Check Point
SOFTWARE TECHNOLOGIES LTD.

WE SECURE THE FUTURE

Check Point Software Technologies

Check Point Cryptographic Library

Cryptographic Module

Version 1.0 (Firmware)

FIPS 140-2 Non-Proprietary Security Policy

Level 1 Validation

Document revision 026, July 2017

Check Point Software Technologies
5 Ha'solelim Street
Tel Aviv, 67897
Israel
http://www.checkpoint.com/

Prepared for Check Point Software Technologies Ltd. by

RYCOMBE CONSULTING

Rycombe Consulting Limited
http://www.rycombe.com
+44 1273 476366

Contents

Figures

1 Introduction

This section identifies the cryptographic module; describes the purpose of this document; provides external references for more information; and explains how the document is organized.

1.1 Identification

Module Name	Check Point Cryptographic Library
Module Version	1.0

1.2 Purpose

This is the non-proprietary FIPS 140-2 Security Policy for the Check Point Cryptographic Library, also referred to as "the module" within this document. This Security Policy details the secure operation of Check Point Cryptographic Library as required in Federal Information Processing Standards Publication 140-2 (FIPS 140-2) as published by the National Institute of Standards and Technology (NIST) of the United States Department of Commerce.

1.3 References

For more information on Check Point products please visit: http://www.checkpoint.com/. For more information on NIST and the Cryptographic Module Validation Program (CMVP), please visit http://csrc.nist.gov/groups/STM/cmvp/index.html.

1.4 Document Organization

This Security Policy document is one part of the FIPS 140-2 Submission Package. This document outlines the functionality provided by the module and gives high-level details on the means by which the module satisfies FIPS 140-2 requirements. With the exception of this Non-Proprietary Security Policy, the FIPS 140-2 Submission documentation may be Check Point proprietary or otherwise controlled and releasable only under appropriate non-disclosure agreements. For access to these documents, please contact Check Point.

The various sections of this document map directly onto the sections of the FIPS 140-2 standard and describe how the module satisfies the requirements of that standard.

1.5 Document Terminology

Term	Description
AES	Advanced Encryption Standard
AH	Authentication Header
ANSI	American National Standards Institute
API	Application Programming Interface
BIOS	Basic Input Output Services
CAVP	Cryptographic Algorithm Validation Program
CMAC	Cipher-based Message Authentication Code
CMSP	Cryptographic Module Security Policy
CMVP	Cryptographic Module Validation Program
CPU	Central Processing Unit (Microprocessor)
CSP	Critical Security Parameters
DES	Data Encryption Standard
DRBG	Deterministic Random-bit Generator
DVD	Digital Video Disc
ECDSA	Elliptic Curve Digital Signature Algorithm
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
ESP	Encapsulating Security Payload
FIPS	Federal Information Processing Standard
GAiA	Check Point proprietary operating environment
HDD	Hard Disk Drive
HMAC	Keyed-Hash Message Authentication Code
IKE	Internet Key Exchange
IPsec	Internet Protocol Security: a protocol suite for securing Internet Protocol (IP) communications by authenticating and encrypting each IP packet of a communication session
KDF	Key Derivation Function
LCD	Liquid Crystal Display
LED	Light Emitting Diode
N/A	Not Applicable
NDRNG	Non-deterministic Random Number Generator
NIST	National Institute of Standards and Technology
OS	Operating System
PCI	Peripheral Component Interconnect
PCIe	Peripheral Component Interconnect Express
RAM	Random-access Memory
RBG	Random Bit Generator
RFC	Request for Comments

TERM	DESCRIPTION
RNG	Random Number Generator
RSA	An algorithm for public-key cryptography. Named after Rivest, Shamir and Adleman who first publicly described it.
SATA	Serial AT Attachment
SCSI	Small Computer System Interface
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
SIC	Secure Internal Communication – a Check Point proprietary protocol
SP	NIST Special Publication document
TLS	Transport Layer Security
Triple-DES	Triple-DES
USB	Universal Serial Bus

Figure 1 Document terminology

2 Check Point Cryptographic Library

This section provides the details of how the module meets the FIPS 140-2 requirements.

2.1 Overview

The module provides cryptographic services to Check Point products.

2.2 Module Specification

The Check Point Cryptographic Library is a firmware module that provides cryptographic services to Check Point products.

The module is classified as a multi-chip standalone module.

The module provides a number of NIST validated cryptographic algorithms for services such as IPsec. The module provides applications with a library interface that enables them to access the various cryptographic algorithm functions supplied by the module.

2.2.1 Hardware and Firmware components

The module is a firmware module that resides on proprietary hardware (see Figure 4). For the purposes of FIPS 140-2 testing, the module is evaluated running on the Check Point 12400 appliance.

The module is packaged as a number of distinct binary images:

File Name(s)	Processor
libcpbcrypt.so, libcpcert.so, libcpprng.so, libcpopenssl.so, vpnd, vpnk, libikev2.so, libcptls.so	Intel Xeon

Figure 2 Module binary images

2.2.2 Cryptographic Boundary

The physical boundary of the module is the case of the hardware appliance on which it is installed. For the purposes of this evaluation, the module was tested on a Check Point 12400 appliance.

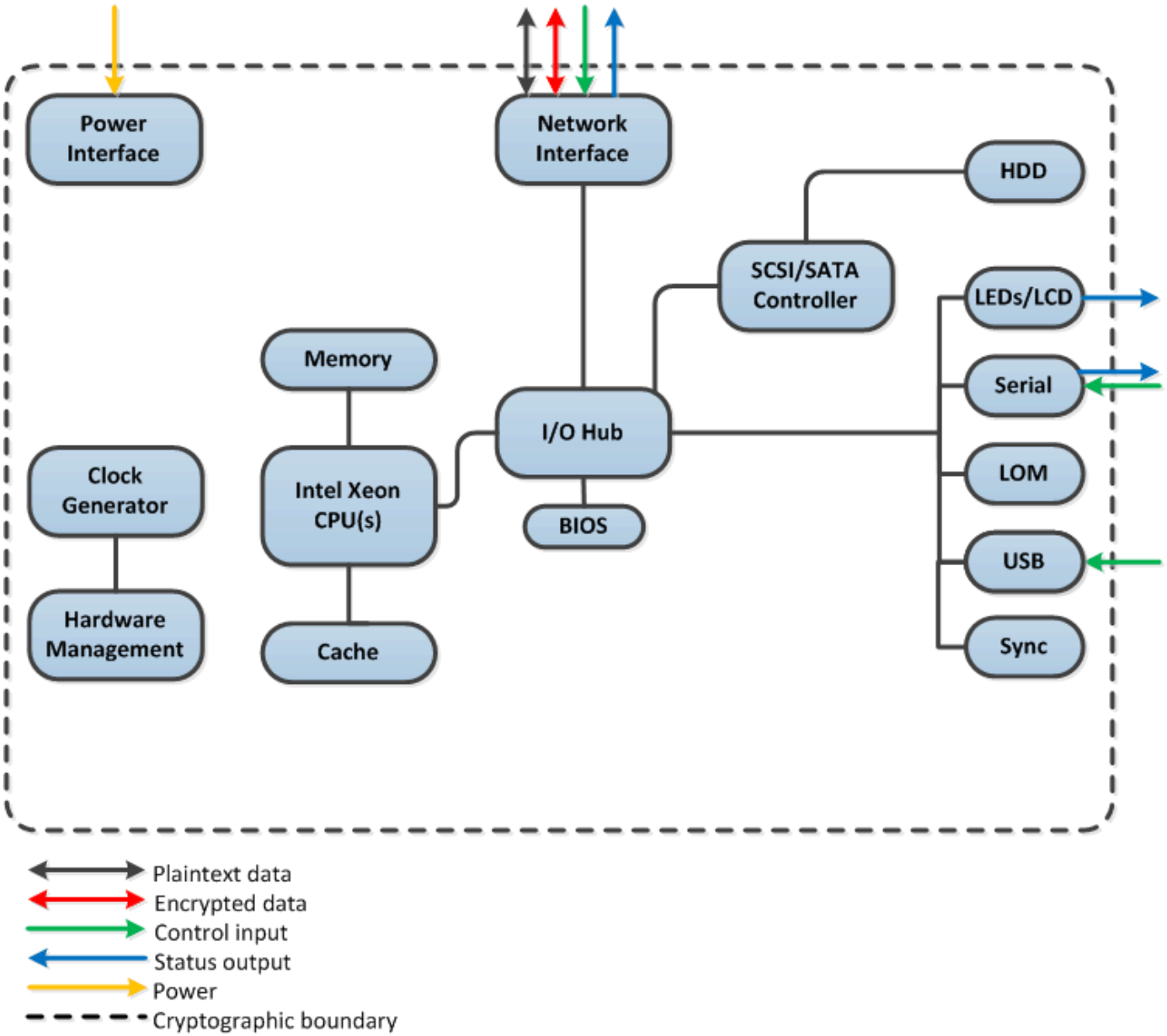

Figure 3 Check Point 12400 appliance hardware block diagram

The module is a firmware module running on a proprietary hardware platform. See Figure 3. The processor of this platform executes all firmware. All firmware components of the module are persistently stored within the device and, while executing, are stored in the device local RAM. The cryptographic boundary of the module includes only the module firmware as listed in Figure 2.

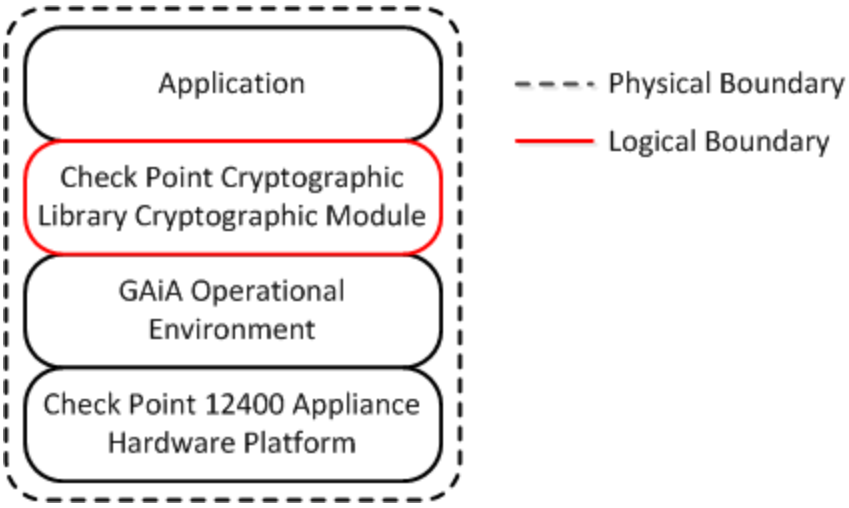

Figure 4 Logical Diagram of the Cryptographic Boundary

2.2.3 Scope of Evaluation

The cryptographic module meets the overall requirements applicable to Level 1 security of FIPS 140-2.

Security Requirements Section	Level
Cryptographic Module Specification	1
Module Ports and Interfaces	1
Roles, Services and Authentication	1
Finite State Model	1
Physical Security	1
Operational Environment	N/A
Cryptographic Key Management	1
EMI/EMC	1
Self-Tests	1
Design Assurance	1
Mitigation of Other Attacks	N/A

Figure 5 Security Level specification per individual areas of FIPS 140-2

2.2.4 Cryptographic Algorithms

2.2.4.1 Approved algorithms

The following table provides details of the approved algorithms that are included within the module:

ALGORITHM TYPE	ALGORITHM	CAVP CERTIFICATE NUMBER	NOTES
Symmetric key	AES	#3418	AES with 128-bit or 256-bit keys using CBC and GCM[1] modes. The modes and sizes are validated for both encryption and decryption.
	Triple-DES	#1929	Three-key Triple-DES (192-bit keys). CBC mode.
Asymmetric Key	RSA	#1750	Key generation (2048–bit keys. Signature generation (2048-bit/3072-bit with either SHA-256, SHA-384 or SHA-512). Signature verification. (1024-bit/2048-bit signature verification with either SHA-1, SHA-256, SHA-384 or SHA-512).
	ECDSA	#685	Supports P-256, P-384, and P-521 curves. FIPS186-4: PKG: CURVES(P-256 P-384 P-521 Testing Candidates) PKV: CURVES(P-256 P-384 P-521) SigGen: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512) SigVer: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512)) SHS: #2824 DRBG: # 823
Hashing	SHS	#2824	SHA-1[2] (disallowed for signature generation), SHA-256, SHA-384, SHA-512.

[1] The module complies with SP 800-52 and is compatible with the specified versions of TLS in Section 4 of RFC 5288. The module complies with RFC 6071 and that an IKEv2 protocol (RFC 7296) shall be used to establish the shared secret SKEYSEED from which the AES GCM encryption keys are derived.

[2] SHA-1 for non-digital signature applications:
SHA-1 is not allowed for digital signature generation. For all other hash function applications, the use of SHA-1 is acceptable. The other applications include HMAC, Key Derivation Functions (KDFs), Random Number Generation (RNGs and RBGs), and

Algorithm Type	Algorithm	CAVP Certificate Number	Notes
Message Authentication Code	HMAC	#2176	HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384.
Random number generator	Hash DRBG	#823	Hash DRBG with SHA-256 and a seed length of 440 bits in accordance with SP800-90A.

Figure 6 Approved Algorithms

The following table lists the key derivation functions (and their associated CVL certificate numbers) implemented by the module.

Approved KDF	CAVP CVL Certificate Number
Transport Layer Security (TLS) v1.0 (SP 800-135)	#514
Internet Key Exchange (IKE) v1 and v2 (SP 800-135)	#514

Figure 7 Approved Key Derivation Functions

For each of these approved Key Derivation Functions the module supports or uses the corresponding protocol. These protocols have not been reviewed or tested by the CAVP or CMVP as testing such protocols is not within the scope of CMVP or CAVP activities.

The following table lists the individual components of FIPS approved and NIST recommended cryptographic algorithms within the module that have been tested:

Component	CAVP CVL Certificate Number	Notes
All of SP800-56A EXCEPT KDF	#920	ECC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: KPG Full Validation Partial Validation) SCHEMES: FullMQV: (KARole: Initiator / Responder) EC: P-256 ED: P-384 EE: P-521 ECDSA #685 SHS #2824 DRBG #823

Figure 8 Approved Components

2.2.4.2 Algorithms allowed in approved mode

- RSA (key wrapping; key establishment methodology provides 112 or 128 bits of encryption strength).

---

hash-only applications (e.g., hashing passwords and using SHA-1 to compute a checksum, such as the approved integrity technique specified in Section 4.6.1 of [FIPS 140-2]).

- Diffie-Hellman (key agreement; key establishment methodology provides between 112 and 128 bits of encryption strength).
- The NDRNG that is used to seed the random number generator.

2.2.4.3 Non-approved algorithms

SHA-1 for digital signature generation is available when the module is installed as described in section 3. However, using this results in a non-approved mode of operation.

2.2.5 Components excluded from the security requirements of the standard

There are no components excluded from the security requirements of the standard.

2.3 Physical ports and logical interfaces

The module's physical boundary is that of the device on which it is installed. The device supports sufficient interfaces to allow operators to initiate cryptographic operations and determine the module status.

The module provides its logical interfaces via Application Programming Interface (API) calls. This logical interface exposes services (described in section 2.4.2) that applications utilize directly.

The logical interfaces provided by the module are mapped onto the FIPS 140-2 logical interfaces: data input, data output, control input, and status output as follows:

FIPS 140-2 Logical Interface	Module Mapping	Appliance Physical Interface
Data Input	Parameters passed to the module via API calls	Network ports
Data Output	Data returned from the module via API calls	Network ports
Control Input	API Calls and/or parameters passed to API calls	USB ports, serial console, network ports, power switch
Status Output	Information received in response to API calls	Network ports, serial console, LCD Display, LEDs.
Power Interface	There is no separate power or maintenance access interface beyond the power interface provided by the device that contains the module	Power connector

Figure 9 Module Interfaces

2.4 Roles, Services and Authentication

2.4.1 Roles

The Cryptographic Module implements both a Crypto Officer role and a User role. Roles are assumed implicitly upon accessing the associated services. Section 2.4.2 summarizes the services available to each role.

ROLE	DESCRIPTION
Crypto Officer	The administrator of the module having full configuration and key management privileges.
User	General User of the module

Figure 10 Roles

2.4.2 Services

Most of the services provided by the module are provided via access to API calls using interfaces exposed by the module.

However, some of the services, such as power-up module integrity testing are performed automatically and so have no function API, but do provide status output.

SERVICE	ROLE	INPUT	OUTPUT	CSP ACCESS	DESCRIPTION
Symmetric Data Encryption and Decryption	User	ESP data.	ESP data.	Session keys, Diffie-Hellman key pairs.	The module supports IPsec/ESP for data encryption and IPsec/ESP for data integrity.
Message Digest	User	Data.	Hash of input data.	None.	Used for data packet integrity within ESP and AH. Used by keyed hash and key generation services. This service provides access to SHA-1, SHA-256, SHA-384 and SHA-512 functionality. SHA-1 is non-approved for digital signature generation.
Keyed Hash	User	Data or message.	Keyed hash of input.	HMAC key.	Used for data packet integrity within ESP and AH.
Digital Signature Generation[3]	User	Data.	Digital signature of data.	Asymmetric key pair (read access)	Used to authenticate to the module during IKE.
Digital Signature Verification	User	Signed data.	Result of verification: Success or failure.	Asymmetric key pair (read access)	Used to authenticate to the module during IKE.

[3] The Digital Signature Generation service can be evoked in a non-approved way if SHA-1 is used. See section 3.1 for more details.

RSA Key Generation	Crypto Officer	Size of key required.	Validated RSA key pair.	Asymmetric key pair	Used to generate RSA key pairs.
ECDSA Key Generation	Crypto Officer	Curve	Validated ECDSA key pair.	Asymmetric key pair	Used to generate ECDSA key pairs.
Symmetric Key Generation	Crypto Officer	Size of key required.	Symmetric key	SP 800-90A Hash_DRBG V & C values. Write access: Session keys; Diffie-Hellman key pairs.	Used to generate symmetric key pairs.
Show Status	User/Crypto Officer	Service inputs.	Service outputs.	All CSPs.	The output indicators described for all services. The Show Status service is provided collectively across all services. Each service provides some status information as part of its service output.
Self-tests	User	Power up the system or power cycle the system.	Success: Module powers up without error.	Integrity Check key.	Self-tests run automatically at power up. Includes KATs for all approved algorithms and ECDSA P-521 integrity check with SHA-512 for integrity testing of the cryptographic module firmware.

Figure 11 Approved Services

2.4.3 Authentication

The module does not support any authentication mechanisms. The module does not perform authentication.

2.5 Physical Security

The Cryptographic Module is a firmware-only cryptographic module. The module was tested on a Check Point 12400 appliance which is built from production-grade components that incorporate standard passivation.

2.6 Operational Environment

The operational environment is the Check Point 12400 appliance that the Check Point Cryptographic Library Cryptographic Module runs on. The module does not provide a general-purpose operating system (OS) to module operators. The module's operational environment includes Check Point's proprietary non-modifiable GAiA Operating System running on (for the purposes of testing) the Check Point 12400 appliance.

The Cryptographic Module is characterized as a firmware module.

2.7 Cryptographic Key Management

2.7.1 Random Number Generators

The module contains an SP 800-90A approved Hash DRBG. Checks are made to ensure that the quality of the entropy remains high enough to be used to seed the DRBG.

Entropy is provided by a CPU time jitter based non-physical true random number generator. The entropy seeds the DRBG via the /dev/random library.

2.7.2 Key Generation

The module generates keys using an approved key generation mechanism made up of an SP 800-90A Hash DRBG and available entropy conditioned by /dev/random supplemented by the standard Linux Kernel RNG built-in noise source (timing events from storage I/O, inter-process calls (IPCs) and human interface devices (if present)). The module uses 440-bits of entropy to generate keys. Symmetric keys generated by the module are unmodified output from the SP 800-90A DRBG. The key generation provides a security strength of 256-bits.

2.7.3 Key Table

The following tables list all of the keys and CSPs within the module, describe their purpose, and describe how each key is generated, entered and output, stored and destroyed.

Note: “Service” keys. A number of the service APIs are for functions that perform cryptographic operations. Some of these accept keys as parameters. There are also APIs for functions that generate keys and pass them back to the calling application. These keys are ephemeral. They are not stored within the module. After these keys have been used by the API functions, they are zeroized within the module. It is the responsibility of the calling application to ensure that it stores, handles and destroys keys appropriately.

KEY	USE	GENERATION/ESTABLISHMENT	INPUT/OUTPUT	STORAGE	DESTRUCTION
Asymmetric Private Key RSA (2048 or 3072-bits) or ECDSA (P-256, P-384, or P-521 curve).	RSA or ECDSA key pair used for authentication (TLS or IKE).	Internally generated by SP 800-90A DRBG.	N/A	Stored on disk in plaintext.	Zeroized by reformatting the module's hard drive containing the module's firmware.
Asymmetric Public Key RSA (1024[4], 2048 or 3072-bits) or ECDSA (P-256, P-384, or P-521 curve).	RSA or ECDSA key pair used for authentication (TLS or IKE).	Internally generated by SP 800-90A DRBG.	N/A	Stored on disk in plaintext.	Zeroized by reformatting the module's hard drive containing the module's firmware.
Diffie-Hellman Private Key 2048, 3072, 4096, 6144, or 8192-bits	Key exchange during IKE.	Generated by SP 800-90A DRBG and established by IKE negotiations.	N/A	Not persistently stored (public parameters stored on disk in plaintext).	Zeroized when the session is terminated or power is removed from the module.
Diffie-Hellman Public Key **2048, 3072, 4096, 6144, or 8192-bits**	Key exchange during IKE.	Generated by SP 800-90A DRBG and established by IKE negotiations.	N/A	Not persistently stored (public parameters stored on disk in plaintext).	Zeroized when the session is terminated or power is removed from the module.
Session keys Triple-DES keys (192 bits), AES (128 bits, 256 bits).	To secure IPsec and TLS traffic (SIC).	Generated by SP 800-90A DRBG and established by IKE negotiations. Generated by TLS handshake.	N/A	Not persistently stored. Cached to disk (plaintext).	Zeroized when the session is terminated or power is removed from the module.
HMAC key (160 bits, 256 bits, 384 bits or 512 bits depending on size	Authenticated TLS traffic.	Generated by SP 800-90A DRBG and established by TLS handshake.	N/A	Cached to disk (plaintext).	Zeroized by reformatting the module's hard drive containing the module's firmware.

[4] 1024-bit RSA only used for signature verification in approved mode of operation

KEY	USE	GENERATION/ESTABLISHMENT	INPUT/OUTPUT	STORAGE	DESTRUCTION
of hash used).					
Integrity Check key ECDSA P-521 curve certificate.	Module firmware integrity check (ECDSA P-521 key).	Generated outside the module and hardcoded into module firmware.	N/A	Hardcoded into the CPHASH binary in plaintext.	Zeroized by reformatting the module's hard drive containing the module's firmware.
SP 800-90A Hash_DRBG V & C values Internal state for the Hash_DRBG	Random bit generator.	Internal state derived from seed value	N/A	RAM only.	Zeroized when the session is terminated or power is removed from the module.

Figure 12 Key Table

2.7.4 Access to Key Material

The following table shows the access that an operator has to specific keys or other critical security parameters when performing each of the services relevant to his/her role.

Services	Role	KEY					
		Asymmetric Key Pair	Diffie-Hellman Key Pairs	Session Keys	HMAC Key	Integrity Check Key	SP 800-90A Hash_DRBG V & C Values
Symmetric Data Encryption and Decryption	User			U			
Message Digest	User						
Keyed Hash	User				U		
Digital Signature Generation	User	U					
Digital Signature Verification	User	U					
RSA Key Generation	CO	W					U
ECDSA Key Generation	CO	W					U
Symmetric Key Generation	CO		W	W			U
Show Status	User						
Self-tests	User					U	

Figure 13 Access to keys by services

Access Rights	Blank	N/A
	R	Read
	W	Write
	U	Use

2.8 Self-Tests

The module implements both power-up and conditional self-tests as required by FIPS 140-2.

The following two sections outline the tests that are performed.

2.8.1 Power-up self-tests

After power-cycling or booting the appliance the module executes the Power-Up Self-Tests with no further inputs or actions by the operator.

The module implements the following power-up self-tests. The module inhibits all data output while it is operating in the Self-Test state.

OBJECT	TEST
SHA-1	Known answer test
SHA-256	Known answer test
SHA-384	Known answer test
SHA-512	Known answer test
AES-128-CBC Encrypt	Known answer test
AES-128-CBC Decrypt	Known answer test
AES-256-CBC Encrypt	Known answer test
AES-256-CBC Decrypt	Known answer test
AES-128-GCM Encrypt	Known answer test
AES-128-GCM Decrypt	Known answer test
AES-256-GCM Encrypt	Known answer test
AES-256-GCM Decrypt	Known answer test
Triple-DES CBC Encrypt	Known answer test
Triple-DES CBC Decrypt	Known answer test
HMAC-SHA-1	Known answer test
HMAC-SHA-256	Known answer test
HMAC-SHA-384	Known answer test
Hash DRBG	Known answer test SP 800-90A Section 11.3 Health Tests (instantiate, reseed and generate)
RSA Signature Generation	Known answer test
RSA Signature Verification	Known answer test
ECDSA Signature Generation	Known answer test
ECDSA Signature Verification	Known answer test
KAS ECC	Primitive Z computation Known answer test
Firmware Integrity Check	ECDSA P-521 integrity check with SHA-512

Figure 14 Power-up self-tests

If any of the power-up KATs fail, the system enters an error state. Any self-test errors are output directly to the console output and specific errors are stored in the filesign.log file. “dmesg” can be run to indicate the status of self-tests.

While in the error state the module inhibits all data output and all cryptographic operations are prohibited. The operator may power cycle the module to re-run the power up self-tests.

2.8.2 Conditional self-tests

The module implements the following conditional self-tests:

EVENT	TEST
Module requests a random number from the FIPS Approved SP800-90 DRBG	A continuous random number generator test
Module requests a random number from the NDRNG used to seed the FIPS Approved SP800-90A DRBG	A continuous random number generator test
RSA key pair is generated	RSA pair-wise consistency test
ECDSA key pair is generated	ECDSA pair-wise consistency test

Figure 15 Conditional self-tests

If any of the conditional self-tests fail, the module shuts down with an error. Any self-test errors are output directly to the console output and specific errors are stored in the filesign.log file. “dmesg” can be run to indicate the status of self-tests.

While in the error state the module inhibits all data output and all cryptographic operations are prohibited. The operator may power cycle the module to restart the module.

2.9 Design Assurance

Check Point employs industry standard best practices in the design, development, production and maintenance of all of its products, including the FIPS 140-2 module.

This includes the use of an industry standard configuration management system that is operated in accordance with the requirements of FIPS 140-2, such that each configuration item that forms part of the module is stored with a label corresponding to the version of the module and that the module and all of its associated documentation can be regenerated from the configuration management system with reference to the relevant version number.

Design documentation for the module is maintained to provide clear and consistent information within the document hierarchy to enable transparent traceability between corresponding areas throughout the document hierarchy, for instance, between elements of this Cryptographic Module Security Policy (CMSP) and the design documentation.

Guidance appropriate to an operator's Role is provided with the module and provides all of the necessary assistance to enable the secure operation of the module by an operator, including the Approved security functions of the module.

Delivery of the Cryptographic Module to customers from the vendor is via secure download. The module firmware downloaded can be verified using SHA-256 hash values that are downloaded separately.

2.10 Mitigation of Other Attacks

The module does not mitigate any other attacks.

3 Secure Operation

This module firmware can be downloaded from the Check Point Secure Knowledge system. A customer will be sent the appropriate link and credentials that they require to do this.

The module is delivered in the fw1_wrapper installation package that will install automatically.

Edit the **/bin/fips** files on SA, GW and opsechost from expert mode > comment the if condition for management, securexl and rtm.

Once installed, the module must be configured to operate in FIPS mode. This is achieved by running the "**fips on**" shell command from a command-line prompt.

To ensure the module is operating in the approved mode, an operator can observe the following approved mode of operation indicator by executing the **ckp_regedit -p "Software/Checkpoint/SIC** CLI command: **FIPS_140=[n]1**

The "cpcrypto ver" command can be used to determine the specific version of the module that has been installed.

The evaluated module was tested on the Check Point 12400 appliance.

3.1 Non-approved mode of operation

If the module is not installed according to the instructions in Section 3 the module will be operating non-compliantly. Installing the module as instructed in Section 3 ensures that there aren't any non-compliant algorithms in the module.

Configuring the module into the approved mode is a one-way operation. After executing the steps to place the module into the approved mode, there is no way to access the above algorithms without completely reinstalling the module in a manner not conformant with this Security Policy. Reinstalling the mode fully zeroizes the existing instantiation of the module.

After the module has been installed according to the instructions provided in Section 3 of this Security Policy, there is only one non-approved algorithm that is not allowed for use in the approved mode that is available to the operator: SHA-1, when used for signature generation. Using SHA-1 for signature generation results in a non-approved mode of operation.

If the operator uses the SHA-1 algorithm with the "Digital Signature Generation" service specified in Figure 11 the module will be operating in the non-approved mode of operation.

3.2 Zeroization

All keys can be zeroized. Ephemeral keys are zeroized by session termination/power cycle. Persistently stored keys can be zeroized by reformatting the hard drive which is not a callable service that the module offers. The module should be under the direct control of the CO when zeroization occurs.