SKU/Features>	The SKU of the license summarizes the features included in the license. For example: `CPSUITE-EVAL-3DES-vNG`

Copy and paste the parameters from the license received from the User Center:

Parameter	Description
`host`	The IP address of the external interface (in quad-dot notation). The last part cannot be 0 or 255.
`expiration date`	The license expiration date. It can be `never`.
`signature`	The license signature string. Case sensitive. The hyphens are optional.
`SKU/features`	A string listing the SKU and the Certificate Key of the license. The SKU of the license summarizes the features included in the license. For example: `CPSB-SWB CPSB-ADNC-M CK0123456789ab`

Example

```
[Expert@HostName:0]# cplic put -l License.lic
Host Expiration SKU
192.168.2.3 14Jan2016  CPSB-SWB CPSB-ADNC-M CK0123456789ab
[Expert@HostName:0]#
```

cpprod_util

Description

This utility lets you work with Check Point Registry (`$CPDIR/registry/HKLM_registry.data`) without manually opening it:

- Shows which Check Point products and features are enabled on this Check Point computer.
- Enables and disables Check Point products and features on this Check Point computer.

Syntax

`cpprod_util CPPROD_GetValue "<Product>" "<Parameter>" {0|1}`
`cpprod_util CPPROD_SetValue "<Product>" "<Parameter>" {1|4} "<Value>" {0|1}`
`cpprod_util -dump`

Parameters

Parameter	Description
`CPPROD_GetValue`	Gets the configuration status of the specified product or feature: ■ 0 - Disabled ■ 1 - Enabled
`CPPROD_SetValue`	Sets the configuration for the specified product or feature.  **Important** - Do not run these commands unless explicitly instructed by Check Point Support or R&D to do so.
`"<Product>"`	Specifies the product or feature.
`"<Parameter>"`	Specifies the configuration parameter for the specified product or feature.
`"<Value>"`	Specifies the value of the configuration parameter for the specified product or feature: ■ One of these integers: 0, 1, 4 ■ A string
`dump`	Creates a dump file of Check Point Registry (`$CPDIR/registry/HKLM_registry.data`) in the current working directory. The name of the output file is `RegDump`.

Notes

- On a Multi-Domain Server, you must run this command in the context of the relevant Domain Management Server.
- If you run the `cpprod_util` command without parameters, it prints:
  - The list of all available products and features (for example, "`FwIsFirewallMgmt`", "`FwIsLogServer`", "`FwIsStandAlone`")
  - The type of the expected argument when you configure a product or feature ("`no-parameter`", "`string-parameter`", or "`integer-parameter`")
  - The type of the returned output ("`status-output`", or "`no-output`")
- To redirect the output of the `cpprod_util` command, you need to redirect the *stderr* to *stdout*:

  ```
  cpprod_util <options> > <output file> 2>&1
  ```

  Example:

  ```
  cpprod_util > /tmp/output_of_cpprod_util.txt 2>&1
  ```

Examples

Example - Checking if this Check Point computer is configured as a Standalone

```
[Expert@MGMT:0]# cpprod_util FwIsStandAlone
0
[Expert@MGMT:0]#
```

Example - Showing a list of all installed Check Point Products Packages on a Security Gateway

```
[Expert@MyGW:0]# cpprod_util CPPROD_GetInstalledProducts
CPFC
IDA
MGMT
FW1
SecurePlatform
CPinfo
DIAG
PPACK
CVPN
[Expert@MyGW:0]#
```

Example - Checking if this Security Gateway is configured as a VSX Gateway

```
[Expert@MyGW:0]# cpprod_util FwIsVSX
0
[Expert@MyGW:0]#
```

Example - Checking if on this Security Gateway the QoS blade is enabled

```
[Expert@MyGW:0]# cpprod_util FwIsFloodGate
1
[Expert@MyGW:0]#
```

Example - Checking if on this Security Gateway the SmartProvisioning is enabled

```
[Expert@MyGW:0]# cpprod_util FwIsAtlasModule
0
[Expert@MyGW:0]#
```

Example - Checking if this Security Gateway is configured in Bridge Mode

```
[Expert@MyGW:0]# cpprod_util FwIsBridge
0
[Expert@MyGW:0]#
```

Example - Checking if this Security Gateway is a member of Full HA cluster

```
[Expert@MyGW:0]# cpprod_util FwIsFullHA
0
[Expert@MyGW:0]#
```

Example - Checking if this Security Gateway is configured with Dynamically Assigned IP (DAIP)

```
[Expert@MyGW:0]# cpprod_util FwIsDAG
0
[Expert@MyGW:0]#
```

Example - Checking if this Security Gateway is configured with IPv6 addresses

```
[Expert@MyGW:0]# cpprod_util FwIsFireWallIPv6
1
[Expert@MyGW:0]#
```

cpstart

Description

Manually starts all Check Point processes and applications.

Syntax

```
cpstart [-fwflag {-default | -proc | -driver}]
```

Parameters

Important - These parameters are for Check Point internal use. Do *not* use them, unless explicitly instructed by Check Point Support or R&D to do so.

Parameter	Description
`-fwflag -default`	Starts Check Point processes and loads the Default Filter policy (`defaultfilter`).
`-fwflag -proc`	Starts Check Point processes.
`-fwflag -driver`	Loads the Check Point kernel modules.

cpstat

Description

Displays the status and statistics information of Check Point applications.

Syntax

```
cpstat [-d] [-h <Host>] [-p <Port>] [-s <SICname>] [-f <Flavor>] [-o
<Polling Interval> [-c <Count>] [-e <Period>]] <Application Flag>
```

Note - You can write the parameters in the syntax in any order.

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. The output shows the SNMP queries and SNMP responses for the applicable SNMP OIDs.
`-h <Host>`	Optional. When you run this command on a Management Server, this parameter specifies the managed Security Gateway. *<Host>* is an IPv4 address, a resolvable hostname, or a DAIP object name. The default is `localhost`.  **Note** - On a Multi-Domain Server, you must run this command in the context of the applicable Domain Management Server:`mdsenv <IP Address or Name of Domain Management Server>`.
`-p <Port>`	Optional. Port number of the Application Monitoring (AMON) server. The default port is 18192.
`-s <SICname>`	Optional. Secure Internal Communication (SIC) name of the Application Monitoring (AMON) server.

Parameter	Description
`-f <Flavor>`	Optional. Specifies the type of the information to collect. If you do not specify a flavor explicitly, the command uses the first flavor in the `<Application Flag>`. To see all flavors, run the `cpstat` command without any parameters.
`-o <Polling Interval>`	Optional. Specifies the polling interval (in seconds) - how frequently the command collects and shows the information. Examples: ■ 0 - The command shows the results only once and the stops (this is the default value). ■ 5 - The command shows the results every 5 seconds in the loop. ■ 30 - The command shows the results every 30 seconds in the loop. ■ N - The command shows the results every N seconds in the loop. Use this parameter together with the "`-c <Count>`" parameter and the "`-e <Period>`" parameter. Example: `cpstat os -f perf -o 2`
`-c <Count>`	Optional. Specifies how many times the command runs and shows the results before it stops. You must use this parameter together with the "`-o <Polling Interval>`" parameter. Examples: ■ 0 - The command shows the results repeatedly every `<Polling Interval>` (this is the default value). ■ 10 - The command shows the results 10 times every `<Polling Interval>` and then stops. ■ 20 - The command shows the results 20 times every `<Polling Interval>` and then stops. ■ N - The command shows the results N times every `<Polling Interval>` and then stops. Example: `cpstat os -f perf -o 2 -c 2`

Parameter	Description
`-e <Period>`	Optional. Specifies the time (in seconds), over which the command calculates the statistics. You must use this parameter together with the "`-o <Polling Interval>`" parameter. You can use this parameter together with the "`-c <Count>`" parameter. Example: `cpstat os -f perf -o 2 -c 2 -e 60`
< Application Flag>	Mandatory. See the table below with flavors for the application flags.

These flavors are available for the application flags

Note - The available flags depend on the enabled Software Blades. Some flags are supported only by a Security Gateway, and some flags are supported only by a Management Server.

Feature or Software Blade	Flag	Flavors
List of enabled Software Blades	`blades`	`fw, ips, av, urlf, vpn, cvpn, aspm, dlp, appi, anti_bot, default, content_ awareness, threat-emulation, default`
Operating System	`os`	`default, ifconfig, routing, routing6, memory, old_memory, cpu, disk, perf, multi_cpu, multi_disk, raidInfo, sensors, power_supply, hw_info, all, average_cpu, average_memory, statistics, updates, licensing, connectivity, vsx`
Firewall	`fw`	`default, interfaces, policy, perf, hmem, kmem, inspect, cookies, chains, fragments, totals, totals64, ufp, http, ftp, telnet, rlogin, smtp, pop3, sync, log_connection, all`
HTTPS Inspection	`https_ inspection`	`default, hsm_status, all`
Identity Awareness	`identityServer`	`default, authentication, logins, ldap, components, adquery, idc, muh`

Feature or Software Blade	Flag	Flavors
Application Control	appi	default, subscription_status, update_status, RAD_status, top_last_hour, top_last_day, top_last_week, top_last_month
URL Filtering	urlf	default, subscription_status, update_status, RAD_status, top_last_hour, top_last_day, top_last_week, top_last_month
IPS	ips	default, statistics, all
Anti-Virus	ci	default
Threat Prevention	antimalware	default, scanned_hosts, scanned_mails, subscription_status, update_status, ab_prm_contracts, av_prm_contracts, ab_prm_contracts, av_prm_contracts
Threat Emulation	threat-emulation	default, general_statuses, update_status, scanned_files, malware_detected, scanned_on_cloud, malware_on_cloud, average_process_time, emulated_file_size, queue_size, peak_size, file_type_stat_file_scanned, file_type_stat_malware_detected, file_type_stat_cloud_scanned, file_type_stat_cloud_malware_scanned, file_type_stat_filter_by_analysis, file_type_stat_cache_hit_rate, file_type_stat_error_count, file_type_stat_no_resource_count, contract, downloads_information_current, downloading_file_information, queue_table, history_te_incidents, history_te_comp_hosts
Threat Extraction	scrub	default, subscription_status, threat_extraction_statistics
Mobile Access	cvpn	cvpnd, sysinfo, products, overall
VSX	vsx	default, stat, traffic, conns, cpu, all, memory, cpu_usage_per_core
IPsec VPN	vpn	default, product, IKE, ipsec, traffic, compression, accelerator, nic, statistics, watermarks, all
Data Loss Prevention	dlp	default, dlp, exchange_agents, fingerprint

Feature or Software Blade	Flag	Flavors
Content Awareness	`ctnt`	`default`
QoS	`fg`	`all`
High Availability	`ha`	`default, all`
Policy Server for Remote Access VPN clients	`polsrv`	`default, all`
Desktop Policy Server for Remote Access VPN clients	`dtps`	`default, all`
LTE / GX	`gx`	`default, contxt_create_info, contxt_delete_info, contxt_update_info, contxt_path_mng_info, GXSA_GPDU_info, contxt_initiate_info, gtpv2_create_info, gtpv2_delete_info, gtpv2_update_info, gtpv2_path_mng_info, gtpv2_cmd_info, all`
Management Server	`mg`	`default, log_server, indexer`
Certificate Authority	`ca`	`default, crl, cert, user, all`
SmartEvent	`cpsemd`	`default`
SmartEvent Correlation Unit	`cpsead`	`default`
Log Server	`ls`	`default`
CloudGuard Controller	`vsec`	`default`
SmartReporter	`svr`	`default`
Provisioning Agent	`PA`	`default`

Feature or Software Blade	Flag	Flavors
Thresholds configured with the `threshold_config` command	`thresholds`	`default, active_thresholds, destinations, error`
Historical status values	`persistency`	`product, TableConfig, SourceConfig`

Examples

Example - Interfaces on a Security Gateway

```
[Expert@MyGW:0]# cpstat -f interfaces fw

Network interfaces
------------------------------------------------------------------------------------------------------------
------------
|Name|IP           |Netmask      |Flags|Peer name|Remote IP|Topology|Proxy name|Slaves|Ports|IPv6
Address|IPv6 Len|
------------------------------------------------------------------------------------------------------------
------------
|eth0|192.168.30.40|255.255.255.0|    0|         |  0.0.0.0|       4|          |      |     |
::|       0|
|eth1| 172.30.60.80|255.255.255.0|    0|         |  0.0.0.0|       4|          |      |     |
::|       0|
|eth2|      0.0.0.0|      0.0.0.0|    0|         |  0.0.0.0|       4|          |      |     |
::|       0|
|eth3|      0.0.0.0|      0.0.0.0|    0|         |  0.0.0.0|       4|          |      |     |
::|       0|
|eth4|      0.0.0.0|      0.0.0.0|    0|         |  0.0.0.0|       4|          |      |     |
::|       0|
|eth5|      0.0.0.0|      0.0.0.0|    0|         |  0.0.0.0|       4|          |      |     |
::|       0|
|eth6|      0.0.0.0|      0.0.0.0|    0|         |  0.0.0.0|       4|          |      |     |
::|       0|
|eth7|      0.0.0.0|      0.0.0.0|    0|         |  0.0.0.0|       4|          |      |     |
::|       0|
------------------------------------------------------------------------------------------------------------
------------

[Expert@MyGW:0]#
```

Example - Policy on a Security Gateway

```
[Expert@MyGW:0]# cpstat -f default fw

Policy name:  MyGW_Policy
Install time: Wed May 23 18:14:32 2018


Interface table
----------------------------------------
|Name|Dir|Total   |Accept|Deny    |Log|
----------------------------------------
|eth0|in | 2393126| 32589| 2360537| 52|
|eth0|out|   33016| 33016|       0|  0|
|eth1|in | 2360350|     0| 2360350|  0|
|eth1|out|       0|     0|       0|  0|
|eth2|in | 2360350|     0| 2360350|  0|
|eth2|out|       0|     0|       0|  0|
|eth3|in | 2348704|     0| 2348704|  1|
|eth3|out|       0|     0|       0|  0|
|eth4|in | 2360350|     0| 2360350|  0|
|eth4|out|       0|     0|       0|  0|
----------------------------------------
|    |   |11855896| 65605|11790291| 53|
----------------------------------------

... ... [truncated for brevity] ... ...

[Expert@MyGW:0]#
```

Example - CPU utilization

```
[Expert@HostName:0]# cpstat -f cpu os
CPU User Time (%):   1
CPU System Time (%): 0
CPU Idle Time (%):   99
CPU Usage (%):       1
CPU Queue Length:    -
CPU Interrupts/Sec:  172
CPUs Number:         8

[Expert@HostName:0]#
```

Example - Performance

```
[Expert@HostName:0]# cpstat os -f perf -o 2 -c 2 -e 60

Total Virtual Memory (Bytes):            12417720320
Active Virtual Memory (Bytes):           3741331456
Total Real Memory (Bytes):               8231063552
Active Real Memory (Bytes):              3741331456
Free Real Memory (Bytes):                4489732096
Memory Swaps/Sec:                        -
Memory To Disk Transfers/Sec:            -
CPU User Time (%):                       0
CPU System Time (%):                     0
CPU Idle Time (%):                       100
CPU Usage (%):                           0
CPU Queue Length:                        -
CPU Interrupts/Sec:                      135
CPUs Number:                             8
Disk Servicing Read\Write Requests Time: -
Disk Requests Queue:                     -
Disk Free Space (%):                     61
Disk Total Free Space (Bytes):           12659716096
Disk Available Free Space (Bytes):       11606188032
Disk Total Space (Bytes):                20477751296


Total Virtual Memory (Bytes):            12417720320
Active Virtual Memory (Bytes):           3741556736
Total Real Memory (Bytes):               8231063552
Active Real Memory (Bytes):              3741556736
Free Real Memory (Bytes):                4489506816
Memory Swaps/Sec:                        -
Memory To Disk Transfers/Sec:            -
CPU User Time (%):                       3
CPU System Time (%):                     0
CPU Idle Time (%):                       97
CPU Usage (%):                           3
CPU Queue Length:                        -
CPU Interrupts/Sec:                      140
CPUs Number:                             8
Disk Servicing Read\Write Requests Time: -
Disk Requests Queue:                     -
Disk Free Space (%):                     61
Disk Total Free Space (Bytes):           12659716096
Disk Available Free Space (Bytes):       11606188032
Disk Total Space (Bytes):                20477751296

[Expert@HostName:0]#
```

cpstop

Description

Manually stops all Check Point processes and applications.

Syntax

```
cpstop [-fwflag {-default | -proc | -driver}]
```

Parameters

Important - These parameters are for Check Point internal use. Do *not* use them, unless explicitly instructed by Check Point Support or R&D to do so.

Parameter	Description
`-fwflag -default`	■ Shuts down Check Point processes ■ Loads the Default Filter policy (`defaultfilter`)
`-fwflag -proc`	■ Shuts down Check Point processes ■ Keeps the currently loaded kernel policy ■ Maintains the Connections table, so that after you run the *"cpstart" on page 207* command, you do not experience dropped packets because they are "out of state" **Note** - Only security rules that do not use user space processes continue to work.
`-fwflag -driver`	Unloads the Check Point kernel modules. Therefore, no policy is loaded.  **Warning** - This leaves your Security Gateway, or a Cluster Member without protection. Before you run this command, we recommend to disconnect your Security Gateway, or a Cluster Member from the network completely.

Example

See these articles:

- sk35496
- sk113045

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.

dynamic_objects

Description

Manages dynamic objects and their applicable ranges of IP addresses on the Security Gateway.

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

Workflow

Step	Description
1	In SmartConsole: 1. Define the applicable dynamic object. 2. Install the Access Control Policy on the Security Gateway.
2	On the Security Gateway, run the `dynamic_objects` command to: 1. Create the applicable dynamic object with the same name 2. Assign the applicable ranges of IP address to the new dynamic object.

Syntax

- To show all configured dynamic objects and their ranges of IP addresses:

```
dynamic_objects -l
```

- To create a new dynamic object (and assign a range of IP addresses to it):

```
dynamic_objects -n <object_name> [-r <FromIP1> <ToIP2> ...
[<FromIPx> <ToIPy>] -a]
```

- To add a new a range of IP addresses to the specific existing dynamic object:

```
dynamic_objects -o <object_name> -r <FromIP1> <ToIP2> ...
[<FromIPx> <ToIPy>] -a
```

- To delete a range of IP addresses from the specific existing dynamic object:

```
dynamic_objects -o <object_name> -r <FromIP1> <ToIP2> ...
[<FromIPx> <ToIPy>] -d
```

- To update the specific existing dynamic object (and assign a different range of IP addresses to it):

```
dynamic_objects -u <object_name> [-r <FromIP1> <ToIP2> ...
[<FromIPx> <ToIPy>]]
```

- To compare the configured dynamic objects and objects configured in SmartConsole:

```
dynamic_objects -c
```

- To delete the specific existing dynamic object (and all ranges of IP addresses assigned to it):

```
dynamic_objects -do <object_name>
```

- To delete all the existing dynamic objects (and all ranges of IP addresses assigned to them):

```
dynamic_objects -e
```

Parameters

Parameter	Description
`<object_name>`	Specifies the name of the object: ■ As defined in SmartConsole ■ As defined with the "`dynamic_objects -n <object name>`" command

`-r <FromIP1> <ToIP2> ... [<FromIPx> <ToIPy>]`	Specifies the ranges of IP addresses in the format of pairs:
	`<From_IP_Address> <To_IP_Address>`
	For example, to specify two ranges, from 192.168.2.30 to 192.168.2.40 and from 192.168.2.50 to 192.168.2.60, enter these four IP addresses:
	`192.168.2.30 192.168.2.40 192.168.2.50 192.168.2.60`

`-a`	Adds the specified ranges of IP addresses to the specified dynamic object.
`-c`	Compare the dynamic objects in the dynamic objects database (`$FWDIR/database/dynamic_objects.db`) and in the `$FWDIR/conf/objects.C` file.
`-d`	Deletes range of IP addresses from the dynamic object.
`-do`	Deletes the specified dynamic object.
`-e`	Deletes all configured dynamic objects from the dynamic objects database (`$FWDIR/database/dynamic_objects.db`).
`-l`	Lists the configured dynamic objects in the dynamic objects database (`$FWDIR/database/dynamic_objects.db`).
`-n`	Creates a new dynamic object.
`-u`	Updates the specified dynamic object. If you specify a range of IP addresses, then the new range replaces all current ranges that are currently assigned to this dynamic object.

Example 1 - Create a new dynamic object named "bigserver" and assign to it the range of IP addresses 192.168.2.30-192.168.2.40

Run either these two commands:

```
dynamic_objects -n bigserver
dynamic_objects -o bigserver -r 192.168.2.30 192.168.2.40 -a
```

Or this single command:

```
dynamic_objects -n bigserver -r 192.168.2.20 192.168.2.40 -a
```

Example 2 - Update the ranges of IP addresses assigned to the dynamic object named "bigserver" from the current range to the new range 192.168.2.60-192.168.2.80

```
dynamic_objects -u bigserver -r 192.168.2.60 192.168.2.80
```

cpwd_admin

Description

The Check Point WatchDog (`cpwd`) is a process that invokes and monitors critical processes such as Check Point daemons on the local computer, and attempts to restart them if they fail.

Among the processes monitored by Watchdog are `fwm`, `fwd`, `cpd`, `DAService`, and others.

The list of monitored processes depends on the installed and configured Check Point products and Software Blades.

The Check Point WatchDog writes monitoring information to the `$CPDIR/log/cpwd.elg log` file.

The `cpwd_admin` utility shows the status of the monitored processes, and configures the Check Point WatchDog.

There are two types of Check Point WatchDog monitoring

Monitoring	Description
Passive	WatchDog restarts the process only when the process terminates abnormally. In the output of the `cpwd_admin list` command, the `MON` column shows `N` for passively monitored processes.
Active	WatchDog checks the process status every predefined interval. WatchDog makes sure the process is alive, as well as properly functioning (not stuck on deadlocks, frozen, and so on). In the output of the `cpwd_admin list` command, the `MON` column shows `Y` for actively monitored processes. The list of actively monitored processes is predefined by Check Point. Users cannot change or configure it.

Syntax

```
cpwd_admin
      config <options>
      del <options>
      detach <options>
      exist
      flist <options>
      getpid <options>
      kill
      list <options>
      monitor_list
      start <options>
      start_monitor
      stop <options>
      stop_monitor
```

Parameters

Parameter	Description
`config <options>`	Configures the Check Point WatchDog. See *"cpwd_admin config" on page 226*.
`del <options>`	Temporarily deletes a monitored process from the WatchDog database of monitored processes. See *"cpwd_admin del" on page 229*.
`detach <options>`	Temporarily detaches a monitored process from the WatchDog monitoring. See *"cpwd_admin detach" on page 230*.
`exist`	Checks whether the WatchDog process `cpwd` is alive. See *"cpwd_admin exist" on page 231*.
`flist <options>`	Saves the status of all monitored processes to a `$CPDIR/tmp/cpwd_list_<Epoch Timestamp>.lst` file. See *"cpwd_admin flist" on page 232*.
`getpid <options>`	Shows the PID of a monitored process. See *"cpwd_admin getpid" on page 234*.

Parameter	Description
`kill <options>`	Terminates the WatchDog process `cpwd`. See *"cpwd_admin kill" on page 235*.  **Important** - Do not run this command unless explicitly instructed by Check Point Support or R&D to do so.
`list`	Prints the status of all monitored processes on the screen. See *"cpwd_admin list" on page 236*.
`monitor_ list`	Prints the status of actively monitored processes on the screen. See *"cpwd_admin monitor_list" on page 239*.
`start <options>`	Starts a process as monitored by the WatchDog. See *"cpwd_admin start" on page 240*.
`start_ monitor`	Starts the active WatchDog monitoring - WatchDog monitors the predefined processes actively. See *"cpwd_admin start_monitor" on page 242*.
`stop <options>`	Stops a monitored process. See *"cpwd_admin stop" on page 243*.
`stop_ monitor`	Stops the active WatchDog monitoring - WatchDog monitors all processes only passively. See *"cpwd_admin stop_monitor" on page 245*.

cpwd_admin config

Description

Configures the Check Point WatchDog.

Important - After changing the WatchDog configuration parameters, you must restart the WatchDog process with the `cpstop` and cpstart commands (which restart *all* Check Point processes).

Syntax

```
cpwd_admin config
      -h
      -a <options>
      -d <options
      -p
      -r
```

Parameters

Parameter	Description
`-h`	Shows built-in usage.
`-a <Configuration_Parameter_1>=<Value_1> <Configuration_Parameter_2>=<Value_2> ... <Configuration_Parameter_N>=<Value_N>`	Adds the WatchDog configuration parameters.  **Note** - Spaces are not allowed between the name of the configuration parameter, the equal sign, and the value.
`-d <Configuration_Parameter_1> <Configuration_Parameter_2> ... <Configuration_Parameter_N>`	Deletes the WatchDog configuration parameters that user added with the "`cpwd_admin config -a`" command.
`-p`	Shows the WatchDog configuration parameters that user added with the "`cpwd_admin config -a`" command.
`-r`	Restores the default WatchDog configuration.

These are the available configuration parameters and the accepted values:

Configuration Parameter	Accepted Values	Description
`default_ctx`	Text string up to 128 characters	On a VSX Gateway, configures the CTX value that is assigned to monitored processes, for which no CTX is specified.
`display_ctx`	■ 0 (default) ■ 1	On a VSX Gateway, configures whether the WatchDog shows the `CTX` column in the output of the `cpwd_admin list` command (between the `APP` and the `PID` columns): ■ 0 - Does not show the `CTX` column ■ 1 - Shows the `CTX` column
`no_limit`	■ Range: -1, 0, >0 ■ Default: 5	If `rerun_mode=1`, specifies the maximal number of times the WatchDog tries to restart a process. ■ -1 - Always tries to restart ■ 0 - Never tries to restart ■ >0 - Tries this number of times
`num_of_procs`	■ Range: 30 - 2000 ■ Default: 2000	Configures the maximal number of processes managed by the WatchDog.
`rerun_mode`	■ 0 ■ 1 (default)	Configures whether the WatchDog restarts processes after they fail: ■ 0 - Does not restart a failed process. Monitor and log only. ■ 1 - Restarts a failed process (this is the default).
`reset_startups`	■ Range: > 0 ■ Default: 3600	Configures the time (in seconds) the WatchDog waits after the process starts and before the WatchDog resets the process's `startup_counter` to 0. To see the process's startup counter, in the output of the `cpwd_admin list` command, refer to the `#START` column.
`sleep_mode`	■ 0 ■ 1 (default)	Configures how the WatchDog restarts the process: ■ 0 - Ignores timeout and restarts the process immediately ■ 1 - Waits for the duration of `sleep_timeout`
`sleep_timeout`	■ Range: 0 - 3600 ■ Default: 60	If `rerun_mode=1`, specifies how much time (in seconds) passes from a process failure until WatchDog tries to restart it.

Configuration Parameter	Accepted Values	Description
`stop_ timeout`	■ Range: > 0 ■ Default: 60	Configures the time (in seconds) the WatchDog waits for a process stop command to complete.
`zero_ timeout`	■ Range: > 0 ■ Default: 7200	After failing `no_limit` times to restart a process, the WatchDog waits `zero_timeout` seconds before it tries again. The value of the `zero_timeout` must be greater than the value of the `timeout`.

The WatchDog saves the user defined configuration parameters in the `$CPDIR/registry/HKLM_ registry.data` file in the "`: (Wd_Config`" section:

```
("CheckPoint Repository Set"
  : (SOFTWARE
    : (CheckPoint
      : (CPshared
        :CurrentVersion (6.0)
        : (6.0
        ... ...
          : (reserved
          ... ...
            : (Wd
                : (Wd_Config
                    :Configuration_Parameter_1 ("[4]Value_1")
                    :Configuration_Parameter_2 ("[4]Value_2")
                )
            )
          ... ...
```

Example

```
[Expert@HostName:0]# cpwd_admin config -p
cpWatchDog doesn't have configuration parameters
[Expert@HostName:0]#
[Expert@HostName:0]# cpwd_admin config -a sleep_timeout=120 no_limit=12
[Expert@HostName:0]#
[Expert@HostName:0]# cpwd_admin config -p
cpWatchDog Configuration parameters are:
sleep_timeout : 120
no_limit : 12
[Expert@HostName:0]#
[Expert@HostName:0]# cpstop ; cpstart
[Expert@HostName:0]#

[Expert@HostName:0]# cpwd_admin config -r
cpWatchDog doesn't have configuration parameters
[Expert@HostName:0]#
[Expert@HostName:0]# cpstop ; cpstart
[Expert@HostName:0]#
[Expert@HostName:0]# cpwd_admin config -p
cpWatchDog doesn't have configuration parameters
[Expert@HostName:0]#
```

cpwd_admin del

Description

Temporarily deletes a monitored process from the WatchDog database of monitored processes.

Notes:

- WatchDog stops monitoring the detached process, but the process stays alive.
- The *"cpwd_admin list" on page 236* command does not show the deleted process anymore.
- This change applies until all Check Point services restart during boot, or with the *"cpstart" on page 207* command.

Syntax on a Security Gateway

```
cpwd_admin del -name <Application Name> [-ctx <VSID>]
```

Parameters

Parameter	Description
<Application Name>	Name of the monitored Check Point process as you see in the output of the *"cpwd_admin list" on page 236* command in the leftmost column `APP`. Examples: ■ `FWM` ■ `FWD` ■ `CPD` ■ `CPM`
`-ctx <VSID>`	On a VSX Gateway, specifies the context of the applicable Virtual System.

Example

```
[Expert@HostName:0]# cpwd_admin del -name FWD
cpwd_admin:
successful Del operation
[Expert@HostName:0]#
```

cpwd_admin detach

Description

Temporarily detaches a monitored process from the WatchDog monitoring.

Notes:

- WatchDog stops monitoring the detached process, but the process stays alive.
- The *"cpwd_admin list" on page 236* command does not show the detached process anymore.
- This change applies until all Check Point services restart during boot, or with the *"cpstart" on page 207* command.

Syntax on a Security Gateway

```
cpwd_admin detach -name <Application Name> [-ctx <VSID>]
```

Parameters

Parameter	Description
<Application Name>	Name of the monitored Check Point process as you see in the output of the *"cpwd_admin list" on page 236* command in the leftmost column `APP`. Examples: ■ `FWM` ■ `FWD` ■ `CPD` ■ `CPM`
`-ctx <VSID>`	On a VSX Gateway, specifies the context of the applicable Virtual System.

Example

```
[Expert@HostName:0]# cpwd_admin detach-name FWD
cpwd_admin:
successful Detach operation
[Expert@HostName:0]#
```

cpwd_admin exist

Description

Checks whether the WatchDog process `cpwd` is alive.

Syntax

```
cpwd_admin exist
```

Example

```
[Expert@HostName:0]# cpwd_admin exist
 cpwd_admin: cpWatchDog is running
[Expert@HostName:0]#
```

cpwd_admin flist

Description

Prints the status of all WatchDog monitored processes on the screen.

Syntax on a Security Gateway

```
cpwd_admin flist [-full] [-ctx <VSID>]
```

Parameters

Parameter	Description
`-full`	Shows the verbose output.
`-ctx <VSID>`	On a VSX Gateway, specifies the context of the applicable Virtual System.

Output

Column	Description
`APP`	Shows the WatchDog name of the monitored process.
`CTX`	On a VSX Gateway, shows the VSID, in which the monitored process runs.
`PID`	Shows the PID of the monitored process.
`STAT`	Shows the status of the monitored process: ■ `E` - executing ■ `T` - terminated
`#START`	Shows how many times the WatchDog started the monitored process.
`START_TIME`	Shows the time when the WatchDog started the monitored process for the last time.
`SLP/LIMIT`	In verbose output, shows the values of the `sleep_timeout` and `no_limit` configuration parameters (see *"cpwd_admin" on page 223*).
`MON`	Shows how the WatchDog monitors this process (see the explanation for the *"cpwd_admin" on page 223*): ■ `Y` - Active monitoring ■ `N` - Passive monitoring
`COMMAND`	Shows the command the WatchDog run to start this process.

Example

```
[Expert@HostName:0]# cpwd_admin flist
/opt/CPshrd-R80.40/tmp/cpwd_list_1564617600.lst
[Expert@HostName:0]#
[Expert@HostName:0]# date --date="@1564617600"
Thu Aug  1 03:00:00 IDT 2019
[Expert@HostName:0]#
```

cpwd_admin getpid

Description

Shows the PID of a WatchDog monitored process.

Syntax for a Security Gateway

```
cpwd_admin getpid -name <Application Name> [-ctx <VSID>]
```

Parameters

Parameter	Description
<Application Name>	Name of the monitored Check Point process as you see in the output of the *"cpwd_admin list" on page 236* command in the leftmost column `APP`. Examples: ■ `FWM` ■ `FWD` ■ `CPD` ■ `CPM`
`-ctx <VSID>`	On VSX Gateway, specifies the context of the applicable Virtual System.

Example

```
[Expert@HostName:0]# cpwd_admin getpid -name FWD
5640
[Expert@HostName:0]#
```

cpwd_admin kill

Description

Terminates the WatchDog process `cpwd`.

Important - Do not run this command unless explicitly instructed by Check Point Support or R&D to do so.

To restart the WatchDog process, you must restart all Check Point services with the *"cpstop" on page 216* and *"cpstart" on page 207* commands.

Syntax

```
cpwd_admin kill
```

cpwd_admin list

Description

Prints the status of all WatchDog monitored processes on the screen.

Syntax on a Security Gateway

```
cpwd_admin list [-full] [-ctx <VSID>]
```

Parameters

Parameter	Description
`-full`	Shows the verbose output.
`-ctx <VSID>`	On a VSX Gateway, specifies the context of the applicable Virtual System.

Output

Column	Description
`APP`	Shows the WatchDog name of the monitored process.
`CTX`	On a VSX Gateway, shows the VSID, in which the monitored process runs.
`PID`	Shows the PID of the monitored process.
`STAT`	Shows the status of the monitored process: ■ `E` - executing ■ `T` - terminated
`#START`	Shows how many times the WatchDog started the monitored process.
`START_ TIME`	Shows the time when the WatchDog started the monitored process for the last time.
`SLP/LIMIT`	In verbose output, shows the values of the `sleep_timeout` and `no_limit` configuration parameters (see *"cpwd_admin" on page 223*).
`MON`	Shows how the WatchDog monitors this process (see the explanation for the *"cpwd_admin" on page 223*): ■ `Y` - Active monitoring ■ `N` - Passive monitoring
`COMMAND`	Shows the command the WatchDog run to start this process.

Examples

Example - Default output on a Security Gateway

```
[Expert@HostName:0]# cpwd_admin list
APP         CTX         PID    STAT  #START  START_TIME             MON  COMMAND
FWK_FORKER  0           4180   E     1       [18:14:04] 23/5/2019   N    fwk_forker
FWK_WD      0           4182   E     1       [18:14:04] 23/5/2019   N    fwk_wd -i 1 -i6 0
CPSICDEMUX  0           5383   E     1       [18:14:14] 23/5/2019   N    cpsicdemux
CPVIEWD     0           5407   E     1       [18:14:15] 23/5/2019   N    cpviewd
HISTORYD    0           5410   E     1       [18:14:15] 23/5/2019   N    cpview_historyd
SXL_STATD   0           5413   E     1       [18:14:15] 23/5/2019   N    sxl_statd
CPD         0           5420   E     1       [18:14:15] 23/5/2019   Y    cpd
MPDAEMON    0           5436   E     1       [18:14:16] 23/5/2019   N    mpdaemon /opt/CPshrd-
R80.40/log/mpdaemon.elg /opt/CPshrd-R80.40/conf/mpdaemon.conf
CI_CLEANUP  0           5626   E     1       [18:14:26] 23/5/2019   N    avi_del_tmp_files
CIHS        0           5628   E     1       [18:14:26] 23/5/2019   N    ci_http_server -j -f
/opt/CPsuite-R80.40/fw1/conf/cihs.conf
FWD         0           5640   E     1       [18:14:26] 23/5/2019   N    fwd
RAD         0           6330   E     1       [18:14:28] 23/5/2019   N    rad
DASERVICE   0           8604   E     1       [18:14:43] 23/5/2019   N    DAService_script
[Expert@HostName:0]#
```

Example - Verbose output on a Security Gateway

```
[Expert@HostName:0]# cpwd_admin list -full
APP        CTX        PID    STAT  #START  START_TIME             SLP/LIMIT  MON
-------------------------------------------------------------------------------
FWK_FORKER 0          4180   E     1       [18:14:04] 23/5/2019   60/5       N
                      PATH = /opt/CPsuite-R80.40/fw1/bin/fwk_forker
                      COMMAND = fwk_forker
-------------------------------------------------------------------------------
FWK_WD     0          4182   E     1       [18:14:04] 23/5/2019   3/u        N
                      PATH = /opt/CPsuite-R80.40/fw1/bin/fwk_wd
                      COMMAND = fwk_wd -i 1 -i6 0
-------------------------------------------------------------------------------
CPSICDEMUX 0          5383   E     1       [18:14:14] 23/5/2019   60/5       N
                      PATH = /opt/CPshrd-R80.40/bin/cpsicdemux
                      COMMAND = cpsicdemux
-------------------------------------------------------------------------------
CPVIEWD    0          5407   E     1       [18:14:15] 23/5/2019   60/5       N
                      PATH = /opt/CPshrd-R80.40/bin/cpviewd
                      COMMAND = cpviewd
-------------------------------------------------------------------------------
HISTORYD   0          5410   E     1       [18:14:15] 23/5/2019   60/5       N
                      PATH = /opt/CPshrd-R80.40/bin/cpview_historyd
                      COMMAND = cpview_historyd
-------------------------------------------------------------------------------
SXL_STATD  0          5413   E     1       [18:14:15] 23/5/2019   60/5       N
                      PATH = /opt/CPsuite-R80.40/fw1/bin/sxl_statd
                      COMMAND = sxl_statd
-------------------------------------------------------------------------------
CPD        0          5420   E     1       [18:14:15] 23/5/2019   60/5       Y
                      PATH = /opt/CPshrd-R80.40/bin/cpd
                      COMMAND = cpd
-------------------------------------------------------------------------------
MPDAEMON   0          5436   E     1       [18:14:16] 23/5/2019   60/5       N
                      PATH = /opt/CPshrd-R80.40/bin/mpdaemon
                      COMMAND = mpdaemon /opt/CPshrd-R80.40/log/mpdaemon.elg /opt/CPshrd-
R80.40/conf/mpdaemon.conf
-------------------------------------------------------------------------------
CI_CLEANUP 0          5626   E     1       [18:14:26] 23/5/2019   60/5       N
                      PATH = /opt/CPsuite-R80.40/fw1/bin/avi_del_tmp_files
                      COMMAND = avi_del_tmp_files
-------------------------------------------------------------------------------
CIHS       0          5628   E     1       [18:14:26] 23/5/2019   60/5       N
                      PATH = /opt/CPsuite-R80.40/fw1/bin/ci_http_server
                      COMMAND = ci_http_server -j -f /opt/CPsuite-R80.40/fw1/conf/cihs.conf
-------------------------------------------------------------------------------
FWD        0          5640   E     1       [18:14:26] 23/5/2019   60/5       N
                      PATH = /opt/CPsuite-R80.40/fw1/bin/fw
                      COMMAND = fwd
-------------------------------------------------------------------------------
RAD        0          6330   E     1       [18:14:28] 23/5/2019   60/5       N
                      PATH = /opt/CPsuite-R80.40/fw1///bin/rad
                      COMMAND = rad
-------------------------------------------------------------------------------
DASERVICE  0          8604   E     1       [18:14:43] 23/5/2019   60/5       N
                      PATH = /opt/CPda/bin/DAService_script
                      COMMAND = DAService_script
[Expert@HostName:0]#
```

cpwd_admin monitor_list

Description

Prints the status of actively monitored processes on the screen.

See the explanation about the active monitoring in *"cpwd_admin" on page 223*.

Syntax

```
cpwd_admin monitor_list
```

Example

```
[Expert@HostName:0]# cpwd_admin monitor_list
cpwd_admin:
APP        FILE_NAME                      NO_MSG_TIMES  LAST_MSG_TIME
CPD        CPD_5420_4714.mntr             0/10          [19:00:33] 31/5/2019
[Expert@HostName:0]#
```

cpwd_admin start

Description

Starts a process as monitored by the WatchDog.

Syntax on a Security Gateway

```
cpwd_admin start -name <Application Name> [-ctx <VSID>] -path "<Full
Path to Executable>" -command "<Command Syntax>" [-env {inherit |
<Env_Var>=<Value>] [-slp_timeout <Timeout>] [-retry_limit {<Limit> |
u}]
```

Parameters

Parameter	Description
`-name <Application Name>`	Name, under which the `cpwd_admin list` command shows the monitored process in the leftmost column `APP`. Examples: ■ `FWM` ■ `FWD` ■ `CPD` ■ `CPM`
`-ctx <VSID>`	On a VSX Gateway, specifies the context of the applicable Virtual System.
`-path "<Full Path to Executable>"`	The full path (with or without Check Point environment variables) to the executable including the executable name. Must enclose in double-quotes. Examples: ■ For FWM: `"$FWDIR/bin/fwm"` ■ For FWD: `"/opt/CPsuite-R80.40/fw1/bin/fw"` ■ For CPD: `"$CPDIR/bin/cpd"` ■ For CPM: `"/opt/CPsuite-R80.40/fw1/scripts/cpm.sh"` ■ For SICTUNNEL: `"/opt/CPshrd-R80.40/bin/cptnl"`

Parameter	Description
`-command "<Command Syntax>"`	The command and its arguments to run. Must enclose in double-quotes. Examples: ■ For FWM: "`fwm`" ■ For FWM on Multi-Domain Server: "`fwm mds`" ■ For FWD: "`fwd`" ■ For CPD: "`cpd`" ■ For CPM: "`/opt/CPsuite-R80.40/fw1/scripts/cpm.sh -s`" ■ For SICTUNNEL: "`/opt/CPshrd-R80.40/bin/cptnl -c "/opt/CPuepm-R80.40/engine/conf/cptnl_srv.conf"`"
`-env {inherit | <Env_Var>=<Value>}`	Configures whether to inherit the environment variables from the shell. ■ `inherit` - Inherits all the environment variables (WatchDog supports up to 80 environment variables) ■ *<Env_Var>=<Value>* - Assigns the specified value to the specified environment variable
`-slp_timeout <Timeout>`	Configures the specified value of the "`sleep_timeout`" configuration parameter. See *"cpwd_admin config" on page 226*.
`-retry_limit {<Limit> | u}`	Configures the value of the "`retry_limit`" configuration parameter. See *"cpwd_admin config" on page 226*. ■ *<Limit>* - Tries to restart the process the specified number of times ■ `u` - Tries to restart the process unlimited number of times

Example

For the list of process and the applicable syntax, see sk97638.

cpwd_admin start_monitor

Description

Starts the active WatchDog monitoring. WatchDog monitors the predefined processes actively.

See the explanation for the *"cpwd_admin" on page 223* command.

Syntax

```
cpwd_admin start_monitor
```

Example

```
[Expert@HostName:0]# cpwd_admin start_monitor
cpwd_admin:
CPWD has started to perform active monitoring on Check Point services/processes
[Expert@HostName:0]#
```

cpwd_admin stop

Description

Stops a WatchDog monitored process.

Syntax on a Security Gateway

```
cpwd_admin stop -name <Application Name> [-ctx <VSID>] [-path "<Full
Path to Executable>" -command "<Command Syntax>" [-env {inherit |
<Env_Var>=<Value>]
```

Parameters

Parameter	Description
`-name <Application Name>`	Name under which the `cpwd_admin list` command shows the monitored process in the leftmost column `APP`. Examples: ■ `FWM` ■ `FWD` ■ `CPD` ■ `CPM`
`-ctx <VSID>`	On a VSX Gateway, specifies the context of the applicable Virtual System.
`-path "<Full Path to Executable>"`	The full path (with or without Check Point environment variables) to the executable including the executable name. Must enclose in double-quotes. Examples: ■ For FWM: "`$FWDIR/bin/fwm`" ■ For FWD: "`/opt/CPsuite-R80.40/fw1/bin/fw`" ■ For CPD: "`$CPDIR/bin/cpd_admin`"
`-command "<Command Syntax>"`	The command and its arguments to run. Must enclose in double-quotes. Examples: ■ For FWM: "`fw kill fwm`" ■ For FWD: "`fw kill fwd`" ■ For CPD: "`cpd_admin stop`"

Parameter	Description
`-env {inherit | <Env_Var>=<Value>}`	Configures whether to inherit the environment variables from the shell. ■ `inherit` - Inherits all the environment variables (WatchDog supports up to 80 environment variables) ■ `<Env_Var>=<Value>` - Assigns the specified value to the specified environment variable

Example

For the list of process and the applicable syntax, see sk97638.

cpwd_admin stop_monitor

Description

Stops the active WatchDog monitoring. WatchDog monitors all processes only passively.

See the explanation for the *"cpwd_admin" on page 223* command.

Syntax

```
cpwd_admin stop_monitor
```

Example

```
[Expert@HostName:0]# cpwd_admin stop_monitor
cpwd_admin:
CPWD has stopped performing active monitoring on Check Point services/processes
[Expert@HostName:0]#
```

fw

Description

- Fetches and unloads Threat Prevention policy.
- Controls the Firewall module.
- Generates the Default Filter policy files.
- Fetches the policy from the Management Server, peer Cluster Member, or local directory.
- Fetches the specified Security or Audit log files from the specified Check Point computer.
- Shows the list of interfaces and their IP addresses.
- Shows information about Check Point computers in High Availability configuration and their states.
- Controls ISP links in ISP Redundancy configuration.
- Kills the specified Check Point processes.
- Shows a list of hosts protected by the Security Gateway.
- Shows the content of Check Point log files.
- Switches the current active log file.
- Shows a list of Security or Audit log files.
- Merges several input log files into a single log file.
- Runs FW Monitor to capture the traffic that passes through the Security Gateway.
- Rebuilds pointer files for Security or Audit log files.
- Manages the Suspicious Activity Monitoring (SAM) rules.
- Manages the Suspicious Activity Policy editor.
- Shows the contents of the Unified Policy kernel tables.
- Shows the currently installed policy.
- Shows and deletes the contents of the specified kernel tables.
- Executes the offline Unified Policy.
- Removes all policies from the Security Gateway or Cluster Member.
- Shows the Security Gateway major and minor version number and build number.

Syntax

```
fw [-d] [-i]
       amw <options>
       ctl <options>
       defaultgen
       fetch <options>
       fetchlogs <options>
       getifs
       hastat <options>
       isp_link <options>
       kill <options>
       lichosts <options>
       log <options>
       logswitch <options>
       lslogs <options>
       mergefiles <options>
       repairlog <options>
       sam <options>
       sam_policy <options>
       showuptables <options>
       stat
       tab <options>
       unloadlocal
       up_execute <options>
       ver <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.

Parameter	Description
`-i`	Specifies the CoreXL Firewall instance. See *"fw -i" on page 250*.
`amw <options>`	Fetches and unloads Threat Prevention policy. See *"fw amw" on page 251*.
`ctl`	Controls the Firewall module. See *"fw ctl" on page 254*.
`defaultgen`	Generates the Default Filter policy files. See *"fw defaultgen" on page 288*.
`fetch <options>`	Fetches the policy from the Management Server, peer Cluster Member, or local directory. See *"fw fetch" on page 290*.
`fetchlogs <options>`	Fetches the specified Security log files (`$FWDIR/log/*.log*`) or Audit log files (`$FWDIR/log/*.adtlog*`) from the specified Check Point computer. See *"fw fetchlogs" on page 292*.
`getifs`	Shows the list with this information: ■ The name of interfaces, to which the Check Point Firewall kernel attached. ■ The IP addresses assigned to the interfaces. See *"fw getifs" on page 294*.
`hastat <options>`	Shows information about Check Point computers in High Availability configuration and their states. See *"fw hastat" on page 295*.
`isp_link <options>`	Controls ISP links in the ISP Redundancy configuration. See *"fw isp_link" on page 296*.
`kill <options>`	Kills the specified Check Point processes. See *"fw kill" on page 297*.
`lichosts <options>`	Shows a list of hosts protected by the Security Gateway. See *"fw lichosts" on page 298*.
`log <options>`	Shows the content of Check Point log files - Security (`$FWDIR/log/*.log`) or Audit (`$FWDIR/log/*.adtlog`). See *"fw log" on page 299*.

Parameter	Description
`logswitch <options>`	Switches the current active log file - Security (`$FWDIR/log/fw.log`) or Audit (`$FWDIR/log/fw.adtlog`). See *"fw logswitch" on page 308*.
`lslogs <options>`	Shows a list of Security log files (`$FWDIR/log/*.log*`) or Audit log files (`$FWDIR/log/*.adtlog*`) residing on the local computer or a remote computer. See *"fw lslogs" on page 311*.
`mergefiles <options>`	Merges several input log files - Security (`$FWDIR/log/*.log`) or Audit (`$FWDIR/log/*.adtlog`) - into a single log file. See *"fw mergefiles" on page 314*.
`monitor <options>`	Runs FW Monitor to capture the traffic that passes through the Security Gateway. See *"fw monitor" on page 317*.
`repairlog <options>`	Rebuilds pointer files for Security log files (`$FWDIR/log/*.log`) or Audit (`$FWDIR/log/*.adtlog`) log files. See *"fw repairlog" on page 347*.
`sam <options>`	Manages the Suspicious Activity Monitoring (SAM) rules. See *"fw sam" on page 348*.
`sam_policy <options>`	Manages the Suspicious Activity Policy editor. See *"fw sam_policy" on page 356*.
`showuptables <options>`	Shows the contents of the Unified Policy kernel tables. See *"fw showuptables" on page 381*.
`stat`	Shows the currently installed policy. See *"fw stat" on page 382*.
`tab <options>`	Shows and deletes the contents of the specified kernel tables. See *"fw tab" on page 384*.
`unloadlocal`	Uninstalls all policies from the Security Gateway or Cluster Member. See *"fw unloadlocal" on page 391*.
`up_execute <options>`	Executes the offline Unified Policy. See *"fw up_execute" on page 395*.
`ver <options>`	Shows the Security Gateway major and minor version number and build number. See *"fw ver" on page 398*.

fw -i

Description

By default, the *"fw" on page 246* 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 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
```

fw amw

Description

Fetches and unloads Threat Prevention policy.

Threat Prevention policy applies to these Software Blades:

- Anti-Bot
- Anti-Spam
- Anti-Virus
- IPS
- Threat Emulation
- Threat Extraction

Syntax

- To fetch the Threat Prevention policy from the Management Server:

```
fw [-d] amw fetch -f [-i] [-n] [-r]
```

- To fetch the Threat Prevention policy from a peer Cluster Member, and, if it fails, then from the Management Server:

```
fw [-d] amw fetch -f -c [-i] [-n] [-r]
```

- To fetch the Threat Prevention policy from the specified Check Point computer(s):

```
fw [-d] amw fetch [-i] [-n] [-r] <Master 1> [<Master 2> ...]
```

- To fetch the Threat Prevention policy stored locally on the Security Gateway:

```
fw [-d] amw fetch local [-nu]
fw [-d] amw fetch localhost [-nu]
```

- To fetch the Threat Prevention policy stored locally on the Security Gateway in the specified directory:

```
fw [-d] amw fetchlocal [-lu] -d <Full Path to Directory>
```

- To unload the current Threat Prevention policy:

```
fw [-d] amw unload
```

Parameters

Parameter	Description
`fw -d amw ...`	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.
`fw amw fetch`	Fetches the Threat Prevention policy from the specified Check Point computer(s). These can be a Management Server, or a peer Cluster Member.
`fw amw fetch local` `fw amw fetch localhost`	Fetches the Threat Prevention policy that is stored locally on the Security Gateway in the `$FWDIR/state/local/AMW/` directory.
`fw amw fetchlocal`	Fetches the Threat Prevention policy that stored locally on the Security Gateway in the specified directory.
`fw amw unload`	Unloads the current Threat Prevention policy from the Security Gateway.  **Important** - This significantly decreases the security on the Security Gateway. This is the same as if you disable the Threat Prevention Software Blades on the Security Gateway.
`-c`	Specifies that you fetch the policy from a peer Cluster Member.  **Notes:** ■ Must also use the "`-f`" parameter. ■ Works only in cluster.
`-f`	Specifies that you fetch the policy from a Management Server listed in the `$FWDIR/conf/masters` file.
`-i`	On a Security Gateway with dynamically assigned IP address (DAIP), specifies to ignore the SIC name and object name.
`-lu`	Specifies to perform a late update - to load signatures just after the Security Gateway copies the policy files to the local directory `$FWDIR/state/local/AMW/`.
`-n`	Specifies not to load the fetched policy, if it is the same as the policy already located on the Security Gateway.
`-nu`	Specifies not to update the currently installed policy.

Parameter	Description
`-r`	On a Cluster Member, specifies to ignore this option in SmartConsole **Install Policy** window: **For gateway clusters, if installation on a cluster member fails, do not install on that cluster**  **Best Practice** - Use this parameter if a peer Cluster Member is **Down**.
`<Master 1> [<Master 2> ...]`	Specifies the Check Point computer(s), from which to fetch the Threat Prevention policy. You can fetch the Threat Prevention policy from the Management Server, or a peer Cluster Member.  **Notes:** ■ If you fetch the Threat Prevention policy from the Management Server, you can enter one of these: • The main IP address of the Management Server object. • The object name of the Management Server. • The hostname that the Security Gateway resolves to the main IP address of the Management Server. ■ If you fetch the Threat Prevention policy from a peer Cluster Member, you can enter one of these: • The main IP address of the Cluster Member object. • The IP address of the Sync interface on the Cluster Member. ■ If the fetch from the first specified *<Master>* fails, the Security Gateway fetches the policy from the second specified *<Master>*, and so on. If the Security Gateway fails to connect to each specified *<Masters>*, the Security Gateway fetches the policy from the `localhost`. ■ If you do not specify the *<Masters>* explicitly, the Security Gateway fetches the policy from the `localhost`.
`-d <Full Path to Directory>`	Specifies local directory on the Security Gateway, from which to fetch the Threat Prevention policy files.

Example

```
[Expert@MyGW:0]# fw amw fetch local
Installing Threat Prevention policy from local
Fetching Threat Prevention policy succeeded
[Expert@MyGW:0]#
```

fw ctl

Description

Controls the Firewall kernel module.

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

Syntax

```
fw [-d] ctl
      arp <options>
      bench <options>
      block <options>
      chain
      conn
      conntab <options>
      cpasstat <options>
      debug <options>
      get <options>
      iflist
      install
      kdebug <options>
      pstat <options>
      set <options>
      tcpstrstat <options>
      uninstall
```

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.

Parameter	Description
`arp <options>`	Shows the configured Proxy ARP entries based on the `$FWDIR/conf/local.arp` file on the Security Gateway. See *"fw ctl arp" on page 257*.
`bench <options>`	Runs the CPU benchmark tests that collect these statistics: ■ FireWall Lock Statistics ■ Outbound Packets Statistics ■ Inbound Packets Statistics See *"fw ctl bench" on page 258*.
`block <options>`	Blocks all connections to, from, and through the Security Gateway. See *"fw ctl block" on page 260*.
`chain`	Shows the list of Firewall Chain Modules. See *"fw ctl chain" on page 261*.
`conn`	Shows the list of Firewall Connection Modules. See *"fw ctl conn" on page 263*.
`conntab <options>`	Shows formatted list of current connections from the **Connections** kernel table (ID 8158). See *"fw ctl conntab" on page 265*.
`cpasstat <options>`	Generates statistics report about Check Point Active Streaming (CPAS). See *"fw ctl cpasstat" on page 269*.
`debug <options>`	Generates kernel debug messages from Check Point Firewall kernel to a debug buffer. See *"'fw ctl debug' and 'fw ctl kdebug'" on page 270*.
`dlpkstat <options>`	Generates statistics report about Data Loss Prevention kernel module. See *"fw ctl dlpkstat" on page 271*.
`get <options>`	Shows the value of the specified kernel parameter. See *"fw ctl get" on page 272*.
`iflist`	Shows the list with this information: ■ The name of interfaces, to which the Check Point Firewall kernel attached. ■ The internal numbers of the interfaces in the Check Point Firewall kernel. See *"fw ctl iflist" on page 274*.

Parameter	Description
`install`	Tells the operating system to start passing packets to Firewall. See *"fw ctl install" on page 275*.
`kdebug <options>`	Generates kernel debug messages from Check Point Firewall kernel to a debug buffer. See *"'fw ctl debug' and 'fw ctl kdebug'" on page 270*.
`leak <options>`	Generates leak detection report. See *"fw ctl leak" on page 276*.
`pstat <options>`	Shows Security Gateway various internal statistics. See *"fw ctl pstat" on page 280*.
`set <options>`	Configures the specified value for the specified kernel parameter. See *"fw ctl set" on page 283*.
`tcpstrstat <options>`	Generates statistics report about TCP Streaming. See *"fw ctl tcpstrstat" on page 285*.
`uninstall`	Tells the operating system to stop passing packets to Firewall, and unloads the current Security Policy. See *"fw ctl uninstall" on page 287*.

fw ctl arp

Description

Shows the configured Proxy ARP entries based on the `$FWDIR/conf/local.arp` file on the Security Gateway.

For more information about the Proxy ARP, see sk30197.

Syntax

```
fw [-d] ctl arp
       [-h]
       [-n]
```

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.
`-h`	Shows the built-in help.
`-n`	Specifies not to resolve hostnames.

fw ctl bench

Description

The benchmark mechanism provides a way to measure the time spent in the code between two points.

This command runs the CPU benchmark tests that collect these statistics:

- FireWall Lock Statistics
- Outbound Packets Statistics
- Inbound Packets Statistics.

Note - This command writes the output of these tests to the `dmesg`.

Syntax

```
fw [-d] ctl bench
      -h
      lock
            [{ioctl | packet} [<Limit>]]
            [stop]
      packet [{<Limit> | stop}]
```

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.
`-h`	Shows the built-in help.

Parameter	Description

`lock`
`[ioctl[ <Limit>]]`
`[packet [<Limit>]]`
`[stop]`
	Runs the lock benchmark that collects the FireWall Lock Statistics.
	Available options:
	■ No parameters - Starts the lock benchmark.
	■ `ioctl` - Calculates the IOCTL flow statistics.
	■ `packet` - Calculates the packet flow statistics.
	■ *<Limit>* - Specifies the time limit (in seconds) for the benchmark to run. Default is 10 seconds. Maximum is 200 seconds.
	■ `stop` - Stops the current lock benchmark.

`packet [{<Limit> | stop}]`	Runs the packet benchmark test that collects these statistics: ■ Outbound Packets Statistics ■ Inbound Packets Statistics Available options: ■ No parameters - Starts the packet benchmark. ■ *<Limit>* - Specifies the time limit (in seconds) for the benchmark to run. Default is 10 seconds. Maximum is 200 seconds. ■ `stop` - Stops the current packet benchmark.

fw ctl block

Description

Blocks all connections to, from, and through the Security Gateway.

Important - The "`fw ctl block on`" command immediately blocks all connections without a prompt and regardless the currently installed policy. To unblock the connections, you must either reboot the Security Gateway, or connect to the Security Gateway over a serial console (or Lights Out Management Card) and run the "`fw ctl block off`" command.

Syntax

```
fw [-d] ctl block
       off
       on
```

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.
`off`	Removes the block of all connections.
`on`	Blocks all connections.

fw ctl chain

Description

Shows the list of Firewall Chain Modules.

This list shows various inspection Chain Modules, through which the traffic passes on this Security Gateway.

The available Chain Modules depend on the configuration and enabled Software Blades.

Important - In Cluster, outputs of this command must be the same on all the Cluster Members.

Syntax

```
fw [-d] ctl chain
```

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 chain
in chain (23):
        0: -7fffffff (0000000000000000) (00000000) SecureXL inbound (sxl_in)
        1: -7ffffffe (0000000000000000) (00000000) SecureXL inbound CT (sxl_ct)
        2: -7f800000 (ffffffff8b6812b0) (ffffffff) IP Options Strip (in) (ipopt_strip)
        3: -7d000000 (ffffffff8a96ee80) (00000003) vpn multik forward in
        4: - 2000000 (ffffffff8a97d830) (00000003) vpn decrypt (vpn)
        5: - 1fffffa (ffffffff8a9533a0) (00000001) l2tp inbound (l2tp)
        6: - 1ffffff8 (ffffffff8b67f0e0) (00000001) Stateless verifications (in) (asm)
        7: - 1ffffff7 (ffffffff8b67ec00) (00000001) fw multik misc proto forwarding
        8: - 1ffffff2 (ffffffff8a982aa0) (00000003) vpn tagging inbound (tagging)
        9: - 1ffffff0 (ffffffff8a983460) (00000003) vpn decrypt verify (vpn_ver)
        10:         0 (ffffffff8b85a950) (00000001) fw VM inbound  (fw)
        11:         1 (ffffffff8a97ed70) (00000003) vpn policy inbound (vpn_pol)
        12:         2 (ffffffff8b681700) (00000001) fw SCV inbound (scv)
        13:         3 (ffffffff8a982130) (00000003) vpn before offload (vpn_in)
        14:         4 (ffffffff8b0fa5c0) (00000003) QoS inbound offload chain module
        15:         5 (ffffffff8b574730) (00000003) fw offload inbound (offload_in)
        16:        10 (ffffffff8b84c9c0) (00000001) fw post VM inbound  (post_vm)
        17:    100000 (ffffffff8b807970) (00000001) fw accounting inbound (acct)
        18:  22000000 (ffffffff8b0fbfc0) (00000003) QoS slowpath inbound chain mod (fg_sched)
        19:  7f730000 (ffffffff8b3d3aa0) (00000001) passive streaming (in) (pass_str)
        20:  7f750000 (ffffffff8b17dff0) (00000001) TCP streaming (in) (cpas)
        21:  7f800000 (ffffffff8b681260) (ffffffff) IP Options Restore (in) (ipopt_res)
        22:  7fb00000 (ffffffff8a9fe8a0) (00000001) Cluster Late Correction (ha_for)
out chain (19):
        0: -7f800000 (ffffffff8b6812b0) (ffffffff) IP Options Strip (out) (ipopt_strip)
        1: -78000000 (ffffffff8a96ee60) (00000003) vpn multik forward out
        2: - 1ffffff (ffffffff8a97fb70) (00000003) vpn nat outbound (vpn_nat)
        3: - 1fffff0 (ffffffff8b168640) (00000001) TCP streaming (out) (cpas)
        4: - 1ffff50 (ffffffff8b3d3aa0) (00000001) passive streaming (out) (pass_str)
        5: - 1ff0000 (ffffffff8a982aa0) (00000003) vpn tagging outbound (tagging)
        6: - 1f00000 (ffffffff8b67f0e0) (00000001) Stateless verifications (out) (asm)
        7:         0 (ffffffff8b85a950) (00000001) fw VM outbound (fw)
        8:        10 (ffffffff8b84c9c0) (00000001) fw post VM outbound  (post_vm)
        9:   2000000 (ffffffff8a982900) (00000003) vpn policy outbound (vpn_pol)
        10:  15000000 (ffffffff8b0fac30) (00000003) QoS outbound offload chain modul (fg_pol)
        11:  1ffffff0 (ffffffff8a951790) (00000001) l2tp outbound (l2tp)
        12:  20000000 (ffffffff8a978280) (00000003) vpn encrypt (vpn)
        13:  21000000 (ffffffff8b0fbfc0) (00000003) QoS slowpath outbound chain mod (fg_sched)
        14:  7f000000 (ffffffff8b807970) (00000001) fw accounting outbound (acct)
        15:  7f700000 (ffffffff8b17cb10) (00000001) TCP streaming post VM (cpas)
        16:  7f800000 (ffffffff8b681260) (ffffffff) IP Options Restore (out) (ipopt_res)
        17:  7f900000 (0000000000000000) (00000000) SecureXL outbound (sxl_out)
        18:  7fa00000 (0000000000000000) (00000000) SecureXL deliver (sxl_deliver)
[Expert@MyGW:0]#
```

fw ctl conn

Description

Shows the list of Firewall Connection Modules.

This list shows various inspection Connection Modules, through which the traffic passes on this Security Gateway.

The available Connection Modules depend on the configuration and enabled Software Blades.

Important - In Cluster, outputs of this command must be the same on all the Cluster Members.

Syntax

```
fw [-d] ctl 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 chain
Registered connections modules:
No. Name             Newconn          Packet           End              Reload           Dup Type Dup
Handler
Connectivity level 0:
1:  Accounting       1:  Accounting       0000000000000000 0000000000000000 FFFFFFFF8B8395A0
0000000000000000 Special  FFFFFFFF8B831720
2:  Authentication   2:  Authentication   FFFFFFFF8B3150A0 0000000000000000 0000000000000000
0000000000000000 Special  FFFFFFFF8B34FCC0
8:  NAT              8:  NAT              0000000000000000 0000000000000000 FFFFFFFF8B6D1AF0
0000000000000000 Special  FFFFFFFF8B6B8410
9:  RTM              9:  RTM              0000000000000000 0000000000000000 0000000000000000
0000000000000000 None
10:  RTM2             10:  RTM2             0000000000000000 0000000000000000 FFFFFFFF8B014970
0000000000000000 None
11:  SPII             11:  SPII             FFFFFFFF8B412060 0000000000000000 FFFFFFFF8B41AF40
FFFFFFFF8B4016A0 None
13:  VPN              13:  VPN              FFFFFFFF8A965440 0000000000000000 FFFFFFFF8AA4CC40
0000000000000000 Special  FFFFFFFF8AA60490
Connectivity level 1:
13:  VPN              13:  VPN              0000000000000000 0000000000000000 0000000000000000
0000000000000000 None
[Expert@MyGW:0]#
```

fw ctl conntab

Description

Shows formatted list of current connections from the **Connections** kernel table (ID 8158).

Use this command if you want to see the simplified information about the current connections.

Best Practices:

- Use the "`fw ctl conntab`" command to see the simplified information about the current connections.
- Use the "`fw tab -t connections -f`" command (*"fw tab" on page 384*) to see the detailed (and more technical) information about the current connections.

Syntax

Important - You can specify many parameters at the same time.

```
fw [-d] ctl conntab
      {-h | -help}
      -sip=<Source IP Address in Decimal Format>
      -sport=<Port Number in Decimal Format>
      -dip=<Destination IP Address>
      -dport=<Port Number in Decimal Format>
      -proto=<Protocol Name>
      -service=<Name of Service>
      -rule=<Rule Number in Decimal Format>
```

Parameters

Parameter	Description
`{-h | -help}`	Shows the built-in usage.
`-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.

Parameter	Description
`-sip=<Source IP Address in Decimal Format>`	Filters the output by the specified Source IP address.
`-sport=<Port Number in Decimal Format>`	Filters the output by the specified Source Port number. See *IANA Service Name and Port Number Registry*.
`-dip=<Destination IP Address in Decimal Format>`	Filters the output by the specified Destination IP address.
`-dport=<Port Number in Decimal Format>`	Filters the output by the specified Destination Port number. See *IANA Service Name and Port Number Registry*.
`-proto=<Protocol Name>`	Filters the output by the specified Protocol name. For example: ■ TCP ■ UDP ■ ICMP See *IANA Protocol Numbers*.
`-service=<Name of Service>`	See the names of Services in SmartConsole, or in the output of this command.
`-rule=<Rule Number in Decimal Format>`	See your Rule Base in SmartConsole, or in the output of the command.

Examples

Example 1 - Default output

```
[Expert@MyGW:0]# fw ctl conntab
<(inbound, src=[192.168.204.1,54201], dest=[192.168.204.40,22], TCP); 3593/3600, rule=2, tcp state=TCP_ESTABLISHED,
service=ssh(481), Ifncin=1, Ifncout=1, conn modules: Authentication, FG-1>
<(outbound, src=[192.168.204.40,59249], dest=[192.168.204.1,53], UDP); 20/40, rule=0, service=domain-udp(335), Ifnsout=1,
conn modules: Authentication, FG-1>
<(outbound, src=[192.168.204.40,37892], dest=[192.168.204.1,53], UDP); 20/40, rule=0, service=domain-udp(335), Ifnsin=1,
Ifnsout=1, conn modules: Authentication, FG-1>
[Expert@MyGW:0]#
```

Example 2 - Filter by a destination port

```
[Expert@MyGW:0]# fw ctl conntab -dport=22
<(inbound, src=[192.168.204.1,54201], dest=[192.168.204.40,22], TCP); 3594/3600, rule=2, tcp state=TCP_ESTABLISHED,
service=ssh(481), Ifncin=1, Ifncout=1, conn modules: Authentication, FG-1>
[Expert@MyGW:0]#
```

Example 3 - Filter by a destination port

```
[Expert@MyGW:0]# fw ctl conntab -dport=53
<(outbound, src=[192.168.204.40,33585], dest=[192.168.204.1,53], UDP); 39/40, rule=0, service=domain-udp(335), Ifnsout=1,
conn modules: Authentication, FG-1>
<(outbound, src=[192.168.204.40,56661], dest=[192.168.204.1,53], UDP); 39/40, rule=0, service=domain-udp(335), Ifnsin=1,
Ifnsout=1, conn modules: Authentication, FG-1>
[Expert@MyGW:0]#
```

Example 4 - Filter by a source port

```
[Expert@MyGW:0]# fw ctl conntab -sport=54201
<(inbound, src=[192.168.204.1,54201], dest=[192.168.204.40,22], TCP); 3600/3600, rule=2, tcp state=TCP_ESTABLISHED,
service=ssh(481), Ifncin=1, Ifncout=1, conn modules: Authentication, FG-1>
[Expert@MyGW:0]#
```

Example 5 - Filter by a protocol

```
[Expert@MyGW:0]# fw ctl conntab -proto=UDP
<(outbound, src=[192.168.204.40,44966], dest=[192.168.204.1,53], UDP); 37/40, rule=0, service=domain-udp(335), Ifnsin=1,
Ifnsout=1, conn modules: Authentication, FG-1>
[Expert@MyGW:0]#
```

Example 6 - Filter by a protocol

```
[Expert@MyGW:0]# fw ctl conntab -proto=TCP
<(inbound, src=[192.168.204.1,54201], dest=[192.168.204.40,22], TCP); 3596/3600, rule=2, tcp state=TCP_ESTABLISHED,
service=ssh(481), Ifncin=1, Ifncout=1, conn modules: Authentication, FG-1>
[Expert@MyGW:0]#
```

Example 7 - Filter by a service

```
[Expert@MyGW:0]# fw ctl conntab -service=domain-udp
<(outbound, src=[192.168.204.40,44966], dest=[192.168.204.1,53], UDP); 35/40, rule=0, service=domain-udp(335), Ifnsin=1,
Ifnsout=1, conn modules: Authentication, FG-1>
[Expert@MyGW:0]#
```

Example 8 - Filter by a rule number

```
[Expert@MyGW:0]# fw ctl conntab -rule=2
<(inbound, src=[192.168.204.1,54201], dest=[192.168.204.40,22], TCP); 3597/3600, rule=2, tcp state=TCP_ESTABLISHED,
service=ssh(481), Ifncin=1, Ifncout=1, conn modules: Authentication, FG-1>
[Expert@MyGW:0]#
```

Example 9 - Filter by a destination IP address, destination port, protocol, and service

```
[Expert@MyGW:0]# fw ctl conntab -dip=192.168.204.40 -dport=22 -proto=TCP -service=ssh
<(inbound, src=[192.168.204.1,54201], dest=[192.168.204.40,22], TCP); 3599/3600, rule=2, tcp state=TCP_ESTABLISHED,
service=ssh(481), Ifncin=1, Ifncout=1, conn modules: Authentication, FG-1>
[Expert@MyGW:0]#
```

Example 10 - Formatted detailed output from the Connections table (for comparison)

```
[Expert@MyGW:0]# fw tab -t connections -f


Formatting table's data - this might take a while...


localhost:

 Date: Sep 10, 2018

11:30:56 5 N/A N/A 192.168.204.40 > N/A LogId: <max_null>; ContextNum: <max_null>; OriginSicName: cn=cp_mgmt,o=MyGW..44jkyv;
: (+)==================================(+); Table_Name: connections; : (+); Attributes: dynamic, id 8158, attributes: keep,
sync, aggressive aging, kbufs 21 22 23 24 25 26 27 28 29 30 31 32 33 34, expires 25, refresh, , hashsize 2097152, unlimited;
LastUpdateTime: 10Sep2018 11:30:56; ProductName: VPN-1 & FireWall-1; ProductFamily: Network;


 11:30:56 5 N/A N/A 192.168.204.40 > N/A LogId: <max_null>; ContextNum: <max_null>; OriginSicName: cn=cp_mgmt,o=MyGW..44jkyv;
: ----------------------------------(+); Direction: 1; Source: 192.168.204.40; SPort: 54201; Dest: 192.168.204.1; DPort: 53;
Protocol: udp; CPTFMT_sep: ;; Type: 131073; Rule: 0; Timeout: 335; Handler: 0; Ifncin: -1; Ifncout: -1; Ifnsin: -1; Ifnsout:
1; Bits: 0000780000000000; Expires: 23/40; LastUpdateTime: 10Sep2018 11:30:56; ProductName: VPN-1 & FireWall-1;
ProductFamily: Network;


 11:30:56 5 N/A N/A 192.168.204.40 > N/A LogId: <max_null>; ContextNum: <max_null>; OriginSicName: cn=cp_mgmt,o=MyGW..44jkyv;
: ----------------------------------(+); Direction: 0; Source: 192.168.204.1; SPort: 53; Dest: 192.168.204.40; DPort: 54201;
Protocol: udp; CPTFMT_sep_1: ->; Direction_1: 1; Source_1: 192.168.204.40; SPort_1: 54201; Dest_1: 192.168.204.1; DPort_1:
53; Protocol_1: udp; FW_symval: 2054; LastUpdateTime: 10Sep2018 11:30:56; ProductName: VPN-1 & FireWall-1; ProductFamily:
Network;


 11:30:56 5 N/A N/A 192.168.204.40 > N/A LogId: <max_null>; ContextNum: <max_null>; OriginSicName: cn=cp_mgmt,o=MyGW..44jkyv;
: ----------------------------------(+); Direction: 1; Source: 192.168.204.40; SPort: 22; Dest: 192.168.204.1; DPort: 54201;
Protocol: tcp; CPTFMT_sep_1: ->; Direction_2: 0; Source_2: 192.168.204.1; SPort_2: 54201; Dest_2: 192.168.204.40; DPort_2:
22; Protocol_2: tcp; FW_symval: 2053; LastUpdateTime: 10Sep2018 11:30:56; ProductName: VPN-1 & FireWall-1; ProductFamily:
Network;


 11:30:56 5 N/A N/A 192.168.204.40 > N/A LogId: <max_null>; ContextNum: <max_null>; OriginSicName: cn=cp_mgmt,o=MyGW..44jkyv;
: ----------------------------------(+); Direction: 0; Source: 192.168.204.1; SPort: 54201; Dest: 192.168.204.40; DPort: 22;
Protocol: tcp; CPTFMT_sep: ;; Type: 114689; Rule: 2; Timeout: 481; Handler: 0; Ifncin: 1; Ifncout: 1; Ifnsin: -1; Ifnsout: -
1; Bits: 02007800000f9000; Expires: 3596/3600; LastUpdateTime: 10Sep2018 11:30:56; ProductName: VPN-1 & FireWall-1;
ProductFamily: Network;


 11:30:56 5 N/A N/A 192.168.204.40 > N/A LogId: <max_null>; ContextNum: <max_null>; OriginSicName: cn=cp_mgmt,o=MyGW..44jkyv;
: ----------------------------------(+); Direction: 0; Source: 192.168.204.1; SPort: 53; Dest: 192.168.204.40; DPort: 44966;
Protocol: udp; CPTFMT_sep_1: ->; Direction_1: 1; Source_1: 192.168.204.40; SPort_1: 44966; Dest_1: 192.168.204.1; DPort_1:
53; Protocol_1: udp; FW_symval: 2054; LastUpdateTime: 10Sep2018 11:30:56; ProductName: VPN-1 & FireWall-1; ProductFamily:
Network;


 11:30:56 5 N/A N/A 192.168.204.40 > N/A LogId: <max_null>; ContextNum: <max_null>; OriginSicName: cn=cp_mgmt,o=MyGW..44jkyv;
: ----------------------------------(+); Direction: 1; Source: 192.168.204.40; SPort: 44966; Dest: 192.168.204.1; DPort: 53;
Protocol: udp; CPTFMT_sep: ;; Type: 131073; Rule: 0; Timeout: 335; Handler: 0; Ifncin: -1; Ifncout: -1; Ifnsin: 1; Ifnsout:
1; Bits: 0000780000000000; Expires: 23/40; LastUpdateTime: 10Sep2018 11:30:56; ProductName: VPN-1 & FireWall-1;
ProductFamily: Network;


[Expert@MyGW:0]#
```

fw ctl cpasstat

Description

Generates statistics report about Check Point Active Streaming (CPAS).

Syntax

```
fw [-d] ctl cpasstat [-r]
```

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.
`-r`	Resets the counters.

'fw ctl debug' and 'fw ctl kdebug'

Description

These commands generate kernel debug messages from Check Point Firewall kernel to a debug buffer.

For more information, see the *R80.40 Next Generation Security Gateway Guide* - Chapter *Kernel Debug on Security Gateway*.

fw ctl dlpkstat

Description

Generates statistics report about Data Loss Prevention, inspected HTTP requests, and Identity Awareness Captive Portal.

This report contains these statistics:

Category	Information
DLP Kernel Statistics Information	Emails and HTTP requests
User Mode Responses Statistics	Emails and HTTP requests
Identity Awareness - Captive Portal	HTTP requests redirected to the Captive Portal
Identity Awareness - Fetch Users Statistics	Synchronous and asynchronous Identity Awareness queries

Best Practice - This report is very useful when you:

- Debug problems with HTTP protocol that occur under traffic stress.
- Examine the traffic shape (for example, to know how many HTTP "POST" and HTTP "GET" requests pass through the Security Gateway).

Syntax

```
fw [-d] ctl dlpkstat [-r]
```

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.
`-r`	Resets the counters.

fw ctl get

Description

Shows the current value of the specified kernel parameter.

Important:

- In Cluster, you must configure all the Cluster Members in the same way.
- In VSX Gateway, the configured values of kernel parameters apply to all existing Virtual Systems and Virtual Routers.

Notes:

- Kernel parameters let you change the advanced behavior of your Security Gateway.
- There are two types of kernel parameters - *integer* and *string*.
- 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`
  - `$FWDIR/boot/modules/fw_kern_64_3_10_64.o`
  - `$FWDIR/boot/modules/fw_kern_64_3_10_64_v6.o`
  - `$PPKDIR/boot/modules/sim_kern_64.o`
  - `$PPKDIR/boot/modules/sim_kern_64_v6.o`
  - `$PPKDIR/boot/modules/sim_kern_64_3_10_64.o`
  - `$PPKDIR/boot/modules/sim_kern_64_3_10_64_v6.o`
- Refer to the related command *"fw ctl set" on page 283*.
- Refer to the related article sk33156: Creating a file with all the kernel parameters and their values

Syntax

```
fw [-d] ctl get
      int <Name of Integer Kernel Parameter> [-a]
      str <Name of String Kernel Parameter> [-a]
```

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.
<Name of Integer Kernel Parameter>	Specifies the name of the integer kernel parameter.
<Name of String Kernel Parameter>	Specifies the name of the string kernel parameter.
`-a`	Specifies to search for this kernel parameter in this order: 1. In `$FWDIR/modules/fw_*.o` 2. In `$PPKDIR/modules/sim_*.o`

Example for an integer kernel parameter

```
[Expert@MyGW:0]# fw ctl get int fw_kdprintf_limit -a
FW:
fw_kdprintf_limit = 100
PPAK 0: fw_kdprintf_limit = 10
[Expert@MyGW:0]#
```

Example for a string kernel parameter

```
[Expert@MyGW:0]# fw ctl get str fileapp_default_encoding_charset -a
FW:
fileapp_default_encoding_charset = 'UTF-8'
PPAK 0: Get failed.
[Expert@MyGW:0]#
```

fw ctl iflist

Description

Shows the list with this information:

- The name of interfaces, to which the Check Point Firewall kernel attached.
- The internal numbers of the interfaces in the Check Point Firewall kernel.

Notes:

- This list shows all detected interfaces, even if there are no IP addresses assigned on them.
- You use this list when you analyze a kernel debug, which shows only the internal numbers of the interfaces (for example, `ifn=2`).
- Related *"cpstat" on page 208* commands:
  - `cpstat -f ifconfig os`
  - `cpstat -f interfaces fw`

Syntax

```
fw [-d] ctl iflist
```

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 iflist
fw ctl iflist
1 : eth0
2 : eth1
3 : eth2
4 : eth3
5 : eth4
6 : eth5
7 : eth6
8 : eth7
[Expert@MyGW:0]#
```

fw ctl install

Description

Tells the operating system to start passing packets to Firewall.

This command runs automatically when the Security Gateway or an administrator runs the *"cpstart" on page 207* command.

Warning

If you run the *"fw ctl uninstall" on page 287* command and then the "`fw ctl install`" command, it does *not* restore the Security Policy.

You must run one of these commands: *"fw fetch" on page 290*, or *"cpstart" on page 207*.

Syntax

```
fw [-d] ctl install
```

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.

fw ctl leak

Description

Generates leak detection report. This report is for Check Point use only.

Important - This command save the report into the active `/var/log/messages` file and the `dmesg` buffer.

Syntax

```
fw [-d] ctl leak
      {-h | -help}
      [{-a | -A}] [-t <Internal Object Type>] [-o <Internal Object
ID>]
            [-d] [-l] [-p]
            [-s]
```

Parameters

Parameter	Description
`fw -d ctl leak ...`	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.
`{-h | -help}`	Shows the built-in help.
`-a`	Specifies to perform leak detection for potential leaks. This parameter is mutually exclusive with the parameter "`-A`".
`-A`	Specifies to perform leak detection for all leaks. This parameter is mutually exclusive with the parameter "`-a`".
`-d`	Dumps object data. This parameter is mutually exclusive with the parameter "`-s`".
`-l`	Prints the action log. This parameter is mutually exclusive with the parameter "`-s`".

Parameter	Description
`-o <Internal Object ID>`	Specifies to perform leak detection for the specified internal object ID.
`-p`	Purges the internal objects from the lists. This parameter is mutually exclusive with the parameter "`-s`".
`-s`	Shows summary only. This parameter is mutually exclusive with the parameters "`-d`", "`-l`", and "`-p`".
`-t <Internal Object Type>`	Specifies the internal object types, for which to perform leak detection. Available internal object types are: ■ `chain` ■ `connh` ■ `cookie` ■ `kbuf` ■ `num` If you do not specify the internal object type explicitly, the command performs leak detection for all internal object types.

Procedure

Step	Description
1	Connect to the command line on the Security Gateway.
2	Log in to the Expert mode.
3	Back up the current */var/log/messages* file: `[Expert@GW_HostName:0]# cp -v /var/log/messages{,_BKP}`
4	Delete the information from the current */var/log/messages* file: `[Expert@GW_HostName:0]# echo '' > /var/log/messages`
5	Delete the information from the current *dmesg* buffer: `[Expert@GW_HostName:0]# dmesg -c`

Step	Description
6	Generate the leak detection report (see the **Syntax** section above): `[Expert@GW_HostName:0]# fw [-d] ctl leak <options>`

7	Make sure the command generated the leak detection report:
	`[Expert@GW_HostName:0]# dmesg`
	`[Expert@GW_HostName:0]# cat /var/log/messages`

8	Collect the leak detection report: `[Expert@GW_HostName:0]# cp -v /var/log/messages{,_LEAK_DETECTION}`
9	Analyze the leak detection report: `/var/log/messages_LEAK_DETECTION`

Example

```
[Expert@MyGW:0]# cp -v /var/log/messages{,_BKP}
`/var/log/messages' -> `/var/log/messages_BKP'
[Expert@MyGW:0]#
[Expert@MyGW:0]# echo '' > /var/log/messages
[Expert@MyGW:0]#
[Expert@MyGW:0]# dmesg -c
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl leak -s
[Expert@MyGW:0]#
[Expert@MyGW:0]# dmesg
[fw4_0];fwleak_report: type chain - 0 objects
[fw4_0];fwleak_report: type cookie - 0 objects
[fw4_0];fwleak_report: type kbuf - 0 objects
[fw4_0];fwleak_report: type connh - 0 objects
[fw4_1];fwleak_report: type chain - 0 objects
[fw4_1];fwleak_report: type cookie - 0 objects
[fw4_1];fwleak_report: type kbuf - 0 objects
[fw4_1];fwleak_report: type connh - 0 objects
[fw4_2];fwleak_report: type chain - 0 objects
[fw4_2];fwleak_report: type cookie - 0 objects
[fw4_2];fwleak_report: type kbuf - 0 objects
[fw4_2];fwleak_report: type connh - 0 objects
[Expert@MyGW:0]#
[Expert@MyGW:0]# cat /var/log/messages
Sep 12 16:09:50 2019 MyGW kernel: [fw4_0];fwleak_report: type chain - 0 objects
Sep 12 16:09:50 2019 MyGW kernel: [fw4_0];fwleak_report: type cookie - 0 objects
Sep 12 16:09:50 2019 MyGW kernel: [fw4_0];fwleak_report: type kbuf - 0 objects
Sep 12 16:09:50 2019 MyGW kernel: [fw4_0];fwleak_report: type connh - 0 objects
Sep 12 16:09:50 2019 MyGW kernel: [fw4_1];fwleak_report: type chain - 0 objects
Sep 12 16:09:50 2019 MyGW kernel: [fw4_1];fwleak_report: type cookie - 0 objects
Sep 12 16:09:50 2019 MyGW kernel: [fw4_1];fwleak_report: type kbuf - 0 objects
Sep 12 16:09:50 2019 MyGW kernel: [fw4_1];fwleak_report: type connh - 0 objects
Sep 12 16:09:50 2019 MyGW kernel: [fw4_2];fwleak_report: type chain - 0 objects
Sep 12 16:09:50 2019 MyGW kernel: [fw4_2];fwleak_report: type cookie - 0 objects
Sep 12 16:09:50 2019 MyGW kernel: [fw4_2];fwleak_report: type kbuf - 0 objects
Sep 12 16:09:50 2019 MyGW kernel: [fw4_2];fwleak_report: type connh - 0 objects
[Expert@MyGW:0]
[Expert@MyGW:0]# cp -v /var/log/messages{,_LEAK_DETECTION}
`/var/log/messages' -> `/var/log/messages_LEAK_DETECTION'
[Expert@MyGW:0]#
```

fw ctl pstat

Description

Shows Security Gateway various internal statistics:

- System Capacity Summary
- Hash kernel memory (hmem) statistics
- System kernel memory (smem) statistics
- Kernel memory (kmem) statistics
- Cookies
- Connections
- Fragments
- NAT
- Handles

Syntax

Important - You can specify many parameters at the same time.

```
fw [-d] ctl pstat [-c] [-h] [-k] [-l] [-m] [-o] [-s] [-v {4 | 6}]
```

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.
`-c`	Shows detailed CoreXL Dispatcher statistics: ■ `fwmultik_global_stats` splits for each CoreXL Firewall instance. ■ `fwmultik_gconn_stats` for each CPU. ■ `fwmultik_stats` for each CPU.
`-h`	Shows additional Hash kernel memory (hmem) statistics.
`-k`	Shows additional Kernel memory (kmem) statistics.

Parameter	Description
`-l`	Shows Handles statistics.
`-m`	Shows general CoreXL Dispatcher statistics.
`-o`	Shows additional Cookies statistics.
`-s`	Shows additional System kernel memory (smem) statistics.
`-v 4` `-v 6`	Shows statistics for IPv4 (`-v 4`) traffic only, or for IPv6 (`-v 4`) traffic only. Default is to show statistics for both IPv4 and IPv6 traffic.

Examples

Example 1 - fw ctl pstat

```
[Expert@MyGW:0]# fw ctl pstat

System Capacity Summary:
  Memory used: 3% (265 MB out of 7117 MB) - below watermark
  Concurrent Connections: Not Available
  Aggressive Aging is enabled, not active

Hash kernel memory (hmem) statistics:
  Total memory allocated: 742391808 bytes in 181248 (4096 bytes) blocks using 1 pool
  Total memory bytes  used:        0   unused: 742391808 (100.00%)   peak: 68247020
  Total memory blocks used:        0   unused:   181248 (100%)   peak:    17227
  Allocations: 2193027 alloc, 0 failed alloc, 2154121 free

System kernel memory (smem) statistics:
  Total memory  bytes  used: 913975068   peak: 1165010872
  Total memory bytes wasted:  7883999
    Blocking  memory  bytes   used:  4896272   peak:  6916084
    Non-Blocking memory bytes used: 909078796   peak: 1158094788
  Allocations: 13217 alloc, 0 failed alloc, 10027 free, 0 failed free
  vmalloc bytes  used: 908585924 expensive: no

Kernel memory (kmem) statistics:
  Total memory  bytes  used: 185761552   peak: 486615148
  Allocations: 2204456 alloc, 0 failed alloc
               2162587 free, 0 failed free
  External Allocations: 0 for packets, 7303643 for SXL

Cookies:
        91808 total, 0 alloc, 0 free,
        2 dup, 91808 get, 0 put,
        182258 len, 909 cached len, 0 chain alloc,
        0 chain free

Connections:
        0 total, 0 TCP, 0 UDP, 0 ICMP,
        0 other, 0 anticipated, 0 recovered, -3 concurrent,
        0 peak concurrent

Fragments:
        0 fragments, 0 packets, 0 expired, 0 short,
        0 large, 0 duplicates, 0 failures

NAT:
        0/0 forw, 0/0 bckw, 0 tcpudp,
        0 icmp, 0-0 alloc

Sync: Run "cphaprob syncstat" for cluster sync statistics.

[Expert@MyGW:0]#
```

fw ctl set

Description

Configures the specified value for the specified kernel parameter.

Important:

- In Cluster, you must configure all the Cluster Members in the same way.
- In VSX Gateway, the configured values of kernel parameters apply to all existing Virtual Systems and Virtual Routers.
- The configuration made with this command does *not* survive reboot.

  To make this configuration permanent, you must edit one of the applicable configuration files:

  - `$FWDIR/boot/modules/fwkern.conf`
  - `$FWDIR/boot/modules/vpnkern.conf`
  - `$PPKDIR/conf/simkern.conf`.

  For more information, see sk26202.

Notes:

- Kernel parameters let you change the advanced behavior of your Security Gateway.
- There are two types of kernel parameters - *integer* and *string*.
- 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`
  - `$FWDIR/boot/modules/fw_kern_64_3_10_64.o`
  - `$FWDIR/boot/modules/fw_kern_64_3_10_64_v6.o`
  - `$PPKDIR/boot/modules/sim_kern_64.o`
  - `$PPKDIR/boot/modules/sim_kern_64_v6.o`
  - `$PPKDIR/boot/modules/sim_kern_64_3_10_64.o`
  - `$PPKDIR/boot/modules/sim_kern_64_3_10_64_v6.o`
- Refer to the related command *"fw ctl get" on page 272*.
- Refer to the related article sk33156: Creating a file with all the kernel parameters and their values

Syntax

```
fw [-d] ctl set
      int <Name of Integer Kernel Parameter> <Integer Value>
      str <Name of String Kernel Parameter> '<String Value>'
```

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.
<Name of Integer Kernel Parameter>	Specifies the name of the integer kernel parameter.
<Integer Value>	Specifies the integer value for the integer kernel parameter.
<Name of String Kernel Parameter>	Specifies the name of the string kernel parameter.
'*<String Value>*'	Specifies the string value for the string kernel parameter.

Example for an integer kernel parameter

```
[Expert@MyGW:0]# fw ctl get int fw_kdprintf_limit
fw_kdprintf_limit = 100
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl set int fw_kdprintf_limit 50
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl get int fw_kdprintf_limit
fw_kdprintf_limit = 50
[Expert@MyGW:0]#
```

Example for a string kernel parameter

```
[Expert@MyGW:0]# fw ctl set str icap_unwrap_append_header_str '__print__'
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl get str icap_unwrap_append_header_str
icap_unwrap_append_header_str = '__print__'
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl set str icap_unwrap_append_header_str ''
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw ctl get str icap_unwrap_append_header_str
icap_unwrap_append_header_str = ''
[Expert@MyGW:0]#
```

fw ctl tcpstrstat

Description

Generates statistics report about TCP Streaming.

Syntax

```
fw [-d] ctl tcpstrstat
       [-p]
       [-r]
```

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.
-p	Shows verbose statistics.
-r	Resets the counters.

Example 1 - Default output

```
[Expert@MyGW:0]# fw ctl tcpstrstat

General Counters:
=================
 Connections:
   Concurrent num of connections .............  0
   Concurrent num of si connections ..........  0
 Packets:
   Total num of packets ......................  2567
   Total packets in bytes ....................  202394
   Concurrent num of async packets ...........  0
 Memory:
   Allocated memory in bytes .................  0
   Referenced skbuffs num ....................  0
   Referenced skbuffs size in bytes ..........  0
   External packet references.................  0
   Allocated memory per connection ...........  0
 Rejected packets/connections:
   Total num of rejected packets .............  0
 Dropped packets/connections:
   Total num of dropped packets ..............  0
 Stripped/Truncated packets:
   Total num of stripped packets .............  0
   Total num of truncated packets ............  0
 Paused packets:
   Total num of c2s|s2c paused packets .......  0 | 0
   Concurrent num of UDP held packets ........  0

Applications Counters:
======================
 Application Name: ASPII_MT
  Connections:
   Total num of connections ..................  954
   Concurrent num of connections .............  0
   Total num of c2s|s2c connections ..........  954 | 954
   Concurrent num of c2s|s2c connections .....  0 | 0
  Packets:
   Total num of c2s|s2c data packets .........  2567 | 0
   Total c2s|s2c data packets in bytes .......  130518 | 0

FastForward Counters:
=====================
  FF connection:
   Total num of c2s|s2c FFconns ..............  0 | 0
   Total num of c2s|s2c saved packets ........  0 | 0
   Total num of c2s|s2c bytes requests .......  0 | 0
   Total num of c2s|s2c saved bytes ..........  0 | 0

[Expert@MyGW:0]#
```

fw ctl uninstall

Description

1. Tells the operating system to stop passing packets to Firewall.
2. Unloads the current Security Policy.
3. Unloads the current Firewall Chain Modules (see *"fw ctl chain" on page 261*).
4. Unloads the current Firewall Connection Modules except for `RTM` (see *"fw ctl conn" on page 263*).

Warnings

1. If you run the "`fw ctl uninstall`" command, the networks behind the Security Gateway become unprotected.
2. If you run the "`fw ctl uninstall`" command and then the *"fw ctl install" on page 275* command, it does not restore the Security Policy.

   You must run one of these commands: *"fw fetch" on page 290*, or *"cpstart" on page 207*.

Syntax

```
fw [-d] ctl uninstall
```

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.

fw defaultgen

Description

Manually generates the Default Filter policy files.

Refer to these related commands:

- *"comp_init_policy" on page 170*
- *"control_bootsec" on page 173*
- *"fwboot default" on page 415*
- *"fwboot bootconf" on page 402*

Syntax

```
fw [-d] defaultgen
```

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.

`defaultgen`	Generates the Default Filter policy files:
	■ For IPv4 traffic:
	`$FWDIR/state/default.bin`
	■ For IPv6 traffic:
	`$FWDIR/state/default.bin6`
	If the Default Filter policy file already exists, the command creates a backup copy (`$FWDIR/state/default.bin.bak` and `$FWDIR/state/default.bin6.bak`).

Example

```
[Expert@MyGW:0]# fw defaultgen
Generating default filter
defaultfilter:
Compiled OK.
defaultfilter:
Compiled OK.
 Backing up default.bin as default.bin.bak
 hostaddr(MyGW) failed
 Backing up default.bin6 as default.bin6.bak
[Expert@MyGW:0]#
```

fw fetch

Description

Fetches the Security Policy from the specified host and installs it to the kernel.

Syntax

- To fetch the policy from the Management Server:

```
fw [-d] fetch -f [-i] [-n] [-r]
```

- To fetch the policy from a peer Cluster Member, and, if it fails, then from the Management Server:

```
fw [-d] fetch -f -c [-i] [-n] [-r]
```

- To fetch the policy from the specified Check Point computer(s):

```
fw [-d] fetch [-i] [-n] [-r] <Master 1> [<Master 2> ...]
```

- To fetch the policy stored locally on the Security Gateway:

```
fw [-d] fetch local [-nu]
```

```
fw [-d] fetch localhost [-nu]
```

- To fetch the policy stored locally on the Security Gateway in the specified directory:

```
fw [-d] fetchlocal -d <Full Path to Directory>
```

Parameters

Parameter	Description
`fw -d fetch...`	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.
`-c`	Specifies that you fetch the policy from a peer Cluster Member.  **Notes:** ■ Must also use the "`-f`" parameter. ■ Works only in cluster.
`-f`	Specifies that you fetch the policy from a Management Server listed in the `$FWDIR/conf/masters` file.

Parameter	Description
`-i`	On a Security Gateway with dynamically assigned IP address (DAIP), specifies to ignore the SIC name and object name.
`-n`	Specifies not to load the fetched policy, if it is the same as the policy already located on the Security Gateway.
`-nu`	Specifies not to update the currently installed policy.
`-r`	On a Cluster Member, specifies to ignore this option in SmartConsole **Install Policy** window: **For gateway clusters, if installation on a cluster member fails, do not install on that cluster**  **Best Practice** - Use this parameter if a peer Cluster Member is **Down**.
<Master 1> [*<Master 2>* ...]	Specifies the Check Point computer(s), from which to fetch the policy. You can fetch the policy from the Management Server, or a peer Cluster Member.  **Notes:** ■ If you fetch the policy from the Management Server, you can enter one of these: • The main IP address of the Management Server object. • The object name of the Management Server. • The hostname that the Security Gateway resolves to the main IP address of the Management Server. ■ If you fetch the policy from a peer Cluster Member, you can enter one of these: • The main IP address of the Cluster Member object. • The IP address of the Sync interface on the Cluster Member. ■ If the fetch from the first specified *<Master>* fails, the Security Gateway fetches the policy from the second specified *<Master>* , and so on. If the Security Gateway fails to connect to each specified *<Masters>*, the Security Gateway fetches the policy from the `localhost`. ■ If you do not specify the *<Masters>* explicitly, the Security Gateway fetches the policy from the `localhost`.
`-d` *<Full Path to Directory>*	Specifies the local directory on the Security Gateway, from which to fetch the policy files.

fw fetchlogs

Description

Fetches the specified Security log files (`$FWDIR/log/*.log*`) or Audit log files (`$FWDIR/log/*.adtlog*`) from the specified Check Point computer.

Syntax

```
fw [-d] fetchlogs [-f <Name of Log File 1>] [-f <Name of Log File 2>]... [-f <Name of Log File N>] <Target>
```

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.
`-f <Name of Log File N>`	Specifies the name of the log file to fetch. Need to specify name only. **Notes:** ■ If you do not specify the log file name explicitly, the command transfers all Security log files (`$FWDIR/log/*.log*`) and all Audit log files (`$FWDIR/log/*.adtlog*`). ■ The specified log file name can include wildcards * and ? (for example, `2017-0?-*.log`). If you enter a wildcard, you must enclose it in double quotes or single quotes. ■ You can specify multiple log files in one command. You must use the `-f` parameter for each log file name pattern. ■ This command also transfers the applicable log pointer files.
`<Target>`	Specifies the remote Check Point computer, with which this local Check Point computer has established SIC trust. ■ If you run this command on a Security Management Server or Domain Management Server, then `<Target>` is the applicable object's name or main IP address of the Check Point Computer as configured in SmartConsole. ■ If you run this command on a Security Gateway or Cluster Member, then `<Target>` is the main IP address of the applicable object as configured in SmartConsole.

Notes:

- This command moves the specified log files from the `$FWDIR/log/` directory on the specified Check Point computer. Meaning, it deletes the specified log files on the specified Check Point computer after it copies them successfully.
- This command moves the specified log files to the `$FWDIR/log/` directory on the local Check Point computer, on which you run this command.
- This command cannot fetch the *active* log files `$FWDIR/log/fw.log` or `$FWDIR/log/fw.adtlog`.

  To fetch these active log files:

  1. Perform log switch on the applicable Check Point computer:

     ```
     fw logswitch [-audit] [-h <IP Address or Hostname>]
     ```

  2. Fetch the rotated log file from the applicable Check Point computer:

     ```
     fw fetchlogs -f <Log File Name> <IP Address or Hostname>
     ```

- This command renames the log files it fetched from the specified Check Point computer. The new log file name is the concatenation of the Check Point computer's name (as configured in SmartConsole), two underscore (_) characters, and the original log file name (for example: `MyGW__2019-06-01_000000.log`).

Example - Fetching log files from a Management Server

```
[Expert@HostName:0]# fw lslogs MyGW
     Size Log file name
        23KB 2019-05-16_000000.log
         9KB 2019-05-17_000000.log
        11KB 2019-05-18_000000.log
      5796KB 2019-06-01_000000.log
      4610KB fw.log
[Expert@HostName:0]#

[Expert@HostName:0]# fw fetchlogs -f 2019-06-01_000000 MyGW
File fetching in process. It may take some time...
File MyGW__2019-06-01_000000.log was fetched successfully
[Expert@HostName:0]#

[Expert@HostName:0]# ls $FWDIR/log/MyGW*
/opt/CPsuite-R80.40/fw1/log/MyGW__2019-06-01_000000.log
/opt/CPsuite-R80.40/fw1/log/MyGW__2019-06-01_000000.logaccount_ptr
/opt/CPsuite-R80.40/fw1/log/MyGW__2019-06-01_000000.loginitial_ptr
/opt/CPsuite-R80.40/fw1/log/MyGW__2019-06-01_000000.logptr
[Expert@HostName:0]#

[Expert@HostName:0]# fw lslogs MyGW
     Size Log file name
        23KB 2019-05-16_000000.log
         9KB 2019-05-17_000000.log
        11KB 2019-05-18_000000.log
      4610KB fw.log
[Expert@HostName:0]#
```

fw getifs

Description

Shows the list with this information:

- The name of interfaces, to which the Check Point Firewall kernel attached.
- The IP addresses assigned to the interfaces.

Notes:

- This list shows only interfaces that have IP addresses assigned on them.
- Related *"cpstat" on page 208* commands:
  - `cpstat -f ifconfig os`
  - `cpstat -f interfaces fw`

Syntax

```
fw [-d] getifs
```

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 getifs
localhost eth0 192.168.30.40 255.255.255.0
localhost eth1 172.30.60.80 255.255.255.0
[Expert@MyGW:0]#
```

fw hastat

Description

Shows information about Check Point computers in High Availability configuration and their states.

Syntax

```
fw hastat [<Target1>] [<Target2>] ... [<TargetN>]
```

Parameters

Parameter	Description
<Target1> *<Target2>* ... *<TargetN>*	Specifies the Check Point computers to query. If you run this command on the Management Server, you can enter the applicable IP address, or the resolvable HostName of the managed Security Gateway or Cluster Member. If you do not specify the target, the command queries the local computer.

fw isp_link

Description

Controls the state of ISP Links in the ISP Redundancy configuration on Security Gateway.

Syntax

```
fw [-d] isp_link
      {-h | -help}
      [<Name of Object>] <Name of ISP Link>
      down
      up
```

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.
`{-h | -help}`	Shows the built-in usage.
<Name of Object>	Only when you run this command on a Management Server: The name of the Security Gateway or Cluster Member object as defined in SmartConsole (from the left navigation panel, click **Gateways & Servers**).
<Name of ISP Link>	The name of the ISP Link as defined in the Security Gateway or Cluster object: 1. In SmartConsole, from the left navigation panel, click **Gateways & Servers**. 2. Open the Security Gateway or Cluster object. 3. From the left tree, click **Other > ISP Redundancy**.
`down`	Changes the state of the specified ISP Link to DOWN.
`up`	Changes the state of the specified ISP Link to UP.

fw kill

Description

Kills the specified Check Point processes.

Important - Make sure the killed process is restarted, or restart it manually. See sk97638.

Syntax

```
fw [-d] kill [-t <Signal Number>] <Name of Process>
```

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.
`-t <Signal Number>`	Specifies which signal to send to the Check Point process. For the list of available signals and their numbers, run the `kill -l` command. For information about the signals, see the manual pages for the `kill` and `signal`. If you do not specify the signal explicitly, the command sends Signal 15 (`SIGTERM`). **Note** - Processes can ignore some signals.
`<Name of Process>`	Specifies the name of the Check Point process to kill. To see the names of the processes, run the `ps auxwf` command.

Example

```
fw kill fwd
```

fw lichosts

Description

Shows IP addresses of internal hosts that Security Gateway detected and counted based on the installed license.

Syntax

```
fw [-d] lichosts [-l] [-x]
```

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.
-l	Shows the output in the long format.
-x	Shows the output in the hexadecimal format.

Example

```
[Expert@MyGW:0]# fw lichosts
License allows an unlimited number of hosts
[Expert@MyGW:0]
```

Related SK article

sk10200 - 'too many internal hosts' error in /var/log/messages on Security Gateway.

fw log

Description

Shows the content of Check Point log files - Security (`$FWDIR/log/*.log`) or Audit (`$FWDIR/log/*.adtlog`).

Syntax

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

```
fw [-d] log [-a] [-b "<Start Timestamp>" "<End Timestamp>"] [-c
<Action>] [{-f | -t}] [-g] [-H] [-h <Origin>] [-i] [-k {<Alert Name> |
all}] [-l] [-m {initial | semi | raw}] [-n] [-o] [-p] [-q] [-S] [-s
"<Start Timestamp>"] [-e "<End Timestamp>"] [-u <Unification Scheme
File>] [-w] [-x <Start Entry Number>] [-y <End Entry Number>] [-z] [-
#] [<Log File>]
```

Parameters

Parameter	Description
`{-h | -help}`	Shows the built-in usage. **Note** - The built-in usage does not show some of the parameters described in this table.
`-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.
`-a`	Shows only Account log entries.
`-b "<Start Timestamp>" "<End Timestamp>"`	Shows only entries that were logged between the specified start and end times. ■ The *<Start Timestamp>* and *<End Timestamp>* may be a date, a time, or both. ■ If date is omitted, then the command assumes the current date. ■ Enclose the "*<Start Timestamp>*" and "*<End Timestamp>* in single or double quotes (`-b 'XX' 'YY'`, or `-b "XX" "YY"`). ■ You cannot use the "`-b`" parameter together with the "`-s`" or "`-e`" parameters. ■ See the date and time format below.

Parameter	Description
`-c <Action>`	Shows only events with the specified action. One of these: ■ `accept` ■ `drop` ■ `reject` ■ `encrypt` ■ `decrypt` ■ `vpnroute` ■ `keyinst` ■ `authorize` ■ `deauthorize` ■ `authcrypt` ■ `ctl` **Notes:** ■ The `fw log` command always shows the **Control** (`ctl`) actions. ■ For *login* action, use the `authcrypt`.
`-e "<End Timestamp>"`	Shows only entries that were logged before the specified time. **Notes:** ■ The *<End Timestamp>* may be a date, a time, or both. ■ Enclose the *<End Timestamp>* in single or double quotes (`-e '...'`, or `-e "..."`). ■ You cannot use the "`-e`" parameter together with the "`-b`" parameter. ■ See the date and time format below.
`-f`	This parameter: 1. Shows the saved entries that match the specified conditions. 2. After the command reaches the end of the currently opened log file, it continues to monitor the log file indefinitely and shows the new entries that match the specified conditions. **Note** - Applies only to the *active* log file `$FWDIR/log/fw.log` or `$FWDIR/log/fw.adtlog`

Parameter	Description
`-g`	Does not show delimiters. The default behavior is: ■ Show a colon (`:`) after a field name ■ Show a semi-colon (`;`) after a field value
`-H`	Shows the High Level Log key.
`-h <Origin>`	Shows only logs that were generated by the Security Gateway with the specified IP address or object name (as configured in SmartConsole).
`-i`	Shows log UID.
`-k {<Alert Name> | all}`	Shows entries that match a specific alert type: ■ *<Alert Name>* - Show only entries that match a specific alert type: • `alert` • `mail` • `snmp_trap` • `spoof` • `user_alert` • `user_auth` ■ `all` - Show entries that match all alert types (this is the default).
`-l`	Shows both the date and the time for each log entry. The default is to show the date only once above the relevant entries, and then specify the time for each log entry.
`-m`	Specifies the log unification mode: ■ `initial` - Complete unification of log entries. The command shows one unified log entry for each ID. This is the default. If you also specify the `-f` parameter, then the output does not show any updates, but shows only entries that relate to the start of new connections. To shows updates, use the `semi` parameter. ■ `semi` - Step-by-step unification of log entries. For each log entry, the output shows an entry that unifies this entry with all previously encountered entries with the same ID. ■ `raw` - No log unification. The output shows all log entries.
`-n`	Does not perform DNS resolution of the IP addresses in the log file (this is the default behavior). This significantly speeds up the log processing.

Parameter	Description
`-o`	Shows detailed log chains - shows all the log segments in the log entry.
`-p`	Does not perform resolution of the port numbers in the log file (this is the default behavior). This significantly speeds up the log processing.
`-q`	Shows the names of log header fields.
`-S`	Shows the Sequence Number.
`-s "<Start Timestamp>"`	Shows only entries that were logged after the specified time. **Notes:** ■ The `<Start Timestamp>` may be a date, a time, or both. ■ If the date is omitted, then the command assumed the current date. ■ Enclose the `<Start Timestamp>` in single or double quotes (`-s '...'`, or `-s "..."`). ■ You cannot use the "`-s`" parameter together with the "`-b`" parameter. ■ See the date and time format below.
`-t`	This parameter: 1. Does not show the saved entries that match the specified conditions. 2. After the command reaches the end of the currently opened log file, it continues to monitor the log file indefinitely and shows the new entries that match the specified conditions. **Note** - Applies only to the *active* log file `$FWDIR/log/fw.log` or `$FWDIR/log/fw.adtlog`
`-u <Unification Scheme File>`	Specifies the path and name of the log unification scheme file. The default log unification scheme file is: `$FWDIR/conf/log_unification_scheme.C`
`-w`	Shows the flags of each log entry (different bits used to specify the "nature" of the log - for example, control, audit, accounting, complementary, and so on).
`-x <Start Entry Number>`	Shows only entries from the specified log entry number and below, counting from the beginning of the log file.
`-y <End Entry Number>`	Shows only entries until the specified log entry number, counting from the beginning of the log file.
`-z`	In case of an error (for example, wrong field value), continues to show log entries. The default behavior is to stop.

Parameter	Description
`-#`	Show confidential logs in clear text.
<Log File>	Specifies the log file to read. If you do not specify the log file explicitly, the command opens the `$FWDIR/log/fw.log` log file. You can specify a switched log file.

Date and Time format

Part of timestamp	Format	Example
Date only	`MMM DD, YYYY`	`June 11, 2018`
Time only Note - In this case, the command assumes the current date.	`HH:MM:SS`	`14:20:00`
Date and Time	`MMM DD, YYYY HH:MM:SS`	`June 11, 2018 14:20:00`

Output

Each output line consists of a single log entry, whose fields appear in this format:

Note - The fields that show depends on the connection type.

```
HeaderDateHour ContentVersion HighLevelLogKey Uuid SequenceNum Flags
Action Origin IfDir InterfaceName LogId ...
```

This table describes some of the fields.

Field Header	Description	Example
`HeaderDateHour`	Date and Time	`12Jun2018 12:56:42`
`ContentVersion`	Version	`5`
`HighLevelLogKey`	High Level Log Key	`<max_null>`, or empty
`Uuid`	Log UUID	`(0x5b1f99cb,0x0,0x3403a8c0,0xc0000000)`
`SequenceNum`	Log Sequence Number	`1`

Field Header	Description	Example
`Flags`	Internal flags that specify the "nature" of the log - for example, control, audit, accounting, complementary, and so on	`428292`
`Action`	Action performed on this connection	■ `accept` ■ `dropreject` ■ `encrypt` ■ `decrypt` ■ `vpnroute` ■ `keyinst` ■ `authorize` ■ `deauthorize` ■ `authcrypt` ■ `ctl`
`Origin`	Object name of the Security Gateway that generated this log	`MyGW`
`IfDir`	Traffic direction through interface: ■ `<` - Outbound (sent by a Security Gateway) ■ `>` - Inbound (received by a Security Gateway)	■ `<` ■ `>`

Field Header	Description	Example
`InterfaceName`	Name of the Security Gateway interface, on which this traffic was logged If a Security Gateway performed some internal action (for example, log switch), then the log entry shows `daemon`	■ `eth0` ■ `daemon` ■ `N/A`
`LogId`	Log ID	`0`
`Alert`	Alert Type	■ `alert` ■ `mail` ■ `snmp_trap` ■ `spoof` ■ `user_alert` ■ `user_auth`
`OriginSicName`	SIC name of the Security Gateway that generated this log	`CN=MyGW,O=MyDomain_ Server.checkpoint.com.s6t98x`
`inzone`	Inbound Security Zone	`Local`
`outzone`	Outbound Security Zone	`External`
`service_id`	Name of the service used to inspect this connection	`ftp`
`src`	Object name or IP address of the connection's source computer	`MyHost`
`dst`	Object name or IP address of the connection's destination computer	`MyFTPServer`

Field Header	Description	Example
proto	Name of the connection's protocol	tcp
sport_svc	Source port of the connection	64933
ProductName	Name of the Check Point product that generated this log	■ VPN-1 & FireWall-1 ■ Application Control ■ FloodGate-1
ProductFamily	Name of the Check Point product family that generated this log	Network

Examples

Example 1 - Show all log entries with both the date and the time for each log entry

```
fw log -l
```

Example 2 - Show all log entries that start after the specified timestamp

```
[Expert@MyGW:0]# fw log -l -s "June 12, 2018 12:33:00"

 12Jun2018 12:33:00 5 N/A 1 accept MyGW > N/A LogId: <max_null>; ContextNum: <max_null>; OriginSicName: CN=MyGW,O=MyDomain_
Server.checkpoint.com.s6t98x; fg-1_client_in_rule_name: Default; fg-1_client_out_rule_name: Default; fg-1_server_in_rule_
name: Host Redirect; fg-1_server_out_rule_name: ; ProductName: FG; ProductFamily: Network;


 12Jun2018 12:33:39 5 N/A 1 drop MyGW < eth0 LogId: 0; ContextNum: <max_null>; OriginSicName: CN=MyGW,O=MyDomain_
Server.checkpoint.com.s6t98x; inzone: Local; outzone: External; service_id: ftp; src: MyGW; dst: MyFTPServer; proto: tcp; UP_
match_table: TABLE_START; ROW_START: 0; match_id: 2; layer_uuid: 4e26fc30-b345-4c96-b8d7-9db6aa7cdd89; layer_name: MyPolicy
Network; rule_uid: 802020d9-5cdc-4c74-8e92-47e1b0eb72e5; rule_name: ; ROW_END: 0; UP_match_table: TABLE_END; UP_action_table:
TABLE_START; ROW_START: 0; action: 0; ROW_END: 0; UP_action_table: TABLE_END; ProductName: VPN-1 & FireWall-1; svc: ftp;
sport_svc: 64933; ProductFamily: Network;


... ... ...


[Expert@MyGW:0]#
```

Example 3 - Show all log entries between the specified timestamps

```
[Expert@MyGW:0]# fw log -l -b "June 12, 2018 12:33:00" 'June 12, 2018 12:34:00'

 12Jun2018 12:33:00 5 N/A 1 accept MyGW > N/A LogId: <max_null>; ContextNum: <max_null>; OriginSicName: CN=MyGW,O=MyDomain_
Server.checkpoint.com.s6t98x; fg-1_client_in_rule_name: Default; fg-1_client_out_rule_name: Default; fg-1_server_in_rule_
name: Host Redirect; fg-1_server_out_rule_name: ; ProductName: FG; ProductFamily: Network;


 12Jun2018 12:33:39 5 N/A 1 drop MyGW < eth0 LogId: 0; ContextNum: <max_null>; OriginSicName: CN=MyGW,O=MyDomain_
Server.checkpoint.com.s6t98x; inzone: Local; outzone: External; service_id: ftp; src: MyGW; dst: MyFTPServer; proto: tcp; UP_
match_table: TABLE_START; ROW_START: 0; match_id: 2; layer_uuid: 4e26fc30-b345-4c96-b8d7-9db6aa7cdd89; layer_name: MyPolicy
Network; rule_uid: 802020d9-5cdc-4c74-8e92-47e1b0eb72e5; rule_name: ; ROW_END: 0; UP_match_table: TABLE_END; UP_action_table:
TABLE_START; ROW_START: 0; action: 0; ROW_END: 0; UP_action_table: TABLE_END; ProductName: VPN-1 & FireWall-1; svc: ftp;
sport_svc: 64933; ProductFamily: Network;


 12Jun2018 12:33:45 5 N/A 1 ctl MyGW > LogId: <max_null>; ContextNum: <max_null>; OriginSicName: CN=MyGW,O=MyDomain_
Server.checkpoint.com.s6t98x; description: Contracts; reason: Could not reach
"https://productcoverage.checkpoint.com/ProductCoverageService". Check DNS and Proxy configuration on the gateway.; Severity:
2; status: Failed; version: 1.0; failure_impact: Contracts may be out-of-date; update_service: 1; ProductName: Security
Gateway/Management; ProductFamily: Network;

[Expert@MyGW:0]#
```

Example 4 - Show all log entries with action "drop"

```
[Expert@MyGW:0]# fw log -l -c drop

 12Jun2018 12:33:39 5 N/A 1 drop MyGW < eth0 LogId: 0; ContextNum: <max_null>; OriginSicName: CN=MyGW,O=MyDomain_
Server.checkpoint.com.s6t98x; inzone: Local; outzone: External; service_id: ftp; src: MyGW; dst: MyFTPServer; proto: tcp; UP_
match_table: TABLE_START; ROW_START: 0; match_id: 2; layer_uuid: 4e26fc30-b345-4c96-b8d7-9db6aa7cdd89; layer_name: MyPolicy
Network; rule_uid: 802020d9-5cdc-4c74-8e92-47e1b0eb72e5; rule_name: ; ROW_END: 0; UP_match_table: TABLE_END; UP_action_table:
TABLE_START; ROW_START: 0; action: 0; ROW_END: 0; UP_action_table: TABLE_END; ProductName: VPN-1 & FireWall-1; svc: ftp;
sport_svc: 64933; ProductFamily: Network;

[Expert@MyGW:0]#
```

Example 5 - Show all log entries with action "drop", show all field headers, and show log flags

```
[Expert@MyGW:0]# fw log -l -q -w -c drop

 HeaderDateHour: 12Jun2018 12:33:39; ContentVersion: 5; HighLevelLogKey: <max_null>; LogUid: ; SequenceNum: 1; Flags: 428292;
Action: drop; Origin: MyGW; IfDir: <; InterfaceName: eth0; Alert: ; LogId: 0; ContextNum: <max_null>; OriginSicName:
CN=MyGW,O=MyDomain_Server.checkpoint.com.s6t98x; inzone: Local; outzone: External; service_id: ftp; src: MyGW; dst:
MyFTPServer; proto: tcp; UP_match_table: TABLE_START; ROW_START: 0; match_id: 2; layer_uuid: 4e26fc30-b345-4c96-b8d7-
9db6aa7cdd89; layer_name: MyPolicy Network; rule_uid: 802020d9-5cdc-4c74-8e92-47e1b0eb72e5; rule_name: ; ROW_END: 0; UP_
match_table: TABLE_END; UP_action_table: TABLE_START; ROW_START: 0; action: 0; ROW_END: 0; UP_action_table: TABLE_END;
ProductName: VPN-1 & FireWall-1; svc: ftp; sport_svc: 64933; ProductFamily: Network;

[Expert@MyGW:0]#
```

Example 6 - Show only log entries from 0 to 10 (counting from the beginning of the log file)

```
[Expert@MyGW:0]# fw log -l -x 0 -y 10

... ...

[Expert@MyGW:0]#
```

fw logswitch

Description

Switches the current active log file:

1. Closes the current active log file
2. Renames the current active log file
3. Creates a new active log file with the default name

Notes:

- By default, this command switches the active Security log file - `$FWDIR/log/fw.log`
- You can specify to switch the active Audit log file - `$FWDIR/log/fw.adtlog`

Syntax

```
fw [-d] logswitch
       [-audit] [<Name of Switched Log>]
       -h <Target> [[+ | -]<Name of Switched Log>]
```

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.
`-audit`	Specifies to switch the active Audit log file (`$FWDIR/log/fw.adtlog`). You can use this parameter only on a Management Server.
`-h <Target>`	Specifies the remote computer, on which to switch the log. **Notes:** ■ The local and the remote computers must have established SIC trust. ■ The remote computer can be a Security Gateway, a Log Server, or a Security Management Server in High Availability deployment. ■ You can specify the remote managed computer by its main IP address or Object Name as configured in SmartConsole.

Parameter	Description

<Name of Switched Log>	Specifies the name of the switched log file.
	Notes:
	■ If you do not specify this parameter, then a default name is:
	`<YYYY-MM-DD_HHMMSS>.log`
	`<YYYY-MM-DD_HHMMSS>.adtlog`
	For example, *2018-03-26_174455.log*
	■ If you specify the name of the switched log file, then the name of the switch log file is:
	`<Specified_Log_Name>.log`
	`<Specified_Log_Name>.adtlog`
	■ The log switch operation fails if the specified name for the switched log matches the name of an existing log file.
	■ The maximal length of the specified name of the switched log file is 230 characters.

+	Specifies to *copy* the active log from the remote computer to the local computer.
	Notes:
	■ If you specify the name of the switched log file, you must write it immediately after *this* + (plus) parameter.
	■ The command copies the active log from the remote computer and saves it in the `$FWDIR/log/` directory on the local computer.
	■ The default name of the saved log file is:
	`<Gateway_Object_Name>__<YYYY-MM-DD_HHMMSS>.log`
	For example, *MyGW__2018-03-26_174455.log*
	■ If you specify the name of the switched log file, then the name of the saved log file is:
	`<Gateway_Object_Name>__<Specified_Log_Name>.log`
	■ When this command copies the log file from the remote computer, it compresses the file.

Parameter	Description

-	Specifies to *transfer* the active log from the remote computer to the local computer.
	Notes:
	■ The command saves the copied active log file in the `$FWDIR/log/` directory on the local computer and then deletes the switched log file on the remote computer.
	■ If you specify the name of the switched log file, you must write it immediately after this - (minus) parameter.
	■ The default name of the saved log file is:
	`<Gateway_Object_Name>__<YYYY-MM-DD_HHMMSS>.log`
	For example, *MyGW__2018-03-26_174455.log*
	■ If you specify the name of the switched log file, then the name of the saved log file is:
	`<Gateway_Object_Name>__<Specified_Log_Name>.log`
	■ When this command transfers the log file from the remote computer, it compresses the file.
	■ As an alternative, you can use the *"fw fetchlogs" on page 292* command.

Compression

When this command transfers the log files from the remote computer, it compresses the file with the `gzip` command (see RFC 1950 to RFC 1952 for details). The algorithm is a variation of LZ77 method. The compression ratio varies with the content of the log file and is difficult to predict. Binary data are not compressed. Text data, such as user names and URLs, are compressed.

Example - Switching the active Security log on a Security Management Server or Security Gateway

```
[Expert@MGMT:0]# fw logswitch
Log file has been switched to: 2018-06-13_182359.log
[Expert@MGMT:0]#
```

Example - Switching the active Security log on a managed Security Gateway and copying the switched log

```
[Expert@MGMT:0]# fw logswitch -h MyGW +
Log file has been switched to: 2018-06-13_185451.log
[Expert@MGMT:0]#
[Expert@MGMT:0]# ls $FWDIR/log/*.log
/opt/CPsuite-R80.40/fw1/log/fw.log
/opt/CPsuite-R80.40/fw1/log/MyGW__2018-06-13_185451.log
[Expert@MGMT:0]#

[Expert@MyGW:0]# ls $FWDIR/log/*.log
/opt/CPsuite-R80.40/fw1/log/fw.log
/opt/CPsuite-R80.40/fw1/log/2018-06-13_185451.log
[Expert@MyGW:0]#
```

fw lslogs

Description

Shows a list of Security log files (`$FWDIR/log/*.log`) and Audit log files (`$FWDIR/log/*.adtlog`) residing on the local computer or a remote computer.

Syntax

```
fw [-d] lslogs [-f <Name of Log File 1>] [-f <Name of Log File 2>] ...
[-f <Name of Log File N>] [-e] [-r] [-s {name | size | stime | etime}]
[<Target>]
```

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.
`-f <Name of Log File>`	Specifies the name of the log file to show. Need to specify name only. **Notes:** ■ If the log file name is not specified explicitly, the command shows all Security log files (`$FWDIR/log/*.log`). ■ File names may include `*` and `?` as wildcards (for example, `2019-0?-*`). If you enter a wildcard, you must enclose it in double quotes or single quotes. ■ You can specify multiple log files in one command. You must use the "`-f`" parameter for each log file name pattern: `-f <Name of Log File 1> -f <Name of Log File 2> ... -f <Name of Log File N>`
`-e`	Shows an extended file list. It includes the following information for each log file: ■ `Size` - The total size of the log file and its related pointer files ■ `Creation Time` - The time the log file was created ■ `Closing Time` - The time the log file was closed ■ `Log File Name` - The file name
`-r`	Reverses the sort order (descending order).

Parameter	Description
`-s {name | size | stime | etime}`	Specifies the sort order of the log files using one of the following sort options: ■ `name` - The file name ■ `size` - The file size ■ `stime` - The time the log file was created (this is the default option) ■ `etime` - The time the log file was closed
<Target>	Specifies the remote Check Point computer, with which this local Check Point computer has established SIC trust. ■ If you run this command on a Security Management Server or Domain Management Server, then *<Target>* is the applicable object's name or main IP address of the Check Point Computer as configured in SmartConsole. ■ If you run this command on a Security Gateway or Cluster Member, then *<Target>* is the main IP address of the applicable object as configured in SmartConsole.

Example 1 - Default output

```
[Expert@HostName:0]# fw lslogs
     Size Log file name
         9KB 2019-06-14_000000.log
        11KB 2019-06-15_000000.log
         9KB 2019-06-16_000000.log
        10KB 2019-06-17_000000.log
         9KB fw.log
[Expert@HostName:0]#
```

Example 2 - Showing all log files

```
[Expert@HostName:0]# fw lslogs -f "*"
     Size Log file name
         9KB fw.adtlog
         9KB fw.log
         9KB 2019-05-29_000000.adtlog
         9KB 2019-05-29_000000.log
         9KB 2019-05-20_000000.adtlog
         9KB 2019-05-20_000000.log
[Expert@HostName:0]#
```

Example 3 - Showing only log files specified by the patterns

```
[Expert@HostName:0]# fw lslogs -f "2019-06-14*" -f '2019-06-15*'
     Size Log file name
         9KB 2019-06-14_000000.adtlog
         9KB 2019-06-14_000000.log
        11KB 2019-06-15_000000.adtlog
        11KB 2019-06-15_000000.log
[Expert@HostName:0]#
```

Example 4 - Showing only log files specified by the patterns and their extended information

```
[Expert@HostName:0]# fw lslogs -f "2019-06-14*" -f '2019-06-15*'
     Size Log file name
         9KB 2019-06-14_000000.adtlog
         9KB 2019-06-14_000000.log
        11KB 2019-06-15_000000.adtlog
        11KB 2019-06-15_000000.log
[Expert@HostName:0]#
```

Example 5 - Showing only log files specified by the patterns, sorting by name in reverse order

```
[Expert@HostName:0]# fw lslogs -f "2019-06-14*" -f '2019-06-15*' -e -s name -r
     Size Creation Time Closing Time Log file name
        11KB 14Jun2018 0:00:00 15Jun2018 0:00:00 2019-06-15_000000.log
        11KB 14Jun2018 0:00:00 15Jun2018 0:00:00 2019-06-15_000000.adtlog
         9KB 13Jun2018 18:23:59 14Jun2018 0:00:00 2019-06-14_000000.log
         9KB 13Jun2018 0:00:00 14Jun2018 0:00:00 2019-06-14_000000.adtlog
[Expert@HostName:0]#
```

Example 6 - Showing only log files specified by the patterns, from a managed Security Gateway with main IP address 192.168.3.53

```
[Expert@MGMT:0]# fw lslogs -f "2019-06-14*" -f '2019-06-15*' 192.168.3.53
     Size Log file name
        11KB 2019-06-15_000000.adtlog
        11KB 2019-06-15_000000.log
         9KB 2019-06-14_000000.log
         9KB 2019-06-14_000000.adtlog
[Expert@MGMT:0]#
```

fw mergefiles

Description

Merges several Security log files (`$FWDIR/log/*.log`) into a single log file.

Merges several Audit log files (`$FWDIR/log/*.adtlog`) into a single log file.

Important:

- Do not merge the *active* Security file `$FWDIR/log/fw.log` with other Security switched log files.

  Switch the active Security file `$FWDIR/log/fw.log` (with the *"fw logswitch" on page 308* command) and only then merge it with other Security switched log files.

- Do not merge the *active* Audit file `$FWDIR/log/fw.adtlog` with other Audit switched log files.

  Switch the active Audit file `$FWDIR/log/fw.adtlog` (with the *"fw logswitch" on page 308* command) and only then merge it with other Audit switched log files.

- This command unifies logs entries with the same Unique-ID (UID). If you rotate the current active log file before all the segments of a specific log arrive, this command merges the records with the same Unique ID from two different files, into one fully detailed record.

- If the size of the final merged log file exceeds 2GB, this command creates a list of merged files, where the size of each merged file size is not more than 2GB.

  The user receives this warning:

  ```
  Warning: The size of the files you have chosen to
  merge is greater than 2GB. The merge will produce
  two or more files.
  ```

  The names of merged files are:

  - *<Name of Merged Log File>*.log
  - *<Name of Merged Log File>*_1.log
  - *<Name of Merged Log File>*_2.log
  - ... ...
  - *<Name of Merged Log File>*_N.log

Syntax

```
fw [-d] mergefiles {-h | -help}
```

```
fw [-d] mergefiles [-r] [-s] [-t <Time Conversion File>] <Name of Log
File 1> <Name of Log File 2> ... <Name of Log File N> <Name of Merged
Log File>
```

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.
`{-h | -help}`	Shows the built-in usage.
`-r`	Removes duplicate entries.
`-s`	Sorts the merged file by the **Time** field in log records.

`-t <Time Conversion File>`	Specifies a full path and name of a file that instructs this command how to adjust the times during the merge.
	This is required if you merge log files from Log Servers configured with different time zones.
	The file format is:
	`<IP Address of Log Server #1> <Signed Date Time #1 in Seconds>`
	`<IP Address of Log Server #2> <Signed Date Time #2 in Seconds>`
	`... ...`
	  **Notes**
	■ You must specify the absolute path and the file name.
	■ The name of the time conversion file cannot exceed 230 characters.

`<Name of Log File 1> ... <Name of Log File N>`	Specifies the log files to merge.  **Notes:** ■ You must specify the absolute path and the name of the input log files. ■ The name of the input log file cannot exceed 230 characters.

Parameter	Description
<Name of Merged Log File>	Specifies the output merged log file.  **Notes:** ■ The name of the merged log file cannot exceed 230 characters. ■ If a file with the specified name already exists, the command stops and asks you to remove the existing file, or to specify another name. ■ The size of the merged log file cannot exceed 2 GB. In such scenario, the command creates several merged log files, each not exceeding the size limit.

Example - Merging Security log files

```
[Expert@HostName:0]# ls -l $FWDIR/*.log
-rw-rw-r-- 1 admin root 189497 Sep  7 00:00 2019-09-07_000000.log
-rw-rw-r-- 1 admin root  14490 Sep  9 09:52 2019-09-09_000000.log
-rw-rw-r-- 1 admin root  30796 Sep 10 10:56 2019-09-10_000000.log
-rw-rw-r-- 1 admin root  24503 Sep 10 13:08 fw.log
[Expert@HostName:0]#
[Expert@HostName:0]# fw mergefiles -s $FWDIR/2019-09-07_000000.log $FWDIR/2019-09-09_000000.log
$FWDIR/2019-09-10_000000.log /var/log/2019-Sep-Merged.log
[Expert@HostName:0]#
[Expert@HostName:0]# ls -l /var/log/2019-Sep-Merged.log*
-rw-rw---- 1 admin root 213688 Sep 10 13:18 /var/log/2019-Sep-Merged.log
-rw-rw---- 1 admin root   8192 Sep 10 13:18 /var/log/2019-Sep-Merged.logLuuidDB
-rw-rw---- 1 admin root     80 Sep 10 13:18 /var/log/2019-Sep-Merged.logaccount_ptr
-rw-rw---- 1 admin root   2264 Sep 10 13:18 /var/log/2019-Sep-Merged.loginitial_ptr
-rw-rw---- 1 admin root   4448 Sep 10 13:18 /var/log/2019-Sep-Merged.logptr
[Expert@HostName: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 *"fw ctl chain" on page 261*).

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 333*).

`-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 345*).
	■ The "`-F`" parameter uses these Kernel Debug Filters:
	For more information, see the *R80.40 Next Generation Security Gateway Guide* - Chapter *Kernel Debug on Security Gateway* - Section *Kernel Debug Filters*.
	• For Source IP address:
	`simple_debug_filter_saddr_<N> "<IP Address>"`
	• For Source Ports:
	`simple_debug_filter_sport_<N> <1-65535>`
	• For Destination IP address:
	`simple_debug_filter_daddr_<N> "<IP Address>"`
	• For Destination Ports:
	`simple_debug_filter_dport_<N> <1-65535>`
	• For Protocol Number:
	`command_simple_debug_filter_proto_<N> <0-254>`
	■ Value 0 is used as "`any`".
	■ 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.

`-H`	Creates an IP address filter. For more information, see *"Kernel Debug Filters" on page 462*. You can specify up to 3 capture filters with this parameter (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. ■ This parameter lets you 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 (the *"fw ctl chain" on page 261* command shows it as `fw VM inbound`) are Pre-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" on page 261*).
	
	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" on page 261* 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" on page 261* 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" on page 261* 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" on page 261* 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 (the *"fw ctl chain" on page 261* command shows it as "`fw VM inbound`") are Pre-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 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 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 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: - 1fffff0 (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: - 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)
 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;"`
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;"`

Which protocol to specify	On which port(s) traffic is captured	Expression
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;"`

Option Description	Expression	Example
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;"`
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;"`

Option Description	Expression	Example
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;"`

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

Option Description	Expression	Example
		TCP Flag: RST + ACK Example: `fw monitor -e "th_flags = 0x14, accept;"`
TCP source port	`th_sport = <Port_Number>`	`fw monitor -e "th_sport = 59259, accept;"`
TCP destination port	`th_dport = <Port_Number>`	`fw monitor -e "th_dport = 22, accept;"`

TCP sequence number (either in Dec or in Hex)	`th_seq = <Number>`	Example for Dec format:
		`fw monitor -e "th_seq = 3937833514, accept;"`
		Example for Hex format:
		`fw monitor -e "th_seq = 0xeab6922a, accept;"`

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;"`

Option Description	Expression	Example
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;"`
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.
<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;"`
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 repairlog

Description

Check Point Security log file (`$FWDIR/log/*.log`) and Audit log files (`$FWDIR/log/*.adtlog`) are databases, with special pointer files.

If these log pointer files become corrupted (which causes the inability to read the log file), this command can rebuild them.

Log File Type	Log File Location	Log Pointer Files

Security log	`$FWDIR/log/*.log`	`*.logptr`
		`*.logaccount_ptr`
		`*.loginitial_ptr`
		`*.logLuuidDB`

Audit log	`$FWDIR/log/*.adtlog`	`*.adtlogptr`
		`*.adtlogaccount_ptr`
		`*.adtloginitial_ptr`
		`*.adtlogLuuidDB`

Syntax

```
fw [-d] repairlog [-u] <Name of Log File>
```

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`	Specifies to rebuild the unification chains in the log file.
<Name of Log File>	The name of the log file to repair.

Example - Repairing the Audit log file

```
fw repairlog -u 2019-06-17_000000.adtlog
```

fw sam

Description

Manages the Suspicious Activity Monitoring (SAM) rules. You can use the SAM rules to block connections to and from IP addresses without the need to change or reinstall the Security Policy. For more information, see sk112061.

You can create the Suspicious Activity Rules in two ways:

- In SmartConsole from Monitoring Results
- In CLI with the `fw sam` command

Notes:

- VSX Gateways and VSX Cluster Members do not support Suspicious Activity Monitoring (SAM) Rules. See sk79700.
- See the *"fw sam_policy" on page 356* and *"sam_alert" on page 425* commands.
- SAM rules consume some CPU resources on Security Gateway.

  Best Practice - Set an expiration that gives you time to investigate, but does not affect performance. Keep only the SAM rules that you need. If you confirm that an activity is risky, edit the Security Policy, educate users, or otherwise handle the risk.
- Logs for enforced SAM rules (configured with the `fw sam` command) are stored in the `$FWDIR/log/sam.dat` file.

  By design, the file is purged when the number of stored entries reaches 100,000.

  This data log file contains the records in one of these formats:

`<type>,<actions>,<expire>,<ipaddr>`
`<type>,<actions>,<expire>,<src>,<dst>,<dport>,<ip_p>`

- SAM Requests are stored on the Security Gateway in the kernel table `sam_requests`.
- IP Addresses that are blocked by SAM rules, are stored on the Security Gateway in the kernel table `sam_blocked_ips`.

Note - To configure SAM Server settings for a Security Gateway or Cluster:

1. Connect with SmartConsole to the applicable Security Management Server or Domain Management Server.
2. From the left navigation panel, click Gateways & Servers.
3. Open the Security Gateway or Cluster object.
4. From the left tree, click **Other > SAM**.
5. Configure the settings.
6. Click **OK**.
7. Install the Access Control Policy on this Security Gateway or Cluster object.

Syntax

- **To add or cancel a SAM rule according to criteria:**

```
fw [-d] sam [-v] [-s <SAM Server>] [-S <SIC Name of SAM Server>]
[-f <Security Gateway>] [-t <Timeout>] [-l <Log Type>] [-C] [-e
<key=val>]+ [-r] -{n|i|I|j|J} <Criteria>
```

- **To delete all SAM rules:**

```
fw [-d] sam [-v] [-s <SAM Server>] [-S <SIC Name of SAM Server>]
[-f <Security Gateway>] -D
```

- **To monitor all SAM rules:**

```
fw [-d] sam [-v] [-s <SAM Server>] [-S <SIC Name of SAM Server>]
[-f <Security Gateway>] [-r] -M -{i|j|n|b|q} all
```

- **To monitor SAM rules according to criteria:**

```
fw [-d] sam [-v] [-s <SAM Server>] [-S <SIC Name of SAM Server>]
[-f <Security Gateway>] [-r] -M -{i|j|n|b|q} <Criteria>
```

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.
`-v`	Enables verbose mode. In this mode, the command writes one message to *stderr* for each Security Gateway, on which the command is enforced. These messages show whether the command was successful or not.
`-s <SAM Server>`	Specifies the IP address (in the X.X.X.X format) or resolvable HostName of the Security Gateway that enforces the command. The default is `localhost`.

Parameter	Description
`-S <SIC Name of SAM Server>`	Specifies the SIC name for the SAM server to be contacted. It is expected that the SAM server has this SIC name, otherwise the connection fails.  **Notes:** ■ If you do not explicitly specify the SIC name, the connection continues without SIC names comparison. ■ For more information about enabling SIC, refer to the OPSEC API Specification. ■ On VSX Gateway, run the *fw vsx showncs -vs <VSID>* command to show the SIC name for the applicable Virtual System.
`-f <Security Gateway>`	Specifies the Security Gateway, on which to enforce the action. `<Security Gateway>` can be one of these: ■ *All* - Default. Specifies to enforce the action on all managed Security Gateways, where SAM Server runs. You can use this syntax only on Security Management Server or Domain Management Server. ■ *localhost* - Specifies to enforce the action on this local Check Point computer (on which the `fw sam` command is executed). You can use this syntax only on Security Gateway or StandAlone. ■ *Gateways* - Specifies to enforce the action on all objects defined as Security Gateways, on which SAM Server runs. You can use this syntax only on Security Management Server or Domain Management Server. ■ *Name of Security Gateway object* - Specifies to enforce the action on this specific Security Gateway object. You can use this syntax only on Security Management Server or Domain Management Server. ■ *Name of Group object* - Specifies to enforce the action on all specific Security Gateways in this Group object.  **Notes:** ■ You can use this syntax only on Security Management Server or Domain Management Server. ■ VSX Gateways and VSX Cluster Members do not support Suspicious Activity Monitoring (SAM) Rules. See sk79700.

Parameter	Description
`-D`	Cancels all inhibit ("`-i`", "`-j`", "`-I`", "`-J`") and notify ("`-n`") parameters.  **Notes:** ■ To "uninhibit" the inhibited connections, run the `fw sam` command with the "`-C`" or "`-D`" parameters. ■ It is also possible to use this command for active SAM requests.
`-C`	Cancels the `fw sam` command to inhibit connections with the specified parameters.  **Notes:** ■ These connections are no longer inhibited (no longer rejected or dropped). ■ The command parameters must match the parameters in the original `fw sam` command, except for the `-t <Timeout>` parameter.
`-t <Timeout>`	Specifies the time period (in seconds), during which the action is enforced. The default is forever, or until you cancel the `fw sam` command.
`-l <Log Type>`	Specifies the type of the log for enforced action: ■ `nolog` - Does not generate Log / Alert at all ■ `short_noalert` - Generates a Log ■ `short_alert` - Generates an Alert ■ `long_noalert` - Generates a Log ■ `long_alert` - Generates an Alert (this is the default)
`-e <key=val>+`	Specifies rule information based on the keys and the provided values. Multiple keys are separated by the plus sign (+). Available keys are (each is limited to 100 characters): ■ `name` - Security rule name ■ `comment` - Security rule comment ■ `originator` - Security rule originator's username
`-r`	Specifies not to resolve IP addresses.
`-n`	Specifies to generate a "Notify" long-format log entry.  **Notes:** ■ This parameter generates an alert when connections that match the specified services or IP addresses pass through the Security Gateway. ■ This action does not inhibit / close connections.

Parameter	Description
`-i`	Inhibits (drops or rejects) new connections with the specified parameters.  **Notes:** ■ Each inhibited connection is logged according to the log type. ■ Matching connections are rejected.
`-I`	Inhibits (drops or rejects) new connections with the specified parameters, and closes all existing connections with the specified parameters.  **Notes:** ■ Matching connections are rejected. ■ Each inhibited connection is logged according to the log type.
`-j`	Inhibits (drops or rejects) new connections with the specified parameters.  **Notes:** ■ Matching connections are dropped. ■ Each inhibited connection is logged according to the log type.
`-J`	Inhibits new connections with the specified parameters, and closes all existing connections with the specified parameters.  **Notes:** ■ Matching connections are dropped. ■ Each inhibited connection is logged according to the log type.
`-b`	Bypasses new connections with the specified parameters.
`-q`	Quarantines new connections with the specified parameters.
`-M`	Monitors the active SAM requests with the specified actions and criteria.
`all`	Gets all active SAM requests. This is used for monitoring purposes only.
`<Criteria>`	Criteria are used to match connections. The criteria and are composed of various combinations of the following parameters: ■ Source IP Address ■ Source Netmask ■ Destination IP Address ■ Destination Netmask ■ Port (see *IANA Service Name and Port Number Registry*) ■ Protocol Number (see *IANA Protocol Numbers*)

Parameter	Description
	Possible combinations are (see the explanations below this table): ■ src <IP> ■ dst <IP> ■ any <IP> ■ subsrc <IP> <Netmask> ■ subdst <IP> <Netmask> ■ subany <IP> <Netmask> ■ srv <Src IP> <Dest IP> <Port> <Protocol> ■ subsrv <Srcip> <Src Netmask> <Dest IP> <Dest Netmask> <Port> <Protocol> ■ subsrvs <Src IP> <Src Netmask> <Dest IP> <Port> <Protocol> ■ subsrvd <Src IP> <Dest IP> <Dest Netmask> <Port> <Protocol> ■ dstsrv <Dest IP> <Port> <Protocol> ■ subdstsrv <Dest IP> <Dest Netmask> <Port> <Protocol> ■ srcpr <IP> <Protocol> ■ dstpr <IP> <Protocol> ■ subsrcpr <IP> <Netmask> <Protocol> ■ subdstpr <IP> <Netmask> <Protocol> ■ generic <key=val>

Explanation for the <Criteria> syntax

Parameter	Description
src <IP>	Matches the Source IP address of the connection.
dst <IP>	Matches the Destination IP address of the connection.
any <IP>	Matches either the Source IP address or the Destination IP address of the connection.
subsrc <IP> <Netmask>	Matches the Source IP address of the connections according to the netmask.
subdst <IP> <Netmask>	Matches the Destination IP address of the connections according to the netmask.

Parameter	Description
`subany <IP> <Netmask>`	Matches either the Source IP address or Destination IP address of connections according to the netmask.
`srv <Src IP> <Dest IP> <Port> <Protocol>`	Matches the specific Source IP address, Destination IP address, Service (port number) and Protocol.
`subsrv <Src IP> <Netmask> <Dest IP> <Netmask> <Port> <Protocol>`	Matches the specific Source IP address, Destination IP address, Service (port number) and Protocol. Source and Destination IP addresses are assigned according to the netmask.
`subsrvs <Src IP> <Src Netmask> <Dest IP> <Port> <Protocol>`	Matches the specific Source IP address, source netmask, destination netmask, Service (port number) and Protocol.
`subsrvd <Src IP> <Dest IP> <Dest Netmask> <Port> <Protocol>`	Matches specific Source IP address, Destination IP, destination netmask, Service (port number) and Protocol.
`dstsrv <Dest IP> <Service> <Protocol>`	Matches specific Destination IP address, Service (port number) and Protocol.
`subdstsrv <Dest IP> <Netmask> <Port> <Protocol>`	Matches specific Destination IP address, Service (port number) and Protocol. Destination IP address is assigned according to the netmask.
`srcpr <IP> <Protocol>`	Matches the Source IP address and protocol.
`dstpr <IP> <Protocol>`	Matches the Destination IP address and protocol.
`subsrcpr <IP> <Netmask> <Protocol>`	Matches the Source IP address and protocol of connections. Source IP address is assigned according to the netmask.
`subdstpr <IP> <Netmask> <Protocol>`	Matches the Destination IP address and protocol of connections. Destination IP address is assigned according to the netmask.

Parameter	Description
`generic <key=val>+`	Matches the GTP connections based on the specified keys and provided values. Multiple keys are separated by the plus sign (+). Available keys are: ■ `service=gtp` ■ `imsi` ■ `msisdn` ■ `apn` ■ `tunl_dst` ■ `tunl_dport` ■ `tunl_proto`

fw sam_policy

Description

Manages the Suspicious Activity Policy editor that lets you work 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:

- *"fw sam" on page 348*
- *"sam_alert" on page 425*

Notes:

- You can run these commands interchangeably: '`fw sam_policy`' and '`fw samp`'.
- You can run these commands in Gaia Clish, or Expert mode.
- Security Gateway stores the SAM Policy rules in the `$FWDIR/database/sam_policy.db` file.
- The SAM Policy management file is `$FWDIR/database/sam_policy.mng`.

Important:

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

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 SAM Policy rules that you need. 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 359*.
`batch`	Adds or deletes many Rate Limiting rules at a time. See *"fw sam_policy batch" on page 372*.
`del <options>`	Deletes one configured Rate Limiting rule one at a time. See *"fw sam_policy del" on page 374*.
`get <options>`	Shows all the configured Rate Limiting rules. See *"fw sam_policy get" on page 377*.

fw sam_policy add

Description

The 'fw sam_policy add' and 'fw6 sam_policy add' commands let you:

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

Notes:

- You can run these commands interchangeably: 'fw sam_policy add' and 'fw samp add'.
- 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.
- The SAM Policy management file is $FWDIR/database/sam_policy.mng.

Important:

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

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 SAM Policy rules that you need. 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 let you:

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

Notes:

- You can run these commands interchangeably: 'fw sam_policy batch' and 'fw samp batch'.
- 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.
- The SAM Policy management file is $FWDIR/database/sam_policy.mng.

Important:

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

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 SAM Policy rules that you need. 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 359*
  - *"fw sam_policy del" on page 374*
- 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 let you:

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

Notes:

- You can run these commands interchangeably: 'fw sam_policy del' and 'fw samp del'.
- 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.
- The SAM Policy management file is $FWDIR/database/sam_policy.mng.

Important:

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

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 SAM Policy rules that you need. 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 del" on the previous page* 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 *add* rule right after the `fw samp del` and `fw6 samp del` command. This flush-only *add* 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 let you:

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

Notes:

- You can run these commands interchangeably: 'fw sam_policy get' and 'fw samp get'.
- 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.
- The SAM Policy management file is $FWDIR/database/sam_policy.mng.

Important:

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

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 SAM Policy rules that you need. 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 359*.
`-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]#
```

fw showuptables

Description

Shows the formatted contents of the Unified Policy kernel tables.

Syntax

```
fw [-d] showuptables
       [-h]
       [-i]
```

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.
`-h`	Shows the built-in usage.
`-i`	Shows the implied rules layers.

fw stat

Description

Shows the following information about the policy on the Security Gateway:

- Name of the installed policy.
- Date of the last policy installation.
- Names of the interfaces protected by the installed policy, and in which direction the policy protects them.

Important - This command is outdated and exists only for backward compatibility with very old versions. Use the "`cpstat -f policy fw`" command instead (see *"cpstat" on page 208*).

Syntax

```
fw [-d] stat [-l | -s] [<Name of Object>]
```

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.
No Parameters	Shows default output - all information is on one line.
`-l`	Shows long output. Shows each interface and its protected traffic direction is on a separate line. In addition, shows this information: ■ `Total` - Number of packets the Security Gateway received on this interface ■ `Reject` - Number of packets the Security Gateway rejected on this interface ■ `Drop` - Number of packets the Security Gateway dropped on this interface ■ `Accept` - Number of packets the Security Gateway accepted on this interface ■ `Log` - Whether Security Gateway sends its logs from this interface (0 - no, 1 - yes)
`-s`	Shows short output. Shows each interface and its protected traffic direction is on a separate line.

Parameter	Description
<Name of Object>	Specifies the name of the Security Gateway or Cluster Member object (as defined in SmartConsole), from which to show the information. Use this parameter only on the Management Server. This requires the established SIC with that Check Point computer.

Example 1 - Default output

```
[Expert@MyGW:0]# fw stat
HOST POLICY DATE
localhost MyGW_Policy 10Sep2018 14:01:25 : [>eth0] [<eth0] [>eth1]
[Expert@MyGW:0]#
```

Example 2 - Short output

```
[Expert@MyGW:0]# fw stat -s
HOST IF POLICY DATE
localhost >eth0 MyGW_Policy 10Sep2018 14:01:25 :
localhost <eth0 MyGW_Policy 10Sep2018 14:01:25 :
localhost >eth1 MyGW_Policy 10Sep2018 14:01:25 :
[Expert@MyGW:0]#
```

Example 3 - Long output

```
[Expert@MyGW:0]# fw stat -l
HOST IF POLICY DATE TOTAL REJECT DROP ACCEPT LOG
localhost >eth0 MyGW_Policy 10Sep2018 14:01:25 : 14377 0 316 14061 1
localhost <eth0 MyGW_Policy 10Sep2018 14:01:25 : 60996 0 0 60996 0
localhost >eth1 MyGW_Policy 10Sep2018 14:01:25 : 304 0 304 0 0
[Expert@MyGW:0]#
```

Example 4 - Long output from the Management Server

```
[Expert@MGMY:0]# fw stat -l MyGW
HOST IF POLICY DATE TOTAL REJECT DROP ACCEPT LOG
MyGW >eth0 MyGW_Policy 12Sep2018 16:34:56 : 120113 0 0 120113 0
MyGW <eth0 MyGW_Policy 12Sep2018 16:34:56 : 10807 0 0 10807 0
MyGW >eth2 MyGW_Policy 12Sep2018 16:34:56 : 3 0 0 3 0
MyGW <eth2 MyGW_Policy 12Sep2018 16:34:56 : 3 0 0 3 0
[Expert@MGMT:0]#
```

fw tab

Description

Shows data from the specified Security Gateway kernel tables.

This command also lets you change the content of dynamic kernel tables. You cannot change the content of static kernel tables.

Kernel tables (also known as State tables) store data that the Firewall and other Software Blades use to inspect packets. These kernel tables are a critical component of Stateful Inspection.

Best Practices:

- Use the "`fw tab -t connections -f`" command to see the detailed (and more technical) information about the current connections in the **Connections** kernel table (ID 8158).
- Use the *"fw ctl conntab" on page 265* command to see the simplified information about the current connections in the **Connections** kernel table (ID 8158).

Syntax

```
fw [-d]
      {-h | -help}
      [-v] [-t <Table>] [-c | -s] [-f] [-o <Output File>] [-r] [-u | -m <Limit>] [-a -e "<Entry>"] [ -x [-e "<Entry>"]] [-y] [<Name of Object>]
```

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.
`{-h | -help}`	Shows the built-in usage.

Parameter	Description

`-t <Table>`	Specifies the kernel table by its name of unique ID.
	To see the names and IDs of the available kernel tables, run:
	`fw tab -s`
	Because the output of this command is very long, we recommend to redirect it to a file. For example:
	`fw tab -s > /tmp/output.txt`

`-a -e "<Entry>"`	Adds the specified entry to the specified kernel table. If a kernel table has the `expire` attribute, when you add an entry with the "`-a -e <Entry>`" parameter, the new entry gets the default table timeout. You can use this parameter only on the local Security Gateway.  **Warning** - If you add a wrong entry, you can make your Security Gateway unresponsive.
`-c`	Shows formatted kernel table data in the common format. This is the default.
`-e "<Entry>"`	Specifies the entry in the kernel table.  **Important** - Each kernel table has its own internal format.
`-f`	Shows formatted kernel table data. For example, shows: ■ All IP addresses and port numbers in the decimal format. ■ All dates and times in human readable format.  **Note** - Each table can use a different style.  **Important** - If the specified kernel table is large, this consumes a large amount of RAM. This can make your Security Gateway unresponsive.
`-o <Output File>`	Saves the output in the specified file in the CL format as a Check Point Firewall log. You can later open this file with the *"fw log" on page 299* command. If you do not specify the full path explicitly, this command saves the output file in the current working directory.

Parameter	Description
`-m <Limit>`	Specifies the maximal number of kernel table entries to show. This command counts the entries from the beginning of the kernel table.
`-r`	Resolves IP addresses in the formatted output.
`-s`	Shows a short summary of the kernel table data.
`-u`	Specifies to show an unlimited number of kernel table entries.  **Important** - If the specified kernel table is large, this consumes a large amount of RAM. This can make your Security Gateway unresponsive.
`-v`	Shows the CoreXL Firewall instance number as a prefix for each line.
`-x [-e <Entry>]`	Deletes all entries or the specified entry from the specified kernel table. You can use this parameter only on the local Security Gateway.  **Warning** - If you delete a wrong entry, you can break the current connections through your Security Gateway. This includes the remote SSH connection.
`-y`	Specifies not to show a prompt before Security Gateway executes a command. For example, this applies to the parameters "`-a`" and "`-x`".
`<Name of Object>`	Specifies the name of the Security Gateway or Cluster Member object (as defined in SmartConsole), from which to show the information. Use this parameter only on the Management Server. This requires the established SIC with that Check Point computer. If you do not use this parameter, the default is `localhost`.

Example 1 - Show the summary of all kernel tables

```
[Expert@MyGW:0]# fw tab -s
HOST                 NAME                                ID #VALS #PEAK #SLINKS
localhost            vsx_firewalled                       0     1     1       0
localhost            firewalled_list                      1     2     2       0
localhost            external_firewalled_list             2     0     0       0
localhost            management_list                      3     2     2       0
localhost            external_management_list             4     0     0       0
localhost            log_server_list                      5     0     0       0
localhost            ips1_sensors_list                    6     0     0       0
localhost            all_tcp_services                     7   141   141       0
localhost            tcp_services                         8     1     1       0
... ...
localhost            connections                       8158     2    56       2
... ...
localhost            up_251_rule_to_clob_uuid         14083     0     0       0
... ...
localhost            urlf_cache_tbl                      29     0     0       0
localhost            proxy_outbound_conn_tbl             30     0     0       0
localhost            dns_cache_tbl                       31     0     0       0
localhost            appi_referrer_table                 32     0     0       0
localhost            uc_hits_htab                        33     0     0       0
localhost            uc_cache_htab                       34     0     0       0
localhost            uc_incident_to_instance_htab        35     0     0       0
localhost            fwx_cntl_dyn_ghtab                  36     0     0       0
localhost            frag_table                          37     0     0       0
localhost            dos_blacklist_notifs                38     0     0       0
[Expert@MyGW:0]#
```

Example 2 - Show the raw data from the Connections table

```
[Expert@MyGW:0]# fw tab -t connections
localhost:
-------- connections --------
dynamic, id 8158, num ents 0, load factor 0.0, attributes: keep, sync, aggressive aging, kbufs 21 22 23 24
25 26 27 28 29 30 31 32 33 34, expires 25, refresh, , hashsize 2097152, unlimited
<00000000, c0a8cc01, 0000d28d, c0a8cc28, 00000016, 00000006; 0001c001, 00044000, 00000002, 000001e1,
00000000, 5b9687cd, 00000000, 28cca8c0, c0000001, 00000001, 00000001, ffffffff, ffffffff, 02007800,
000f9000, 00000080, 00000000, 00000000, 38edac90, ffffc200, 00000000, 00000000, 00000000, 00000000,
00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000;
1996/3600>
<00000001, c0a8cc28, 00000016, c0a8cc01, 0000d28d, 00000006> -> <00000000, c0a8cc01, 0000d28d, c0a8cc28,
00000016, 00000006> (00000805)
<00000000, c0a8cc01, 0000c9f6, c0a8cc28, 00000016, 00000006; 0001c001, 00044000, 00000002, 000001e1,
00000000, 5b9679de, 00000000, 28cca8c0, c0000001, 00000001, 00000001, ffffffff, ffffffff, 02007800,
000f9000, 00000080, 00000000, 00000000, 38edaa98, ffffc200, 00000000, 00000000, 00000000, 00000000,
00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000;
3597/3600>
<00000001, c0a8cc28, 00000016, c0a8cc01, 0000c9f6, 00000006> -> <00000000, c0a8cc01, 0000c9f6, c0a8cc28,
00000016, 00000006> (00000805)
[Expert@MyGW:0]#
```

Example 3 - Show the formatted data from the Connections table

```
[Expert@MyGW:0]# fw tab -t connections -f
 Using cptfmt
Formatting table's data - this might take a while...

localhost:
 Date: Sep 10, 2018
 20:30:48 5 N/A  N/A    192.168.204.40 > N/A  LogId: <max_null>; ContextNum: <max_null>; OriginSicName:
cn=cp_mgmt,o=MyGW..44jkyv;  : (+)==================================(+); Table_Name: connections; : (+);
Attributes: dynamic, id 8158, attributes: keep, sync, aggressive aging, kbufs 21 22 23 24 25 26 27 28 29 30
31 32 33 34, expires 25, refresh, , hashsize 2097152, unlimited; LastUpdateTime: 10Sep2018 20:30:48;
ProductName: VPN-1 & FireWall-1; ProductFamily: Network;

 20:30:48 5 N/A  N/A    192.168.204.40 > N/A  LogId: <max_null>; ContextNum: <max_null>; OriginSicName:
cn=cp_mgmt,o=MyGW..44jkyv; : ----------------------------------(+); Direction: 1; Source: 192.168.204.40;
SPort: 55411; Dest: 192.168.204.1; DPort: 53; Protocol: udp; CPTFMT_sep: ;; Type: 131073; Rule: 0; Timeout:
335; Handler: 0; Ifncin: -1; Ifncout: -1; Ifnsin: 1; Ifnsout: 1; Bits: 0000780000000000; Expires: 2/40;
LastUpdateTime: 10Sep2018 20:30:48; ProductName: VPN-1 & FireWall-1; ProductFamily: Network;

 20:30:48 5 N/A  N/A    192.168.204.40 > N/A  LogId: <max_null>; ContextNum: <max_null>; OriginSicName:
cn=cp_mgmt,o=MyGW..44jkyv; : ----------------------------------(+); Direction: 0; Source: 192.168.204.1;
SPort: 53901; Dest: 192.168.204.40; DPort: 22; Protocol: tcp; CPTFMT_sep: ;; Type: 114689; Rule: 2;
Timeout: 481; Handler: 0; Ifncin: 1; Ifncout: 1; Ifnsin: -1; Ifnsout: -1; Bits: 02007800000f9000; Expires:
2002/3600; LastUpdateTime: 10Sep2018 20:30:48; ProductName: VPN-1 & FireWall-1; ProductFamily: Network;

 20:30:48 5 N/A  N/A    192.168.204.40 > N/A  LogId: <max_null>; ContextNum: <max_null>; OriginSicName:
cn=cp_mgmt,o=MyGW..44jkyv; : ----------------------------------(+); Direction: 1; Source: 192.168.204.40;
SPort: 22; Dest: 192.168.204.1; DPort: 53901; Protocol: tcp; CPTFMT_sep_1: ->; Direction_1: 0; Source_1:
192.168.204.1; SPort_1: 53901; Dest_1: 192.168.204.40; DPort_1: 22; Protocol_1: tcp; FW_symval: 2053;
LastUpdateTime: 10Sep2018 20:30:48; ProductName: VPN-1 & FireWall-1; ProductFamily: Network;

 20:30:48 5 N/A  N/A    192.168.204.40 > N/A  LogId: <max_null>; ContextNum: <max_null>; OriginSicName:
cn=cp_mgmt,o=MyGW..44jkyv; : ----------------------------------(+); Direction: 0; Source: 192.168.204.1;
SPort: 51702; Dest: 192.168.204.40; DPort: 22; Protocol: tcp; CPTFMT_sep: ;; Type: 114689; Rule: 2;
Timeout: 481; Handler: 0; Ifncin: 1; Ifncout: 1; Ifnsin: -1; Ifnsout: -1; Bits: 02007800000f9000; Expires:
3600/3600; LastUpdateTime: 10Sep2018 20:30:48; ProductName: VPN-1 & FireWall-1; ProductFamily: Network;

 20:30:48 5 N/A  N/A    192.168.204.40 > N/A  LogId: <max_null>; ContextNum: <max_null>; OriginSicName:
cn=cp_mgmt,o=MyGW..44jkyv; : ----------------------------------(+); Direction: 1; Source: 192.168.204.40;
SPort: 22; Dest: 192.168.204.1; DPort: 51702; Protocol: tcp; CPTFMT_sep_1: ->; Direction_1: 0; Source_1:
192.168.204.1; SPort_1: 51702; Dest_1: 192.168.204.40; DPort_1: 22; Protocol_1: tcp; FW_symval: 2053;
LastUpdateTime: 10Sep2018 20:30:48; ProductName: VPN-1 & FireWall-1; ProductFamily: Network;

 20:30:48 5 N/A  N/A    192.168.204.40 > N/A  LogId: <max_null>; ContextNum: <max_null>; OriginSicName:
cn=cp_mgmt,o=MyGW..44jkyv; : ----------------------------------(+); Direction: 0; Source: 192.168.204.1;
SPort: 53; Dest: 192.168.204.40; DPort: 55411; Protocol: udp; CPTFMT_sep_1: ->; Direction_2: 1; Source_2:
192.168.204.40; SPort_2: 55411; Dest_2: 192.168.204.1; DPort_2: 53; Protocol_2: udp; FW_symval: 2054;
LastUpdateTime: 10Sep2018 20:30:48; ProductName: VPN-1 & FireWall-1; ProductFamily: Network;
[Expert@MyGW:0]#
```

Example 4 - Show only two entries from the Connections table

```
[Expert@MyGW:0]# fw tab -t connections -m 2
localhost:
-------- connections --------
dynamic, id 8158, num ents 0, load factor 0.0, attributes: keep, sync, aggressive aging, kbufs 21 22 23 24
25 26 27 28 29 30 31 32 33 34, expires 25, refresh, , hashsize 2097152, unlimited
<00000000, c0a8cc01, 0000d28d, c0a8cc28, 00000016, 00000006; 0001c001, 00044000, 00000002, 000001e1,
00000000, 5b9687cd, 00000000, 28cca8c0, c0000001, 00000001, 00000001, ffffffff, ffffffff, 02007800,
000f9000, 00000080, 00000000, 00000000, 38edac90, ffffc200, 00000000, 00000000, 00000000, 00000000,
00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000;
1961/3600>
<00000001, c0a8cc28, 00000016, c0a8cc01, 0000d28d, 00000006> -> <00000000, c0a8cc01, 0000d28d, c0a8cc28,
00000016, 00000006> (00000805)
...(4 More)
[Expert@MyGW:0]#
```

Example 5 - Show the raw data from the Connections table and show the IDs of CoreXL Firewall instances for each entry

```
[Expert@MyGW:0]# fw tab -t 8158 -v
localhost:
-------- connections --------
dynamic, id 8158, num ents 6, load factor 0.0, attributes: keep, sync, aggressive aging, kbufs 21 22 23 24
25 26 27 28 29 30 31 32 33 34, expires 25, refresh, , hashsize 2097152, unlimited
[fw_0] <00000001, c0a80335, 00004710, c0a803f0, 00008652, 00000006> -> <00000000, c0a803f0, 00008652,
c0a80335, 00004710, 00000006> (00000805)
[fw_0] <00000001, c0a80335, 00008adf, c0a803f0, 0000470f, 00000006; 0002d001, 00046000, 10000000, 0000000e,
00000000, 5b9a4129, 00030000, 3503a8c0, c0000000, ffffffff, ffffffff, 00000001, 00000001, 00000800,
00000000, 80008080, 00000000, 00000000, 338ea330, ffffc200, 00000000, 00000000, 00000000, 00000000,
00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000;
3162/3600>
[fw_0] <00000000, c0a803f0, 00008652, c0a80335, 00004710, 00000006; 0001c001, 00044000, 12000000, 0000000f,
00000000, 5b8fed6a, 00030001, 3503a8c0, c0000000, 00000001, 00000001, ffffffff, ffffffff, 00000800,
08000000, 00000080, 00000000, 00000000, 337b0978, ffffc200, 00000000, 00000000, 00000000, 00000000,
00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000;
3599/3600>
[fw_0] <00000000, c0a803f0, 0000470f, c0a80335, 00008adf, 00000006> -> <00000001, c0a80335, 00008adf,
c0a803f0, 0000470f, 00000006> (00000806)
[fw_0] <00000001, c0a80334, 00004710, c0a803f0, 0000a659, 00000006> -> <00000000, c0a803f0, 0000a659,
c0a80334, 00004710, 00000006> (00000805)
[fw_0] <00000000, c0a803f0, 0000a659, c0a80334, 00004710, 00000006; 0001c001, 00044100, 12000000, 0000000f,
00000000, 5b8feabb, 0000007a, 3403a8c0, c0000000, ffffffff, ffffffff, ffffffff, ffffffff, 00000000,
10000000, 04000080, 00000000, 00000000, 3364aed0, ffffc200, 00000000, 00000000, 00000000, 00000000,
00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000;
3484/3600>
[fw_1] <00000001, c0a80334, 00004710, c0a803f0, 0000bc74, 00000006> -> <00000000, c0a803f0, 0000bc74,
c0a80334, 00004710, 00000006> (00000805)
[fw_1] <00000001, c0a80335, 00000016, ac14a810, 0000e056, 00000006> -> <00000000, ac14a810, 0000e056,
c0a80335, 00000016, 00000006> (00000805)
[fw_1] <00000000, ac14a810, 0000e056, c0a80335, 00000016, 00000006; 0001c001, 00044000, 00000003, 000001df,
00000000, 5b9a3832, 00030000, 3503a8c0, c0000001, 00000001, 00000001, ffffffff, ffffffff, 00000800,
08000000, 00000080, 00000000, 00000000, 33410370, ffffc200, 00000000, 00000000, 00000000, 00000000,
00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000;
3600/3600>
[fw_1] <00000000, c0a803f0, 0000bc74, c0a80334, 00004710, 00000006; 0001c001, 00044100, 12000000, 0000000f,
00000000, 5b8fe89b, 00000001, 3403a8c0, c0000001, ffffffff, ffffffff, ffffffff, ffffffff, 00000000,
10000000, 04000080, 00000000, 00000000, 335841e0, ffffc200, 00000000, 00000000, 00000000, 00000000,
00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000;
3600/3600>
[fw_2] <00000000, c0a803f0, 0000ab74, c0a80335, 00004710, 00000006; 0001c001, 00044000, 12000000, 0000000f,
00000000, 5b8fed7e, 00030000, 3503a8c0, c0000002, 00000001, 00000001, ffffffff, ffffffff, 00000800,
08000000, 00000080, 00000000, 00000000, 33337660, ffffc200, 00000000, 00000000, 00000000, 00000000,
00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000;
3556/3600>
[fw_2] <00000001, c0a80335, 00004710, c0a803f0, 0000ab74, 00000006> -> <00000000, c0a803f0, 0000ab74,
c0a80335, 00004710, 00000006> (00000805)
[fw_2] <00000001, c0a80335, 00001fb4, 00000000, 00001fb4, 00000011> -> <00000000, 00000000, 00001fb4,
c0a80335, 00001fb4, 00000011> (00000805)
[fw_2] <00000000, 00000000, 00001fb4, c0a80335, 00001fb4, 00000011; 00010001, 00004000, 00000003, 00000028,
00000000, 5b8fed76, 00030000, 3503a8c0, c0000002, 00000001, ffffffff, ffffffff, ffffffff, 00000800,
08000000, 00000084, 00000000, 00000000, 336d4e30, ffffc200, 00000000, 00000000, 00000000, 00000000,
00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000; 38/40>
[fw_2] <00000000, 00000000, 00001fb4, c0a80334, 00001fb4, 00000011; 00010001, 00004100, 00000003, 00000028,
00000000, 5b8fed72, 0000025f, 3403a8c0, c0000002, ffffffff, ffffffff, ffffffff, ffffffff, 00000000,
10000000, 04000084, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000,
00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000; 39/40>
[fw_2] <00000001, c0a80334, 00001fb4, 00000000, 00001fb4, 00000011> -> <00000000, 00000000, 00001fb4,
c0a80334, 00001fb4, 00000011> (00000805)
Table fetched in 3 chunks
[Expert@MyGW:0]#
```

fw unloadlocal

Description

Uninstalls all policies from the Security Gateway or Cluster Member.

Warning

1. The "`fw unloadlocal`" command prevents all traffic from passing through the Security Gateway (Cluster Member), because it disables the IP Forwarding in the Linux kernel on the Security Gateway (Cluster Member).
2. The "`fw unloadlocal`" command removes all policies from the Security Gateway (Cluster Member). This means that the Security Gateway (Cluster Member) accepts all incoming connections destined to all active interfaces without any filtering or protection enabled.

Notes

- If you need to remove the current policy, but keep the Security Gateway (Cluster Member) protected, then run the *"comp_init_policy" on page 170* command on the Security Gateway (Cluster Member).
- To load the policies on the Security Gateway (Cluster Member), run one of these commands on the Security Gateway (Cluster Member), or reboot:
  - *"fw fetch" on page 290*
  - *"cpstart" on page 207*

Syntax

```
fw [-d] unloadlocal
```

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]# cpstat -f policy fw

Product name: Firewall
Policy name: My_Policy
Policy install time: Tue Oct 23 18:23:14 2018
... ... ...
[Expert@MyGW:0]#

[Expert@MyGW:0]# sysctl -a | grep forwarding | grep -v bridge
net.ipv6.conf.bond0.forwarding = 1
net.ipv6.conf.eth1.forwarding = 1
net.ipv6.conf.eth3.forwarding = 1
net.ipv6.conf.eth2.forwarding = 1
net.ipv6.conf.eth4.forwarding = 1
net.ipv6.conf.eth5.forwarding = 1
net.ipv6.conf.eth0.forwarding = 1
net.ipv6.conf.eth6.forwarding = 1
net.ipv6.conf.default.forwarding = 1
net.ipv6.conf.all.forwarding = 1
net.ipv6.conf.lo.forwarding = 1
net.ipv4.conf.bond0.mc_forwarding = 0
net.ipv4.conf.bond0.forwarding = 1
net.ipv4.conf.eth1.mc_forwarding = 0
net.ipv4.conf.eth1.forwarding = 1
net.ipv4.conf.eth2.mc_forwarding = 0
net.ipv4.conf.eth2.forwarding = 1
net.ipv4.conf.eth0.mc_forwarding = 0
net.ipv4.conf.eth0.forwarding = 1
net.ipv4.conf.lo.mc_forwarding = 0
net.ipv4.conf.lo.forwarding = 1
net.ipv4.conf.default.mc_forwarding = 0
net.ipv4.conf.default.forwarding = 1
net.ipv4.conf.all.mc_forwarding = 0
net.ipv4.conf.all.forwarding = 1
[Expert@MyGW:0]#

[Expert@MyGW:0]# fw unloadlocal

Uninstalling Security Policy from all.all@MyGW
Done.
[Expert@MyGW:0]#

[Expert@MyGW:0]# cpstat -f policy fw

Product name: Firewall
Policy name:
Policy install time:
... ... ...
[Expert@MyGW:0]#

[Expert@MyGW:0]# sysctl -a | grep forwarding | grep -v bridge
net.ipv6.conf.bond0.forwarding = 0
net.ipv6.conf.eth1.forwarding = 0
net.ipv6.conf.eth3.forwarding = 0
net.ipv6.conf.eth2.forwarding = 0
net.ipv6.conf.eth4.forwarding = 0
net.ipv6.conf.eth5.forwarding = 0
net.ipv6.conf.eth0.forwarding = 0
net.ipv6.conf.eth6.forwarding = 0
net.ipv6.conf.default.forwarding = 0
net.ipv6.conf.all.forwarding = 0
net.ipv6.conf.lo.forwarding = 0
net.ipv4.conf.bond0.mc_forwarding = 0
net.ipv4.conf.bond0.forwarding = 0
net.ipv4.conf.eth1.mc_forwarding = 0
net.ipv4.conf.eth1.forwarding = 0
net.ipv4.conf.eth2.mc_forwarding = 0
net.ipv4.conf.eth2.forwarding = 0
net.ipv4.conf.eth0.mc_forwarding = 0
net.ipv4.conf.eth0.forwarding = 0
net.ipv4.conf.lo.mc_forwarding = 0
net.ipv4.conf.lo.forwarding = 0
```

```
net.ipv4.conf.default.mc_forwarding = 0
net.ipv4.conf.default.forwarding = 0
net.ipv4.conf.all.mc_forwarding = 0
net.ipv4.conf.all.forwarding = 0
[Expert@MyGW:0]#

[Expert@MyGW:0]# fw fetch localhost
Installing Security Policy My_Policy on all.all@MyGW
Fetching Security Policy from localhost succeeded
[Expert@MyGW:0]#
```

fw up_execute

Description

Executes the offline Unified Policy.

Syntax

```
fw [-d] up_execute ipp=<IANA Protocol Number> [src=<Source IP>]
[dst=<Destination IP>] [sport=<Source Port>] [dport=<Destination
Port>] [protocol=<IANA Protocol Name>]
[application=<Application/Category Name 1>
[application=<Application/Category Name 2> ...]]
```

Parameters

Parameter	Description
No Parameters	Shows the built-in usage.
`-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.
`ipp=<IANA Protocol Number>`	IANA Protocol Number in the Hexadecimal format.  **Important** - This parameter is always mandatory. For example: ■ TCP = 6 ■ UDP = 17 ■ ICMP = 1 See *IANA Protocol Numbers*.
`src=<Source IP>`	Source IP address.
`dst=<Destination IP>`	Destination IP address.
`sport=<Source Port>`	Source Port number in the Decimal format. See *IANA Service Name and Port Number Registry*.

Parameter	Description
`dport=<Destination Port>`	Destination Port number in the Decimal format. **Important** - This parameter is mandatory for the TCP (6) and UDP (17) protocols. See *IANA Service Name and Port Number Registry*.
`protocol=<IANA Protocol Name>`	Name of the protocol. For example: ■ TCP ■ UDP ■ ICMP ■ HTTP See *IANA Protocol Numbers*.
`application=<Application/Category Name>`	Name of the Application/Category as defined in SmartConsole. You can specify multiple applications.

Example 1

```
[Expert@MyGW:0]# fw up_execute src=126.200.49.240 dst=10.1.1.1 ipp=1

Rulebase execution ended successfully.
Overall status:
----------------
Active clob mask: 0
Required clob mask: 0
Match status: MATCH
Match action: Accept

Per Layer:
------------
Layer name: Network
Layer id: 0
Match status: MATCH
Match action: Accept
Matched rule: 2
Possible rules: 2 16777215

[Expert@MyGW:0]#
```

Example 2

```
[Expert@MyGW:0]# fw up_execute src=10.1.1.1 ipp=6 dport=8080 protocol=HTTP application=Facebook
application=Opera

Rulebase execution ended successfully.
Overall status:
----------------
Active clob mask: 0
Required clob mask: 0
Match status: MATCH
Match action: Accept

Per Layer:
------------
Layer name: Network
Layer id: 0
Match status: MATCH
Match action: Accept
Matched rule: 2
Possible rules: 2 16777215

[Expert@MyGW:0]#
```

fw ver

Description

Shows this information about the Security Gateway software:

- Major version
- Minor version
- Build number
- Kernel build number

Syntax

```
fw [-d] ver [-k] [-f <Output File>]
```

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.
`ver`	Shows: ■ Major version ■ Minor version ■ Build number
`-k`	■ Shows: ■ Major version ■ Minor version ■ Build number ■ Kernel build number
`-f <Output File>`	Saves the output to the specified file. If you do not specify the full path explicitly, this command saves the output file in the current working directory.

Example 1

```
[Expert@MyGW:0]# fw ver -k
This is Check Point's software version R80.40 - Build 123
[Expert@MyGW:0]#
```

Example 2

```
[Expert@MyGW:0]# fw ver -k
This is Check Point's software version R80.40 - Build 456
[Expert@MyGW:0]#
```

fwboot

Description

Configures Check Point boot options.

Important - Most of these commands are for Check Point use only.

Syntax

```
[Expert@HostName:0]# $FWDIR/boot/fwboot
      bootconf <options>
      corexl <options>
      cpuid <options>
      default <options>
      fwboot_ipv6 <options>
      fwdefault <options>
      ha_conf <options>
      ht <options>
      multik_reg <options>
      post_drv <options>
```

Parameters

Parameter	Description
`bootconf <options>`	Shows and configures the security boot options. See *"fwboot bootconf" on page 402*.
`corexl <options>`	Configures and monitors the CoreXL. See *"fwboot corexl" on page 406*.
`cpuid <options>`	Shows the number of available CPUs and CPU cores on this Security Gateway. See *"fwboot cpuid" on page 413*.
`default <options>`	Loads the specified Default Filter policy on this Security Gateway. See *"fwboot default" on page 415*.

Parameter	Description
`fwboot_ipv6 <options>`	Shows the internal memory address of the hook function for the specified CoreXL Firewall instance. See *"fwboot fwboot_ipv6" on page 416*.
`fwdefault <options>`	Loads the specified Default Filter policy on this Security Gateway. See *"fwboot fwdefault" on page 417*.
`ha_conf <options>`	Configures the cluster mechanism during boot. See *"fwboot ha_conf" on page 418*.
`ht <options>`	Shows and configures the SMT (HyperThreading) feature (sk93000) boot options. See *"fwboot ht" on page 419*.
`multik_reg <options>`	Shows the internal memory address of the registration function for the specified CoreXL Firewall instance. See *"fwboot multik_reg" on page 422*.
`post_drv <options>`	Loads the Firewall driver for CoreXL during boot. See *"fwboot post_drv" on page 424*.

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 406*
  - *"control_bootsec" on page 173*

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.
`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`.

Parameter	Description
`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`.
`set_corexl {0 | 1}`	Enables or disables CoreXL: ■ 0 - disables ■ 1 - enables  **Notes:** ■ In the `$FWDIR/boot/boot.conf` file, refer to the value of the `COREXL_INSTALLED`. ■ To configure CoreXL, use the *"cpconfig" on page 189* menu.
`set_core_override <number>`	Configures the number of overriding CPU cores. The SMT (HyperThreading) feature (sk93000) uses this configuration to set the number of CPU cores after reboot.  **Note** - In the `$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.

Parameter	Description
`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 `$FWDIR/boot/boot.conf` file, refer to the value of the `CTL_IPFORWARDING`.
`set_ipv6 {0 | 1}`	Enables or disables the IPv6 Support: ■ 0 - disables ■ 1 - enables **Notes:** ■ In the `$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 *R80.40 Gaia Administration Guide*.
`set_kernnum <number>`	Configures the number of IPv4 CoreXL Firewall instances. **Notes:** ■ In the `$FWDIR/boot/boot.conf` file, refer to the value of the `KERN_INSTANCE_NUM`. ■ To configure CoreXL, use the *"cpconfig" on page 189* menu.
`set_kern6num <number>`	Configures the number of IPv6 CoreXL Firewall instances. **Notes:** ■ In the `$FWDIR/boot/boot.conf` file, refer to the value of the `KERN6_INSTANCE_NUM`. ■ To configure CoreXL, use the *"cpconfig" on page 189* 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 189* menu.
- After all changes in CoreXL configuration on the Security Gateway, you must reboot it.
- In 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

Parameter	Description
No Parameters	Shows the built-in help with available parameters.

`core_count`	Returns the number of CPU cores on this computer.
	Example
	`[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]#`

Parameter	Description

`curr_ instance4_ count`	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]#`

`curr_ instance6_ count`	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]#`

`def_by_ allowed [n]`	Sets the default configuration for CoreXL according to the specified allowed number of CPU cores.
`default`	Sets the default configuration for CoreXL.

`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]#`

Parameter	Description

`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 180* 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]#`

`[-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 180* 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]#`

Parameter	Description

`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_ instances 4_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]#`

`max_ instances 4_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]#`

Parameter	Description

`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]#`

`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.

Parameter	Description

`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]#`

Parameter	Description

`--full`	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]#`

`ht_aware`	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]#`

`-n`	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]#`

`--possible`	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]#`

fwboot default

Description

Loads the specified Default Filter policy on this Security Gateway.

Notes:

- You must run this command from the Expert mode.
- This command is the same as the *"fwboot default" above* command.
- Refer to these related commands:
  - *"fw defaultgen" on page 288*
  - *"fwboot bootconf" on page 402*
  - *"control_bootsec" on page 173*
  - *"comp_init_policy" on page 170*

Syntax

```
[Expert@HostName:0]# $FWDIR/boot/fwboot default <Default Filter Policy File>
```

Parameters

Parameter	Description
No Parameters	Shows the built-in help with available parameters.
<Default Filter Policy File>	Specifies the full path and name of the Default Filter policy file. The default is `$FWDIR/boot/default.bin`

Example

```
[Expert@MyGW:0]# $FWDIR/boot/fwboot default $FWDIR/boot/default.bin
FW-1: Default filter installed successfully
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw stat
HOST POLICY DATE
localhost defaultfilter 13Sep2018 14:27:23 : [>eth0] [<eth0]
[Expert@MyGW:0]
```

fwboot fwboot_ipv6

Description

Shows the internal memory address of the hook function for the specified CoreXL Firewall instance.

Important - This command is for Check Point use only.

Syntax

```
[Expert@HostName:0]# $FWDIR/boot/fwboot fwboot_ipv6 <Number of CoreXL Firewall instance> hook [-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.
-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 fwboot_ipv6 0 hook
0xffffffff89f8fc00
[Expert@MyGW:0]#

[Expert@MyGW:0]# $FWDIR/boot/fwboot fwboot_ipv6 1 hook
0xffffffff8cd71c00
[Expert@MyGW:0]#

[Expert@MyGW:0]# $FWDIR/boot/fwboot fwboot_ipv6 2 hook
0xffffffff8fb53c00
[Expert@MyGW:0]#
```

fwboot fwdefault

Description

Loads the specified Default Filter policy on this Security Gateway.

Notes:

- You must run this command from the Expert mode.
- This command is the same as the *"fwboot default" on page 415*command.
- Refer to these related commands:
  - *"fw defaultgen" on page 288*
  - *"fwboot bootconf" on page 402*
  - *"control_bootsec" on page 173*
  - *"comp_init_policy" on page 170*

Syntax

```
[Expert@HostName:0]# $FWDIR/boot/fwboot fwdefault <Default Filter Policy File>
```

Parameters

Parameter	Description
No Parameters	Shows the built-in help with available parameters.
<Default Filter Policy File>	Specifies the full path and name of the Default Filter policy file. The default file is `$FWDIR/boot/default.bin`

Example

```
[Expert@MyGW:0]# $FWDIR/boot/fwboot fwdefault $FWDIR/boot/default.bin
FW-1: Default filter installed successfully
[Expert@MyGW:0]#
[Expert@MyGW:0]# fw stat
HOST POLICY DATE
localhost defaultfilter 13Sep2018 14:27:23 : [>eth0] [<eth0]
[Expert@MyGW:0]
```

fwboot ha_conf

Description

Configures the cluster mechanism during boot.

Important - This command is for Check Point use only.

Notes:

- You must run this command from the Expert mode.
- Refer to these related commands:
  - *"fw defaultgen" on page 288*
  - *"fwboot bootconf" on page 402*
  - *"control_bootsec" on page 173*
  - *"comp_init_policy" on page 170*
- To install a cluster, see the *R80.40 Installation and Upgrade Guide*.
- To configure a cluster , see the *R80.40 Installation and Upgrade Guide* and *R80.40 ClusterXL Administration Guide*.

Syntax

```
[Expert@HostName:0]# $FWDIR/boot/fwboot ha_conf
```

fwboot ht

Description

Shows and configures the boot options for the SMT (HyperThreading) feature (sk93000).

Important - This command is for Check Point use only. To configure SMT (HyperThreading) feature, follow sk93000.

Note - You must run this command from the Expert mode.

Syntax

```
[Expert@HostName:0]# $FWDIR/boot/fwboot ht
      --core_override [<number>]
      --disable
      --eligible
      --enable
      --enabled
      --supported
```

Parameters

Parameter	Description
No Parameters	Shows the built-in help with available parameters.
`--core_override [<number>]`	Shows or configures the number of overriding CPU cores. The SMT feature uses this configuration to set the number of CPU cores after reboot.
`--disable`	Disables the SMT feature.

Parameter	Description

`--eligible`	Returns a number that shows if this system is eligible for the SMT feature. Run:
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot ht --eligible`
	`[Expert@MyGW:0]# echo $?`
	■ If you get 1 - The system is eligible for the SMT.
	■ If you get 0 - The system is *not* eligible for the SMT.
	The possible causes are:
	• The system is not a Check Point appliance.
	• The system does not support the SMT.
	• The system does not run Gaia OS.
	• The appliance runs Gaia OS with 32-bit kernel and has more than 4 CPU cores.

`--enable`	Enables the SMT feature.

`--enabled`	Returns a number that shows if SMT feature is enabled on this system. Run:
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot ht --enabled`
	`[Expert@MyGW:0]# echo $?`
	■ If you get 1 - The SMT is enabled.
	■ If you get 0 - The SMT is disabled.
	The possible causes are:
	• The system does not run Gaia OS.
	• The SMT is disabled in software.

Parameter	Description

`--supported`	Returns a number that shows if this system supports the SMT feature.
	Run:
	`[Expert@MyGW:0]# $FWDIR/boot/fwboot ht --supported`
	`[Expert@MyGW:0]# echo $?`
	■ If you get 1 - System supports the SMT.
	■ If you get 0 - System does *not* support the SMT.
	The possible causes are:
	• The system's CPU does not support the SMT.
	• The SMT is disabled in the system's BIOS.
	• The SMT is disabled in software.

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" on page 216* and *"cpstart" on page 207* 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.

sam_alert

Description

For SAM v1, this utility executes Suspicious Activity Monitoring (SAM) actions according to the information received from the standard input.

For SAM v2, this utility executes Suspicious Activity Monitoring (SAM) actions with User Defined Alerts mechanism.

Notes:

- VSX Gateways and VSX Cluster Members do not support Suspicious Activity Monitoring (SAM) Rules. See sk79700.
- You can run this command only in the Expert mode.
- On a Multi-Domain Server, you must run this command in the context of the applicable Domain Management Server:

  ```
  mdsenv <IP Address or Name of Domain Management Server>
  ```

- See the *"fw sam" on page 348* and *"fw sam_policy" on page 356* commands.

SAM v1 syntax

Syntax for SAM v1

```
sam_alert [-v] [-o] [-s <SAM Server>] [-t <Time>] [-f <Security Gateway>] [-C] {-n|-i|-I} {-src|-dst|-any|-srv}
```

Parameters for SAM v1

Parameter	Description
`-v`	Enables the verbose mode for the `fw sam` command.
`-o`	Specifies to print the input of this tool to the standard output (to use with pipes in a CLI syntax).
`-s <SAM Server>`	Specifies the SAM Server to be contacted. Default is localhost.
`-t <Time>`	Specifies the time (in seconds), during which to enforce the action. The default is forever.

Parameter	Description
`-f <Security Gateway>`	Specifies the Security Gateway, on which to run the operation.  **Important** - If you do not specify the target Security Gateway explicitly, this command applies to all managed Security Gateways.
`-C`	Cancels the specified operation.
`-n`	Specifies to notify every time a connection, which matches the specified criteria, passes through the Security Gateway.
`-i`	Inhibits (drops or rejects) connections that match the specified criteria.
`-I`	Inhibits (drops or rejects) connections that match the specified criteria and closes all existing connections that match the specified criteria.
`-src`	Matches the source address of connections.
`-dst`	Matches the destination address of connections.
`-any`	Matches either the source or destination address of connections.
`-srv`	Matches specific source, destination, protocol and port.

SAM v2 syntax

Syntax for SAM v2

```
sam_alert -v2 [-v] [-O] [-S <SAM Server>] [-t <Time>] [-f <Security
Gateway>] [-n <Name>] [-c "<Comment">] [-o <Originator>] [-l {r |
a}] -a {d | r| n | b | q | i} [-C] {-ip |-eth} {-src|-dst|-any|-srv}
```

Parameters for SAM v2

Parameter	Description
`-v2`	Specifies to use SAM v2.
`-v`	Enables the verbose mode for the `fw sam` command.
`-O`	Specifies to print the input of this tool to the standard output (to use with pipes in a CLI syntax).
`-S <SAM Server>`	the SAM server to be contacted. Default is localhost
`-t <Time>`	Specifies the time (in seconds), during which to enforce the action. The default is forever.
`-f <Security Gateway>`	Specifies the Security Gateway, on which to run the operation.  **Important** - If you do not specify the target Security Gateway explicitly, this command applies to all managed Security Gateways.
`-n <Name>`	Specifies the name for the SAM rule. Default is empty.
`-c "<Comment>"`	Specifies the comment for the SAM rule. Default is empty. You must enclose the text in the double quotes or single quotes.
`-o <Originator>`	Specifies the originator for the SAM rule. Default is `sam_alert`.
`-l {r | a}`	Specifies the log type for connections that match the specified criteria: ■ `r` - Regular ■ `a` - Alert Default is `None`.

Parameter	Description
`-a {d | r| n | b | q | i}`	Specifies the action to apply on connections that match the specified criteria: ■ `d` - Drop ■ `r` - Reject ■ `n` - Notify ■ `b` - Bypass ■ `q` - Quarantine ■ `i` - Inspect
`-C`	Specifies to close all existing connections that match the criteria.
`-ip`	Specifies to use IP addresses as criteria parameters.
`-eth`	Specifies to use MAC addresses as criteria parameters.
`-src`	Matches the source address of connections.
`-dst`	Matches the destination address of connections.
`-any`	Matches either the source or destination address of connections.
`-srv`	Matches specific source, destination, protocol and port.

Example

See sk110873: How to configure Security Gateway to detect and prevent port scan.

stattest

Description

Check Point AMON client to query SNMP OIDs.

You can use this command as an alternative to the standard SNMP commands for debug purposes - to make sure the applicable SNMP OIDs provide the requested information.

Notes:

- You can run this command only in the Expert mode.
- On a Multi-Domain Server, you must run this command in the context of the applicable Domain Management Server:

```
mdsenv <IP Address or Name of Domain Management Server>
```

Syntax

- **To query a Regular OID:**

```
stattest get [-d] [-h <Host>] [-p <Port>] [-x <Proxy Server>] [-v <VSID>] [-t <Timeout>] <Regular_OID_1> <Regular_OID_2> ... <Regular_OID_N>
```

These are specified in the SNMP MIB files.

For Check Point MIB files, see sk90470.

- **To query a Statistical OID:**

```
stattest get [-d] [-h <Host>] [-p <Port>] [-x <Proxy Server>] -l <Polling Interval> -r <Polling Duration> [-v <VSID>] [-t <Timeout>] <Statistical_OID_1> <Statistical_OID_2> ... <Statistical_OID_N>
```

Statistical OIDs take some time to "initialize".

For example, to calculate an average, it is necessary to collect enough samples.

Check Point statistical OIDs are registered in the `$CPDIR/conf/statistical_oid.conf` file.

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.
`-h <Host>`	Specifies the remote Check Point host to query by its IP address or resolvable hostname.

Parameter	Description
`-p <Port>`	Specifies the port number, on which the AMON server listens. Default port is 18192.
`-x <Proxy Server>`	Specifies the Proxy Server by its IP address or resolvable hostname. **Note** - Use only when you query a remote host.
`-l <Polling Interval>`	Specifies the time in seconds between queries. **Note** - Use only when you query a Statistical OID.
`-r <Polling Duration>`	Specifies the time in seconds, during which to run consecutive queries. **Note** - Use only when you query a Statistical OID.
`-v <VSID>`	On a VSX Gateway, specifies the context of a Virtual Device to query.
`-t <Timeout>`	Specifies the session timeout in milliseconds.
`<Regular_OID_1> <Regular_OID_2> ... <Regular_OID_N>`	Specifies the Regular OIDs to query. **Notes:** ■ OID must not start with period. ■ Separate the OIDs with spaces. ■ You can specify up to 100 OIDs.
`<Statistical_OID_1> <Statistical_OID_2> ... <Statistical_OID_N>`	Specifies the Statistical OIDs to query. **Notes:** ■ OID must not start with period. ■ Separate the OIDs with spaces. ■ You can specify up to 100 OIDs.

Example - Query a Regular OID

Query the CPU Idle utilization at the OID 1.3.6.1.4.1.2620.1.6.7.2.3 (`procIdleTime`).

```
[Expert@HostName]# stattest get 1.3.6.1.4.1.2620.1.6.7.4.2
```

Example - Query a Statistical OID

Query the CPU Idle utilization at the OID 1.3.6.1.4.1.2620.1.6.7.2.3 (`procIdleTime`).

Information is collected with intervals of 5 seconds during 5 seconds

```
[Expert@HostName]# stattest get -l 5 -r 5 1.3.6.1.4.1.2620.1.6.7.2.3
```

usrchk

Description

Controls the UserCheck daemon (*usrchkd*).

Syntax

```
usrchk
      hits <options>
      incidents <options>
      debug <options>
```

Note - You can also enter partial names of the sub-commands and their options.

Parameters

Parameter	Description
No Parameter	Shows the built-in help. This applies to sub-commands as well. For example, run just the "`usrchk hits`" command.
`hits <options>`	Shows user hits (violations). The available options are:

	■ Show user hits:
	• List all existing hits:
	`usrchk hits list all`
	• Show hits for a specified user:
	`usrchk hits list user <UserName>`
	• Show hits for a specified interaction object:
	`usrchk hits list uci <Name of UserCheck Interaction Object>`

	■ Clear user hits:
	• Clear all existing hits:
	`usrchk hits clear all`
	• Clear hits for a specified user:
	`usrchk hits clear user <UserName>`
	• Clear hits for a specified interaction object:
	`usrchk hits clear uci <Name of UserCheck Interaction Object>`

	■ Database operations:
	• Reload hits from the database:
	`usrchk hits db reload`
	• Update hits changes in the database:
	`usrchk hits db reload update`

Parameter	Description
`incidents <options>`	Sends emails to users about incidents. The available option is: ■ Send emails to users about their expiring email violations: `usrchk incidents expiring`
`debug <options>`	Controls the debug of the UserCheck daemon. The available options are:
	■ Enable the debug: `usrchk debug on` **Important** - After you run this command "`usrchk debug on`", you must run the command "`usrchk debug set ...`" to configure the required filter. **Important** - When you enable the debug, it affects the performance of the *usrchkd* daemon. Make sure to disable the debug after you complete your troubleshooting.
	■ Disable the debug: `usrchk debug off`

Parameter	Description

	■ Filter which debug logs UserCheck writes to the log file based on the specified Debug Topics and Severity:
	`usrchk debug set <Topic Name> <Severity>`
	The available Debug Topics are:
	• `all`
	• Check Point Support provides more specific topics, based on the reported issue
	The available Severities are:
	• `all`
	• `critical`
	• `events`
	• `important`
	• `surprise`
	  **Best Practice** - We recommend to enable all Topics and all Severities. Run:
	`usrchk debug set all all`

	■ Show the UserCheck current debug status: `usrchk debug stat`
	■ Unset the specified Debug Topic(s): `usrchk debug unset <Topic Name>`
	■ Reset all debug topics: `usrchk debug reset`
	■ Rotate the UserCheck log files: `usrchk debug`
	■ Show the memory consumption by the *usrchkd* daemon: `usrchk debug memory`

Parameter	Description
	■ Show and set the number of indentation spaces in the `$FWDIR/log/usrchk.elg` file. `usrchk debug spaces [<0 - 5>]` You can specify the number of spaces: • 0 (this is the default) • 1 • 2 • 3 • 4 • 5

Notes:

■ To show all UserCheck interaction objects, run:

```
usrchk hits list all
```

■ You can only run a command that contains "`user <UserName>`" if:

- Identity Awareness is enabled on the Security Gateway.
- User object is used in the same policy rules as UserCheck objects.

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:

- `$FWDIR/modules/fw_kern_64.o`
- `$FWDIR/modules/fw_kern_64_v6.o`
- `$PPKDIR/modules/sim_kern_64.o`
- `$PPKDIR/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 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`.

  This requires a maintenance window, because the new values of the kernel parameters take effect only after a reboot.
- In Cluster, you must configure all the Cluster Members in the same way

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

Step	Description
1	Connect to the command line on your Security Gateway or Cluster Member.
2	Log in to the Expert mode.
3	Get the list of the available integer kernel parameters and their values: `modinfo -p $FWDIR/boot/modules/fw_kern*.o | sort -u | grep _type | 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`
4	Analyze the output file: `/var/log/fw_integer_kernel_parameters.txt`

Viewing the current value of a Firewall *integer* kernel parameter

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

3	Check the current value of an integer kernel parameter:
	`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	Description
1	Connect to the command line on your Security Gateway or Cluster Member.
2	Log in to Gaia Clish or the Expert mode.

3	Set the new value for an integer kernel parameter:
	`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]#`

4	Make sure the new value is set:
	`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 instructions are provided in various SK articles in Check Point Support Center, and provided by Check Point Support.

For more information, see sk26202: Changing the kernel global parameters for Check Point Security Gateway.

Step	Description
1	Connect to the command line on your Security Gateway or Cluster Member.
2	Log in to the Expert mode.

3	See if the configuration file already exists.
	■ For Firewall kernel parameters:
	`ls -l $FWDIR/boot/modules/fwkern.conf`
	■ For VPN kernel parameters:
	`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**.
	■ For Firewall kernel parameters:
	`touch $FWDIR/boot/modules/fwkern.conf`
	■ For VPN kernel parameters:
	`touch $FWDIR/boot/modules/vpnkern.conf`

5	Back up the current configuration file.
	■ For Firewall kernel parameters:
	`cp -v $FWDIR/boot/modules/fwkern.conf{,_BKP}`
	■ For VPN kernel parameters:
	`cp -v $FWDIR/boot/modules/vpnkern.conf{,_BKP}`

Step	Description

6	Edit the current configuration file.
	■ For Firewall kernel parameters:
	`vi $FWDIR/boot/modules/fwkern.conf`
	■ For VPN kernel parameters:
	`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). ■ To add an integer kernel parameter: *<Name_of_Integer_Kernel_Parameter>=<Integer_Value>* ■ To add a string kernel parameter:  **Note** - You must write the value in single quotes, or double-quotes. *<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 Vi editor.
9	Reboot the Security Gateway or Cluster Member.  **Important** - In cluster, this can cause a failover.
10	Connect to the command line on your Security Gateway or Cluster Member.
11	Log in to Gaia Clish or the Expert mode.

Step	Description

12	Make sure the new value of the kernel parameter is set:
	■ For an integer kernel parameter, run:
	`fw ctl get int <Name of Integer Kernel Parameter> [-a]`
	■ For a string kernel parameter, run:
	`fw ctl get str <Name of String Kernel Parameter> [-a]`

Working with String Kernel Parameters

Viewing the list of the available Firewall *string* kernel parameters and their values

Step	Description
1	Connect to the command line on your Security Gateway or Cluster Member.
2	Log in to the Expert mode.
3	Get the list of the available integer kernel parameters and their values: `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`
4	Analyze the output file: `/var/log/fw_string_kernel_parameters.txt`

Viewing the current value of a Firewall *string* kernel parameter

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

3	Check the current value of a string kernel parameter:
	`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	Description
1	Connect to the command line on your Security Gateway or Cluster Member.
2	Log in to Gaia Clish or the Expert mode.

3	Set the new value for a string kernel parameter:
	  **Note** - You must write the value in single quotes, or double-quotes.
	`fw ctl set str <Name of String Kernel Parameter> '<String Text>'`
	or
	`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]#`

4	Make sure the new value is set:
	`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]#`

Removing the current value from a Firewall *string* kernel parameter *temporarily*

Important - This change does *not* survive reboot.

Step	Description
1	Connect to the command line on your Security Gateway or Cluster Member.
2	Log in to Gaia Clish 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.
	`fw ctl set str '<Name of String Kernel Parameter>'`
	or
	`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]#`

4	Make sure the value is cleared (the new value is empty):
	`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 *cannot* configure SecureXL kernel parameters on-the-fly with the "`fw ctl set`" command.

  You must configure them only permanently 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 Cluster, you must configure all the Cluster Members in the same way

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`

Viewing the list of the available SecureXL *integer* kernel parameters and their values

Step	Description
1	Connect to the command line on your Security Gateway or Cluster Member.
2	Log in to the Expert mode.
3	Get the list of the available integer kernel parameters and their values: `modinfo -p $PPKDIR/boot/modules/sim_kern*.o | sort -u | grep _type | 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`
4	Analyze the output file: `/var/log/sxl_integer_kernel_parameters.txt`

Viewing the list of the available SecureXL *string* kernel parameters and their values

Step	Description
1	Connect to the command line on your Security Gateway or Cluster Member.
2	Log in to the Expert mode.
3	Get the list of the available integer kernel parameters and their values: `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`
4	Analyze the output file: `/var/log/sxl_string_kernel_parameters.txt`

Configuring a value for a SecureXL kernel parameter *permanently*

For more information, see sk26202: Changing the kernel global parameters for Check Point Security Gateway.

Step	Description
1	Connect to the command line on your Security Gateway or Cluster Member.
2	Log in to the Expert mode.
3	See if the configuration file already exists: `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**: `touch $PPKDIR/conf/simkern.conf`
5	Back up the current configuration file: `cp -v $PPKDIR/conf/simkern.conf{,_BKP}`
6	Edit the current configuration file: `vi $PPKDIR/conf/simkern.conf`

Step	Description

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).
	■ To add an *integer* kernel parameter:
	`<Name_of_SecureXL_Integer_Kernel_Parameter>=<Integer_Value>`
	■ To add a *string* kernel parameter:
	  **Note** - You must write the value in single quotes, or double-quotes.
	`<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 Vi editor.
9	Reboot the Security Gateway or Cluster Member.  **Important** - In cluster, this can cause a failover.
10	Connect to the command line on your Security Gateway or Cluster Member.
11	Log in to Gaia Clish or the Expert mode.

12	Make sure the new value of the kernel parameter is set:
	■ For an integer kernel parameter, run:
	`fw ctl get int <Name of Integer Kernel Parameter> [-a]`
	■ For a string kernel parameter, run:
	`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 467*, or the *"Kernel Debug Procedure with Connection Life Cycle" on page 470*.

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 FW 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 this parameter, because it disables even the basic default debug messages.

Table: Parameters of the 'fw ctl debug' command (continued)

Parameter	Description
`-d <Strings to Search>`	When this parameter is specified, 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.
`-s "<String to Stop Debug>"`	When this parameter is specified, 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.

Table: Parameters of the 'fw ctl debug' command (continued)

Parameter	Description

`{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`

`-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 317*.
	■ `-i <Name of Filter File>` - Specifies the file that contains the INSPECT filter.
	■ `-i -` - Specifies that the INSPECT filter arrives from the standard input. You are prompted 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 462*.

`-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.
`-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.

Table: Parameters of the 'fw ctl debug' command (continued)

Parameter	Description
`-p <List of Fields>`	By default, when the Security Gateway prints the debug messages, the messages start with the applicable CPU ID and CoreXL FW 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.
`/<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.

Table: Parameters of the 'fw ctl debug' command (continued)

Parameter	Description
`-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:

- The 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
- The Security Gateway applies these debug filters to both the non-accelerated and accelerated traffic.
- The Security Gateway applies these debug filters to *"Kernel Debug Procedure with Connection Life Cycle" on page 470*.

Best Practice - It is usually simpler to set the Connection Tuple and Host IP Address filters from within the *"'fw ctl debug' and 'fw ctl kdebug'" on page 270* command. To filter the kernel debug by a VPN Peer, use the procedure below.

To configure debug filter of the type "By connection tuple parameters":

The 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*)

This debug filter lets you 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 you need the final filter to match both directions of the connection, you need 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":

This debug filter lets you 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":

This debug filter lets you 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

You need the kernel debug to show 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 470*.

Important:

- Kernel debug increases the load on the Security Gateway CPU. Schedule a maintenance window.
- In Cluster, you must perform these steps on all the Cluster Members in the same way.

Step	Description
1	Connect to the command line on the Security Gateway.
2	Log in to the Expert mode.
3	Reset the kernel debug options: `fw ctl debug 0`
4	Reset the kernel debug filters: `fw ctl set int simple_debug_filter_off 1`
5	Configure the applicable kernel debug filters. See *"Kernel Debug Filters" on page 462*.
6	Allocate the kernel debug buffer for every CoreXL FW instance: `fw ctl debug -buf 8200`
7	Make sure the kernel debug buffer was allocated: `fw ctl debug | grep buffer`
8	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 477*.
9	Examine the list of the debug flags that are enabled in the specified kernel modules: `fw ctl debug -m <module>`
10	Start the kernel debug: `fw ctl kdebug -T -f > /var/log/kernel_debug.txt`

Step	Description
11	Replicate the issue, or wait for the issue to occur.
12	Stop the kernel debug: Press the **CTRL+C** keys
13	Reset the kernel debug options: `fw ctl debug 0`
14	Reset the kernel debug filters: `fw ctl set int simple_debug_filter_off 1`
15	Analyze the debug output file: `/var/log/kernel_debug.txt`

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 477*).

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 462*.
	  **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	Description
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 477*.
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>
;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)

Syntax:

```
fw ctl debug -m accel_apps + {all | <List of Debug Flags>}
```

Flag	Description
av_lite	Messages from the lite Content Inspection (Anti-Virus) module
cmi_lite	Messages from the lite Context Management Interface / Infrastructure module
error	General errors
warning	General warnings

Module 'accel_pm_mgr' (Accelerated Pattern Match Manager)

Syntax:

```
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:

```
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')
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:

```
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:

```
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
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

Flag	Description
`uf`	URL filters and URL cache
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'cluster' (ClusterXL)

Syntax:

```
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 502*
  - The debug flag "`sync`" in *"Module 'CPAS' (Check Point Active Streaming)" on page 488*
- 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:

  ```
  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:

  ```
  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
`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)

Flag	Description
`io`	Information about sending of packets through cluster interfaces
`log`	Creating and sending of logs by cluster **Note** - Also enable the debug flag "`log`" in *"Module 'fw' (Firewall)" on page 502*.
`mac`	Current configuration of and detection of cluster interfaces **Note** - Also 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
`pivot`	Operation of ClusterXL in Load Sharing Unicast mode (Pivot mode)
`pnote`	Registration and monitoring of Critical Devices (pnotes)
`select`	Packet selection (includes the Decision Function)
`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

Module 'cmi_loader' (Context Management Interface / Infrastructure Loader)

Syntax:

```
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 489*.
`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
`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:

```
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 502*.
`tcp`	TCP processing messages
`tcpinfo`	TCP processing messages - more detailed description
`timer`	Reports of internal timer ticks **Warning** - Prints many messages, without real content.
`warning`	General warnings

Module 'cpcode' (Data Loss Prevention - CPcode)

Syntax:

```
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 493*
- *"Module 'dlpk' (Data Loss Prevention - Kernel Space)" on page 495*
- *"Module 'dlpuk' (Data Loss Prevention - User Space)" on page 496*

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
`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 *R80.40 Threat Prevention Administration Guide*.

Syntax:

```
fw ctl debug -m CPSSH + {all | <List of Debug Flags>}
```

Important - Also enable the debug flag "`cpsshi`" in *"Module 'fw' (Firewall)" on page 502*.

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 492*.
`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 521*.
`mux_conn_app`	Information about connections  **Note** - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 521*.

Flag	Description
`mux_decrypt_app`	Information about decryption of connections  **Note** - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 521*.
`mux_encrypt_app`	Information about encryption of connections  **Note** - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 521*.
`mux_inf`	Internal flow  **Note** - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 521*.
`mux_stream`	Internal flow  **Note** - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 521*.
`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:

```
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:

```
fw ctl debug -m dlpda + {all | <List of Debug Flags>}
```

Note - Also see:

- *"Module 'cpcode' (Data Loss Prevention - CPcode)" on page 489*
- *"Module 'dlpk' (Data Loss Prevention - Kernel Space)" on page 495*
- *"Module 'dlpuk' (Data Loss Prevention - User Space)" on page 496*

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)
`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`')

Flag	Description
`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:

```
fw ctl debug -m dlpk + {all | <List of Debug Flags>}
```

Note - Also see:

- *"Module 'cpcode' (Data Loss Prevention - CPcode)" on page 489*
- *"Module 'dlpda' (Data Loss Prevention - Download Agent for Content Awareness)" on page 493*
- *"Module 'dlpuk' (Data Loss Prevention - User Space)" on page 496*

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:

```
fw ctl debug -m dlpuk + {all | <List of Debug Flags>}
```

Note - Also see:

- *"Module 'cpcode' (Data Loss Prevention - CPcode)" on page 489*
- *"Module 'dlpda' (Data Loss Prevention - Download Agent for Content Awareness)" on page 493*
- *"Module 'dlpk' (Data Loss Prevention - Kernel Space)" on page 495*

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
`warning`	General warnings

Module 'DOMO' (Domain Objects)

Syntax:

```
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:

```
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
`multik`	Processing of connections in CoreXL
`pkt`	Packet recording mechanism
`policy`	QoS policy rules matching
`qosaccel`	Acceleration of QoS traffic
`rates`	Rule and connection rates (IQ Engine behavior and status)

Flag	Description
`rtm`	Failures in information gathering in the Real Time Monitoring module **Note** - Also see *"Module 'RTM' (Real Time Monitoring)" on page 527*.
`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:

```
fw ctl debug -m FILE_SECURITY + {all | <List of Debug Flags>}
```

Note - Also see *"Module 'WSIS' (Web Intelligence Infrastructure)" on page 546*.

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:

```
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:

```
fw ctl debug -m fw + {all | <List of Debug Flags>}
```

Flag	Description
`acct`	Accounting data in logs for Application Control (also enable the debug of *"Module 'APPI' (Application Control Inspection)" on page 481*)
`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
`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 514*
`cookie`	Virtual de-fragmentation , cookie issues (cookies in the data structure that holds the packets)
`corr`	Correction layer

Flag	Description
cpsshi	SSH Inspection **Important** - Also enable all the debug flags in *"Module 'CPSSH' (SSH Inspection)" on page 490*.
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 462*)
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 (also AMD)
error	General errors
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)

Flag	Description
`handlers`	Operations related to the Context Management Interface / Infrastructure Loader  **Note** - Also see *"Module 'cmi_loader' (Context Management Interface / Infrastructure Loader)" on page 487*.
`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)
`ld`	Kernel dynamic tables infrastructure (reads from / writes to the tables)  **Warning** - Security Gateway can freeze / 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

Flag	Description
`machine`	INSPECT Virtual Machine (actual assembler commands being processed)  **Warning** - Security Gateway can freeze / 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** - Also see *"Module 'MALWARE' (Threat Prevention)" on page 518*.
`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 317*)  **Note** - Also enable the debug flag "`misc`" in this module.
`monitorall`	Prints output similar to the "`fw monitor -p all`" command (see *"fw monitor" on page 317*)  **Note** - Also enable the debug flag "`misc`" in this module.
`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) Also always enable the debug flag '`sip`' in this module

Flag	Description
`multik`	CoreXL-related  **Note** - This debug flag enables all the debug flags in the *"Module 'multik' (Multi-Kernel Inspection - CoreXL)" on page 519*, except for the debug flag "`packet`".
`nac`	Network Access Control (NAC) feature in Identity Awareness
`nat`	NAT issues - basic information
`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)
`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*

Flag	Description
`sip`	VoIP traffic - SIP and H.323  **Note** - Also see: ■ *"Module 'h323' (VoIP H.323)" on page 510* ■ *"Module 'WS_SIP' (Web Intelligence VoIP SIP Parser)" on page 544*
`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** - Also see *"Module 'synatk' (Accelerated SYN Defender)" on page 532*.
`sync`	Synchronization operations in Check Point cluster  **Note** - Also see the debug flag "`sync`" in *"Module 'CPAS' (Check Point Active Streaming)" on page 488*.
`tcpstr`	TCP streaming mechanism
`te`	Prints the name of an interface for incoming connection from Threat Emulation Machine
`tlsparser`	*Currently is not used*
`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

Flag	Description
`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:

```
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
`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:

```
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:

```
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 488*.
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 487*.
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
icap	Processing of ICAP connections
info	General information
memory	Memory allocation operations

Flag	Description
`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:

```
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
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:

```
fw ctl debug -m kiss + {all | <List of Debug Flags>}
```

Note - Also see *"Module 'kissflow' (Kernel Infrastructure Flow)" on page 517*.

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)
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
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
ioctl	IOCTL control messages (communication between the kernel and daemons)
kqstats	Kernel Worker thread statistics (resetting, initializing, turning off)

Flag	Description
`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
`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

Flag	Description
`worker`	Kernel Worker - queuing and dequeuing

Module 'kissflow' (Kernel Infrastructure Flow)

Syntax:

```
fw ctl debug -m kissflow + {all | <List of Debug Flags>}
```

Note - Also see *"Module 'kiss' (Kernel Infrastructure)" on page 514*.

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:

```
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:

```
fw ctl debug -m multik + {all | <List of Debug Flags>}
```

Note - When you enable the debug flag 'multik' in the *"Module 'fw' (Firewall)" on page 502*, it enables all the debug flags in this debug module, except for the debug flag 'packet'.

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)
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

Flag	Description
`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:

```
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 488*.
`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 525*.
`proxy_tp`	Proxy tunnel parser
`stream`	General information about the data stream
`test`	*Currently is not used*
`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)

Flag	Description
udp	UDP parser
warning	General warnings
ws	Web Intelligence

Module 'NRB' (Next Rule Base)

Syntax:

```
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 481*.
`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 493* ■ *"Module 'dlpk' (Data Loss Prevention - Kernel Space)" on page 495* ■ *"Module 'dlpuk' (Data Loss Prevention - User Space)" on page 496*
`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
`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`')

Flag	Description
`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:

```
fw ctl debug -m PSL + {all | <List of Debug Flags>}
```

Note - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 521*.

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:

```
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:

```
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`'
`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)

Flag	Description
`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:

```
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:

```
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:

```
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 *R80.40 Performance Tuning Administration Guide* - Chapter *SecureXL* - Section *Accelerated SYN Defender*.

Syntax:

```
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 'UC' (UserCheck)

Syntax:

```
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:

```
fw ctl debug -m UP + {all | <List of Debug Flags>}
```

Note - Also see:

- *"Module 'upconv' (Unified Policy Conversion)" on page 536*
- *"Module 'UPIS' (Unified Policy Infrastructure)" on page 537*

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
`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

Flag	Description
`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:

```
fw ctl debug -m upconv + {all | <List of Debug Flags>}
```

Note - Also see:

- *"Module 'UP' (Unified Policy)" on page 534*
- *"Module 'UPIS' (Unified Policy Infrastructure)" on page 537*

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:

```
fw ctl debug -m UPIS + {all | <List of Debug Flags>}
```

Note - Also see:

- *"Module 'UP' (Unified Policy)" on page 534*
- *"Module 'upconv' (Unified Policy Conversion)" on page 536*

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
`error`	General errors
`fwapp`	Information about policy installation for the FireWall application
`info`	General information
`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)
`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`')

Flag	Description
`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:

```
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 509*
`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
`mspi`	Information related to creation and destruction of MSA / MSPI
`multicast`	VPN multicast
`multik`	information related to interaction between VPN and CoreXL
`nat`	NAT issues , cluster IP manipulation (Cluster Virtual IP address <=> Member IP address)
`om_alloc`	Allocation of Office Mode IP addresses

Flag	Description
`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
`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:

```
fw ctl debug -m WS + {all | <List of Debug Flags>}
```

Notes:

- Also see *"Module 'WSIS' (Web Intelligence Infrastructure)" on page 546*.
- 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)
`crumb`	*Currently is not used*
`error`	General errors (the connection is probably rejected)
`event`	Events
`fatal`	Fatal errors

Flag	Description
`flow`	*Currently is not used*
`global`	Handling of global structure (usually, related to policy)
`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
`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

Flag	Description
`warning`	General warnings

Module 'WS_SIP' (Web Intelligence VoIP SIP Parser)

Syntax:

```
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)
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 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

Flag	Description
`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:

```
fw ctl debug -m WSIS + {all | <List of Debug Flags>}
```

Note - Also see *"Module 'WS' (Web Intelligence)" on page 541*.

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`	*Currently is not used*
`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`')
`verbose`	Prints additional information (used with other debug flags)
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Running Check Point Commands in Shell Scripts

To run Check Point commands in shell scripts, you need to add the call for Check Point shell script `/etc/profile.d/CP.sh` to your shell script.

Add this call right under the sha-bang line.

```
#!/bin/bash
source /etc/profile.d/CP.sh
<Check Point commands>
[mandatory last new line]
```