QUANTUM

04 July 2024

NEXT GENERATION SECURITY GATEWAY

R80.40

Administration Guide

CHECK POINT™

Check Point Copyright Notice

Important Information

Latest Software

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

Certifications

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

Check Point R80.40

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

Latest Version of this Document in English

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

Feedback

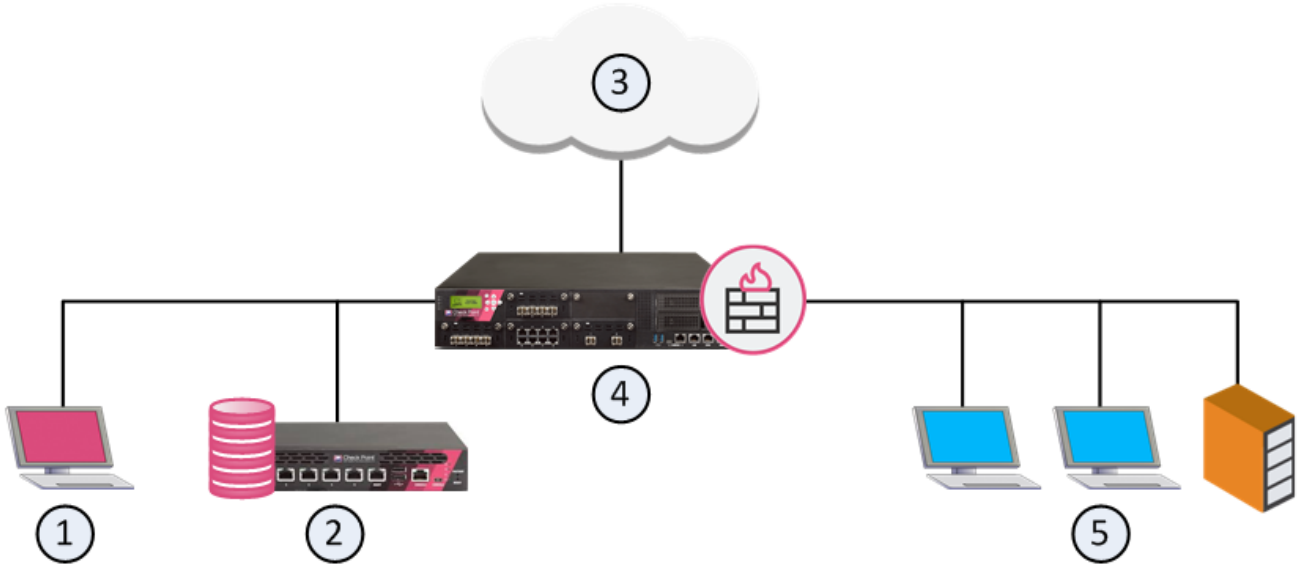
Check Point is engaged in a continuous effort to improve its documentation.
Please help us by sending your comments.

Revision History

Date	Description
26 October 2023	Updated: ■ *"Command Line Reference" on page 217* - removed all commands from this chapter (refer to the *R80.40 CLI Reference Guide*)
23 October 2022	Updated: ■ *"Working with FutureX HSM" on page 94* ■ *"Controlling ISP Redundancy from CLI" on page 161* ■ *"SecureXL Kernel Parameters" on page 235* ■ *"Kernel Debug Procedure" on page 261* ■ *"Kernel Debug Modules and Debug Flags" on page 271*
20 August 2020	Updated: ■ *"UserCheck" on page 41*
19 June 2020	Updated: ■ *"Security Policy" on page 13*
19 June 2020	Updated: ■ *"Hardware Security Module (HSM)" on page 53* (the entire chapter)
28 January 2020	First release of this document

Table of Contents

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 that manages 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.

  How to get there:

  1. Connect with SmartConsole to the Management Server.
  2. From the left navigation panel, click **Security Policies**.
  3. In the **Access Control** section, click **Policy**.

  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).

  How to get there:

  1. Connect with SmartConsole to the Management Server.
  2. From the left navigation panel, click **Security Policies**.
  3. In the **Access Control** section, click **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

  How to get there:

  1. Connect with SmartConsole to the Management Server.
  2. From the left navigation panel, click **Security Policies**.
  3. In the **Access Control** section, click **Desktop**.
  4. Click **Open Desktop Policy in SmartDashboard**.
  5. From the top, click the **Desktop** tab.

  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.

How to get there:

1. Connect with SmartConsole to the Management Server.
2. From the left navigation panel, click **Security Policies**.
3. In the **Threat Prevention** section, click **Policy**.

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.

How to get there:

1. Connect with SmartConsole to the Management Server.
2. From the left navigation panel, click **Security Policies**.
3. In the **HTTPS Inspection** section, click **Policy**.

   Note - In addition, in the **HTTPS Tools** section, click **Additional Settings**.

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.

How to get there:

1. Connect with SmartConsole to the Management Server.
2. From the left navigation panel, click **Manage & Settings**.
3. From the left tree, click **Blades**.
4. In the **Data Loss Prevention** section, click **Configure in SmartDashboard**.
5. From the top, click the **Data Loss Prevention** tab.
6. From the left tree, click **Policy**.

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 *R80.40 Security Management Administration Guide*.

Creates a policy for traffic to or from specific geographical or political locations.

How to get there:

1. Connect with SmartConsole to the Management Server.
2. From the left navigation panel, click **Security Policies**.
3. In the **Access Control** section, click **Policy**.
4. Follow sk126172 to use **Updatable Objects** in the **Source** and **Destination** columns.

For additional information, see sk126172 and 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.

How to get there:

1. Connect with SmartConsole to the Management Server.
2. From the left navigation panel, click **Manage & Settings**.
3. From the left tree, click **Blades**.
4. In the **Mobile Access** section, click **Configure in SmartDashboard**.
5. From the top, click the **Mobile Access** tab.
6. From the left tree, click **Policy**.

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 Security 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 13* > Section *Access Control Policy* > Section *Desktop Rule Base*
- *R81 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 *R81 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 28*
- *"Anti-Virus Software Blade" on page 29*
- *"Threat Extraction Software Blade" on page 30*
- *"Threat Emulation Software Blade" on page 31*
- *"IPS Software Blade" on page 33*

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 41*.

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 41*.

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 41*.

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 41*.

Mail Transfer Agent (MTA)

The Threat Emulation Software Blade requires this feature 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 IP addresses. Identity Awareness maps users and computer identities. This lets you enforce Access Control policy rules and audit data based on identity.

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

Identity Awareness uses the Source and Destination IP addresses of network traffic to identify users and computers.

You can use these elements as matching criteria in the `Source` and `Destination` fields of the Access Control policy rules:

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

With Identity Awareness, 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 39*) 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 41*.

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 41*.

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 35*) 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 41*.

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.

These Software Blades require the UserCheck feature:

- *"Threat Emulation Software Blade" on page 31*
- *"Threat Extraction Software Blade" on page 30*
- *"Anti-Bot Software Blade" on page 28*
- *"Anti-Virus Software Blade" on page 29*
- *"Data Loss Prevention Software Blade" on page 39*
- *"Application Control Software Blade" on page 37*
- *"URL Filtering Software Blade" on page 38*

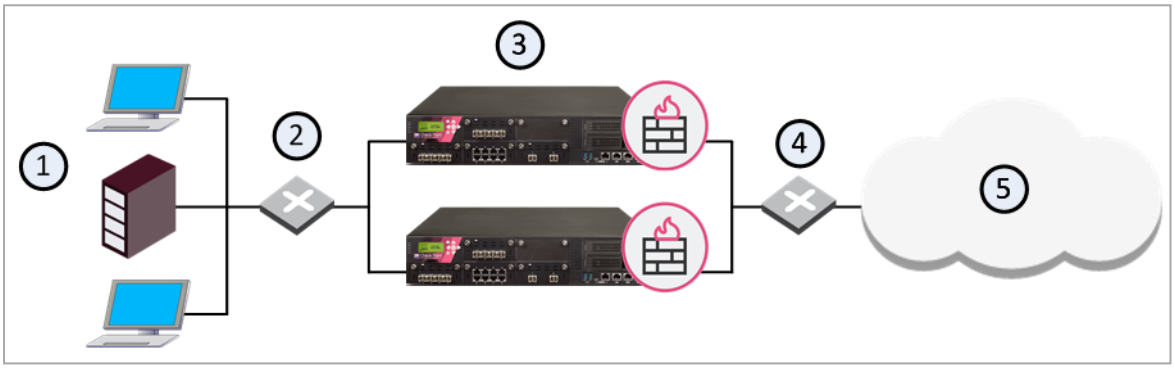
For more information, see:

- The *R80.40 Security Management Administration Guide* > Chapter *Creating an Access Control Policy* > Section *The Columns of the Access Control Rule Base*
- sk83700 - How to customize and localize the UserCheck portal

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*.

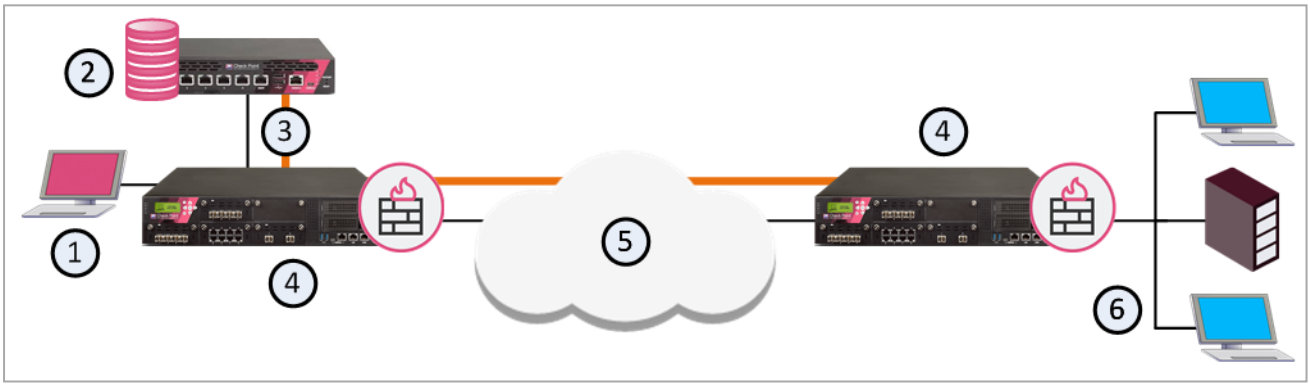
QoS Software Blade

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.

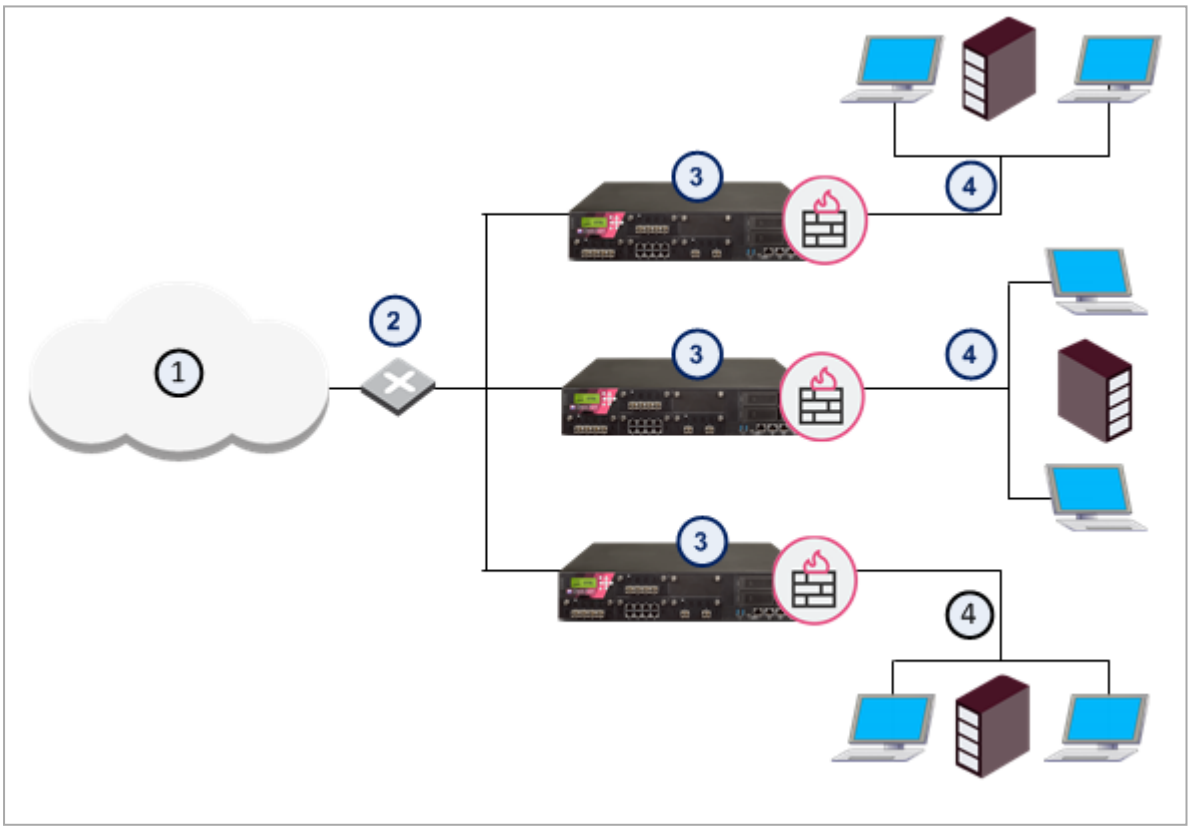
Each **Virtual System** works as a Security Gateway, typically protecting a specified network. When packets arrive at the VSX Gateway, it sends traffic to the Virtual System protecting the destination network. The Virtual System inspects all traffic and allows or rejects it according to rules defined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

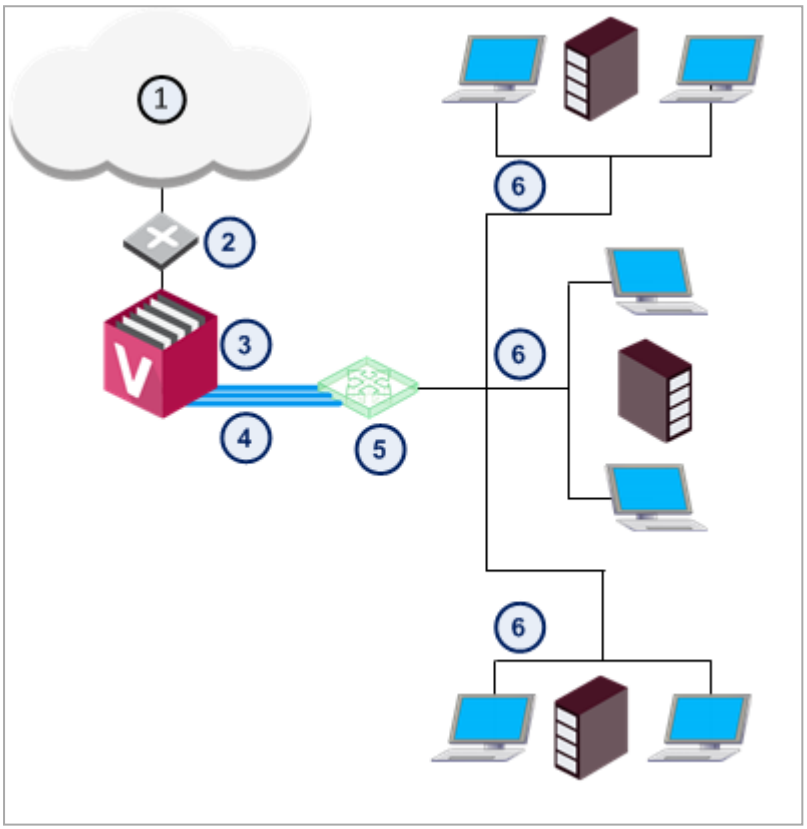


Item	Description
1	Internet

Item	Description
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.

Item	Description
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 a 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 configures more than one traffic queue for each network interface.

For each interface, more than one CPU core is used for acceleration.

Note - Multi-Queue is applicable only if SecureXL is enabled (this is the default).

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 a 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 a 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)

In This Section:

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, the HSM holds these objects for outbound HTTPS Inspection:

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

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

2. Two to three RSA 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.

You can use these HSM solutions to work with the Check Point Security Gateway:

- **Gemalto Luna SP SafeNet HSM**

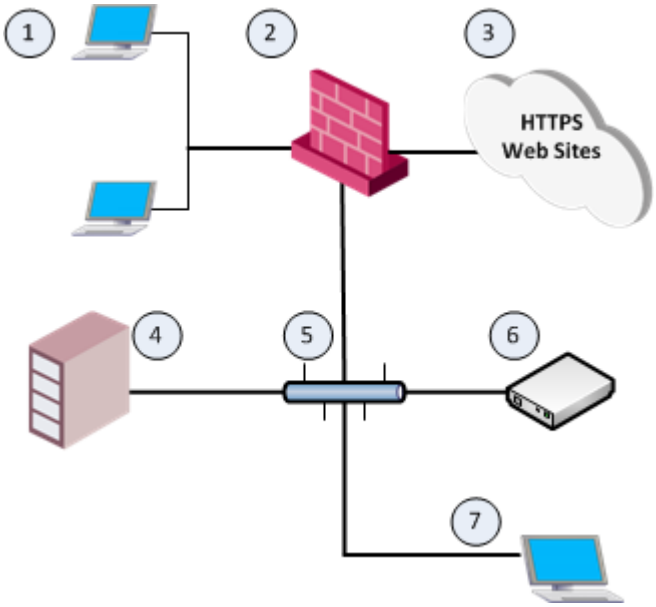
  See *"Working with Gemalto HSM (New Procedure)" on page 61*.

- **FutureX**

  See *"Working with FutureX HSM" on page 94*.

Note - For other HSM vendors that use PKCS#11 API, contact Check Point Solution Center through a local Check Point Office.

The Check Point Environment with an HSM



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.

Generic Workflow

In This Section:

This section contains generic workflows for an HSM environment.

Workflow for Configuring a Check Point Security Gateway to Work with HSM

Follow the steps below on the Security Gateway and Cluster Members that must work with an HSM.

Note - Instructions for specific HSM vendors are located in the corresponding sections.

Generic Step 1 of 3: Configure the HTTPS Inspection to work without the HSM Server

Important:

- In a Cluster, you must configure all the Cluster Members in the same way.
- In a VSX environment, you must perform this step in the context of *each* Virtual System (on the VSX Gateway or *each* VSX Cluster Member).

Step	Instructions
1	In SmartConsole, configure the HTTPS Inspection. See the *R80.40 Security Management Administration Guide* > Chapter HTTPS Inspection.

2	On the Security Gateway / *each* Cluster Member, disable the HSM in the `$FWDIR/conf/hsm_configuration.C` file.
	a. Connect to the command line.
	b. Log in to the Expert mode.
	c. Edit the file:
	`vi $FWDIR/conf/hsm_configuration.C`
	d. Configure the value "`no`" for the parameter "`enabled`":
	`:enabled ("no")`
	e. Save the changes in the file and exit the editor.

Step	Instructions
3	In SmartConsole, install the applicable Access Control Policy on the Security Gateway (Cluster).
4	Make sure that HTTPS Inspection works correctly without the HSM Server: a. From an internal computer, connect to any HTTPS web site. b. On the internal computer, in the web browser, you must receive the signed CA certificate from the Security Gateway (Cluster).

Generic Step 2 of 3: Install and configure the PKCS#11 library supplied by the HSM vendor

Important:

- In a Cluster, you must configure all the Cluster Members in the same way.
- In a VSX environment, you must perform this step in the context of the VSX Gateway or *each* VSX Cluster Member (context of VS 0).
- You must get the HSM Client package from the HSM vendor.

Step	Instructions
1	Unpack and install the HSM Client package supplied by the HSM vendor.
2	Transfer the required PKCS#11 library file to the **/usr/lib/hsm_client/** directory. **Important** - For security reasons, only the root user has permissions to access this directory. You must transfer the physical file into this directory. Do **not** create a symbolic link.
3	Transfer other tools or files supplied by the HSM vendor that are required to configure the PKCS#11 library.
4	Configure the required connection or trust between with the HSM Server.
5	**Optional:** Make sure there is a trusted link with the HSM Server that is based on the PKCS#11 library. **Note** - Use the applicable tool supplied by the HSM vendor. You can also examine the trust with the Check Point command "`cpstat`").

Generic Step 3 of 3: Configure the HTTPS Inspection to work with the HSM Server for Outbound HTTPS Inspection

Important:

- In a Cluster, you must configure all the Cluster Members in the same way.
- In a VSX environment, you must perform this step in the context of *each* Virtual System (on the VSX Gateway or *each* VSX Cluster Member).

Notes:

- In this step, you configure the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway / *each* Cluster Member.
- After you apply the HSM configuration for the first time, you can get an HSM connection error.
  Most common scenario is when you configure several Security Gateways (Cluster Members) to use the same HSM Server, and they access it at the same time.
  In this case:
  a. Run the "`fw fetch local`" command on the Security Gateway (Cluster Member) that has an HSM connection issue.
  In a VSX environment, run this command in the context of the problematic VSX Virtual System.
  b. When you see "`HSM on`" on the screen, continue to configure the next Security Gateway, Cluster Member, or VSX Virtual System.
- After any change in the `$FWDIR/conf/hsm_configuration.C` file, you must do one of these:
  - Fetch the local policy with the "`fw fetch local`" command.
  - In SmartConsole install the policy on the Security Gateway / Cluster / VSX Virtual System object.
- If the HSM Server is **not** available when you fetch the local policy or install the policy in SmartConsole, the HTTPS Inspection **cannot** inspect the Outbound HTTPS traffic. As a result, internal computers behind the Security Gateway / Cluster / VSX Virtual System **cannot** access HTTPS web sites.
  In addition, see *"Disabling Communication from the Security Gateway to the HSM Server" on page 112*.

Configuration steps:

Step	Instructions
1	Connect to the command line on the Security Gateway / *each* Cluster Member.
2	Log in to the Expert mode.
3	Back up the `$FWDIR/conf/hsm_configuration.C` file.
4	Edit the `$FWDIR/conf/hsm_configuration.C` file.

Step	Instructions

5	Configure the required values for these attributes
	(see the corresponding sections for HSM vendors):
	`(:enabled ("no") # "yes" / "no"`
	`:hsm_vendor_name ("")`
	`:lib_filename ("")`
	`:CA_cert_public_key_handle (0)`
	`:CA_cert_private_key_handle (0)`
	`:CA_cert_buffer_handle (0)`
	`:token_id ("")`
	`)`
	
	Notes:
	■ The "**:enabled ()**" attribute must have the value of either "**yes**" (to enable the HSM), or "**no**" (to disable the HSM).
	■ The "**:hsm_vendor_name ()**" attribute must contain the required name of the HSM vendor.
	■ The "**:lib_filename ()**" attribute must contain the name of the PKCS#11 library of your HSM vendor (located in the **/usr/lib/hsm_client/** directory).
	■ The "**:CA_cert_<XXX> ()**" attributes must have the required values of handles.
	■ The "**:token_id ()**" attribute must contain the password for the partition on the HSM Server.
	
	Example:
	`(:enabled ("yes")`
	`:hsm_vendor_name ("FutureX HSM")`
	`:lib_filename ("libfxpkcs11.so")`
	`:CA_cert_public_key_handle (2)`
	`:CA_cert_private_key_handle (1)`
	`:CA_cert_buffer_handle (3)`
	`:token_id ("safest")`
	`)`

6	To apply the new configuration, restart all Check Point services with this command: `cprestart` **Important** - This blocks all traffic until all services restart. In a cluster, this can cause a failover.

Step	Instructions
7	Make sure that the Security Gateway / *each* Cluster Member can connect to the HSM Server and that HTTPS Inspection is activated successfully on the outbound traffic. Run this command: `cpstat https_inspection -f all` The output must show: ■ `HSM partition access (Accessible/Not Accessible): Accessible` ■ `Outbound status (HSM on/HSM off/HSM error): HSM on` For more information, see *"Monitoring HTTPS Inspection with HSM in CLI" on page 135*.
8	Make that HTTPS Inspection is activated successfully on the outbound traffic: a. From an internal computer, connect to any HTTPS web site. b. On the internal computer, in the web browser, you must receive the signed CA certificate from the HSM Server.

Workflow for Configuring an HSM Client Workstation

HSM Client workstation is an external computer, on which you install the HSM Client software of your HSM vendor.

HSM Client workstation can run on Windows, Linux, or other operating system, as required by the HSM vendor.

You use the HSM Client workstation to:

- Create a CA Certificate on the HSM Server.

  Check Point Security Gateways / Cluster Members use this CA Certificate for HTTPS Inspection when it needs to store and access SSL keys on the HSM Server.

- Manage keys for a fake certificate created by the Check Point Security Gateway / Cluster Members.

Important - You must get the HSM Client package from the HSM vendor.

Working with Gemalto HSM (New Procedure)

In This Section:

Important - There are two different procedures to configure a Check Point Security Gateway / ClusterXL to work with the **Gemalto HSM Server**:

Environment	Procedure to Follow
R80.40 with the Jumbo Hotfix Accumulator *Take 53 or higher*	Procedure below
R80.40, or R80.40 with the Jumbo Hotfix Accumulator *Take lower than 53*	*"Working with Gemalto HSM (Previous Procedure)" on page 77*

Prerequisites

1. R80.40 Management Server (see sk160736).
2. R80.40 Security Gateway or Cluster Members (see sk160736).
3. **Mandatory:** R80.40 Jumbo Hotfix Accumulator *Take 53* or higher (see Jumbo Hotfix Accumulator for R80.40).

Configuration Steps

Use this workflow to configure a Check Point Security Gateway / ClusterXL to work with the **Gemalto HSM Server**.

Step 1 of 5: Extracting the Gemalto Help Package

Use the Gemalto configuration documents to configure the Gemalto HSM environment.

Step	Instructions
1	Download this package: Gemalto SafeNet HSM Help package (**007-011136-012_Net_HSM_6.2.2_Help_RevA**) **Note** - Software Subscription or Active Support plan is required to download this package.
2	Use a Windows-based computer.
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 of 5: Configuring the Gemalto HSM Server to Work with Security Gateway / ClusterXL

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

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

Step	Instructions

4	Complete the configuration of the Gemalto HSM Server to work with the Check Point Security Gateway / ClusterXL:
	
	a. Set the applicable partition to be active and auto-activated.
	Run these commands in LunaSH:
	`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.
	
	b. Disable the validation of the client source IP address by NTLS upon an NTLA client connection.
	Run this command in LunaSH:
	`lunash:> ntls ipcheck disable`
	
	Note - This allows the HSM Server to accept traffic from Check Point Cluster Members that hide this traffic behind a Cluster VIP address, and from a Check Point Security Gateway hidden behind NAT.

Step 3 of 5: Configuring the Gemalto HSM Client workstation

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

Check Point Security Gateway / ClusterXL uses this CA Certificate for HTTPS Inspection to store and to access SSL keys on the Gemalto HSM Server.

Note - You can also use Check Point Security Gateway / ClusterXL with the installed HSM Client package as an HSM Client workstation.

Step	Instructions
1	Get this HSM Client package from the Gemalto vendor: **610-012382-017_SW_Client_HSM_6.2.2_RevA**
2	Install a Windows-based or Linux-based computer to use as a Gemalto HSM Client Workstation.
3	Install the HSM Client package on the computer: From the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to the *Installation Guide* > *SafeNet HSM Client Software Installation*.
4	Establish a Trust Link between the Gemalto HSM Client Workstation and the Gemalto HSM Server. From the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to the *Configuration Guide* > *[Step 7] Create a trusted link and register Client and Appliance with each other*. On the Gemalto HSM Client Workstation, run in LunaCM: `lunacm:> clientconfig deploy -c <IP Address of HSM Client Workstation> -n <IP Address of HSM Server> -par <Partition Name> -pw <Partition Password>`

Step 4 of 5: Creating the CA Certificate on the Gemalto HSM Server

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

2	Use the "`cmu generatekeypair`" command to create a key pair.
	From the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to the *Utilities Reference Guide* > *Certificate Management Utility (CMU)* > *cmu generatekeypair*.
	Example:
	`# cd /usr/safenet/lunaclient/bin`
	`# ./cmu generatekeypair -modulusBits=2048 -publicExponent=65537 -labelPublic="CAPublicKeyPairLabel" -labelPrivate="CAPrivateKeyPairLabel" -sign=T -verify=T`

3	When prompted, enter the password for the partition on Gemalto HSM Server (you configured it in *"Step 2 of 5: Configuring the Gemalto HSM Server to Work with Security Gateway / ClusterXL" on page 63*). Example: `Enter a password for the token in slot 0:`
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	View the handles of the key pair you created.
	From the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to the *Utilities Reference Guide* > *Certificate Management Utility (CMU)* > *cmu list*.
	`# ./cmu list`
	Example output:
	`Enter password for token in slot 0 : <Password for the Partition>`
	`handle=17 label=CAPrivateKeyPairLabel`
	`handle=18 label=CAPublicKeyPairLabel`

Step	Instructions
6	Use the handle numbers from the previous step to create the CA certificate. From the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to the *Utilities Reference Guide* > *Certificate Management Utility (CMU)* > *cmu selfsigncertificate* Example: `# ./cmu selfsigncertificate -privatehandle=17 CN="www.gemltoHSM.cp" -sha256WithRSA -startDate 20190720 -endDate 20240720 -serialNum=111aaa -keyusage digitalsignature,keycertsign,crlsign -basicconstraints=critical,ca:true`

7	View the handles of the CA certificate you created.
	`# ./cmu list`
	Example output:
	`Enter password for token in slot 0 : <Password for the Partition>`
	`handle=13 label=www.myhsm.cp`
	`handle=17 label=CAPrivateKeyPairLabel`
	`handle=18 label=CAPublicKeyPairLabel`
	Important - You use the numbers of these three handles later when you configure the `$FWDIR/conf/hsm_configuration.C` file on the Check Point Security Gateway / *each* Cluster Member.

8	Export the CA certificate to a file. From the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to the *Utilities Reference Guide* > *Certificate Management Utility (CMU)* > *cmu export* `# ./cmu export -handle=<Handle Number> -outputfile=<Name of Output File>`

Step 5 of 5: Configuring the Security Gateway / ClusterXL to Work with the Gemalto HSM Server

This step has three sub-steps.

Sub-Step 5-A: Configuring HTTPS Inspection on the Security Gateway / Cluster Members to work *without* the Gemalto HSM Server

Important:

- In a Cluster, you must configure all the Cluster Members in the same way.
- In a VSX environment, you must perform this step in the context of *each* Virtual System (on the VSX Gateway or *each* VSX Cluster Member).

Step	Instructions
1	In SmartConsole, enable and configure the HTTPS Inspection. See the *R80.40 Security Management Administration Guide* > Chapter HTTPS Inspection.

2	On the Security Gateway (*each* Cluster Member), disable the HSM in the `$FWDIR/conf/hsm_configuration.C` file.
	a. Connect to the command line.
	b. Log in to the Expert mode.
	c. Edit the file:
	`vi $FWDIR/conf/hsm_configuration.C`
	d. Configure the value "`no`" for the parameter "`enabled`":
	`:enabled ("no")`
	e. Save the changes in the file and exit the editor.

3	In SmartConsole, install the applicable Access Control Policy on the Security Gateway / ClusterXL object.
4	Make sure that HTTPS Inspection works correctly without the HSM Server: a. From an internal computer, connect to any HTTPS web site. b. On the internal computer, in the web browser, you should receive the signed CA certificate from the Security Gateway / ClusterXL.

Sub-Step 5-B: Installing the Gemalto HSM Simplified Client Software Packages on the Security Gateway (Cluster Members)

Important:

- In a Cluster, you must configure all the Cluster Members in the same way.
- In a VSX environment, you must perform this step in the context of the VSX Gateway or *each* VSX Cluster Member (context of VS 0).

Notes:

- For more information, see the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*.
  For information about establishing a Trust Link, go to the *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 new Trust Link, you have to delete the current Trust Link.
  See *"Deleting a Trust Link with the HSM Server" on page 75*.

Step	Instructions
1	Open the Gemalto HSM Client package you received from Gemalto: **610-012382-017_SW_Client_HSM_6.2.2_RevA** Go to this directory: `linux` > `32`
2	Install the HSM Client package. From the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to the *Installation Guide* > *SafeNet HSM Client Software Installation*.
3	In the Expert mode, copy the **libCryptoki2.so** file to the **/usr/lib/hsm_client/** directory: `cp -v /usr/safenet/lunaclient/lib/libCryptoki2.so /usr/lib/hsm_client/` **Important** - For security reasons, only the root user has permissions to access this directory. You must copy the physical file into this directory. Do **not** create a symbolic link.

Step	Instructions
4	Establish a Trust Link between the Gemalto HSM Client on the Security Gateway / *each* Cluster Member and the Gemalto HSM Server. From the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to the *Configuration Guide* > *[Step 7] Create a trusted link and register Client and Appliance with each other*. On the Security Gateway / *each* Cluster Member, run in LunaCM: `lunacm:> clientconfig deploy -c <IP Address of Security Gateway or Cluster Member> -n <IP Address of HSM Server> -par <Partition Name> -pw <Partition Password>`
5	Examine the partition access on the Security Gateway / *each* Cluster Member: `# /usr/safenet/lunaclient/bin/vtl verify`

Sub-Step 5-C: Configuring HTTPS Inspection on the Security Gateway / Cluster Members to work *with* the Gemalto HSM Server

Important:

- In a Cluster, you must configure all the Cluster Members in the same way.
- In a VSX environment, you must perform this step in the context of the VSX Gateway or *each* VSX Cluster Member (context of VS 0).

Notes:

- After you apply the HSM configuration for the first time, you may get an HSM connection error.
  Most common scenario is when you configure several Security Gateways (Cluster Members) to use the same HSM Server, and they access it at the same time.
  In this case:
  a. Run the "`fw fetch local`" command on the Security Gateway (Cluster Member) that has an HSM connection issue.
     In a VSX environment, run this command in the context of the problematic VSX Virtual System.
  b. Wait until you see "`HSM on`".
  c. Continue to configure the next Security Gateway, Cluster Member, or VSX Virtual System.
- After any change in the `$FWDIR/conf/hsm_configuration.C` file, you must do one of these:
  - Fetch the local policy with the "`fw fetch local`" command
  - In SmartConsole, install the policy on the Security Gateway / ClusterXL / VSX Virtual System object.
- If the HSM Server is **not** available when you fetch the local policy or install the policy in SmartConsole, the HTTPS Inspection **cannot** inspect the Outbound HTTPS traffic. As a result, internal computers behind the Security Gateway / ClusterXL / VSX Virtual System **cannot** to access HTTPS web sites.
  In addition, see *"Disabling Communication from the Security Gateway to the HSM Server" on page 112*.

Step	Instructions
1	Connect to the command line on the Security Gateway / *each* Cluster Member.
2	Log in to the Expert mode.
3	Back up the `$FWDIR/conf/hsm_configuration.C` file: `cp -v $FWDIR/conf/hsm_configuration.C{,_BKP}`
4	Edit the `$FWDIR/conf/hsm_configuration.C` file: `vi $FWDIR/conf/hsm_configuration.C`

Step	Instructions

5	Configure the required values for these attributes:
	`(`
	`:enabled ("yes")`
	`:hsm_vendor_name ("Luna Gemalto HSM")`
	`:lib_filename ("libCryptoki2.so")`
	`:CA_cert_public_key_handle (<`**`Number of "Public" Handle for CA certificate`**`>)`
	`:CA_cert_private_key_handle (<`**`Number of "Private" Handle for CA certificate`**`>)`
	`:CA_cert_buffer_handle (<`**`Number of Handle for CA certificate`**`>)`
	`:token_id ("<`**`Password for Partition on Gemalto HSM Server`**`>")`
	`)`

Step	Instructions

	Notes:
	■ The "**:enabled ()**" attribute must have the value of either "**yes**" (to enable the HSM), or "**no**" (to disable the HSM).
	■ The "**:hsm_vendor_name ()**" attribute must contain the string "**Luna Gemalto HSM**" (or must be empty).
	■ The "**:lib_filename ()**" attribute must contain the name of the PKCS#11 library of the Gemalto HSM vendor (located in the **/usr/lib/hsm_client/** directory on the Security Gateway / *each* Cluster Member.
	■ The "**:CA_cert_<XXX> ()**" attributes must have the required values of handles from the output of the "`cmu list`" command on the Gemalto HSM Server.
	See *"Step 4 of 5: Creating the CA Certificate on the Gemalto HSM Server" on page 66*.
	■ The "**:token_id ()**" attribute must have the contain the password for the partition on the Gemalto HSM Server.
	See *"Step 2 of 5: Configuring the Gemalto HSM Server to Work with Security Gateway / ClusterXL" on page 63*.
	
	Example:
	`(`
	`:enabled ("yes")`
	`:hsm_vendor_name ("Gemalto HSM")`
	`:lib_filename ("libCryptoki2.so")`
	`:CA_cert_public_key_handle (17)`
	`:CA_cert_private_key_handle (18)`
	`:CA_cert_buffer_handle (13)`
	`:token_id ("p@ssw0rd")`
	`)`

Step	Instructions

6	Apply the new configuration.
	■ If you explicitly defined (or changed) the value of the "**:hsm_vendor_name ()**" attribute the string "**Gemalto HSM**", then restart all Check Point services with this command:
	`cprestart`
	Important - This blocks all traffic until all services restart. In a cluster, this can cause a failover.
	■ If you did **not** define the value of the "**:hsm_vendor_name ()**" attribute (it is empty), then fetch the local policy with this command:
	`fw fetch local`

7	Make sure that the Security Gateway / *each* Cluster Member can connect to the HSM Server and that HTTPS Inspection is activated successfully on the outbound traffic. Run this command: `cpstat https_inspection -f all` The output must show: ■ `HSM partition access (Accessible/Not Accessible): Accessible` ■ `Outbound status (HSM on/HSM off/HSM error): HSM on` For more information, see *"Monitoring HTTPS Inspection with HSM in CLI" on page 135*.
8	Make that HTTPS Inspection is activated successfully on the outbound traffic: a. From an internal computer, connect to any HTTPS web site. b. On the internal computer, in the web browser, you should receive the signed CA certificate from the HSM Server.

Additional Actions for a Gemalto HSM Server

Deleting a Trust Link with the HSM Server

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

Use Case: When you replace or reconfigure a Check Point Security Gateway, or an HSM Server.

Step	Instructions

1	Delete the current Trust Link on the Check Point Security Gateway / *each* Cluster Member:
	a. Connect to the command line.
	b. Log in to the Expert mode.
	c. Go to the SafeNet HSM Simplified Client installation directory:
	`cd /usr/safenet/lunaclient/bin/`
	d. Delete the old Trust Link:
	`./vtl deleteServer -n <IP Address of HSM Server>`

2	Delete the current Trust Link on the HSM Appliance:
	a. Connect to the HSM Appliance over SSH.
	b. Examine the list of configured HSM Client Workstations:
	`lunash:> client list`
	c. Delete the Check Point HSM Client Workstation:
	`lunash:> client delete -client <Name of HSM Client>`

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

Configuring a Second Interface on a Gemalto HSM Appliance for NTLS

Step	Instructions
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 the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation* > *LunaSH Command Reference Guide* > *LunaSH Commands*.

Working with Gemalto HSM (Previous Procedure)

In This Section:

Important - There are two different procedures to configure a Check Point Security Gateway (Cluster) to work with the **Gemalto HSM Server**:

Environment	Procedure to Follow
R80.40 with the Jumbo Hotfix Accumulator *Take 53 or higher*	*"Working with Gemalto HSM (New Procedure)" on page 61*
R80.40, or R80.40 with the Jumbo Hotfix Accumulator *Take lower than 53*	Procedure below

Prerequisites

1. R80.40 Management Server (see sk160736).
2. R80.40 Security Gateway or Cluster Members (see sk160736).
3. **Optional:** R80.40 Jumbo Hotfix Accumulator *Take lower than 53* (see Jumbo Hotfix Accumulator for R80.40).

Configuration Steps

Step 1 of 5: Extracting the Gemalto Help Package

Use the Gemalto configuration documents to configure the Gemalto HSM environment.

Step	Instructions
1	Download this package: Gemalto SafeNet HSM Help package (**007-011136-012_Net_HSM_6.2.2_Help_RevA**) **Note** - Software Subscription or Active Support plan is required to download this package.
2	Use a Windows-based computer.
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 of 5: Configuring the Gemalto HSM Appliance Server to Work with Security Gateway

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

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

Step	Instructions

4	Complete the configuration of the Gemalto HSM Appliance Server to work with the Check Point Security Gateway (Cluster):
	a. Set the applicable partition to be active and auto-activated.
	Run these commands in LunaSH:
	`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.
	b. Disable the validation of the client source IP address by NTLS during an NTLA client connection.
	Run this command in LunaSH:
	`lunash:> ntls ipcheck disable`
	Note - This allows the HSM Server to accept traffic from Check Point Cluster Members that hide this traffic behind a Cluster VIP address, and from a Check Point Security Gateway hidden behind NAT.

Step 3 of 5: Configuring the Gemalto HSM Client Workstation

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

Check Point Security Gateway (Cluster Members) uses this CA Certificate for HTTPS Inspection to store and to access SSL keys on the Gemalto HSM Server.

Note - You can also use Check Point Security Gateway (Cluster Members) with the installed HSM Client package as an HSM Client workstation.

Step	Instructions
1	Download this software package: **SafeNet HSM Client for Workstation** **Note** - Software Subscription or Active Support plan is required to download this package.
2	Install a Windows-based or Linux-based computer to use as a Gemalto HSM Client Workstation.
3	Install the HSM Client package on the computer: From the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to the *Installation Guide* > *SafeNet HSM Client Software Installation*.
4	Establish a Trust Link between the Gemalto HSM Client Workstation and the Gemalto HSM Server. From the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to the *Configuration Guide* > *[Step 7] Create a trusted link and register Client and Appliance with each other*. On the Gemalto HSM Client Workstation, run in LunaCM: `lunacm:> clientconfig deploy -c <IP Address of HSM Client Workstation> -n <IP Address of HSM Server> -par <Partition Name> -pw <Partition Password>` **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 of 5** > *"Sub-Step 5-C: Establishing a Trust Link between the Security Gateway (Cluster Members) and the Gemalto HSM Appliance Server" on page 86*.

Step 4 of 5: Creating the CA Certificate on the Gemalto HSM Server

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

2	Use the "`cmu generatekeypair`" command to create a key pair.
	From the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to the *Utilities Reference Guide* > *Certificate Management Utility (CMU)* > *cmu generatekeypair*.
	Example:
	`# cd /usr/safenet/lunaclient/bin`
	`# ./cmu generatekeypair -modulusBits=2048 -publicExponent=65537 -labelPublic="CAPublicKeyPairLabel" -labelPrivate="CAPrivateKeyPairLabel" -sign=T -verify=T`

3	Enter the password for the partition on Gemalto HSM Server (you configured it in *"Step 2 of 5: Configuring the Gemalto HSM Appliance Server to Work with Security Gateway" on page 79*). Example: `Enter a password for the token in slot 0:`
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	Examine the handles of the key pair you created.
	From the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to the *Utilities Reference Guide* > *Certificate Management Utility (CMU)* > *cmu list*.
	`# ./cmu list`
	Example output:
	`Enter password for token in slot 0 : <Password for the Partition>`
	`handle=17 label=CAPrivateKeyPairLabel`
	`handle=18 label=CAPublicKeyPairLabel`

Step	Instructions
6	Use the handle numbers from the previous step to create the CA certificate. From the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*, go to the *Utilities Reference Guide* > *Certificate Management Utility (CMU)* > *cmu selfsigncertificate* Example: `# ./cmu selfsigncertificate -privatehandle=17 -CN="www.myhsm.cp" -sha1WithRSA -startDate 20190720 -endDate 20240101 -serialNum=123456789abcdef`

7	Examine the handles of the CA certificate you created.
	`# ./cmu list`
	Example output:
	`Enter password for token in slot 0 : <Password for the Partition>`
	`handle=13 label=www.myhsm.cp`
	`handle=17 label=CAPrivateKeyPairLabel`
	`handle=18 label=CAPublicKeyPairLabel`
	Important - You use the numbers of these three handles when you configure the `$FWDIR/conf/hsm_configuration.C` file on the Check Point Security Gateway (Cluster Members).

Step 5 of 5: Configuring the Security Gateway to Work with the Gemalto HSM Server

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

This step has four sub-steps.

Sub-Step 5-A: Configuring HTTPS Inspection on the Security Gateway (Cluster Members) to work *without* the Gemalto HSM Server

Important:

- In a Cluster, you must configure all the Cluster Members in the same way.
- In a VSX environment, you must perform this step in the context of *every* Virtual System (on the VSX Gateway or *every* VSX Cluster Member).

Step	Instructions
1	In SmartConsole, enable and configure the HTTPS Inspection. See the *R80.40 Security Management Administration Guide* > Chapter HTTPS Inspection.

2	On the Security Gateway (*every* Cluster Member), disable the HSM in the `$FWDIR/conf/hsm_configuration.C` file.
	a. Connect to the command line.
	b. Log in to the Expert mode.
	c. Edit the file:
	`vi $FWDIR/conf/hsm_configuration.C`
	d. Configure the value "`no`" for the parameter "`enabled`":
	`:enabled ("no")`
	e. Save the changes in the file and exit the editor.

3	In SmartConsole, install the applicable Access Control Policy on the Security Gateway (Cluster).
4	Make sure that HTTPS Inspection works correctly without the HSM Server: a. From an internal computer, connect to any HTTPS web site. b. On the internal computer, in the web browser, you must receive the signed CA certificate from the Security Gateway (Cluster).

Sub-Step 5-B: Installing the Gemalto HSM Simplified Client Software Packages on the Security Gateway (Cluster Members)

Important:

- In a Cluster, you must configure all the Cluster Members in the same way.
- In a VSX environment, you must perform this step in the context of the VSX Gateway or *every* VSX Cluster Member (context of VS 0).

Step	Instructions
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	Transfer the software package to the Check Point Security Gateway (*every* Cluster Member) to some directory.
3	Connect to the command line on the Check Point Security Gateway (*every* Cluster Member).
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`

Sub-Step 5-C: Establishing a Trust Link between the Security Gateway (Cluster Members) and the Gemalto HSM Appliance Server

Important:

- In a Cluster, you must configure all the Cluster Members in the same way.
- In a VSX environment, you must perform this step in the context of the VSX Gateway or *every* VSX Cluster Member (context of VS 0).

Notes:

- For more information, see the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation*.
  For information about establishing a Trust Link, go to the *Appliance Administration Guide* > *Configuration without One-step NTLS* > *[Step 7] Create a Network Trust Link Between the Client and the Appliance*.
- If it is necessary to establish new Trust Link, you have to delete the current Trust Link.
  See *"Deleting a Trust Link with the HSM Server" on page 92*.

Step	Instructions
1	Connect to the command line on the Check Point Security Gateway (*every* Cluster Member).
2	Log in to the Expert mode.
3	Go to the SafeNet HSM Simplified Client installation directory: `# cd /usr/safenet/lunaclient/bin/`
4	Import the HSM Appliance Server certificate (`server.pem`) from the HSM Appliance to the Security Gateway (Cluster Member): **Important** - The period at the end is part of the syntax. `# scp admin@<IP Address of HSM Appliance>:server.pem .`
5	Register the HSM Appliance Server certificate (`server.pem`) on the Security Gateway (Cluster Member): `# ./vtl addServer -n <IP Address of HSM Appliance> -c server.pem`

Step	Instructions
6	Create a certificate and private key for the Check PointSecurity Gateway (Cluster Member): `# ./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 (Cluster Member)>Key.pem` ■ The certificate file is created and written to: `/usr/safenet/lunaclient/cert/client/<IP Address of CP Gateway (Cluster Member)>.pem`
7	Transfer the certificate file that you created from the Security Gateway (Cluster Member) to the HSM Appliance: **Important** - The colon at the end is part of the syntax. `# scp <IP Address of CP Gateway (Cluster Member)>.pem admin@<IP Address of HSM Appliance>:`
8	Connect to the command line on the HSM Appliance and log in to LunaSH.
9	Register the Check Point Security Gateway (*every* Cluster Member) on the HSM Appliance Server: `lunash:> client register -client <Desired Name of HSM Client> -ip <IP Address of CP Gateway (Cluster Member)>`
10	Restart the Network Trust Link service: `lunash:> service restart ntls`
11	Confirm the Check Point Security Gateway (Cluster Member) registration as a trusted HSM client: `lunash:> client list`

Step	Instructions
12	Assign the Check Point Security Gateway (Cluster Member) to the applicable partition: `lunash:> client assignPartition -client <Configured Name of HSM Client> -partition <Partition Name>`
13	Examine the partition access: `lunash:> client show -client <Configured Name of HSM Client>`
14	On the Check Point Security Gateway (*every* Cluster Member), examine the access to its partition on the HSM Appliance Server : `# ./vtl verify`

Sub-Step 5-D: Configuring HTTPS Inspection on the Security Gateway (Cluster Members) to work *with* the Gemalto HSM Server

Important:

- In a Cluster, you must configure all the Cluster Members in the same way.
- In a VSX environment, you must perform this step in the context of the VSX Gateway or *every* VSX Cluster Member (context of VS 0).

Notes:

- After you apply the HSM configuration for the first time, you can get an HSM connection error.
  Most common scenario is when you configure several Security Gateways (Cluster Members) to use the same HSM Server, and they access it at the same time.
  In this case:
  a. Run the "`fw fetch local`" command on the Security Gateway (Cluster Member) that has an HSM connection issue.
  In a VSX environment, run this command in the context of the problematic VSX Virtual System.
  b. When you see "`HSM on`" on the screen, continue to configure the next Security Gateway, Cluster Member, or VSX Virtual System.
- After any change in the `$FWDIR/conf/hsm_configuration.C` file, you must fetch the local policy (with the "`fw fetch local`" command) or install the policy on the Security Gateway (Cluster, VSX Virtual System) in SmartConsole.
- If the HSM Server is **not** available when you fetch the local policy or install the policy in SmartConsole, the HTTPS Inspection *cannot* inspect the Outbound HTTPS traffic. As a result, internal computers behind the Security Gateway (Cluster, VSX Virtual System) *cannot*access HTTPS web sites.
  In addition, see *"Disabling Communication from the Security Gateway to the HSM Server" on page 112*.

Step	Instructions
1	Connect to the command line on the Security Gateway (*every* Cluster Member).
2	Log in to the Expert mode.
3	Back up the `$FWDIR/conf/hsm_configuration.C` file: `cp -v $FWDIR/conf/hsm_configuration.C{,_BKP}`

Step	Instructions
4	Edit the `$FWDIR/conf/hsm_configuration.C` file: `vi $FWDIR/conf/hsm_configuration.C`

5	Configure the required values for these attributes:
	`(`
	`:enabled ("yes")`
	`:CA_cert_public_key_handle (<`**`Number of "Public" Handle for CA certificate`**`>)`
	`:CA_cert_private_key_handle (<`**`Number of "Private" Handle for CA certificate`**`>)`
	`:CA_cert_buffer_handle (<`**`Number of Handle for CA certificate`**`>)`
	`:token_id ("<`**`Password for Partition on Gemalto HSM Server`**`>")`
	`)`

	Notes:
	■ The "**:enabled ()**" attribute must have the value of either "**yes**" (to enable the HSM), or "**no**" (to disable the HSM).
	■ The "**:CA_cert_<XXX> ()**" attributes must have the required values of handles from the output of the "`cmu list`" command on the Gemalto HSM Server.
	See *"Step 4 of 5: Creating the CA Certificate on the Gemalto HSM Server" on page 82*.
	■ The "**:token_id ()**" attribute must have the contain the password for the partition on the Gemalto HSM Server.
	See *"Step 2 of 5: Configuring the Gemalto HSM Appliance Server to Work with Security Gateway" on page 79*.
	Example:
	`(`
	`:enabled ("yes")`
	`:CA_cert_public_key_handle (17)`
	`:CA_cert_private_key_handle (18)`
	`:CA_cert_buffer_handle (13)`
	`:token_id ("p@ssw0rd")`
	`)`

Step	Instructions
6	Fetch the local policy: `fw fetch local`
7	Make sure that the Security Gateway (*every* Cluster Member) can connect to the HSM Server and that HTTPS Inspection is activated successfully on the outbound traffic. Run this command: `cpstat https_inspection -f all` The output must show: ■ `HSM partition access (Accessible/Not Accessible): Accessible` ■ `Outbound status (HSM on/HSM off/HSM error): HSM on` For more information, see *"Monitoring HTTPS Inspection with HSM in CLI" on page 135*.
8	Make that HTTPS Inspection is activated successfully on the outbound traffic: a. From an internal computer, connect to any HTTPS web site. b. On the internal computer, in the web browser, you must receive the signed CA certificate from the HSM Server.

Additional Actions for a Gemalto HSM Server

Deleting a Trust Link with the HSM Server

If it is necessary to establish new Trust Link between a Check Point Security Gateway and an HSM Server, you have to delete the current Trust Link.

Use Case: When you replace or reconfigure a Check Point Security Gateway, or an HSM Server.

Step	Instructions

1	Delete the current Trust Link on the Check Point Security Gateway (*every* Cluster Member):
	a. Connect to the command line.
	b. Log in to the Expert mode.
	c. Go to the SafeNet HSM Simplified Client installation directory:
	`cd /usr/safenet/lunaclient/bin/`
	d. Delete the current Trust Link:
	`./vtl deleteServer -n <IP Address of HSM Server>`

2	Delete the current Trust Link on the HSM Appliance:
	a. Connect to the HSM Appliance over SSH.
	b. Examine the list of configured HSM Client Workstations:
	`lunash:> client list`
	c. Delete the Check Point HSM Client Workstation:
	`lunash:> client delete -client <Name of HSM Client>`

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

Configuring a Second Interface on a Gemalto HSM Appliance for NTLS

Step	Instructions
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 the *Gemalto SafeNet Network HSM 6.2.2 Product Documentation > LunaSH Command Reference Guide > LunaSH Commands*.

Working with FutureX HSM

Use this workflow to configure a Check Point Security Gateway / ClusterXL to work with the **FutureX HSM Server**.

Prerequisites

FutureX Software Packages

The FutureX vendor supplies all these packages.

Package	Files	Description
FutureX PKCS11 Library	■ `fxpkcs11-windows-4.20-4afd.zip` ■ `fxpkcs11-redhat-4.20-4afd.tar` ■ `fxpkcs11-linux-4.20-4afd.tar` ■ `fxpkcs11-mac-4.20-4afd.tar`	Contains the FutureX PKCS #11 Library. Install on the: ■ FutureX HSM Client Workstation ■ Check Point Security Gateway / *each* Cluster Member.
FutureX CLI Utility	■ `fxcl-hsm-windows-1.2.4.2-37a8.zip` ■ `fxcl-hsm-redhat-1.2.4.2-37a8.tar` ■ `fxcl-hsm-linux-1.2.4.2-37a8.tar` ■ `fxcl-hsm-mac-1.2.4.2-37a8.tar` ■ `fxcli-hsm-commands.txt`	Contains the FutureX CLI Utility to manage keys and certificates. Install on the FutureX HSM Client Workstation.
FutureX Certificates		FutureX certificates for trust between the FutureX HSM Client Workstation and the FutureX HSM Server.

Configuration Steps

Use this workflow to configure a Check Point Security Gateway / ClusterXL to work with the **FutureX HSM Server**.

Step 1 of 3: Configuring the FutureX HSM Client Workstation

You use the FutureX HSM Client Workstation to:

- Create a CA Certificate on the FutureX HSM Server. The Check Point Security Gateway uses this CA Certificate for HTTPS Inspection to store and access SSL keys on the FutureX HSM Server.
- Manage keys for fake certificate the Check Point Security Gateway / ClusterXLcreated.

Step	Instructions
1	Install a computer to use as a FutureX HSM Client Workstation. Get the applicable HSM Client package from the FutureX vendor. A FutureX HSM Client Workstation can run these operating systems (for more information, contact the FutureX vendor): ■ Windows (contact FutureX vendor to download and install the "**FXTools**" package from the FutureX portal). ■ Red Hat Linux ■ Ubuntu Linux ■ Debian Linux ■ macOS
2	Transfer the applicable **FutureX PKCS #11 Library** package to the FutureX HSM Client Workstation. ■ For Windows OS: **fxpkcs11-windows-<BUILD>.zip** ■ For Red Hat Linux OS: **fxpkcs11-redhat-<BUILD>.tar** ■ For Ubuntu and Debian Linux OS: **fxpkcs11-linux-<BUILD>.tar** ■ For macOS: **fxpkcs11-mac-<BUILD>.tar** **Important** - Make sure to transfer the file in the binary mode.

Step	Instructions
3	Extract the contents of the **FutureX PKCS #11 Library** package to some directory on the FutureX HSM Client Workstation. In the instructions below, we show this directory as: **<PKCS#11 Dir>**. **Important:** ■ The FutureX PKCS #11 Library package (**fxpkcs11-<OS>-<BUILD>**) contains the nested directory called "**fxpkcs11**". You must extract the contents of this nested directory "**fxpkcs11**" into the **<PKCS#11 Dir>** directory. ■ The nested directory "**fxpkcs11**" contains the nested directories called "**x64**" (for 64-bit OS) and "**x86**" (for 32-bit OS). You must extract the contents of the applicable nested directory "**x64**" or "**x86**" into the **<PKCS#11 Dir>** directory.
4	Transfer the certificates you received from the FutureX vendor to some directory on the FutureX HSM Client Workstation.

Step	Instructions

5	Prepare the HSM Client to work with the PKCS#11 manager:
	a. On a Linux-based HSM Client, install the OpenSSL package:
	■ On Ubuntu and Debian Linux, run:
	`sudo apt-get install openssl`
	■ On Red Hat Linux, run:
	`sudo yum install openssl`
	b. Make sure this FutureX PKCS #11 Library file is located in the **<PKCS#11 Dir>** directory:
	■ On Linux OS:
	`libfxpkcs11.so`
	■ On Windows OS:
	`fxpkcs11.dll`
	c. Make sure the configuration file **fxpkcs11.cfg** is located in the applicable directory:
	■ On Linux OS:
	Transfer this file from the **<PKCS#11 Dir>** directory to the **/etc/** directory
	(you must edit the copied file in the **/etc/** directory).
	■ On Windows OS:
	Keep this file in the **<PKCS#11 Dir>** directory.
	d. Configure these settings in the file **fxpkcs11.cfg**:
	■ **<LOG-FILE>**
	Set the path to the log file in this attribute.
	■ **<ADDRESS>**
	Set the IP address of the FutureX HSM Server in this attribute.
	■ **<PROD-PORT>**
	Set the port on the FutureX HSM Server in this attribute.
	You can use the default port 9100, or configure a different port.
	If you use a Cloud FutureX HSM, get the port number from the FutureX Support.
	Additional related attributes:
	■ **<PROD-TLS-CA>**
	Contains the path to the Certificate Authority certificate file.
	This attribute can appear multiple times.
	You can put all the certificates of the CA chain.
	■ **<PROD-TLS-CERT>**
	Contains the path to the client certificate file.
	■ **<PROD-TLS-KEY>**
	Contains the path to the client private key file.

Step	Instructions

6	Test the PKCS#11 Library:
	a. To test the configuration, run the tool **configTest** from the **<PKCS#11 Dir>** directory.
	b. To manage keys, run the tool **PKCS11Manager** from the **<PKCS#11 Dir>** directory.
	c. Examine the log file you configured in the **<LOG-FILE>** attribute in the **fxpkcs11.cfg** file.
	Important - If you have problems with the location of the configuration file or the PKCS #11 Library file, you can set these environmental variables:
	■ **FXPKCS11_CFG** to contain the full path to the configuration file **fxpkcs11.cfg**
	■ **FXPKCS11_MODULE** to contain the full path to the PKCS #11 Library file
	To set an environmental variable:
	■ On Linux OS, use this command:
	`export VARIABLE=VALUE`
	Example:
	`export FXPKCS11_CFG=/home/user/fxpkcs11.cfg`
	■ On Windows OS, use this command:
	`set VARIABLE=VALUE`
	Example:
	`set FXPKCS11_`
	`CFG=C:\Users\Futurex\Desktop\fxpkcs11.cfg`

7	For more information about the configuration of PKCS#11 on the FutureX HSM Client Workstation: a. Log in to the FutureX portal. b. Go to: *DEVELOPER DOCUMENTATION >* *GENERAL PURPOSE >* *General Purpose Technical Reference >* *PKCS #11 Technical Reference*

Step	Instructions
8	Transfer the applicable FutureX CLI Utility package to the FutureX HSM Client Workstation. ■ For Windows OS: **fxcl-hsm-windows-<BUILD>.zip** ■ For Red Hat Linux OS: **fxcl-hsm-redhat-<BUILD>.tar** ■ For Ubuntu and Debian Linux OS: **fxcl-hsm-linux-<BUILD>.tar** ■ For macOS: **fxcl-hsm-mac-<BUILD>.tar** **Important** - Make sure to transfer the file in the binary mode.
9	Extract the contents of the FutureX CLI Utility package to some directory on the FutureX HSM Client Workstation. In the instructions below, we show this directory as: **<CLI Dir>**. **Important:** ■ The FutureX CLI Utility package (**fxcl-hsm-<OS>-<BUILD>**) contains the nested directory called "**fxcl**". ■ The nested directory **fxcl** contains the nested directories called "**x64**" (for 64-bit OS) and "**x86**" (for 32-bit OS). ■ The nested directories **x64** and **x86** contain the nested directories called "**OpenSSL-1.0.x**" and "**OpenSSL-1.1.x**". You must extract the contents of the applicable nested directory "**OpenSSL-1.0.x**" or "**OpenSSL-1.1.x**" into the **<CLI Dir>** directory. Administrator decides, which version of the OpenSSL to use (for more information, contact the FutureX vendor).
10	Transfer these certificates to the **<CLI Dir>** directory on the FutureX HSM Client Workstation: ■ The Client certificate (denoted below as **<Client Certificate>**) ■ The CA certificate (denoted below as **<CA Certificate>**)

Step	Instructions

11	Establish a connection between the FutureX HSM Client and the FutureX HSM Server:
	a. Go to the **<CLI Dir>** directory.
	b. Start the shell:
	`fxcli-hsm`
	c. Run these commands in the order they are listed:
	`tls pki -f <Client Certificate> -p safest`
	`tls ca -f <CA Certificate>`
	`connect tcp -c <IP Address of URL of HSM Server>:<Port on HSM Server>`
	`login user -u <Username> -p <Password (default is "safest")>`
	`exit`

12	You can use these tools on the FutureX HSM Client Workstation to manage keys and certificates that are stored on the FutureX HSM Server: a. **PKCS11Manager** ■ Run this command from the **<PKCS#11 Dir>** directory. ■ This tool can create keys and browse the content of the HSM partition (that stores keys and certificates). ■ Follow the tool's menu to see the available options. b. **fxcli-hsm** ■ Run this command from the **<CLI Dir>** directory. ■ To see all available commands in this shell, run: `help` ■ To see all available options for a command in this shell, either run only the command, or the command with the "`-h`" option.

Step 2 of 3: Creating the CA Certificate on the FutureX HSM Server

Step	Instructions
1	On the FutureX HSM Client Workstation, open the FutureX CLI utility.

2	Get the list of available slots.
	Run one of these commands:
	`keytable list`
	`keytable reload`

Step	Instructions

3	Generate the key pair for the CA certificate:
	`generate -a rsa -b 2048 --slot <Slot or Label of CA Certificate> --name <Name of CA Certificate Private Key File> --tpk-slot <Slot or Label of CA Certificate Public Key> --tpk-name <Name of CA Certificate Public Key File> -u sign,verify`
	Example:
	`generate -a RSA -b 2048 --slot 0 --name CAPrivateKey --tpk-slot 1 --tpk-name CAPublicKey -u sign,verify`
	Important - Do **not** use the "`... slot next`" option, because it can override keys for a fake certificate the Check Point Security Gateway / ClusterXL created.

4	Generate the CA certificate: `x509 sign --private-slot <Slot or Label of Private Key> --dn "<Distinguished Name of CA Certificate>" --ca 1 --key-usage DigitalSignature --key-usage CrlSign --key-usage KeyCertSign --save-slot <Slot to Save the CA Certificate> --save-name <Label of CA Certificate File> -o <Full Path and Name of CA Certificate File>.cer --validity-period '<Period>'` Example: `x509 sign --private-slot 0 --dn "CN=www.futurexhsm.cp" --ca 1 --key-usage DigitalSignature --key-usage KeyCertSign --key-usage CrlSign --save-slot 2 --save-name CACert -o Z:\FutureXhsm.cer --validity-period '5 years'` **Important** - Do **not** use the "`... slot next`" option, because it can override keys for a fake certificate the Check Point Security Gateway / ClusterXL created.

5	Get the list of slots used for the CA certificate and CA certificate's key pair.
	Run one of these commands:
	`keytable list`
	`keytable reload`
	Note - The command "`keytable list`" shows the slot numbers as the PKCS#11 handles plus one. For example, it shows slot 0 as handle 1, slot 1 as handle 2, and so on.

Step	Instructions

6	Write down the handles of the:
	■ CA certificate
	■ CA certificate public key
	■ CA certificate private key
	Example:
	`CAPublicKey (1)`
	`CAPrivateKey (2)`
	`CACert (3)`
	Important - You use the numbers of these three handles when you configure the `$FWDIR/conf/hsm_configuration.C` file on the Check Point Security Gateway / ClusterXL.

Step 3 of 3: Configuring the Security Gateway / ClusterXL to Work with the FutureX HSM Server

This step has four sub-steps.

Sub-Step 3-A: Configuring HTTPS Inspection on the Security Gateway / Cluster Members to work *without* the FutureX HSM Server

Important:

- In a Cluster, you must configure all the Cluster Members in the same way.
- In a VSX environment, you must perform this step in the context of *each* Virtual System (on the VSX Gateway or *each* VSX Cluster Member).

Step	Instructions
1	In SmartConsole, configure the HTTPS Inspection. See the *R80.40 Security Management Administration Guide* > Chapter HTTPS Inspection.

2	On the Security Gateway / *each* Cluster Member, disable the HSM in the `$FWDIR/conf/hsm_configuration.C` file.
	a. Connect to the command line.
	b. Log in to the Expert mode.
	c. Edit the file:
	`vi $FWDIR/conf/hsm_configuration.C`
	d. Configure the value "`no`" for the parameter "`enabled`":
	`:enabled ("no")`
	e. Save the changes in the file and exit the editor.

3	In SmartConsole, install the applicable Access Control Policy on the Security Gateway / ClusterXL object.
4	Make sure that HTTPS Inspection works correctly without the HSM Server: a. From an internal computer, connect to any HTTPS web site. b. On the internal computer, in the web browser, you must receive the signed CA certificate from the Security Gateway / ClusterXL.

Sub-Step 3-B: Installing the required software packages on the Security Gateway (Cluster Members)

Important:

- In a Cluster, you must configure all the Cluster Members in the same way.
- In a VSX environment, you must perform this step in the context of the VSX Gateway or *each* VSX Cluster Member (context of VS 0).

Step	Instructions
1	Transfer the FutureX PKCS #11 binary files to the Security Gateway / *each* Cluster Member: a. Open the **FutureX PKCS#11 Library** package. b. Go to this folder: **fxpkcs11-linux-<BUILD>** > **fxpkcs11** > **x86** > **OpenSSL-1.1.x** c. Transfer these files to the Security Gateway (*each* Cluster Member) to the **/usr/lib/hsm_client/** directory: ▪ **libfxpkcs11.so** ▪ **configTest** **Important** - Make sure to transfer the files in the binary mode.
2	Transfer the FutureX PKCS #11 configuration file to the Security Gateway / *each* Cluster Member: a. Open the **FutureX PKCS#11 Library** package. b. Go to this folder: **fxpkcs11-linux-<BUILD>** > **fxpkcs11** c. Transfer this file to the Security Gateway / *each* Cluster Member to the **/etc/** directory: **fxpkcs11.cfg**

Sub-Step 3-C: Configuring a connection between the Security Gateway / Cluster Members and the FutureX HSM Server

To establish a connection between a Check PointSecurity Gateway (HSM client) to a FutureX HSM server, you must create certificate files for the TLS authentication between the Check PointSecurity Gateway and the FutureX HSM server. These are the options to create the required certificate files:

- Create the certificates on the HSM (the most common method).
- Get the certificates from the FutureX vendor.
- Enabling the "`Anonymous`" setting on the HSM server, so that mutual authentication is not required (see the FutureX Integration Guide).

Important:

- In a Cluster, you must configure all the Cluster Members in the same way.
- In a VSX environment, you must perform this step in the context of the VSX Gateway or *each* VSX Cluster Member (context of VS 0).

Step	Instructions
1	Transfer the FutureX certificate files you received from the FutureX vendor to the Security Gateway / *each* Cluster Member to the **/usr/futurex/** directory.
2	Connect to the command line on the Security Gateway / *each* Cluster Member.
3	Log in to the Expert mode.
4	Back up the configuration file **/etc/fxpkcs11.cfg**: `cp -v /etc/fxpkcs11.cfg{,_BKP}`
5	Edit the configuration file **/etc/fxpkcs11.cfg**: `vi /etc/fxpkcs11.cfg`

<table>
<tr><th>Step</th><th>Instructions</th></tr>
<tr><td>6</td><td>Configure these attribute values:
<table>
<tr><th>Attribute</th><th>Attribute Value</th></tr>
<tr><td><LOG-FILE></td><td><code>/var/log/fxpkcs11.log</code></td></tr>
<tr><td><ADDRESS></td><td>The IP address (or URL) of the FutureX HSM Server.</td></tr>
<tr><td><PROD-PORT></td><td>The port on the FutureX HSM Server.
You can use the default port 9100, or configure a different port.</td></tr>
<tr><td><PROD-TLS-CA></td><td>The path to the Certificate Authority certificate file.
This attribute can appear multiple times.</td></tr>
<tr><td><PROD-TLS-CERT></td><td>The path to the client certificate file.</td></tr>
<tr><td><PROD-TLS-KEY></td><td>The path to the client private key file.</td></tr>
</table></td></tr>
<tr><td>7</td><td>Save the changes in the file and exit the editor.</td></tr>
<tr><td>8</td><td>Create the required symbolic link:
<code>ln -s /var/log/fxpkcs11.log /tmp/fxpkcs11.log</code></td></tr>
</table>

Sub-Step 3-D: Configuring HTTPS Inspection on the Security Gateway / Cluster Members to work *with* the FutureX HSM Server

Important:

- In a Cluster, you must configure all the Cluster Members in the same way.
- In a VSX environment, you must perform this step in the context of the VSX Gateway or *each* VSX Cluster Member (context of VS 0).

Notes:

- After you apply the HSM configuration for the first time, you can get an HSM connection error.
  Most common scenario is when you configure several Security Gateways / Cluster Members to use the same HSM Server, and they access it at the same time.
  In this case:
  a. Run the "`fw fetch local`" command on the Security Gateway / Cluster Member that has an HSM connection issue.
  In a VSX environment, run this command in the context of the problematic VSX Virtual System.
  b. When you see "`HSM on`" on the screen, continue to configure the next Security Gateway / *each* Cluster Member / VSX Virtual System.
- After any change in the `$FWDIR/conf/hsm_configuration.C` file, you must do one of these:
  - Fetch the local policy with the "`fw fetch local`" command
  - In SmartConsole, install the policy on the Security Gateway / ClusterXL / VSX Virtual System object.
- If the HSM Server is **not** available when you fetch the local policy or install the policy in SmartConsole, the HTTPS Inspection **cannot** inspect the Outbound HTTPS traffic. As a result, internal computers behind the Security Gateway / ClusterXL / VSX Virtual System **cannot** access HTTPS web sites.
  In addition, see *"Disabling Communication from the Security Gateway to the HSM Server" on page 112*.

Step	Instructions
1	Connect to the command line on the Security Gateway / *each* Cluster Member.
2	Log in to the Expert mode.
3	Back up the `$FWDIR/conf/hsm_configuration.C` file: `cp -v $FWDIR/conf/hsm_configuration.C{,_BKP}`
4	Edit the `$FWDIR/conf/hsm_configuration.C` file: `vi $FWDIR/conf/hsm_configuration.C`

Step	Instructions

5	Configure the required values for these attributes:
	`(`
	`:enabled ("yes")`
	`:hsm_vendor_name ("FutureX HSM")`
	`:lib_filename ("")`
	`:CA_cert_public_key_handle (<`**`Number of "CAPublicKey" Handle for CA certificate`**`>)`
	`:CA_cert_private_key_handle (<`**`Number of "CAPrivateKey" Handle for CA certificate`**`>)`
	`:CA_cert_buffer_handle (<`**`Number of "CACert" Handle for CA certificate`**`>)`
	`:token_id ("<`**`Password for Partition on FutureX HSM Server`**`>")`
	`)`

Step	Instructions

	Notes:
	■ The "**:enabled ()**" attribute must have the value of either "**yes**" (to enable the HSM), or "**no**" (to disable the HSM).
	■ The "**:hsm_vendor_name ()**" attribute must contain the string "**FutureX HSM**".
	■ The "**:lib_filename ()**" attribute must contain the full path to the file **libfxpkcs11.so** (from the FutureX PKCS #11 Library) on the Security Gateway / *each* Cluster Member.
	You must configure this full path explicitly, if this file is **not** located at the default path: **/usr/lib/libfxpkcs11.so**
	■ The "**:CA_cert_<XXX> ()**" attributes must have the required values of handles from the output of the "`keytable`" command on the FutureX HSM Server.
	See *"Step 2 of 3: Creating the CA Certificate on the FutureX HSM Server" on page 100*.
	■ The "**:token_id ()**" attribute must have the contain the password for the partition on the FutureX HSM Server.
	Example:
	`(:enabled ("yes")`
	`:hsm_vendor_name ("FutureX HSM")`
	`:lib_filename ("")`
	`:CA_cert_public_key_handle (1)`
	`:CA_cert_private_key_handle (2)`
	`:CA_cert_buffer_handle (3)`
	`:token_id ("p@ssw0rd")`
	`)`

Step	Instructions

6	Apply the new configuration.
	■ If you explicitly defined (or changed) the value of the "**:hsm_vendor_name ()**" attribute to the string "**FutureX HSM**", then restart all Check Point services with this command:
	`cprestart`
	Important - This blocks all traffic until all services restart. In a cluster, this can cause a failover.
	■ If the value of the "**:hsm_vendor_name ()**" attribute already contained the string "**FutureX HSM**", then fetch the local policy with this command:
	`fw fetch local`

7	Make sure that the Security Gateway / *each* Cluster Member can connect to the HSM Server and that HTTPS Inspection is activated successfully on the outbound traffic. Run this command: `cpstat https_inspection -f all` The output must show: ■ `HSM partition access (Accessible/Not Accessible): Accessible` ■ `Outbound status (HSM on/HSM off/HSM error): HSM on` For more information, see *"Monitoring HTTPS Inspection with HSM in CLI" on page 135*.
8	Make that HTTPS Inspection is activated successfully on the outbound traffic: a. From an internal computer, connect to any HTTPS web site. b. On the internal computer, in the web browser, you must receive the signed CA certificate from the HSM Server.

Note - If there is a connectivity issue from the Check Point Security Gateway / Cluster Member to the FutureX HSM Server, then perform these steps on the Security Gateway / Cluster Member:

1. Examine the **/var/log/fxpkcs11.log** file.
   If you do not see a root cause in this log file, continue to the next step to configure verbose logs.
2. Configure these logging settings in the **/etc/fxpkcs11.cfg** file to see more information in the log file:
   - **LOG-TRAFFIC: YES**
   - **LOG-MODE: INFO**
     or
     LOG-MODE: ERROR

Disabling Communication from the Security Gateway to the HSM Server

You can disable communication from the Check Point Security Gateway / Cluster Members to an HSM Server. For example, when the HSM Server is under maintenance.

Important:

- In a Cluster, you must configure all the Cluster Members in the same way.
- In a VSX environment, you must perform this step in the context of *each* Virtual System (on the VSX Gateway or *each* VSX Cluster Member).

Step	Instructions
1	Connect to the command line on the Security Gateway / *each* Cluster Member.
2	Log in to the Expert mode.
3	Edit the `$FWDIR/conf/hsm_configuration.C` file: `vi $FWDIR/conf/hsm_configuration.C`
4	Configure the value "`no`" for the parameter "`enabled`": `:enabled ("no")`
5	Save the changes in the file and exit the editor.
6	Fetch the local policy: `fw fetch local`

Monitoring HTTPS Inspection When Security Gateway Works with HSM

When HTTPS Inspection daemon **wstlsd** initializes on a Check Point Security Gateway / Cluster Member, it checks whether this Security Gateway / Cluster Member is configured to work with an HSM Server.

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

  See *"Monitoring HTTPS Inspection with HSM in SmartConsole Logs" on page 114*.

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

  See *"Monitoring HTTPS Inspection with HSM over SNMP" on page 118*.

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

  See *"Monitoring HTTPS Inspection with HSM in CLI" on page 135*.

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 Server in SmartConsole:

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

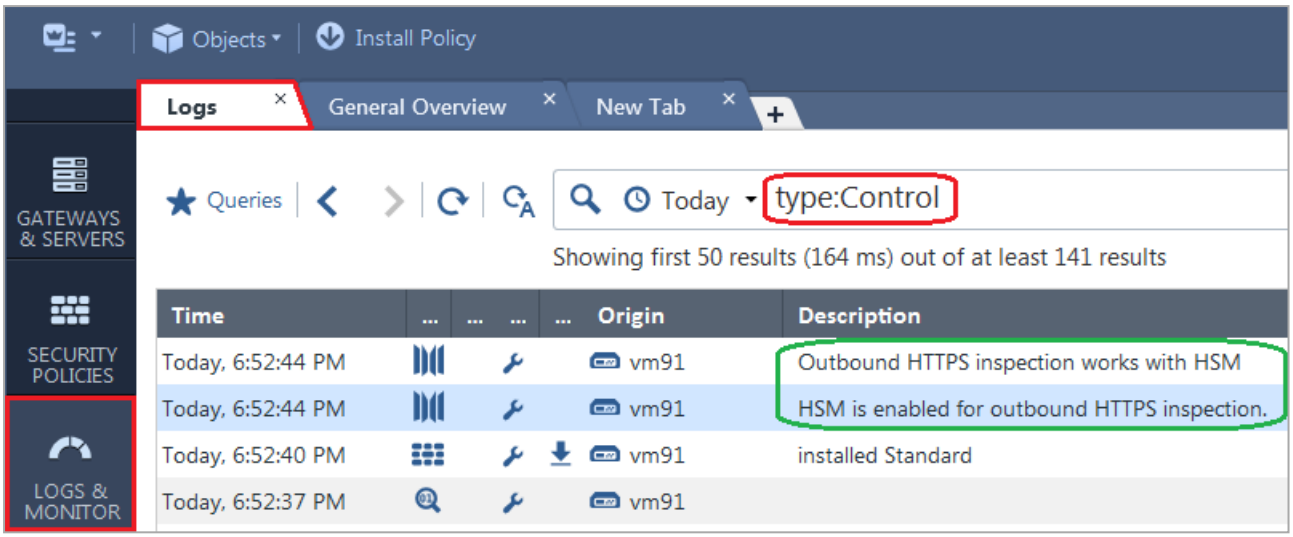
Possible logs are:

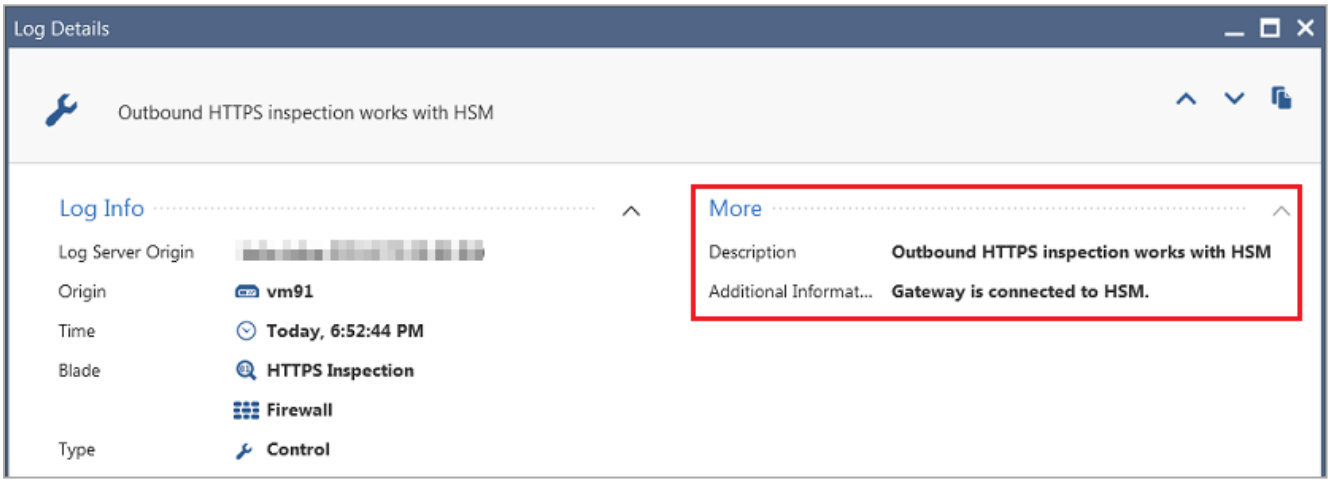
Log Description	Log Additional Information	Explanation
`HSM is enabled for outbound HTTPS inspection with <HSM Vendor>`		■ The value of the **:enabled()** attribute is set to **"yes"** in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway / Cluster Member. ■ The *<HSM Vendor>* is the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

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

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 HSM Server on the Security Gateway / Cluster Member 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 / Cluster Member.
	`Off`	HTTPS Inspection feature is **not** configured on the Security Gateway / Cluster Member.

"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 / Cluster Member.
	`HTTPS Inspection is off`	HTTPS Inspection feature is **not** configured on the Security Gateway / Cluster Member.

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

"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 with <HSM Vendor>`	■ The value of the **:enabled()** attribute is set to **"yes"** in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway / Cluster Member. ■ The *<HSM Vendor>* is the value of the **":hsm_vendor_name()"** attribute in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway / Cluster Member.

SNMP OID	Returned strings	Explanation
	`HSM is disabled for HTTPS inspection`	One of these: ■ The HSM Client software packages are **not** installed on the Security Gateway / Cluster Member. ■ The `$FWDIR/conf/hsm_configuration.C` file does **not** exist on the Security Gateway / Cluster Member. ■ The HTTPS Inspection daemon **wstlsd** could **not** read the value of the "**:enabled()**" attribute in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway / Cluster Member. ■ The "**:enabled()**" attribute is corrupted in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway / Cluster Member.

SNMP OID	Returned strings	Explanation
	■ The HSM Client software packages are **not** installed on the Security Gateway / Cluster Member. ■ The `$FWDIR/conf/hsm_configuration.C` file does **not** exist on the Security Gateway / Cluster Member.	

SNMP OID	Returned strings	Explanation
	■ The HTTPS Inspection daemon **wstlsd** could **not** read the value of the "**:enabled ()**" attribute in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway / Cluster Member.	

SNMP OID	Returned strings	Explanation
	■ The "**:enabled ()**" attribute is corrupted in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway / Cluster Member. **Important** - In these cases, outbound HTTPS Inspection works without the HSM Server, and SSL keys are stored on the Security Gateway / Cluster Member.	

"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 / Cluster Member could **not** check the access to its partition on the HSM Server. Most probably, because HSM configuration is disabled on the Security Gateway / Cluster Member.
	`Accessible`	Security Gateway / Cluster Member could access its partition on the HSM Server.
	`Not Accessible`	Security Gateway / Cluster Member could **not** access its partition on the HSM Server because of 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 / Cluster Member could **not** check the access to its partition on the HSM Server.
	`Gateway can access HSM partition for HTTPS inspection`	Security Gateway / Cluster Member could access its partition on the HSM Server.
	`Gateway cannot access HSM partition for HTTPS inspection: <Error Message>`	Security Gateway / Cluster Member could **not** access its partition on the HSM Server because of 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	`N / A`	When the HTTPS Inspection daemon **wstlsd** starts, it is necessary to wait for one minute or less, until you can get the actual status.

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

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 / Cluster Member. 2. An error occurred. **Important** - In this case, outbound HTTPS Inspection does **not** work, and HTTPS traffic does **not** pass through.

Note - The conditions for the returned strings are calculated on the Security Gateway / Cluster Member 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	`Cannot get HTTPS inspection outbound status. Process may be under initialization. Please try again in a minute.`	When the HTTPS Inspection daemon **wstlsd** starts, it is necessary to wait for one minute or less, until you can get the actual status.

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

SNMP OID	Returned strings	Explanation
	`Outbound HTTPS inspection is off due to HSM error: <Error Message>`	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 / Cluster Member. 2. An error occurred. **Important** - In this case, outbound HTTPS Inspection does **not** work, and HTTPS traffic does **not** pass through. 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`

SNMP OID	Returned strings	Explanation
		■ `Error generating key pair on HSM server`

Note - The conditions for the returned strings are calculated on the Security Gateway / Cluster Member 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 with <HSM Vendor>`" 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 with Gemalto HSM
.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 with
Gemalto HSM
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 / Cluster Member to see the HTTPS Inspection status and the status of connection to the HSM 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 with
Gemalto HSM
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 with
Gemalto HSM
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 / Cluster Member.
	`Off`	HTTPS Inspection feature is **not** configured on the Security Gateway / Cluster Member.

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 / Cluster Member.
	`HTTPS Inspection is off`	HTTPS Inspection feature is **not** configured on the Security Gateway / Cluster Member.

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

Explanation about the "HSM configuration status description"

Item	Possible returned strings	Explanation
`HSM enabled description`	`HSM is enabled for HTTPS inspection with <HSM Vendor>`	■ The value of the **:enabled()** attribute is set to **"yes"** in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway / Cluster Member. ■ The *<HSM Vendor>* is the value of the "**:hsm_vendor_name ()**" attribute in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway / Cluster Member.
	`HSM is disabled for HTTPS inspection`	One of these: ■ The HSM Client software packages are **not** installed on the Security Gateway / Cluster Member. ■ The `$FWDIR/conf/hsm_configuration.C` file does **not** exist on the Security Gateway / Cluster Member. ■ The HTTPS Inspection daemon **wstlsd** could **not** read the value of the "**:enabled ()**" attribute in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway / Cluster Member. ■ The "**:enabled()**" attribute is corrupted in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway / Cluster Member. **Important** - In these cases, outbound HTTPS Inspection works without the HSM Server, and SSL keys are stored on the Security Gateway (Cluster Members).

Explanation about the "HSM partition access status"

Item	Possible returned strings	Explanation
`HSM partition access (Accessible/Not Accessible)`	`N/A`	Security Gateway / Cluster Member could **not** check the access to its partition on the HSM Server.
	`Accessible`	Security Gateway / Cluster Member could access its partition on the HSM Server.
	`Not Accessible`	Security Gateway / Cluster Member could **not** access its partition on the HSM Server because of an error. **Important** - In this case, outbound HTTPS Inspection does **not** work, and HTTPS traffic does **not** pass through.

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 / Cluster Member could **not** check the access to its partition on the HSM Server. Most probably, because HSM configuration is disabled on the Security Gateway / Cluster Member.
	`Gateway can access HSM partition for HTTPS inspection`	Security Gateway / Cluster Member could access its partition on the HSM Server.
	`Gateway cannot access HSM partition for HTTPS inspection: <Error Message>`	Security Gateway / Cluster Member could **not** access its partition on the HSM Server because of 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 / Cluster Member. 2. An error occurred. 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` **Important** - In this case, outbound HTTPS Inspection does **not** work, and HTTPS traffic does **not** pass through.

Explanation about the "Outbound HTTPS Inspection status"

Item	Possible returned strings	Explanation
`Outbound status (HSM on/HSM off/HSM error)`	`N / A`	When the HTTPS Inspection daemon **wstlsd** starts, it is necessary to wait for one minute or less, until you can get the actual status.

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

Note - The conditions for the returned strings are calculated on the Security Gateway / Cluster Member 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`	`Cannot get HTTPS inspection outbound status. Process may be under initialization. Please try again in a minute.`	When the HTTPS Inspection daemon **wstlsd** starts, it is necessary to wait for one minute or less, until you can get the actual status.
	`Outbound HTTPS inspection works with HSM`	All these conditions were met: 1. The value of the **:enabled()** attribute is set to **"yes"** in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway / Cluster Member. 2. Security Gateway / Cluster Member could connect to the HSM Server.
	`Outbound HTTPS inspection works without HSM`	The value of the **:enabled()** attribute is set to **"no"** in the `$FWDIR/conf/hsm_configuration.C` file on the Security Gateway / Cluster Member, or this file does **not** exist.

Item	Possible returned strings	Explanation
	`Outbound HTTPS inspection is off due to HSM error: <Error Message>`	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 / Cluster Member. 2. An error occurred. 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` **Important** - In this case, outbound HTTPS Inspection does **not** work, and HTTPS traffic does **not** pass through.

Note - The conditions for the returned strings are calculated on the Security Gateway / Cluster Member 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:

Important - ISP Redundancy is **not** supported if Dynamic Routing is configured (Known Limitation PMTR-68991).

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

Introduction

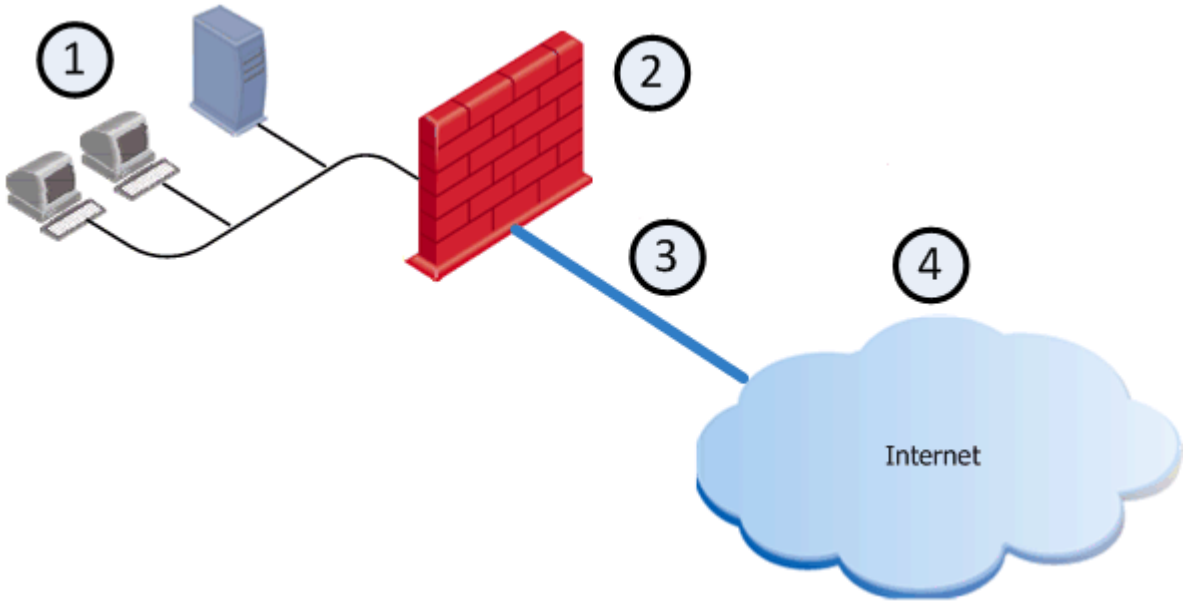
ISP Redundancy connects a Security Gateway to the Internet through redundant Internet Service Provider (ISP) links.

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.

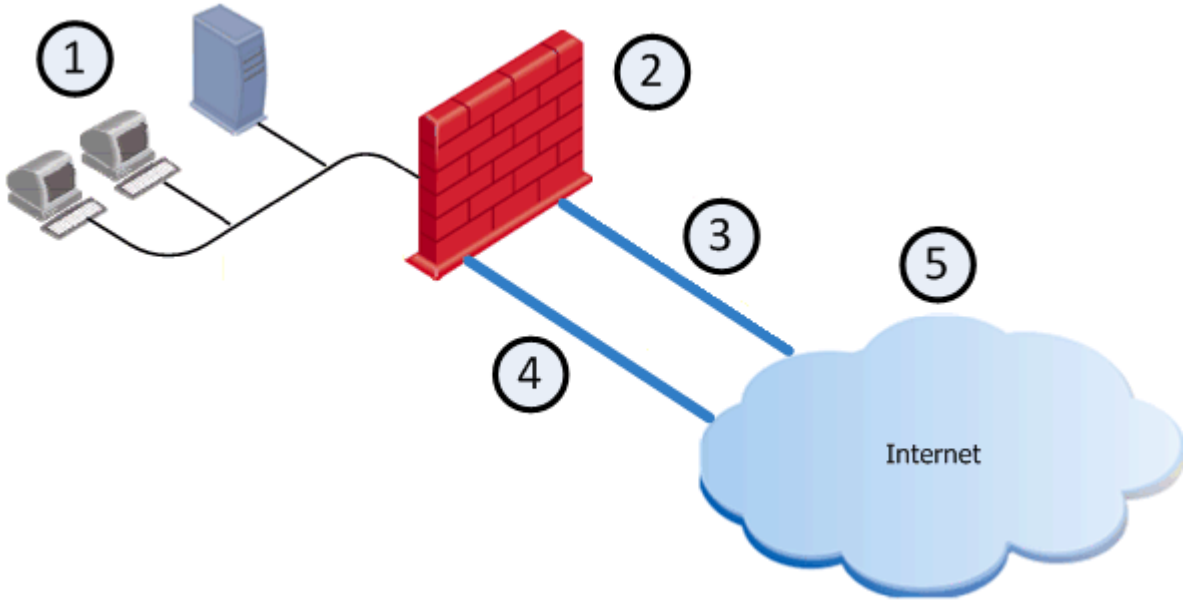
Example of a typical deployment with a single ISP link



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.



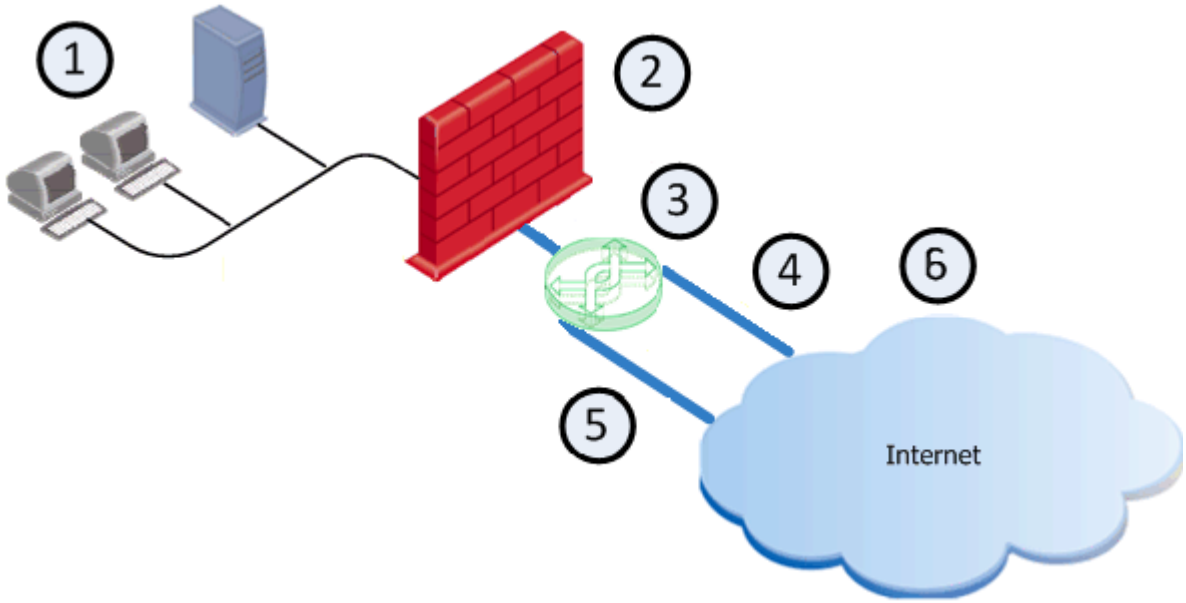
Item	Description
1	Internal network
2	Security Gateway
3	ISP A
4	ISP B
5	Internet

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 161*).

  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 161*).

     - 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 the 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 must 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** work 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.

Warning - We do **not** recommend that you make any changes in this script.

Mirror and Decrypt

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

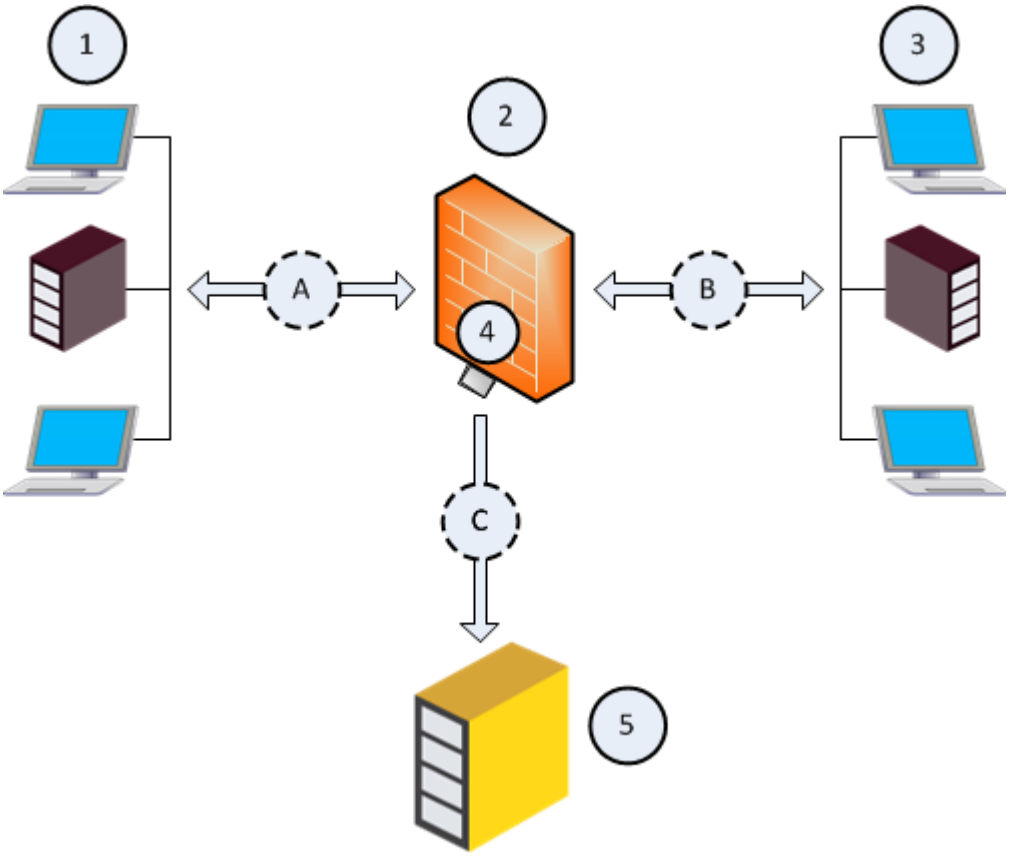
Action	Description
Only mirror of all traffic	Your Security Gateway or Cluster clones all traffic (including HTTPS without decryption) that passes through it, and sends it out of the designated physical interface.
Mirror and Decrypt of HTTPS traffic	Your Security Gateway or Cluster clones all HTTPS traffic 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

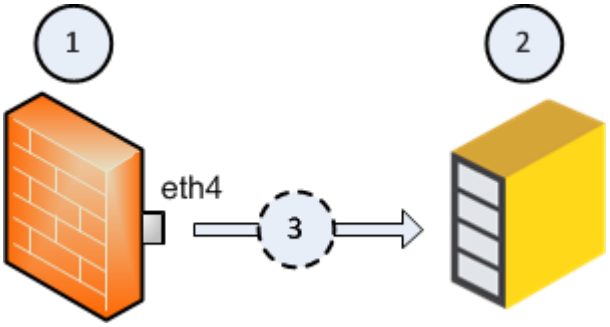
Traffic	Source MAC address of the decrypted and mirrored packets the Security Gateway and Cluster Members send
Mirror only of all traffic	MAC address of the designated physical interface.
Mirror and Decrypt of HTTPS traffic	00:00:00:00:00:00:

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 VSX 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	Instructions
1	Read and follow the *"Mirror and Decrypt Requirements" on page 165*.
2	Prepare the Security Gateway, or *each* cluster member. See *"Preparing the Security Gateway or each Cluster Member" on page 167*.
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 168*.

Preparing the Security Gateway or each Cluster Member

Step	Instructions
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*.

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	Instructions
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	Instructions
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	Instructions
a	From the left navigation panel, click **Gateways & Servers**.
b	Open the object of the Security Gateway, or Cluster.
c	From the left tree, 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	Instructions
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 in SmartConsole:

Step	Instructions
a	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**.

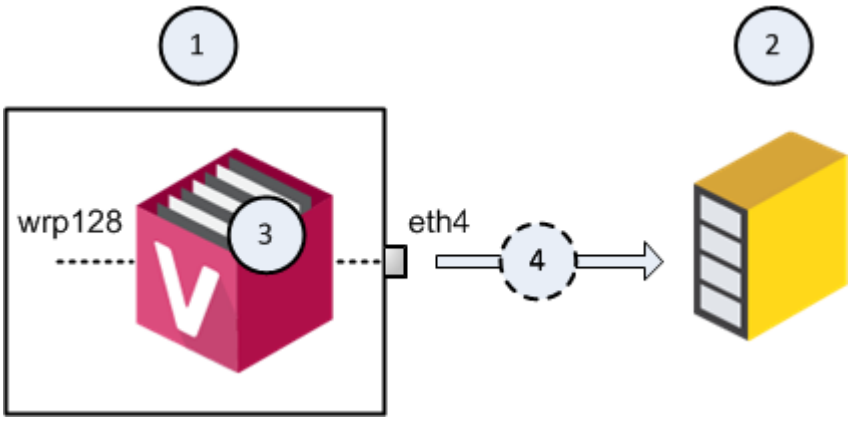
Step	Instructions
d	Select the existing policy and click **Edit** (the pencil icon). Alternatively, create a new policy.
e	From the navigation tree of the **Policy** window, click **General**.
f	In the **Policy Types** section, make sure you select **only** the Access Control.
g	In Access Control section, 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	Instructions
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, 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	Instructions
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

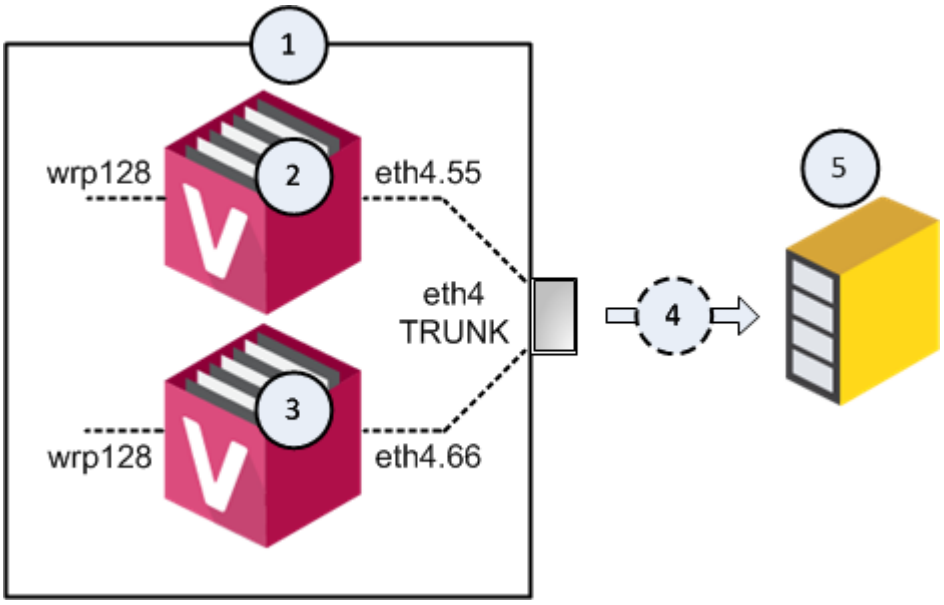
Example topology for one Virtual System:



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**).

Item	Description
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	Instructions
1	Read and follow the *"Mirror and Decrypt Requirements" on page 165*.
2	Prepare the VSX Gateway, or *each* VSX Cluster Member. See *"Preparing the VSX Gateway or each VSX Cluster Member" on page 177*.
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 178*. ■ *"Configuring Mirror and Decrypt in SmartConsole for Several Virtual Systems" on page 184*.

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*.

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	Instructions
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	Instructions
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	Instructions
a	In SmartConsole, open the Virtual System object.
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 165*. 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	Instructions
a	From the left navigation panel, click **Gateways & Servers**.
b	Open the Virtual System object.
c	From the left tree, 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.

Step	Instructions
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 in SmartConsole:

Step	Instructions
a	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**.

Step	Instructions
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	Instructions
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, 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	Instructions
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	Instructions
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	Instructions
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	Instructions
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	Instructions
a	In SmartConsole, open the Virtual System object.
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 165*. 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	Instructions
a	From the left navigation panel, click **Gateways & Servers**.
b	Open the Virtual System object.
c	From the left tree, 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.
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	Instructions
a	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.
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.

Step	Instructions
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**
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

Step	Instructions
	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, 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.
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.

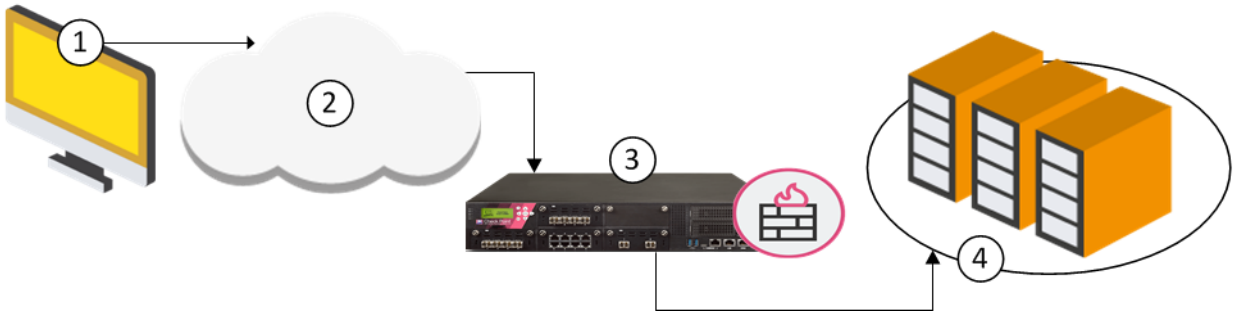
ConnectControl does not consume more memory or CPU processing power on Security Gateway or Cluster Members.

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 Security 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 Security Gateway 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 Security 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 Security Gateway 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 configured earlier (or define a new **Server Group** object).

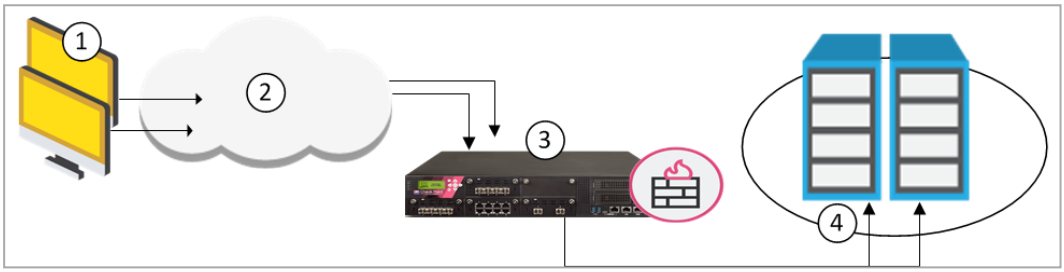
The members of the group must be hosts, Security Gateways, or OSE devices.

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 Security 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.

The Check Point Security Gateway listens 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.

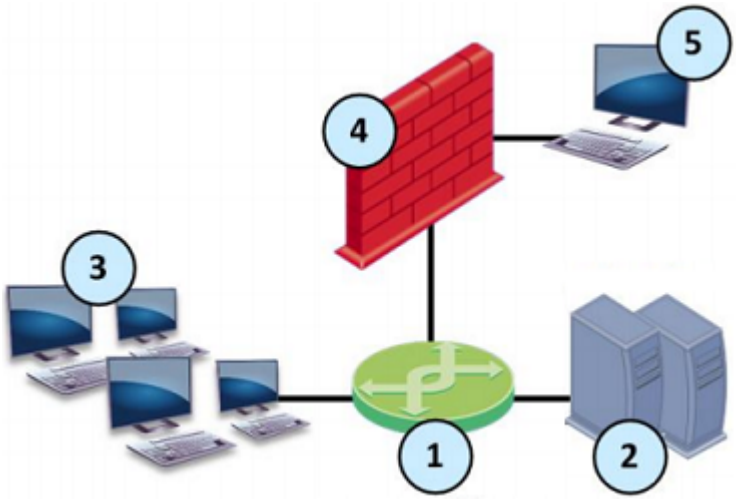
You can use the Monitor Mode:

- 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

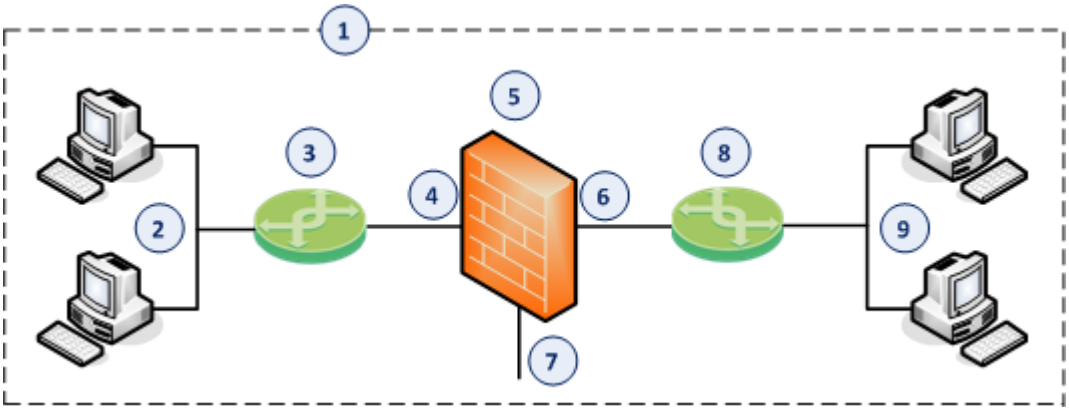
Introduction to Bridge Mode

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 subordinate interfaces, the Security Gateway (or Cluster Members) inspects it and passes it to the second bridge subordinate 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 subordinate interface **(4)** on the Security Gateway in Bridge Mode.
4	One bridged subordinate interface (for example, `eth1`) on the Security Gateway in Bridge Mode.
5	Security Gateway in Bridge Mode.
6	Another bridged subordinate 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 subordinate 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 in the *R80.40 CLI Reference Guide*:

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

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 a 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	Instructions
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	Instructions

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	Instructions
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	Instructions
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	Instructions
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`

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	Instructions
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	Instructions
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*.

Working with Kernel Parameters on Security Gateway

This section describes what are kernel parameters, and how to view and configure their values.

Introduction to Kernel Parameters

Kernel parameters let you change the advanced behavior of your Security Gateway.

These are the supported types of kernel parameters:

Type	Description
Integer	Accepts only one integer value.
String	Accepts only a plain-text string.

Important:

- In Cluster, you must see and configure the same value for the same kernel parameter on *each* Cluster Member.
- In VSX Gateway, the configured values of kernel parameters apply to all existing Virtual Systems and Virtual Routers.

Security Gateway gets the names and the default values of the kernel parameters from these kernel module files:

- `$FWDIR/boot/modules/fw_kern_64.o`
- `$FWDIR/boot/modules/fw_kern_64_v6.o`
- `$PPKDIR/boot/modules/sim_kern_64.o`
- `$PPKDIR/boot/modules/sim_kern_64_v6.o`

Firewall Kernel Parameters

To change the internal default behavior of Firewall or to configure special advanced settings for Firewall, you can use Firewall kernel parameters.

The names of applicable Firewall kernel parameters and their values appear in various SK articles in *Check Point Support Center*, and provided by *Check Point Support*.

Important:

- The names of Firewall kernel parameters are case-sensitive.
- You can configure most of the Firewall kernel parameters on-the-fly with the "`fw ctl set`" command.
  This change does **not** survive a reboot.
- You can configure some of the Firewall kernel parameters only permanently in the special configuration file `$FWDIR/boot/modules/fwkern.conf` with the "`fw ctl set -f`" command.
  This requires a maintenance window, because the new values of the kernel parameters take effect only after a reboot.
- You can configure some of the Firewall kernel parameters only permanently in the special configuration files - `$FWDIR/boot/modules/fwkern.conf` or `$FWDIR/boot/modules/vpnkern.conf`. You must manually edit these files.
  This requires a maintenance window, because the new values of the kernel parameters take effect only after a reboot.
- In a Cluster, you must configure all the Cluster Members in the same way.

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	Instructions
1	Connect to the command line on your Security Gateway or Cluster Member.
2	Log in to the Expert mode.
3	Make sure you can get the list of the available integer kernel parameters and their values without errors: **Note** - The configuration of your Security Gateway might not support all kernel parameters. As a result, the Security Gateway might fail to get the value of some kernel parameters. `modinfo -p $FWDIR/boot/modules/fw_kern*.o | sort -u | grep ':int param' | awk 'BEGIN {FS=":"} ; {print $1}' | xargs -n 1 fw ctl get int`
4	If in the previous step there were **no** errors, get the list of the available integer kernel parameters and their values, and save the list to a file: `modinfo -p $FWDIR/boot/modules/fw_kern*.o | sort -u | grep ':int param' | awk 'BEGIN {FS=":"} ; {print $1}' | xargs -n 1 fw ctl get int 1>> /var/log/fw_integer_kernel_parameters.txt 2>> /var/log/fw_integer_kernel_parameters.txt`
5	Analyze the output file: `/var/log/fw_integer_kernel_parameters.txt`

Viewing the current value of a Firewall *integer* kernel parameter

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

3	Get 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	Instructions
1	Connect to the command line on your Security Gateway or Cluster Member.
2	Log in to Gaia Clish or the Expert mode.

3	Configure 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 configured:
	`fw ctl get int <Name of Integer Kernel Parameter>`
	Example:
	`[Expert@MyGW:0]# fw ctl get int send_buf_limit`
	`send_buf_limit = 100`
	`[Expert@MyGW:0]#`

Configuring a value for a Firewall *integer* kernel parameter *permanently*

To make a kernel parameter configuration permanent (to survive reboot), you must edit one of the applicable configuration files:

- For Firewall kernel parameters:

  `$FWDIR/boot/modules/fwkern.conf`

- For VPN kernel parameters:

  `$FWDIR/boot/modules/vpnkern.conf`

The exact parameters appear in various SK articles in *Check Point Support Center*, and provided by *Check Point Support*.

Short procedure for the "fwkern.conf" file

Step	Instructions
1	Connect to the command line on your Security Gateway or Cluster Member.
2	Log in to the Expert mode.
3	Back up the current configuration file, if it exists: `cp -v $FWDIR/boot/modules/fwkern.conf{,_BKP}`

4	Configure the required Firewall kernel parameter with the assigned value in the exact format specified below.
	`fw ctl set -f int <Name_of_Integer_Kernel_Parameter> <Integer_Value>`
	Example:
	`[Expert@MyGW:0]# fw ctl set -f int send_buf_limit 100`
	`"fwkern.conf" was updated successfully`
	`[Expert@MyGW:0]#`

5	Examine the configuration file. `cat $FWDIR/boot/modules/fwkern.conf`
6	Reboot the Security Gateway or Cluster Member. **Important** - In cluster, this can cause a failover.
7	Connect to the command line on your Security Gateway or Cluster Member.
8	Log in to Gaia Clish or the Expert mode.

Step	Instructions
9	Make sure the new value of the kernel parameter is configured: `fw ctl get int <Name of Integer Kernel Parameter> [-a]`

Long procedure for the "fwkern.conf" and "vpnkern.conf" files

For more information, see sk26202: Changing the kernel global parameters for Check Point Security Gateway.

Step	Instructions
1	Connect to the command line on your Security Gateway 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	Instructions

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). `<Name_of_Integer_Kernel_Parameter>=<Integer_Value>`
8	Save the changes in the file and exit the 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 configured: `fw ctl get int <Name of Integer Kernel Parameter> [-a]`

Working with String Kernel Parameters

Viewing the list of the available Firewall *string* kernel parameters and their values

Step	Instructions
1	Connect to the command line on your Security Gateway or Cluster Member.
2	Log in to the Expert mode.
3	Make sure you can get the list of the available integer kernel parameters and their values without errors: **Note** - The configuration of your Security Gateway might not support all kernel parameters. As a result, the Security Gateway might fail to get the value of some kernel parameters. `modinfo -p $FWDIR/boot/modules/fw_kern*.o | sort -u | grep ':string param' | awk 'BEGIN {FS=":"} ; {print $1}' | xargs -n 1 fw ctl get str`
4	If in the previous step there were **no** errors, get the list of the available string kernel parameters and their values, and save the list to a file: `modinfo -p $FWDIR/boot/modules/fw_kern*.o | sort -u | grep ':string param' | awk 'BEGIN {FS=":"} ; {print $1}' | xargs -n 1 fw ctl get str 1>> /var/log/fw_string_kernel_parameters.txt 2>> /var/log/fw_string_kernel_parameters.txt`
5	Analyze the output file: `/var/log/fw_string_kernel_parameters.txt`

Viewing the current value of a Firewall *string* kernel parameter

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

3	Get 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	Instructions
1	Connect to the command line on your Security Gateway or Cluster Member.
2	Log in to Gaia Clish or the Expert mode.

3	Configure the new value for a string kernel parameter:
	Note - You must write the value in single quotes, or double quotes.
	`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]#`

Configuring a value for a Firewall *string* kernel parameter *permanently*

To make a kernel parameter configuration permanent (to survive reboot), you must edit one of the applicable configuration files:

- For Firewall kernel parameters:

  `$FWDIR/boot/modules/fwkern.conf`

- For VPN kernel parameters:

  `$FWDIR/boot/modules/vpnkern.conf`

The exact parameters appear in various SK articles in *Check Point Support Center*, and provided by *Check Point Support*.

Short procedure for the "fwkern.conf" file

Step	Instructions
1	Connect to the command line on your Security Gateway or Cluster Member.
2	Log in to the Expert mode.
3	Back up the current configuration file, if it exists: `cp -v $FWDIR/boot/modules/fwkern.conf{,_BKP}`

4	Configure the required Firewall kernel parameter with the assigned value in the exact format specified below.
	Note - You must write the value in single quotes, or double quotes.
	`fw ctl set -f str <Name_of_String_Kernel_Parameter> '<String_Text>'`
	or
	`fw ctl set -f str <Name_of_String_Kernel_Parameter> "<String_Text>"`
	Example:
	`[Expert@MyGW:0]# fw ctl set -f str ws_debug_ip_str '1.1.1.1'`
	`"fwkern.conf" was updated successfully`
	`[Expert@MyGW:0]#`

5	Examine the configuration file. `cat $FWDIR/boot/modules/fwkern.conf`
6	Reboot the Security Gateway or Cluster Member. **Important** - In cluster, this can cause a failover.

Step	Instructions
7	Connect to the command line on your Security Gateway or Cluster Member.
8	Log in to Gaia Clish or the Expert mode.
9	Make sure the new value of the kernel parameter is set: `fw ctl get str <Name of String Kernel Parameter> [-a]`

Long procedure for the "fwkern.conf" and "vpnkern.conf" files

For more information, see sk26202: Changing the kernel global parameters for Check Point Security Gateway.

Step	Instructions
1	Connect to the command line on your Security Gateway 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	Instructions

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).
	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.
12	Make sure the new value of the kernel parameter is set: `fw ctl get str <Name of String Kernel Parameter> [-a]`

Removing the current value from a Firewall *string* kernel parameter *temporarily*

Important - This change does **not** survive reboot.

Step	Instructions
1	Connect to the command line on your Security Gateway or Cluster Member.
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 can configure SecureXL kernel parameters in the current session with the "`fw ctl set`" command.
  This change does **not** survive reboot.
- To configure SecureXL kernel parameters permanently, you must configure them in the special configuration file - `$PPKDIR/conf/simkern.conf`
  Schedule a maintenance window, because this procedure requires a reboot.
- For some SecureXL kernel parameters, you **cannot** get their current value on-the-fly with the "`fw ctl get`" command (see sk43387).
- In a Cluster, you must configure all the Cluster Members in the same way.

Examples of SecureXL kernel parameters

Type	Name

Integer	`num_of_sxl_devices`
	`sim_ipsec_dont_fragment`
	`tcp_always_keepalive`
	`sim_log_all_frags`
	`simple_debug_filter_dport_1`
	`simple_debug_filter_proto_1`

String	`simple_debug_filter_addr_1`
	`simple_debug_filter_daddr_2`
	`simlinux_excluded_ifs_list`

Working with Integer Kernel Parameters

Viewing the list of the available SecureXL *integer* kernel parameters and their values

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member.
2	Log in to the Expert mode.
3	Make sure you can get the list of the available integer kernel parameters and their values without errors: **Note** - The configuration of your Security Gateway might not support all kernel parameters. As a result, the Security Gateway might fail to get the value of some kernel parameters. `modinfo -p $PPKDIR/boot/modules/sim_kern*.o | sort -u | grep ':int param' | awk 'BEGIN {FS=":"} ; {print $1}' | xargs -n 1 fw ctl get int`
4	If in the previous step there were **no** errors, get the list of the available integer kernel parameters and their values, and save the list to a file: `modinfo -p $PPKDIR/boot/modules/sim_kern*.o | sort -u | grep ':int param' | awk 'BEGIN {FS=":"} ; {print $1}' | xargs -n 1 fw ctl get int 1>> /var/log/sxl_integer_kernel_parameters.txt 2>> /var/log/sxl_integer_kernel_parameters.txt`
5	Analyze the output file: `/var/log/sxl_integer_kernel_parameters.txt`

Viewing the current value of a SecureXL *integer* kernel parameter

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

3	Get 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 sim_ipsec_dont_fragment`
	`sim_ipsec_dont_fragment = 1`
	`[Expert@MyGW:0]#`

Configuring a value for a SecureXL *integer* kernel parameter *temporarily*

Important - This change does **not** survive reboot.

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

3	Configure 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 sim_ipsec_dont_fragment 0`
	`Set operation succeeded`
	`[Expert@MyGW:0]#`

4	Make sure the new value is configured:
	`fw ctl get int <Name of Integer Kernel Parameter>`
	Example:
	`[Expert@MyGW:0]# fw ctl get int sim_ipsec_dont_fragment`
	`sim_ipsec_dont_fragment = 0`
	`[Expert@MyGW:0]#`

Configuring a value for a SecureXL *integer* kernel parameter *permanently*

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member.
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. The same syntax applies to the Security Gateway / each Cluster Member: `vi $PPKDIR/conf/simkern.conf`
7	Add the required SecureXL kernel parameter with the assigned value in the exact format specified below. **Important** - This configuration file does not support space characters, tabulation characters, and comments (lines that contain the # character). `<Name_of_SecureXL_Integer_Kernel_Parameter>=<Integer_Value>`
8	Save the changes in the file and exit the editor.
9	Reboot. `reboot` **Important** - In cluster, this can cause a failover.
10	Connect to the command line on your Security Gateway / each Cluster Member.
11	Log in to Gaia Clish or the Expert mode.

Step	Instructions
12	Make sure the new value of the kernel parameter is configured: `fw ctl get int <Name of Integer Kernel Parameter> [-a]`

Working with String Kernel Parameters

Viewing the list of the available SecureXL *string* kernel parameters and their values

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member.
2	Log in to the Expert mode.
3	Make sure you can get the list of the available integer kernel parameters and their values without errors: **Note** - The configuration of your Security Gateway might not support all kernel parameters. As a result, the Security Gateway might fail to get the value of some kernel parameters. `modinfo -p $PPKDIR/boot/modules/sim_kern*.o | sort -u | grep ':string param' | awk 'BEGIN {FS=":"} ; {print $1}' | xargs -n 1 fw ctl get str`
4	If in the previous step there were **no** errors, get the list of the available string kernel parameters and their values, and save the list to a file: `modinfo -p $PPKDIR/boot/modules/sim_kern*.o | sort -u | grep ':string param' | awk 'BEGIN {FS=":"} ; {print $1}' | xargs -n 1 fw ctl get str 1>> /var/log/sxl_string_kernel_parameters.txt 2>> /var/log/sxl_string_kernel_parameters.txt`
5	Analyze the output file: `/var/log/sxl_string_kernel_parameters.txt`

Viewing the current value of a SecureXL *string* kernel parameter

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

3	Get 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 sim_ipsec_dont_fragment`
	`sim_ipsec_dont_fragment = 1`
	`[Expert@MyGW:0]#`

Configuring a value for a SecureXL *string* kernel parameter *temporarily*

Important - This change does **not** survive reboot.

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

3	Configure the new value for a string kernel parameter.
	Note - You must write the value in single quotes, or double quotes. Use **one** of these syntax options.
	`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 fwkdebug_print_connkey_on_str 'Packet accepted'`
	`Set operation succeeded`
	`[Expert@MyGW:0]#`

4	Make sure the new value is configured:
	`fw ctl get str <Name of String Kernel Parameter>`
	Example:
	`[Expert@MyGW:0]# fw ctl get str fwkdebug_print_connkey_on_str`
	`fwkdebug_print_connkey_on_str = 'Packet accepted'`
	`[Expert@MyGW:0]#`

Configuring a value for a SecureXL *string* kernel parameter *permanently*

Step	Instructions
1	Connect to the command line on your Security Gateway / each Cluster Member.
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. The same syntax applies to the Security Gateway / each Cluster Member: `vi $PPKDIR/conf/simkern.conf`

7	Add the required SecureXL kernel parameter with the assigned value in the exact format specified below.
	Important - This configuration file does not support space characters, tabulation characters, and comments (lines that contain the # character).
	Note - You must write the value in single quotes, or double quotes. Use **one** of these syntax options.
	`<Name_of_SecureXL_String_Kernel_Parameter>='<String_Text>'`
	or
	`<Name_of_SecureXL_String_Kernel_Parameter>="<String_Text>"`

8	Save the changes in the file and exit the editor.
9	Reboot. `reboot` **Important** - In cluster, this can cause a failover.

Step	Instructions
10	Connect to the command line on your Security Gateway / each Cluster Member.
11	Log in to Gaia Clish or the Expert mode.
12	Make sure the new value of the kernel parameter is configured: `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 261*, or the *"Kernel Debug Procedure with Connection Life Cycle" on page 264*.

Step	Action	Description
1	Configure the applicable debug settings: a. Restore the default settings. b. Allocate the debug buffer.	In this step, you prepare the kernel debug options: a. Restore the default debug settings, so that any other debug settings do not interfere with the kernel debug. b. Allocate the kernel debug buffer, in which Security Gateway holds the applicable debug messages.
2	Configure the applicable kernel debug modules and their debug flags.	In this step, you prepare the applicable kernel debug modules and their debug flags, so that Security Gateway collects only applicable debug messages.
3	Start the collection of the kernel debug into an output file.	In this step, you configure Security Gateway to write the debug messages from the kernel debug buffer into an output file.
4	Stop the kernel debug.	In this step, you configure Security Gateway to stop writing the debug messages into an output file.
5	Restore the default kernel debug settings.	In this step, you restore the default kernel debug options.

To see the built-in help for the kernel debug

```
fw ctl debug -h
```

To restore the default kernel debug settings

- To reset all debug flags and enable only the default debug flags in all kernel modules:

  ```
  fw ctl debug 0
  ```

- To disable all debug flags including the default flags in all kernel modules:

  Best Practice - Do **not** run this command, because it disables even the basic default debug messages.

  ```
  fw ctl debug -x
  ```

To allocate the kernel debug buffer

```
fw ctl debug -buf 8200 [-v {"<List of VSIDs>" | all}] [-k]
```

Notes:

- Security Gateway allocates the kernel debug buffer with the specified size for *every* CoreXL Firewall instance.
- The maximal supported buffer size is 8192 kilobytes..

To configure the debug modules and debug flags

- General syntax:

```
fw ctl debug [-d <Strings to Search>] [-v {"<List of VSIDs>" | all}] -m <Name of Debug Module> {all | + <List of Debug Flags> | - <List of Debug Flags>}
```

```
fw ctl debug [-s "<String to Stop Debug>"] [-v {"<List of VSIDs>" | all}] -m <Name of Debug Module> {all | + <List of Debug Flags> | - <List of Debug Flags>}
```

- To see a list of all debug modules and their flags:

  Note - The list of kernel modules depends on the Software Blades you enabled on the Security Gateway.

```
fw ctl debug -m
```

- To see a list of debug flags that are already enabled:

```
fw ctl debug
```

- To enable all debug flags in the specified kernel module:

```
fw ctl debug -m <Name of Debug Module> all
```

- To enable the specified debug flags in the specified kernel module:

```
fw ctl debug -m <Name of Debug Module> + <List of Debug Flags>
```

- To disable the specified debug flags in the specified kernel module:

```
fw ctl debug -m <Name of Debug Module> - <List of Debug Flags>
```

To collect the kernel debug output

- General syntax (only supported parameters are listed):

```
fw ctl kdebug [-p <List of Fields>] [-T] -f > /<Path>/<Name of Output File>
```

```
fw ctl kdebug [-p <List of Fields>] [-T] -f -o /<Path>/<Name of Output File> -m <Number of Cyclic Files> [-s <Size of Each Cyclic File in KB>]
```

- To start the collection of the kernel debug into an output file:

```
fw ctl kdebug -T -f > /<Path>/<Name of Output File>
```

- To start collecting the kernel debug into cyclic output files:

```
fw ctl kdebug -T -f -o /<Path>/<Name of Output File> -m <Number of Cyclic Files> [-s <Size of Each Cyclic File in KB>]
```

Parameters

Note - Only supported parameters are listed.

Table: Parameters of the 'fw ctl debug' command

Parameter	Description
`0 | -x`	Controls how to disable the debug flags: ■ `0` Resets all debug flags and enables only the default debug flags in all kernel modules. ■ `-x` Disables all debug flags, including the default flags in all kernel modules. **Best Practice** - Do **not** use the "`-x`" parameter, because it disables even the basic default debug messages.

Table: Parameters of the 'fw ctl debug' command (continued)

Parameter	Description
`-d <Strings to Search>`	When you specify this parameter, the Security Gateway: 1. Examines the applicable debug messages based on the enabled kernel debug modules and their debug flags. 2. Collects only debug messages that contain at least one of the specified strings into the kernel debug buffer. 3. Writes the entire kernel debug buffer into the output file. **Notes:** ■ These strings can be any plain text (not a regular expression) that you see in the debug messages. ■ Separate the applicable strings by commas without spaces: `-d String1,String2,...,StringN` ■ You can specify up to 10 strings, up to 250 characters in total.
`-s "<String to Stop Debug>"`	When you specify this parameter, the Security Gateway: 1. Collects the applicable debug messages into the kernel debug buffer based on the enabled kernel debug modules and their debug flags. 2. Does not write any of these debug messages from the kernel debug buffer into the output file. 3. Stops collecting all debug messages when it detects the first debug message that contains the specified string in the kernel debug buffer. 4. Writes the entire kernel debug buffer into the output file. **Notes:** ■ This one string can be any plain text (not a regular expression) that you see in the debug messages. ■ String length is up to 50 characters.
`-m <Name of Debug Module>`	Specifies the name of the kernel debug module, for which you print or configure the debug flags.

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`

Table: Parameters of the 'fw ctl debug' command (continued)

Parameter	Description
`-v {"<List of VSIDs>" | all}`	Specifies the list of Virtual Systems. A VSX Gateway automatically filters the collected kernel debug information for debug messages only for these Virtual Systems. ■ `-v "<List of VSIDs>"` Monitors the messages only from the specified Virtual Systems. To specify the Virtual Systems, enter their VSID number separated with commas and without spaces: `"VSID1[,VSID2,VSID3,...,VSIDn]"` Example: `-v "1,3,7"` ■ `-v all` Monitors the messages from all configured Virtual Systems. **Notes:** ■ This parameter is supported only in VSX mode. ■ This parameter and the `-k` parameter are mutually exclusive.

Table: Parameters of the 'fw ctl debug' command (continued)

Parameter	Description

`-e <Expression>`
`-i <Name of Filter File>`
`-i -`
`-u`
	Specifies the INSPECT filter for the debug:
	■ `-e <Expression>`
	Specifies the INSPECT filter. See the *R80.40 CLI Reference Guide* > section "fw monitor".
	■ `-i <Name of Filter File>`
	Specifies the file that contains the INSPECT filter.
	■ `-i -`
	Specifies that the INSPECT filter arrives from the standard input.
	The Security Gateway prompts to enter the INSPECT filter on the screen.
	■ `-u` - Removes the INSPECT debug filter.
	Notes:
	■ These are *legacy* parameters ("`-e`" and "`-i`").
	■ When you use these parameters ("`-e`" and "`-i`"), the Security Gateway cannot apply the specified INSPECT filter to the accelerated traffic.
	■ For new debug filters, see *"Kernel Debug Filters" on page 256*.

`-z`	The Security Gateway processes some connections in both SecureXL code and in the Host appliance code (for example, Passive Streaming Library (PSL) - an IPS infrastructure, which transparently listens to TCP traffic as network packets, and rebuilds the TCP stream out of these packets.). The Security Gateway processes some connections in only in the Host appliance code. When you use this parameter, kernel debug output contains the debug messages only from the Host appliance code.

Table: Parameters of the 'fw ctl debug' command (continued)

Parameter	Description
`-k`	The Security Gateway processes some connections in both kernel space code and in the user space code (for example, Web Intelligence). The Security Gateway processes some connections only in the kernel space code. When you use this parameter, kernel debug output contains the debug messages only from the kernel space. **Notes:** ■ This parameter is not supported in the VSX mode, in which the Firewall works in the user space. ■ This parameter and the `-v` parameter are mutually exclusive.
`-p <List of Fields>`	By default, when the Security Gateway prints the debug messages, the messages start with the applicable CPU ID and CoreXL Firewall instance ID. You can print additional fields in the beginning of each debug message. **Notes:** ■ These fields are available: `all`, `proc`, `pid`, `date`, `mid`, `type`, `freq`, `topic`, `time`, `ticks`, `tid`, `text`, `errno`, `host`, `vsid`, `cpu`. ■ When you specify the applicable fields, separate them with commas and without spaces: `Field1,Field2,...,FieldN` ■ The more fields you specify, the higher the load on the CPU and on the hard disk.
`-T`	Prints the time stamp in microseconds in front of each debug message. **Best Practice** - Always use this parameter to make the debug analysis easier.
`-f`	Collects the debug data until you stop the kernel debug in one of these ways: ■ When you press the **CTRL+C** keys. ■ When you run the "`fw ctl debug 0`" command. ■ When you run the "`fw ctl debug -x`" command. ■ When you kill the "`fw ctl kdebug`" process.

Table: Parameters of the 'fw ctl debug' command (continued)

Parameter	Description
`/<Path>/<Name of Output File>`	Specifies the path and the name of the debug output file. **Best Practice** - Always use the largest partition on the disk - `/var/log/`. Security Gateway can generate many debug messages within short time. As a result, the debug output file can grow to large size very fast.
`-o /<Path>/<Name of Output File> -m <Number of Cyclic Files> [-s <Size of Each Cyclic File in KB>]`	Saves the collected debug data into cyclic debug output files. When the size of the current `<Name of Output File>` reaches the specified `<Size of Each Cyclic File in KB>` (more or less), the Security Gateway renames the current `<Name of Output File>` to `<Name of Output File>.0` and creates a new `<Name of Output File>`. If the `<Name of Output File>.0` already exists, the Security Gateway renames the `<Name of Output File>.0` to `<Name of Output File>.1`, and so on - until the specified limit `<Number of Cyclic Files>`. When the Security Gateway reaches the `<Number of Cyclic Files>`, it deletes the oldest files. The valid values are: ■ `<Number of Cyclic Files>` - from 1 to 999 ■ `<Size of Each Cyclic File in KB>` - from 1 to 2097150

Kernel Debug Filters

By default, kernel debug output contains information about all processed connections.

You can configure filters for kernel debug to collect debug messages only for the applicable connections.

There are three types of debug filters:

- By connection tuple parameters
- By an IP address parameter
- By a VPN peer parameter

To configure these kernel debug filters, assign the applicable values to the applicable kernel parameters **before** you start the kernel debug.

You assign the values to the applicable kernel parameters temporarily with the "`fw ctl set`" command.

Notes:

- A Security Gateway supports:
  - up to **five** Connection Tuple filters in total (from all types)
  - up to **three** Host IP Address filters
  - up to **two** VPN Peer filters
- A Security Gateway applies these debug filters to both the non-accelerated and accelerated traffic.
- A Security Gateway applies these debug filters to *"Kernel Debug Procedure with Connection Life Cycle" on page 264*.

Best Practice - It is usually simpler to set the Connection Tuple and Host IP Address filters from within the "`fw ctl debug`" command (see the *R80.40 CLI Reference Guide*). To filter the kernel debug by a VPN Peer, use the procedure below.

To configure debug filter of the type "By connection tuple parameters":

A Security Gateway processes connections based on the 5-tuple:

- Source IP address
- Source Port (see *IANA Service Name and Port Number Registry*)
- Destination IP address
- Destination Port (see *IANA Service Name and Port Number Registry*)
- Protocol Number (see *IANA Protocol Numbers*)

With this debug filter you can filter by these tuple parameters:

Tuple Parameter	Syntax for Kernel Parameters
Source IP address	`fw ctl set str simple_debug_filter_saddr_<N> "<IPv4 or IPv6 Address>"`
Source Ports	`fw ctl set int simple_debug_filter_sport_<N> <1-65535>`
Destination IP address	`fw ctl set str simple_debug_filter_daddr_<N> "<IPv4 or IPv6 Address>"`
Destination Ports	`fw ctl set int simple_debug_filter_dport_<N> <1-65535>`
Protocol Number	`fw ctl set int simple_debug_filter_proto_<N> <0-254>`

Notes:

1. *<N>* is an integer between 1 and 5. This number is an index for the configured kernel parameters of this type.
2. When you specify IP addresses, you must enclose them in double quotes.
3. When you configure kernel parameters with the *same* index <N>, the debug filter is a logical "**AND**" of these kernel parameters.
   In this case, the final filter matches only *one* direction of the processed connection.
   - Example 1 - packets from the source IP address X to the destination IP address Y:

     ```
     simple_debug_filter_saddr_1 <Value X>
     AND
     simple_debug_filter_daddr_1 <Value Y>
     ```
   - Example 2 - packets from the source IP address X to the destination port Y:

     ```
     simple_debug_filter_saddr_1 <Value X>
     AND
     simple_debug_filter_dport_1 <Value Y>
     ```
4. When you configure kernel parameters with the *different* indices <N>, the debug filter is a logical "OR" of these kernel parameters.
   This means that if it is necessary the final filter matches both directions of the connection, then it is necessary to configure the applicable debug filters for both directions.
   - Example 1 - packets either from the source IP address X, or to the destination IP address Y:

     ```
     simple_debug_filter_saddr_1 <Value X>
     OR
     simple_debug_filter_daddr_2 <Value Y>
     ```
   - Example 2 - packets either from the source IP address X, or to the destination port Y:

     ```
     simple_debug_filter_saddr_1 <Value X>
     OR
     simple_debug_filter_dport_2 <Value Y>
     ```
5. For information about the Port Numbers, see *IANA Service Name and Port Number Registry*.
6. For information about the Protocol Numbers, see *IANA Protocol Numbers*.

To configure debug filter of the type "By an IP address parameter":

With this debug filter you can filter by one IP address, which is either the source or the destination IP address of the packet.

Syntax for Kernel Parameters:

```
fw ctl set str simple_debug_filter_addr_<N> "<IPv4 or IPv6 Address>"
```

Notes:

1. *<N>* is an integer between 1 and 3.
   This number is an index for the configured kernel parameters of this type.
2. You can configure one, two, or three of these kernel parameters at the same time.
   - Example 1:
     Configure one IP address (`simple_debug_filter_addr_1`).
   - Example 2:
     Configure two IP addresses (`simple_debug_filter_addr_1` and `simple_debug_filter_addr_2`).
     This would match packets, where any of these IP addresses appears, either as a source or a destination.
3. You must enclose the IP addresses in double quotes.

To configure debug filter of the type "By a VPN peer parameter":

With this debug filter you can filter by one IP address.

Syntax for Kernel Parameters:

```
fw ctl set str simple_debug_filter_vpn_<N> "<IPv4 or IPv6 Address>"
```

Notes:

1. *<N>* is an integer - 1 or 2.
   This number is an index for the configured kernel parameters of this type.
2. You can configure one or two of these kernel parameters at the same time.
   - Example 1:
     Configure one VPN peer (`simple_debug_filter_vpn_1`).
   - Example 2:
     Configure two VPN peers (`simple_debug_filter_vpn_1` and `simple_debug_filter_vpn_2`).
3. You must enclose the IP addresses in double quotes.

To disable all debug filters:

You can disable all the configured debug filters of all types.

Syntax for Kernel Parameter:

```
fw ctl set int simple_debug_filter_off 1
```

Usage Example

It is necessary to show in the kernel debug the information about the connection from Source IP address 192.168.20.30 from any Source Port to Destination IP address 172.16.40.50 to Destination Port 80 (192.168.20.30:<Any> --> 172.16.40.50:80).

Run these commands **before** you start the kernel debug:

```
fw ctl set int simple_debug_filter_off 1
fw ctl set str simple_debug_filter_saddr_1 "192.168.20.30"
fw ctl set str simple_debug_filter_daddr_1 "172.16.40.50"
fw ctl set str simple_debug_filter_saddr_2 "172.16.40.50"
fw ctl set str simple_debug_filter_daddr_2 "192.168.20.30"
fw ctl set int simple_debug_filter_dport_1 80
fw ctl set int simple_debug_filter_sport_2 80
```

Important - In the above example, two Connection Tuple filters are used ("`..._1`" and "`..._2`") - one for each direction, because we want the debug filter to match both directions of this connection.

Kernel Debug Procedure

Alternatively, use the *"Kernel Debug Procedure with Connection Life Cycle" on page 264*.

Important:

- Debug increases the load on the CPU on the Security Gateway / Cluster Members. Schedule a maintenance window.
- We strongly recommend to connect over serial console to your Security Gateway / each Cluster Member.
  This is to prevent a possible issue when you cannot work with the CLI because of a high load on the CPU.
- In Cluster, you must perform these steps on all the Cluster Members in the same way.

Step	Instructions
1	Connect to the command line on the Security Gateway / each Cluster Member over SSH, or console.
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 256*.
6	Allocate the kernel debug buffer for each CoreXL Firewall 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 271*. **Important** - The CPU load increases at this point because the Firewall kernel starts to write **some** debug messages to the `/var/log/messages` file and the `dmesg` buffer.

Step	Instructions
9	Examine the list of the debug flags that are enabled in the specified kernel modules: `fw ctl debug -m <module>`
10	Save the kernel debug output to a file: `fw ctl kdebug -T -f > /var/log/kernel_debug.txt` **Important** - The CPU load increases even more at this point because the Firewall starts to write **all** debug messages to the output file.
11	Replicate the issue, or wait for the issue to occur.
12	Stop the kernel debug output: Press the **CTRL+C** keys. **Important** - This does not stop all CPU load yet because the Firewall kernel continues to write **some** debug messages to the `/var/log/messages` file and the `dmesg` buffer.
13	Reset the kernel debug options: `fw ctl debug 0` **Important** - This stops all CPU load from the kernel debug.
14	Reset the kernel debug filters: `fw ctl set int simple_debug_filter_off 1`
15	Transfer this file from the Security Gateway / each Cluster Member to your computer: `/var/log/kernel_debug.txt` **Best Practice** - Compress this file with the "`tar -zxvf`" command and transfer it from the Security Gateway / each Cluster Member to your computer. If you transfer to an FTP server, do so in the binary mode.
16	Analyze the debug output file.

Example - Connection 192.168.20.30:<Any> --> 172.16.40.50:80

```
[Expert@GW:0]# fw ctl debug 0
Defaulting all kernel debugging options
Debug state was reset to default.
[Expert@GW:0]#
[Expert@GW:0]# fw ctl set int simple_debug_filter_off 1
[Expert@GW:0]#
[Expert@GW:0]# fw ctl set str simple_debug_filter_saddr_1 "192.168.20.30"
[Expert@GW:0]#
[Expert@GW:0]# fw ctl set str simple_debug_filter_daddr_2 "192.168.20.40"
[Expert@GW:0]#
[Expert@GW:0]# fw ctl set int simple_debug_filter_dport_1 80
[Expert@GW:0]#
[Expert@GW:0]# fw ctl debug -buf 8200
Initialized kernel debugging buffer to size 8192K
[Expert@GW:0]#
[Expert@GW:0]# fw ctl debug | grep buffer
Kernel debugging buffer size: 8192KB
[Expert@GW:0]#
[Expert@GW:0]# fw ctl debug -m fw + conn drop
Updated kernel's debug variable for module fw
Debug flags updated.
[Expert@GW:0]#
[Expert@GW:0]# fw ctl debug -m fw
Kernel debugging buffer size: 8192KB
Module: fw
Enabled Kernel debugging options: error warning conn drop
Messaging threshold set to type=Info freq=Common
[Expert@GW:0]#
[Expert@GW:0]# fw ctl kdebug -T -f > /var/log/kernel_debug.txt
... ... Replicate the issue, or wait for the issue to occur ... ...
... ... Press CTRL+C ... ...
[Expert@GW:0]#
[Expert@GW:0]# fw ctl debug 0
Defaulting all kernel debugging options
Debug state was reset to default.
[Expert@GW:0]#
[Expert@GW:0]# fw ctl set int simple_debug_filter_off 1
[Expert@GW:0]#
[Expert@GW:0]# ls -l /var/log/kernel_debug.txt
-rw-rw---- 1 admin root 1630619 Apr 12 19:49 /var/log/kernel_debug.txt
[Expert@GW:0]#
```

Kernel Debug Procedure with Connection Life Cycle

Introduction

R80.20 introduced a new debug tool called Connection Life Cycle.

This tool generates a formatted debug output file that presents the debug messages hierarchically by connections and packets:

- The first hierarchy level shows connections.
- After you expand the connection, you see all the packets of this connection.

Important - You must use this tool in the Expert mode together with the regular kernel debug flags (see *"Kernel Debug Modules and Debug Flags" on page 271*).

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 256*.
	Important - If you do not specify filters, then the tool prints debug messages for *all* traffic. This causes high load on the CPU and increases the time to format the debug output file.
	Each filter must contain these five numbers (5-tuple) separated with commas:
	`"<Source IP Address>,<Source Port>,<Destination IP Address>,<Destination Port>,<Protocol Number>"`
	Example of capturing traffic from IP 192.168.20.30 from any port to IP 172.16.40.50 to port 22 over the TCP protocol:
	`-f "192.168.20.30,0,172.16.40.50,22,6"`

Table: Parameters of the 'conn_life_cycle.sh' script (continued)

Parameter	Description
	Notes: ■ The tool supports up to **five** of such filters. ■ The tool treats the value 0 (zero) as "any". ■ If you specify two or more filters, the tool performs a logical "OR" of all the filters on each packet. If the packet matches at least one filter, the tool prints the debug messages for this packet. ■ "*<Source IP Address>*" and "*<Destination IP Address>*" - IPv4 or IPv6 address ■ "*<Source Port>*" and "*<Destination Port>*" - integers from 1 to 65535 (see *IANA Service Name and Port Number Registry*) ■ *<Protocol Number>* - integer from 0 to 254 (see *IANA Protocol Numbers*)
`-o /<Path>/<Name of Raw Debug Output File>`	Mandatory. Specifies the absolute path and the name of the raw debug output file. Example: `-o /var/log/kernel_debug.txt`
`-o /<Path>/<Name of Formatted Debug Output File>`	Mandatory. Specifies the absolute path and the name of the formatted debug output file (to analyze by an administrator). Example: `-o /var/log/kernel_debug_formatted.txt`

Procedure

Important - In cluster, you must perform these steps on all the Cluster Members in the same way.

Step	Instructions
1	Connect to the command line on the Security Gateway.
2	Log in to the Expert mode.
3	Enable the applicable debug flags in the applicable kernel modules: `fw ctl debug -m <module> {all | + <flags>}` See *"Kernel Debug Modules and Debug Flags" on page 271*.
4	Examine the list of the debug flags that are enabled in the specified kernel modules: `fw ctl debug -m <module>`
5	Start the debug capture: `conn_life_cycle.sh -a start -o /var/log/kernel_debug.txt -T -f "<Filter1>" [... [-f "<FilterN>"]]`
6	Replicate the issue, or wait for the issue to occur.
7	Stop the debug capture and prepare the formatted debug output: `conn_life_cycle.sh -a stop -o /var/log/kernel_debug_formatted.txt`
8	Transfer the formatted debug output file from your Security Gateway to your desktop or laptop computer: `/var/log/kernel_debug_formatted.txt`
9	Examine the formatted debug output file in an advanced text editor like Notepad++ (click **Language > R > Ruby**), or any other Ruby language viewer.

Example

Collecting the kernel debug for TCP connection from IP 172.20.168.15 (any port) to IP 192.168.3.53 and port 22

```
[Expert@GW:0]# fw ctl debug -m fw + conn drop
Updated kernel's debug variable for module fw
Debug flags updated.
[Expert@GW:0]#
[Expert@GW:0]# fw ctl debug -m fw
Kernel debugging buffer size: 50KB
HOST:
Module: fw
Enabled Kernel debugging options: error warning conn drop
Messaging threshold set to type=Info freq=Common
[Expert@GW:0]#
[Expert@GW:0]# conn_life_cycle.sh -a start -o /var/log/kernel_debug.txt -T -f
"172.20.168.15,0,192.168.3.53,22,6"
Set operation succeeded
Set operation succeeded
Set operation succeeded
Set operation succeeded
Set operation succeeded
Set operation succeeded
Set operation succeeded
Initialized kernel debugging buffer to size 8192K
Set operation succeeded
Capturing started...
[Expert@GW:0]#

... ... Replicate the issue, or wait for the issue to occur ... ...

[Expert@GW:0]#
[Expert@GW:0]# conn_life_cycle.sh -a stop -o /var/log/kernel_debug_formatted.txt
Set operation succeeded
Defaulting all kernel debugging options
Debug state was reset to default.
Set operation succeeded
doing unification...
Openning host debug file /tmp/tmp.KiWmF18217... OK
New unified debug file: /tmp/tmp.imzMZ18220... OK
prepare unification
performing unification
Done :-)
doing grouping...
wrapping connections and packets...
Some of packets lack description, probably because they were already handled when the feature
was enabled.
[Expert@GW:0]#
[Expert@GW:0]# fw ctl debug -m fw
Kernel debugging buffer size: 50KB
HOST:
Module: fw
Enabled Kernel debugging options: error warning
Messaging threshold set to type=Info freq=Common
[Expert@GW:0]
[Expert@GW:0] ls -l /var/log/kernel_debug.*
-rw-rw---- 1 admin root 40960 Nov 26 13:02 /var/log/kernel_debug.txt
-rw-rw---- 1 admin root 24406 Nov 26 13:02 /var/log/kernel_debug_formatted.txt
[Expert@GW:0]
```

Opening the kernel debug in Notepad++

Everything is collapsed:

```
   Connection with 1st packet already in handling so no conn details
[+]
{++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
++++++++++++++++++++++++++++++++++
```

Opened the first hierarchy level to see the connection:

```
   Connection with 1st packet already in handling so no conn details
[-]
{++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
++++++++++++++++++++++++++++++++++
;26Nov2018 13:02:06.736016;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is  INBOUND;
[+]{------------------------------------------------------------ packet begins ----------------
--------------------------------------
```

Opened the second hierarchy level to see the packets of this connection:

```
   Connection with 1st packet already in handling so no conn details
[-]
{++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
;26Nov2018 13:02:06.736016;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is  INBOUND;
[-]{---------------------------------------------------- packet begins ----------------
;26Nov2018 13:02:06.736021;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is entering  CHAIN_
MODULES_ENTER;
;26Nov2018 13:02:06.736035;[cpu_2];[fw4_1];#fwconn_lookup_cache: conn <dir 0,
172.20.168.15:57821 -> 192.168.3.53:22 IPP 6>;
;26Nov2018 13:02:06.736046;[cpu_2];[fw4_1];#<1c001,44000,2,1e2,0,UUID: 5bfbc2a2-0000-0000-c0-
a8-3-35-1-0-0-c0, 1,1,ffffffff,ffffffff,40800,0,80,OPQS:
[0,ffffc20033d220f0,0,0,0,0,ffffc20033958648,0,0,0,ffffc200325d57b0,0,0,0,0,0],0,0,0,0,0,0,0,
0,0,0,0,0,0,0>
;26Nov2018 13:02:06.736048;[cpu_2];[fw4_1];CONN LIFE CYCLE: lookup: found;
;26Nov2018 13:02:06.736053;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is entering  VM_ENTER;
;26Nov2018 13:02:06.736055;[cpu_2];[fw4_1];#
;26Nov2018 13:02:06.736060;[cpu_2];[fw4_1];#Before VM: <dir 0, 172.20.168.15:57821 ->
192.168.3.53:22 IPP 6> (len=40) TCP flags=0x10 (ACK), seq=686659054, ack=4181122096, data
end=686659054 (ifn=1) (first seen) (looked up) ;
;26Nov2018 13:02:06.736068;[cpu_2];[fw4_1];#After  VM: <dir 0, 172.20.168.15:57821 ->
192.168.3.53:22 IPP 6> (len=40) TCP flags=0x10 (ACK), seq=686659054, ack=4181122096, data
end=686659054 ;
;26Nov2018 13:02:06.736071;[cpu_2];[fw4_1];#VM Final action=ACCEPT;
;26Nov2018 13:02:06.736072;[cpu_2];[fw4_1];# -----  Stateful VM inbound Completed -----
;26Nov2018 13:02:06.736075;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is exiting  VM_EXIT;
;26Nov2018 13:02:06.736081;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is entering  POST VM_
ENTER;
;26Nov2018 13:02:06.736083;[cpu_2];[fw4_1];#
;26Nov2018 13:02:06.736085;[cpu_2];[fw4_1];#fw_post_vm_chain_handler: (first_seen 32, new_
conn 0, is_my_ip 0, is_first_packet 0);
;26Nov2018 13:02:06.736089;[cpu_2];[fw4_1];#Before POST VM: <dir 0, 172.20.168.15:57821 ->
192.168.3.53:22 IPP 6> (len=40) TCP flags=0x10 (ACK), seq=686659054, ack=4181122096, data
end=686659054 (ifn=1) (first seen) (looked up) ;
;26Nov2018 13:02:06.736095;[cpu_2];[fw4_1];#After  POST VM: <dir 0, 172.20.168.15:57821 ->
192.168.3.53:22 IPP 6> (len=40) TCP flags=0x10 (ACK), seq=686659054, ack=4181122096, data
end=686659054 ;
;26Nov2018 13:02:06.736097;[cpu_2];[fw4_1];#POST VM Final action=ACCEPT;
;26Nov2018 13:02:06.736098;[cpu_2];[fw4_1];# -----  Stateful POST VM inbound Completed -----
;26Nov2018 13:02:06.736101;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is exiting  POST VM_
EXIT;
;26Nov2018 13:02:06.736104;[cpu_2];[fw4_1];#fwconnoxid_msg_get_cliconn: warning - failed to
get connoxid message.;
;26Nov2018 13:02:06.736107;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is entering  CPAS_ENTER;
;26Nov2018 13:02:06.736110;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is exiting  CPAS_EXIT;
;26Nov2018 13:02:06.736113;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is exiting  CHAIN_
MODULES_EXIT;
;26Nov2018 13:02:06.736116;[cpu_2];[fw4_1];Packet 0xffff8101ea45e680 is  ACCEPTED;
}
;26Nov2018 13:02:06.770652;[cpu_2];[fw4_1];Packet 0xffff8101ea128580 is  INBOUND;
```

Kernel Debug Modules and Debug Flags

This section describes the Kernel Debug Modules and their Debug Flags.

To see the available kernel debug modules and their debug flags, run:

```
fw ctl debug -m
```

List of kernel debug modules (in alphabetical order):

Module 'accel_apps' (Accelerated Applications)

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')

Flag	Description
`track`	Use only for very limited important debug prints, so it can be used in a loaded environment - Content-Disposition, Content-Type, extension validation, extension matching
`uf`	URL filters and URL cache
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'cluster' (ClusterXL)

Syntax:

```
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 297*
  - The debug flag "`sync`" in *"Module 'CPAS' (Check Point Active Streaming)" on page 283*
- 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)

Flag	Description
`ifstate`	Interface state (all the operations and checks on interfaces)
`io`	Information about sending of packets through cluster interfaces
`log`	Creating and sending of logs by cluster **Note** - Also enable the debug flag "`log`" in *"Module 'fw' (Firewall)" on page 297*.
`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 284*.
`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)

Flag	Description
`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 297*.
`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 288*
- *"Module 'dlpk' (Data Loss Prevention - Kernel Space)" on page 290*
- *"Module 'dlpuk' (Data Loss Prevention - User Space)" on page 291*

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 297*.

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 287*.
`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 318*.
`mux_conn_app`	Information about connections **Note** - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 318*.
`mux_decrypt_app`	Information about decryption of connections **Note** - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 318*.
`mux_encrypt_app`	Information about encryption of connections **Note** - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 318*.

Flag	Description
`mux_inf`	Internal flow **Note** - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 318*.
`mux_stream`	Internal flow **Note** - Also see *"Module 'MUX' (Multiplexer for Applications Traffic)" on page 318*.
`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 284*
- *"Module 'dlpk' (Data Loss Prevention - Kernel Space)" on page 290*
- *"Module 'dlpuk' (Data Loss Prevention - User Space)" on page 291*

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

Flag	Description
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`verbose`	Prints additional information (used with other debug flags)
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module '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 284*
- *"Module 'dlpda' (Data Loss Prevention - Download Agent for Content Awareness)" on page 288*
- *"Module 'dlpuk' (Data Loss Prevention - User Space)" on page 291*

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 284*
- *"Module 'dlpda' (Data Loss Prevention - Download Agent for Content Awareness)" on page 288*
- *"Module 'dlpk' (Data Loss Prevention - Kernel Space)" on page 290*

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

Flag	Description
`qosaccel`	Acceleration of QoS traffic
`rates`	Rule and connection rates (IQ Engine behavior and status)
`rtm`	Failures in information gathering in the Real Time Monitoring module **Note** - Also see *"Module 'RTM' (Real Time Monitoring)" on page 324*.
`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 345*.

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 275*)
`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 311*

Flag	Description
`cookie`	Virtual de-fragmentation , cookie issues (cookies in the data structure that holds the packets)
`corr`	Correction layer
`cpsshi`	SSH Inspection **Important** - Also enable all the debug flags in *"Module 'CPSSH' (SSH Inspection)" on page 285*.
`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 256*)
`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

Flag	Description
`ftp`	Processing of FTP Data connections (used to call applications over FTP Data - i.e., Anti-Virus)
`handlers`	Operations related to the Context Management Interface / Infrastructure Loader **Note** - Also see *"Module 'cmi_loader' (Context Management Interface / Infrastructure Loader)" on page 281*.
`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 or hang due to very high CPU load!.
`leaks`	Memory leak detection mechanism
`link`	Creation of links in Connections kernel table (ID 8158)
`log`	Everything related to calls in the log

Flag	Description
`machine`	INSPECT Virtual Machine (actual assembler commands being processed) **Warning** - Security Gateway can freeze or hang due to very high CPU load!.
`mail`	Issues with e-mails over POP3, IMAP
`malware`	Matching of connections to Threat Prevention Layers (multiple rulebases) **Note** - Also see *"Module 'MALWARE' (Threat Prevention)" on page 315*.
`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 the *R80.40 CLI Reference Guide* > section "fw monitor") **Note** - Also enable the debug flag "`misc`" in this module.
`monitorall`	Prints output similar to the "`fw monitor -p all`" command (see the *R80.40 CLI Reference Guide* > section "fw monitor") **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
`multik`	CoreXL-related **Note** - This debug flag enables all the debug flags in the *"Module 'multik' (Multi-Kernel Inspection - CoreXL)" on page 316*, except for the debug flag "`packet`".

Flag	Description
`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 306* ■ *"Module 'WS_SIP' (Web Intelligence VoIP SIP Parser)" on page 343*
`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 329*.
`sync`	Synchronization operations in Check Point cluster **Note** - Also see the debug flag "`sync`" in *"Module 'CPAS' (Check Point Active Streaming)" on page 283*.
`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

Flag	Description
`warning`	General warnings
`wire`	Wire-mode Virtual Machine chain module
`xlate`	NAT issues - basic information
`xltrc`	NAT issues - additional information - going through NAT rulebase
`zeco`	Memory allocations in the Zero-Copy kernel module

Module 'gtp' (GPRS Tunneling Protocol)

Syntax:

```
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

Flag	Description
`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 283*.
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 281*.
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

Flag	Description
`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 314*.

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

Flag	Description
`ioctl`	IOCTL control messages (communication between the kernel and daemons)
`kqstats`	Kernel Worker thread statistics (resetting, initializing, turning off)
`kw`	Kernel Worker state and Pattern Matcher inspection
`leak`	Memory leak detection mechanism
`memory`	Memory allocation operations
`memprof`	Memory allocation operations in the Memory Profiler (when the kernel parameter `fw_conn_mem_prof_enabled=1`)
`misc`	CPU counters, Memory counters, getting/setting of global kernel parameters
`mtctx`	Multi-threaded context - memory allocation, reference count
`packet`	Internal parsing operations on packets
`pcre`	Perl Compatible Regular Expressions (execution, memory allocation)
`pm`	Pattern Matcher compilation and execution
`pmdump`	Pattern Matcher DFA (dumping XMLs of DFAs)
`pmint`	Pattern Matcher compilation
`pools`	Memory pool allocation operations
`queue`	Kernel Worker thread queues
`rem`	Regular Expression Matcher - Pattern Matcher 2nd tier (slow path)
`salloc`	System Memory allocation
`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*

Flag	Description
`thread`	Kernel thread that supplies low level APIs to the kernel thread
`timers`	Internal timers
`usrmem`	User Space platform memory usage
`vbuf`	Virtual buffer
`warning`	General warnings
`worker`	Kernel Worker - queuing and dequeuing

Module 'kissflow' (Kernel Infrastructure Flow)

Syntax:

```
fw ctl debug -m kissflow + {all | <List of Debug Flags>}
```

Note - Also see *"Module 'kiss' (Kernel Infrastructure)" on page 311*.

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 297*, 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

Flag	Description
`state`	Starting and stopping of CoreXL Firewall instances, establishment of relationship between CoreXL Firewall instances
`temp_conns`	Temporary connections
`uid`	Cross-instance Unique IDs
`vpn_multik`	MultiCore VPN (see sk118097)

Module 'MUX' (Multiplexer for Applications Traffic)

R80.20 introduced a new layer between the Streaming layer and the Applications layer - MUX (Multiplexer).

Applications are registered to the Streaming layer through the MUX layer.

The MUX layer chooses to work over PSL (passive streaming) or CPAS (active streaming).

Syntax:

```
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 283*.
`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 322*.
`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

Flag	Description
`tol`	Test Object List algorithm (to determine whether an application is malicious or not)
`udp`	UDP parser
`warning`	General warnings
`ws`	Web Intelligence

Module 'NRB' (Next Rule Base)

Syntax:

```
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 275*.
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 288* ■ *"Module 'dlpk' (Data Loss Prevention - Kernel Space)" on page 290* ■ *"Module 'dlpuk' (Data Loss Prevention - User Space)" on page 291*
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

Flag	Description
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`verbose`	Prints additional information (used with other debug flags)
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module '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 318*.

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 'TPUTILS' (Threat Prevention Utilities)

Syntax

- On the Security Gateway / each Cluster Member, run in the Expert mode:

```
fw ctl debug -m TPUTILS + {all | <List of Debug Flags>}
```

- On the Scalable Platform Security Group, run in the Expert mode:

```
g_fw ctl debug -m TPUTILS + {all | <List of Debug Flags>}
```

Flag	Description
`address`	Information about connection's IP address
`bloom`	Bloom filter operations
`coverage`	Coverage times (entering, blocking, and time spent)
`error`	General errors (the connection is probably rejected)
`global`	Handling of global structure (usually, related to policy)
`info`	General information
`memory`	Memory allocation operations
`module`	Operations in the TPUTILS module (initialization, module loading, calls to the module, policy loading, and so on)
`subject`	Prints the debug subject of each debug message
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`uuid`	Session UUID
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'UC' (UserCheck)

Syntax:

```
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 334*
- *"Module 'UPIS' (Unified Policy Infrastructure)" on page 335*

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*

Flag	Description
`prob_impl`	Implied matched rules
`rulebase`	Unified Policy rulebase
`sec_rb`	Secondary NRB rulebase operations
`stats`	Statistics about connections, transactions
`subject`	Prints the debug subject of each debug message
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`urlf_ssl`	*Currently is not used*
`verbose`	Prints additional information (used with other debug flags)
`vpn`	VPN classifier
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'upconv' (Unified Policy Conversion)

Syntax:

```
fw ctl debug -m upconv + {all | <List of Debug Flags>}
```

Note - Also see:

- *"Module 'UP' (Unified Policy)" on page 332*
- *"Module 'UPIS' (Unified Policy Infrastructure)" on page 335*

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 332*
- *"Module 'upconv' (Unified Policy Conversion)" on page 334*

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

Flag	Description
`tables`	Operations on kernel tables
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`topo`	Information about topology and Anti-Spoofing of interfaces; about Address Range objects
`upapp`	Information about policy installation for Unified Policy application
`update`	Information about policy installation for CMI Update application
`verbose`	Prints additional information (used with other debug flags)
`vpn`	VPN classifier
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'VPN' (Site-to-Site VPN and Remote Access VPN)

Syntax:

```
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 304*
`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

Flag	Description
`nat`	NAT issues , cluster IP manipulation (Cluster Virtual IP address <=> Member IP address)
`om_alloc`	Allocation of Office Mode IP addresses
`osu`	Cluster Optimal Service Upgrade (see sk107042)
`packet`	Events that can happen for every packet, unless covered by more specific debug flags
`pcktdmp`	Prints the encrypted packets before the encryption Prints the decrypted packets after the decryption
`policy`	Events that can happen only for a special packet in a connection, usually related to policy decisions or logs / traps
`queue`	Handling of Security Association (SA) queues
`rdp`	Processing of Check Point RDP connections
`ref`	Reference counting for MSA / MSPI, when storing or deleting Security Associations (SAs)
`resolver`	VPN Link Selection table and Certificate Revocation List (CRL), which is also part of the peer resolving mechanism
`rsl`	Operations on Range Skip List
`sas`	Information about keys and Security Associations (SAs)
`sr`	SecureClient / SecureRemote related issues
`tagging`	Sets the VPN policy of a connection according to VPN communities, VPN Policy related information
`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

Flag	Description
`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 345*.
- 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`')

Flag	Description
`uuid`	Session UUID
`vs`	Prints the VSID of the debugged Virtual System
`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

Flag	Description
`pkt_dump`	Packet dump
`policy`	Policy (installation and enforcement)
`regexp`	Regular Expression library
`report_mgr`	Report manager (errors and logs)
`session`	Session layer
`spii`	Stateful Protocol Inspection Infrastructure (INSPECT streaming)
`ssl_insp`	HTTPS Inspection
`sslt`	SSL Tunneling (SSLT)
`stat`	Memory usage statistics
`stream`	Stream virtualization
`subject`	Prints the debug subject of each debug message
`timestamp`	Prints the timestamp for each debug message (changes when you enable the debug flag '`coverage`')
`uuid`	Session UUID
`vs`	Prints the VSID of the debugged Virtual System
`warning`	General warnings

Module 'WSIS' (Web Intelligence Infrastructure)

Syntax:

```
fw ctl debug -m WSIS + {all | <List of Debug Flags>}
```

Note - Also see *"Module 'WS' (Web Intelligence)" on page 340*.

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

Glossary

A

Anti-Bot
Check Point Software Blade on a Security Gateway that blocks botnet behavior and communication to Command and Control (C&C) centers. Acronyms: AB, ABOT.

Anti-Spam
Check Point Software Blade on a Security Gateway that provides comprehensive protection for email inspection. Synonym: Anti-Spam & Email Security. Acronyms: AS, ASPAM.

Anti-Virus
Check Point Software Blade on a Security Gateway that uses real-time virus signatures and anomaly-based protections from ThreatCloud to detect and block malware at the Security Gateway before users are affected. Acronym: AV.

Application Control
Check Point Software Blade on a Security Gateway that allows granular control over specific web-enabled applications by using deep packet inspection. Acronym: APPI.

Audit Log
Log that contains administrator actions on a Management Server (login and logout, creation or modification of an object, installation of a policy, and so on).

B

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

C

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

Cluster Member
Security Gateway that is part of a cluster.

Compliance
Check Point Software Blade on a Management Server to view and apply the Security Best Practices to the managed Security Gateways. This Software Blade includes a library of Check Point-defined Security Best Practices to use as a baseline for good Security Gateway and Policy configuration.

Content Awareness
Check Point Software Blade on a Security Gateway that provides data visibility and enforcement. Acronym: CTNT.

CoreXL
Performance-enhancing technology for Security Gateways on multi-core processing platforms. Multiple Check Point Firewall instances are running in parallel on multiple CPU cores.

CoreXL Firewall Instance
On a Security Gateway with CoreXL enabled, the Firewall kernel is copied multiple times. Each replicated copy, or firewall instance, runs on one processing CPU core. These firewall instances handle traffic at the same time, and each firewall instance is a complete and independent firewall inspection kernel. Synonym: CoreXL FW Instance.

CoreXL SND
Secure Network Distributer. Part of CoreXL that is responsible for: Processing incoming traffic from the network interfaces; Securely accelerating authorized packets (if SecureXL is enabled); Distributing non-accelerated packets between Firewall kernel instances (SND maintains global dispatching table, which maps connections that were assigned to CoreXL Firewall instances). Traffic distribution between CoreXL Firewall instances is statically based on Source IP addresses, Destination IP addresses, and the IP 'Protocol' type. The CoreXL SND does not really "touch" packets. The decision to stick to a particular FWK daemon is done at the first packet of connection on a very high level, before anything else. Depending on the SecureXL settings, and in most of the cases, the SecureXL can be offloading decryption calculations. However, in some other cases, such as with Route-Based VPN, it is done by FWK daemon.

CPUSE
Check Point Upgrade Service Engine for Gaia Operating System. With CPUSE, you can automatically update Check Point products for the Gaia OS, and the Gaia OS itself.

D

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

Data Loss Prevention
Check Point Software Blade on a Security Gateway that detects and prevents the unauthorized transmission of confidential information outside the organization. Acronym: DLP.

Data Type
Classification of data in a Check Point Security Policy for the Content Awareness Software Blade.

Distributed Deployment
Configuration in which the Check Point Security Gateway and the Security Management Server products are installed on different computers.

Dynamic Object
Special object type, whose IP address is not known in advance. The Security Gateway resolves the IP address of this object in real time.

E

Endpoint Policy Management
Check Point Software Blade on a Management Server to manage an on-premises Harmony Endpoint Security environment.

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

G

Gaia
Check Point security operating system that combines the strengths of both SecurePlatform and IPSO operating systems.

Gaia Clish
The name of the default command line shell in Check Point Gaia operating system. This is a restricted shell (role-based administration controls the number of commands available in the shell).

Gaia Portal
Web interface for the Check Point Gaia operating system.

H

Hotfix
Software package installed on top of the current software version to fix a wrong or undesired behavior, and to add a new behavior.

HTTPS Inspection
Feature on a Security Gateway that inspects traffic encrypted by the Secure Sockets Layer (SSL) protocol for malware or suspicious patterns. Synonym: SSL Inspection. Acronyms: HTTPSI, HTTPSi.

I

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

Identity Awareness
Check Point Software Blade on a Security Gateway that enforces network access and audits data based on network location, the identity of the user, and the identity of the computer. Acronym: IDA.

Identity Logging
Check Point Software Blade on a Management Server to view Identity Logs from the managed Security Gateways with enabled Identity Awareness Software Blade.

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

IPS
Check Point Software Blade on a Security Gateway that inspects and analyzes packets and data for numerous types of risks (Intrusion Prevention System).

IPsec VPN
Check Point Software Blade on a Security Gateway that provides a Site to Site VPN and Remote Access VPN access.

J

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

K

Kerberos
An authentication server for Microsoft Windows Active Directory Federation Services (ADFS).

L

Log Server
Dedicated Check Point server that runs Check Point software to store and process logs.

Logging & Status
Check Point Software Blade on a Management Server to view Security Logs from the managed Security Gateways.

M

Management Interface
(1) Interface on a Gaia Security Gateway or Cluster member, through which Management Server connects to the Security Gateway or Cluster member. (2) Interface on Gaia computer, through which users connect to Gaia Portal or CLI.

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

Manual NAT Rules
Manual configuration of NAT rules by the administrator of the Check Point Management Server.

Mobile Access
Check Point Software Blade on a Security Gateway that provides a Remote Access VPN access for managed and unmanaged clients. Acronym: MAB.

Multi-Domain Log Server
Dedicated Check Point server that runs Check Point software to store and process logs in a Multi-Domain Security Management environment. The Multi-Domain Log Server consists of Domain Log Servers that store and process logs from Security Gateways that are managed by the corresponding Domain Management Servers. Acronym: MDLS.

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

N

Network Object
Logical object that represents different parts of corporate topology - computers, IP addresses, traffic protocols, and so on. Administrators use these objects in Security Policies.

Network Policy Management
Check Point Software Blade on a Management Server to manage an on-premises environment with an Access Control and Threat Prevention policies.

O

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

P

Provisioning
Check Point Software Blade on a Management Server that manages large-scale deployments of Check Point Security Gateways using configuration profiles. Synonyms: SmartProvisioning, SmartLSM, Large-Scale Management, LSM.

Q

QoS
Check Point Software Blade on a Security Gateway that provides policy-based traffic bandwidth management to prioritize business-critical traffic and guarantee bandwidth and control latency.

R

Rule
Set of traffic parameters and other conditions in a Rule Base (Security Policy) that cause specified actions to be taken for a communication session.

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

S

SecureXL
Check Point product on a Security Gateway that accelerates IPv4 and IPv6 traffic that passes through a Security Gateway.

Security Gateway
Dedicated Check Point server that runs Check Point software to inspect traffic and enforce Security Policies for connected network resources.

Security Management Server
Dedicated Check Point server that runs Check Point software to manage the objects and policies in a Check Point environment within a single management Domain. Synonym: Single-Domain Security Management Server.

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

SIC
Secure Internal Communication. The Check Point proprietary mechanism with which Check Point computers that run Check Point software authenticate each other over SSL, for secure communication. This authentication is based on the certificates issued by the ICA on a Check Point Management Server.

SmartConsole
Check Point GUI application used to manage a Check Point environment - configure Security Policies, configure devices, monitor products and events, install updates, and so on.

SmartDashboard
Legacy Check Point GUI client used to create and manage the security settings in versions R77.30 and lower. In versions R80.X and higher is still used to configure specific legacy settings.

SmartProvisioning
Check Point Software Blade on a Management Server (the actual name is "Provisioning") that manages large-scale deployments of Check Point Security Gateways using configuration profiles. Synonyms: Large-Scale Management, SmartLSM, LSM.

SmartUpdate
Legacy Check Point GUI client used to manage licenses and contracts in a Check Point environment.

Software Blade
Specific security solution (module): (1) On a Security Gateway, each Software Blade inspects specific characteristics of the traffic (2) On a Management Server, each Software Blade enables different management capabilities.

Standalone
Configuration in which the Security Gateway and the Security Management Server products are installed and configured on the same server.

T

Threat Emulation
Check Point Software Blade on a Security Gateway that monitors the behavior of files in a sandbox to determine whether or not they are malicious. Acronym: TE.

Threat Extraction
Check Point Software Blade on a Security Gateway that removes malicious content from files. Acronym: TEX.

U

Updatable Object
Network object that represents an external service, such as Microsoft 365, AWS, Geo locations, and more.

URL Filtering
Check Point Software Blade on a Security Gateway that allows granular control over which web sites can be accessed by a given group of users, computers or networks. Acronym: URLF.

User Directory
Check Point Software Blade on a Management Server that integrates LDAP and other external user management servers with Check Point products and security solutions.

V

VSX
Virtual System Extension. Check Point virtual networking solution, hosted on a computer or cluster with virtual abstractions of Check Point Security Gateways and other network devices. These Virtual Devices provide the same functionality as their physical counterparts.

VSX Gateway
Physical server that hosts VSX virtual networks, including all Virtual Devices that provide the functionality of physical network devices. It holds at least one Virtual System, which is called VS0.

Z

Zero Phishing
Check Point Software Blade on a Security Gateway (R81.20 and higher) that provides real-time phishing prevention based on URLs. Acronym: ZPH.