Hewlett-Packard Development Company, L.P.
NonStop Volume Level Encryption (NSVLE)
Product No: T0867 SW Version: 2.0

FIPS 140–2 Non–Proprietary Security Policy

FIPS Security Level: 1
Document Version: 1.3

Hewlett-Packard Development Company,
3000 Hanover Street Palo Alto, CA 94394
Phone: (650) 857-1501
http://www.hp.com

TableofContents

TableofFigures

ListofTables

1 Introduction

1.1 Purpose

This is a non–proprietary Cryptographic Module Security Policy for the NonStop Volume Level Encryption (NSVLE) from Hewlett-Packard Development Company, L.P. This Security Policy describes how the NonStop Volume Level Encryption meets the security requirements of FIPS 140–2 and how to run the module in a secure FIPS 140–2 mode. This policy was prepared as part of the Level 1 FIPS 140–2 validation of the module.

FIPS 140–2 (Federal Information Processing Standards Publication 140–2 – *Security Requirements for Cryptographic Modules*) details the U.S. and Canadian Government requirements for cryptographic modules. More information about the FIPS 140–2 standard and validation program is available on the Cryptographic Module Validation Program (CMVP) website, which is maintained by the National Institute of Standards and Technology (NIST) and the Communication Security Establishment Canada (CSEC): http://csrc.nist.gov/groups/STM/cmvp.

The NonStop Volume Level Encryption is referred to in this document as NSVLE or the module.

1.2 References

This document deals only with operations and capabilities of the module in the technical terms of a FIPS 140–2 cryptographic module security policy. More information is available on the module from the following sources:

- The HP website (http://www.hp.com) contains information on the full line of products from HP.
- For any related questions regarding the NSVLE please contact HP.

1.3 Document Organization

The Security Policy document is one document in a FIPS 140–2 Submission Package. In addition to this document, the Submission Package contains:

- Vendor Evidence Document
- Finite State Model
- Other supporting documentation as additional references

With the exception of this Non–Proprietary Security Policy, the FIPS 140–2 Submission Package is proprietary to HP and is releasable only under appropriate non–disclosure agreements. For access to these documents, please contact HP.

2 NSVLE

This section describes the NonStop Volume Level Encryption (NSVLE) module from HP.

2.1 Overview

Hewlett–Packard's NonStop technology has been leading the way for over three decades, providing reliable solutions for the most demanding enterprises. The NonStop platform is used in complex computing environments, where business–critical applications need 24 x 7 availability, extreme scalability, and fault–tolerance. NonStop plays an important role in major industries and markets, including finance, healthcare, telecommunications, manufacturing, retail, and government.

HP NonStop Volume Level Encryption, or NSVLE, is a fully integrated encryption solution using FIPS–Approved algorithms to protect data from threats such as theft and unauthorized disclosure. NSVLE is intended to be used by systems with a NonStop infrastructure that include at a minimum the following components:

- HP Integrity NonStop NS–Series or BladeSystem servers for application support
- HP Enterprise Secure Key Manager (ESKM) for FIPS-Approved key generation and retrieval (FIPS Validation Certificate#1922)
- HP Storage Cluster I/O Modules (CLIM) for connecting disk and tape media
- Storage devices such as SAS (Serial Attached SCSI[1]) disks, StorageWorks XP Disk Arrays, and LTO[2] Generation 5 tape devices

Within this infrastructure, NSVLE resides as a software module on each Storage CLIM and provides data at rest encryption for supported SAS and Fibre Channel connected storage devices. Figure 1 provides an illustration of the NonStop architecture.

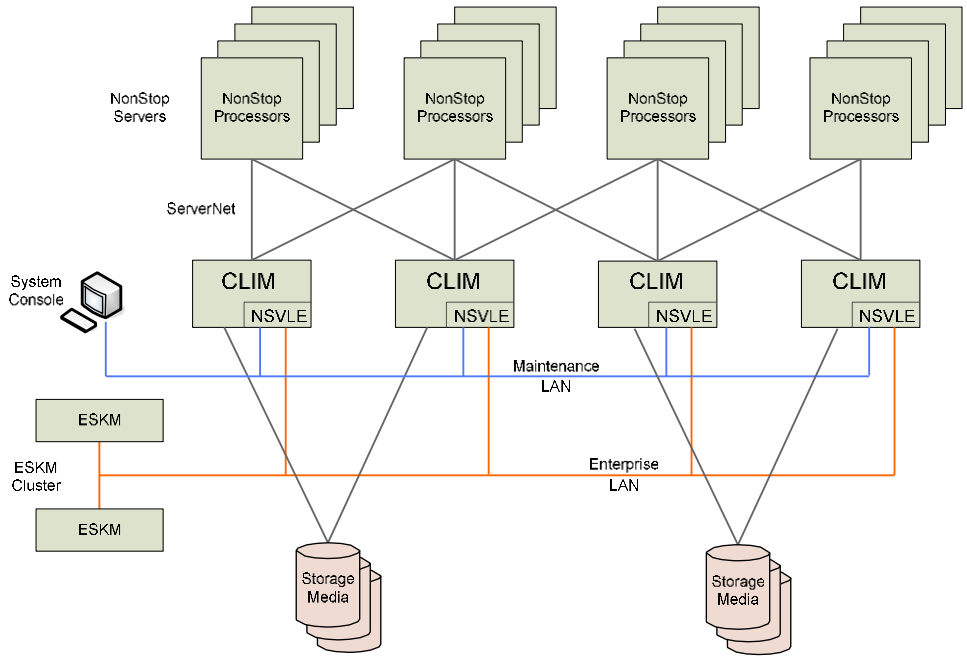

Figure 1 – Architecture of NSVLE Environment

[1] SCSI – Small Computer System Interface
[2] LTO – Linear Tape-Open

CLIMs, specifically Storage CLIMs in this document, are HP ProLiant class servers that act as adapters for disks and tapes, using advanced caching technology that speeds up processing. Communication between NonStop servers and CLIMs is done with a combination of the Maintenance LAN and ServerNet, the core interconnectivity technology for NonStop systems. The CLIMs are connected directly to the Enterprise LAN so that they can communicate with the key manager (ESKM) cluster. Interactions between the ESKM and CLIM must be authenticated using certificates and encrypted through TLS[3], so that the CLIM can securely receive keys from the ESKM.

NSVLE provides both initial volume encryption and then subsequent ongoing encryption of data with key rotation, all the while keeping data online, even during write operations. Existing applications can read and write data normally while data is automatically encrypted and decrypted as it passes through the CLIM.

The NonStop Volume Level Encryption is validated at the following FIPS 140–2 section levels:

Table 1 – Security Level Per FIPS 140–2 Section

Section	Section Title	Level
1	Cryptographic Module Specification	1
2	Cryptographic Module Ports and Interfaces	1
3	Roles, Services, and Authentication	1
4	Finite State Model	1
5	Physical Security	N/A
6	Operational Environment	1
7	Cryptographic Key Management	1
8	EMI/EMC[4]	1
9	Self–tests	1
10	Design Assurance	1
11	Mitigation of Other Attacks	N/A

2.2 Module Specification

The NonStop Volume Level Encryption is a software module with a multi–chip standalone embodiment. The overall FIPS security level of the module is 1. The following sections will define the physical and logical boundaries of the NSVLE module.

2.2.1 Physical Cryptographic Boundary

As a software cryptographic module, there are no physical protection mechanisms implemented. The module must rely on the physical characteristics of host systems, which were not tested as part of this FIPS 140–2 validation. NSVLE has been tested on the HP Storage CLIM hardware platform, which is a HP ProLiant class server. The physical boundary of the cryptographic module is the HP ProLiant chassis, which encloses the complete set of hardware and software, including the operating system and the module. See Figure 2 below for a block diagram of a ProLiant host system.

[3] TLS – Transport Layer Security
[4] EMI/EMC – Electromagnetic Interference / Electromagnetic Compatibility

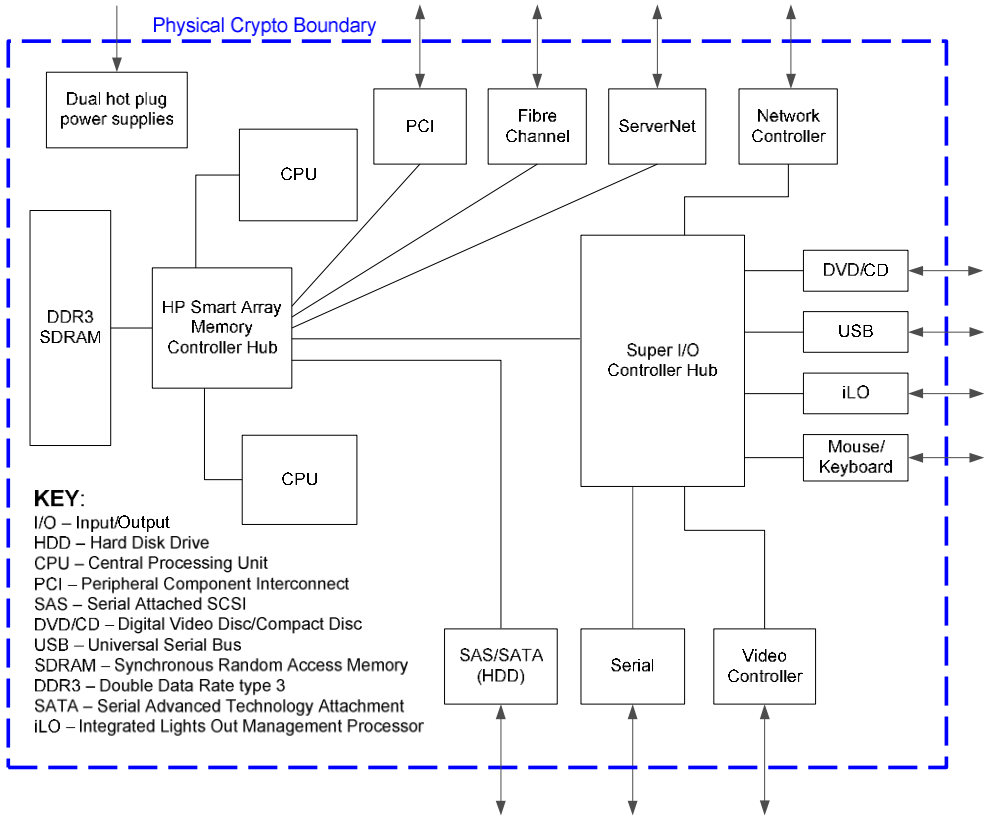

Figure 2 – Block Diagram of an HP ProLiant System

2.2.2 Logical Cryptographic Boundary

NSVLE is a software implementation, as shown in Figure 3, which resides on each CLIM and consists of three components:

- Kryptonmod – Cryptographic engine (loadable kernel module) providing data encryption and decryption of disk data using the XTS[5]-AES[6]and AES-CBC[7] algorithms.
- ESKM client API[8] library (libicapi)– Cryptographic engine providing Enterprise Secure Key Manager interface support to allow key generation, secure key retrieval, and secure communication with the ESKM appliance. An ESKM appliance is a FIPS-validated server that can create, store, and manage millions of encryption keys.
- KM[9]client application – A client application that communicates with the NonStop Servers and external ESKM cluster node. KM client application interfaces with the ESKM client API library to request keys and key attributes from the ESKM appliance via a TLS session.

Figure 4 shows a logical block diagram of the module executing in memory and interacting with CLIM host application software.

[5] XTS – XEX (XOR Encrypt XOR) Tweakable Block Cipher with Ciphertext Stealing
[6] AES – Advanced Encryption Standard
[7] CBC – Cipher Block Chaining
[8] API – Application programming interface
[9] KM – Key Manager

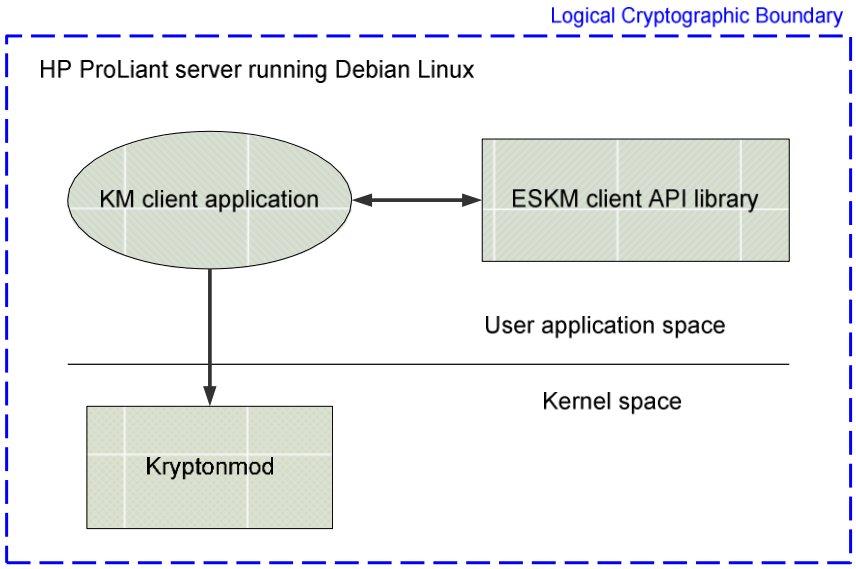

Figure 3 – NSVLE Logical Boundary

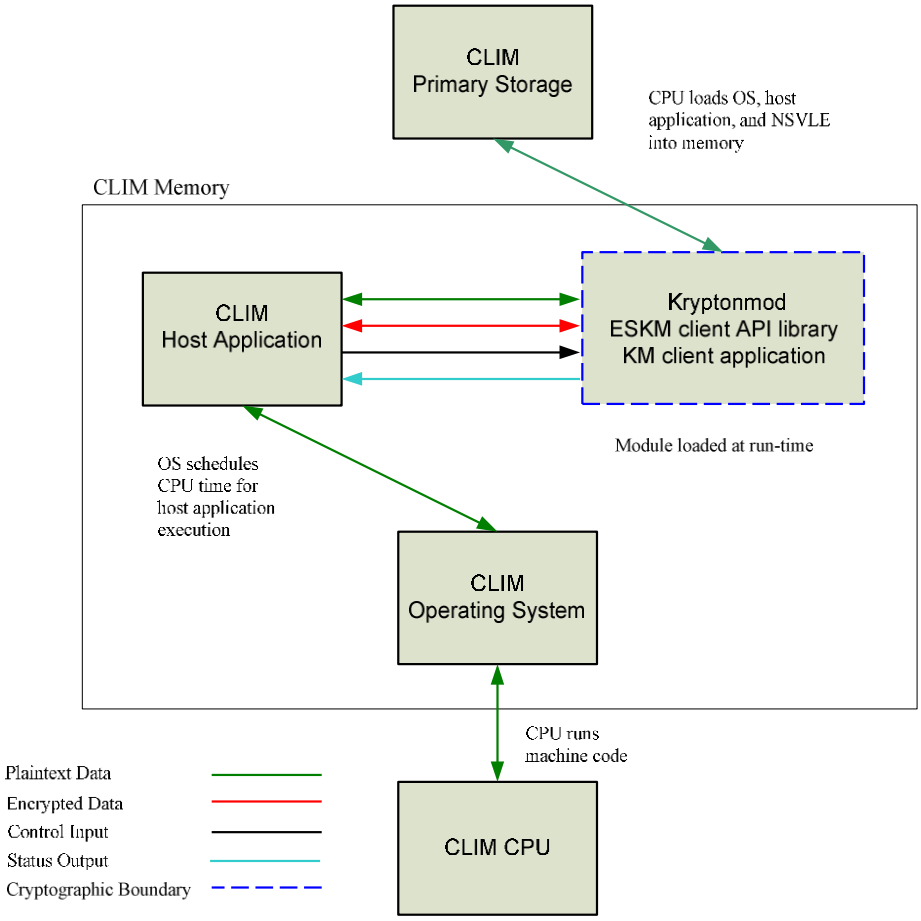

Figure 4 – NSVLE Logical Operating Environment

2.3 Module Interfaces

The module's logical interfaces exist in the software as both an API as well as a Command Line interface (CLI). Physically, ports and interfaces are considered to be those of the host server. The CLI, API, and physical interfaces can be categorized into following interfaces defined by FIPS 140–2:

- Data Input Interface
- Data Output Interface
- Data Control Interface
- Status Output Interface
- Power Interface

A mapping of the FIPS 140–2 logical interfaces, the physical interfaces, and the module can be found in the following table:

Table 2 – FIPS 140–2 Logical Interfaces

FIPS 140–2 Interface	Physical Interface	Logical Interface
Data Input	Mouse/Keyboard, Serial, USB, ServerNet, Fibre Channel, SAS/SATA, PCI, and DVD/CD drive	Arguments for API calls that contain data to be used or processed by the module
Data Output	Monitor, Serial, USB, ServerNet, Fibre Channel, SAS/SATA, PCI, and DVD/CD drive	Arguments for API calls that contain module response data to be used or processed by the caller
Control Input	Mouse/Keyboard, Serial, iLO, and Network	CLI, API Function calls and arguments that initiate and control the operation of the module
Status Output	Monitor, Serial, USB, Network, ServerNet, Fibre Channel, SAS/SATA, and PCI	Return values from API function calls, error messages
Power	Power ports	Not Applicable

2.4 Roles and Services

NSLVE does not perform authentication of any operators. It relies on the authentication mechanisms supported by the operating system on which it runs. The module supports the following roles: Crypto–Officer (CO) and User. Both roles are implicitly assumed when services are executed.

Note 1: The following definitions are used in Table 3 and Table 4 for "CSP[10] and Type of Access".

R – Read: *The plaintext CSP is read by the service.*
W – Write: *The CSP is established, generated, modified, or zeroized by the service.*
X – Execute: *The CSP is used within an Approved or allowed security function or authentication mechanism.*

Note 2: Input and Output parameters of an API call that are not specifically plaintext, ciphertext, or a CSP are NOT itemized in the "Input" and "Output" columns, since it is assumed that most API calls will have such parameters.

[10] CSP – Critical Security Parameter

2.4.1 Crypto Officer Role

The Crypto Officer role has the ability to utilize the module via the KM client application CLI. Descriptions of the services available to the Crypto Officer role are provided in the table below.

Table 3 – Mapping of Crypto Officer Services to Inputs, Outputs, CSPs, and Type of Access

Service	Description	Input	Output	CSP and Type of Access
Create Key	Requests a new key from an ESKM	CLI command parameters	Success or error code	KCBC – W KXTS – W
Key Attribute Services	Set or get key attributes of an existing key	CLI command parameters	Status	No CSP access
Clone Key	Request ESKM to clone a preexisting key	CLI command parameters	Success or error code	No CSP access
Get Key	Retrieves a key from the ESKM server and pass it to Kryptonmod	CLI command parameters	Success or error code, key	KCBC – W KXTS – W

2.4.2 User Role

The User role has the ability to use all services offered by the module's ESKM client API library and Kryptonmod software components. Descriptions of these services are provided in Table 4 below.

Table 4 – Mapping of User Services to Inputs, Outputs, CSPs, and Type of Access

Service	Description	Input	Output	CSP and Type of Access
Load Kryptonmod	Load Kryptonmod during boot	None	Status	No CSP access
Key Container Services	Create, prepare, or delete an entry in Kryptonmod container table	API call parameters	Status	No CSP access
Decrypt	Decrypt data using specified key	API call parameters, key, ciphertext	Plaintext	KCBC – R, X KXTS – R, X
Encrypt	Encrypts data using specified key	API call parameters key, plaintext	Ciphertext	KCBC – R, X KXTS – R, X
Export Generic Key	Exports a "generic" key from the cryptographic module. This key is not allowed to be directly used by the cryptographic module.	API call parameters	Status, key	X
Key Info Services	Return the status, length, or algorithm type of a key	API call parameters	Status	No CSP access

Service	Description	Input	Output	CSP and Type of Access
Get Table	Returns table sans keys or encryption context	None	Status	No CSP access
Get Status	Returns the current module status (version, self–test results, algorithm support)	None	Status	No CSP access
Set Key	Populates Key Container with a key	API call parameters, key	Status	KCBC – W KXTS – W
Self–Test	Perform encrypt/decrypt using test data	API call parameters	Status	No CSP access

2.5 Physical Security

NonStop Volume Level Encryption is a software–only module and does not include physical security mechanisms. Thus, the FIPS 140–2 requirements for physical security are not applicable.

2.6 Operational Environment

The module was tested on an HP ProLiant DL380 Gen8 server with an Intel(R) Xeon(R) E5-2658 processor running the Debian Linux HPTE[12] Version 5.0.0 OS. For FIPS 140–2 compliance, this is considered to be a single user operating system. As such, all keys, intermediate values, and other CSPs remain only in the process space of the operator using the module. The operating system uses native memory management mechanisms to ensure that outside processes cannot access the process space used by the module.

2.7 Cryptographic Key Management

The module implements the following FIPS–Approved algorithms (Note: all algorithm implementations are in the ESKM client API library software component of NSVLE except for those specifically identified as belonging to the Kryptonmod component):

Table 5 – FIPS–Approved Algorithm Implementations

Algorithm	Certificate Number	OBJECT
Symmetric Key Algorithm		
AES CBC[13] 128 and 256-bit	2375	libicapi
AES CBC 256-bit, XTS 256-bit	2376	kryptonmod
Triple-DES 168–bit CBC	1486	libicapi
Secure Hashing Algorithm (SHA)		
SHA-1[14], SHA-256	2047	libicapi
Message Authentication Code (MAC) Function		
HMAC using SHA-1	1477	libicapi
Deterministic Random Bit Generator (DRBG)		

[12] HPTE – Hewlett-Packard Telco Extensions
[13] CBC – Cipher–Block Chaining
[14] Refer to NIST Special Publication 800-131A for guidance on transitions to the use of stronger cryptographic keys and robust algorithms

Algorithm	Certificate Number	OBJECT
SP800-90A, CTR DRBG using AES 256-bit	311	libicapi
Asymmetric Key Algorithm		
RSA[16] (PKCS[15] #1.5) sign/verify: 2048, 3072, 4096-bit	1230	libicapi
TLS Key derivation		
TLS KDF	228	libicapi

The following non–Approved algorithm implementations are supported by the module (note that these algorithms are allowed for use in the FIPS–Approved mode of operation):

a. RSA key wrapping (key establishment methodology provides between 112 and 150 bits of encryption strength)
b. MD5 for use within the TLS Key Derivation Function (KDF)
c. Non-Deterministic Random Number Generator (NDRNG) used to seed the approved DRBG

[15] PKCS – Public–Key Cryptography Standards
[16] Refer to NIST Special Publication 800-131A for guidance on transitions to the use of stronger cryptographic keys and robust algorithms

The CSPs supported by the module are shown in the table below.

Note: The "Input" and "Output" columns in Table 6 are in reference to the module's logical boundary.

Table 6 – List of Cryptographic Keys, Cryptographic Key Components, and CSPs

CSP	CSP Type	Use	Generation / Input	Output	Storage	Zeroization
KCBC (AES CBC)	AES 256–bit key	Kryptonmod symmetric encryption and decryption	Enters encrypted	Not output	Plaintext in volatile memory	Delete Key Container service
KXTS (XTS-AES)	AES 256–bit key	Kryptonmod symmetric encryption and decryption	Enters encrypted	Not output	Plaintext in volatile memory	Delete Key Container service
DRBG Seed	DRBG seed material	Used for SP 800-90A CTR_DRBG	Generated internally	Not output	Plaintext in volatile memory	Power cycle the host computer or API call
DRBG V	DRBG "internal state" secret value	Used for SP 800-90A CTR_DRBG	Generated internally	Not output	Plaintext in volatile memory	Power cycle the host computer or API call
DRBG key value	DRBG "internal state" secret value (AES 256-bit key)	Used for SP 800-90A CTR_DRBG	Generated internally	Not output	Plaintext in volatile memory	Power cycle the host computer or API call
TLS_PM (TLS Premaster Secret)	384–bit key material	Derive the master secret during TLS session negotiation	Enters encrypted or generated internally	Exits encrypted	Plaintext in volatile memory	Power cycle the host computer or API call
TLS_M (TLS Master Secret)	384–bit key material	Input into the session key creation process within TLS	Generated internally	Not output	Plaintext in volatile memory	Power cycle the host computer or API
AES Key	AES 256–bit key	Symmetric encryption and decryption (e.g. during TLS session negotiation)	Generated internally	Not output	Plaintext in volatile memory	Power cycle the host computer or API call
TDES Key	TDES 168–bit key	Symmetric encryption and decryption(e.g. during TLS session negotiation)	Generated internally	Not output	Plaintext in volatile memory	Power cycle the host computer or API call

CSP	CSP Type	Use	Generation / Input	Output	Storage	Zeroization
HMAC Key	HMAC key	Data authentication (e.g. during TLS session negotiation)	Generated internally	Not output	Plaintext in volatile memory	Power cycle the host computer or API call
Software Integrity Key	HMAC key	Authenticate NSVLE during power–on self–test	Not input or generated	Not output	Plaintext in volatile memory	Not applicable per FIPS 140–2 Implementation Guidance Section 7.4
RSA Private Key	2048, 3072, or 4096 bit RSA private key	Used to authenticate or provide confidentiality to data (e.g. during TLS session negotiation)	Enters in plaintext	Not output	Plaintext in volatile memory	Power cycle the host computer or API call

The public keys supported by the module are shown in the table below:

Table 7 – Cryptographic Module Public Keys

Key	Key Type	Use	Generation / Input	Output	Storage	Zeroization
RSA Public Key	2048, 3072, or 4096 bit RSA public key	Used to authenticate or provide confidentiality to data (e.g. during TLS session negotiation)	Enters in plaintext	Not output	Plaintext in volatile memory	Power cycle the host computer or API call

2.8 Self–Tests

The NonStop Volume Level Encryption performs the following self–tests at power–up:

- Software integrity test using HMAC–SHA–1
- Known Answer Tests (KATs)
    - AES–CBC 128 and 256 bit key encrypt/decrypt
    - XTS-AES 256 bit key encrypt/decrypt (kryptonmod)
    - AES-CBC 256 bit key encrypt/decrypt (kryptonmod)
    - Triple–DES CBC 168 bit key encrypt/decrypt
    - SHA–1
    - HMAC SHA–1
    - HMAC SHA-256[1]
    - RSA signature generation/verification
    - SP 800-90A CTR_DRBG

The NonStop Volume Level Encryption performs the following conditional self–tests:

- Continuous DRBG test
- Continuous random number generator test on non-Approved RNG

2.9 Mitigation of Other Attacks

This section is not applicable. The module does not claim to mitigate any attacks beyond the FIPS 140–2 Level 1 requirements for this validation.

[1] The SHA-256 KAT is tested as part of the HMAC SHA-256 KAT.

3 Secure Operation

The NonStop Volume Level Encryption meets Level 1 requirements for FIPS 140–2. The sections below describe how to securely operate the module.

3.1 Initial Setup

The CLIM's Debian Linux operating system is set for single–user mode at the factory before it is delivered to the end–user, who has no ability to make modifications. Therefore, from a FIPS 140–2 perspective, it is considered to be a single–user operating system. The NSVLE software module is installed at the factory as part of the entire CLIM software component. The initial setup for NSVLE involves obtaining and installing a NSVLE license that will enable encryption on the CLIM, and then configuring the KM client application and other NonStop components, such as ESKM.

3.2 Secure Management

The module always operates in FIPS–Approved mode when used as specified within this Security Policy. This module can only operate in FIPS-Approved mode. The software can not be loaded in non FIPS-Approved mode.

3.2.1 Initialization

In order to use NSVLE on a NonStop system with installed HP Storage CLIMs, the following security relevant tasks will need to be completed.

- Install the NSVLE license so that encryption can be enabled. See the *NonStop Volume Level Encryption Guide* for details.
- Create a client certificate for the CLIM and have it signed by the ESKM Certificate Authority. See the *NonStop CLIM Installation and Configuration Guide* for details.
- Install the signed client certificate on the CLIM.
- Configure CLIM/ESKM LAN connection.
- Configure ESKM server settings (e.g. enable TLS with client certificate authentication). See the *HP Enterprise Secure Key Manager Users Guide* for details.
- Configure the storage devices for encryption.
- The Crypto-Officer shall ensure that the ESKM client application module is launched using skm command with the – loadKrypton argument. Failure to do this will result in the module not functioning.
- Verification of proper installation and start-up of the module can be verified by viewing the log file. The status message indicating successful start up and operation is, "Krypton driver successfully loaded". This indicates that the module was started appropriately and has passed all FIPS power-on self-tests.

The library will only use FIPS–approved algorithms for cryptographic purposes when used as defined within this Security Policy.

3.2.2 Assumptions

The module must be managed in accordance with all delivery, operation, and user guidance. To ensure the secure operation of the module, the following assumptions are made:

- Crypto–Officers are non–hostile, appropriately trained, and follow all administrative guidance.
- Crypto–Officers will not modify any boot scripts which come preinstalled on the CLIM platforms.
- The module will be used only as specified within this FIPS 140-2 Security Policy document.

4 Acronyms

This section describes the acronyms.

Table 8 – Acronyms

Acronym	Definition
AES	Advanced Encryption Standard
AMD	Advanced Micro Devices
ANSI	American National Standards Institute
API	Application Programming Interface
CBC	Cipher Block Chaining
CLI	Command Line Interface
CLIM	Cluster I/O Module
CMVP	Cryptographic Module Validation Program
CO	Crypto Officer
CSEC	Communications Security Establishment Canada
CSP	Critical Security Parameter
DRBG	Deterministic Random Bit Generator
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
ESKM	Enterprise Secure Key Manager
FIPS	Federal Information Processing Standard
HMAC	(Keyed–) Hash Message Authentication Code
HP	Hewlett–Packard
HPTE	Hewlett-Packard Telco Extensions
iLO	Integrated Lights Out Management Processor
KAT	Known Answer Test
KDF	Key Derivation Function
KM	Key Manager
LTO	Linear Tape-Open
MAC	Message Authentication Code
NIST	National Institute of Standards and Technology
NSVLE	NonStop Volume Level Encryption
NVLAP	National Voluntary Laboratory Accreditation Program
OS	Operating System
PKCS	Public–Key Cryptography Standards
PRNG	Pseudo Random Number Generator

Acronym	Definition
RNG	Random Number Generator
RSA	Rivest, Shamir, and Adleman
SAS	Serial Attached SCSI
SATA	Serial Advanced Technology Attachment
SCSI	Small Computer System Interface
SHA	Secure Hash Algorithm
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TDES	Triple Data Encryption Standard
XTS	XEX (XOR Encrypt XOR) Tweakable Block Cipher with Ciphertext Stealing