L3Harris Common Data Loader Version 02.01 Security Target

Version 0.4
12/12/24

Prepared for:

L3Harris

2400 Palm Bay Rd NE
Palm Bay, FL 32905

Prepared By:

www.gossamersec.com

LIST OF TABLES

1. Security Target Introduction

This section identifies the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization. The TOE is L3Harris Common Data Loader Version 02.01 provided by L3Harris. The TOE is being evaluated as a full drive encryption solution.

The Security Target contains the following additional sections:

- Conformance Claims (Section 2)
- Security Objectives (Section 3)
- Extended Components Definition (Section 4)
- Security Requirements (Section 5)
- TOE Summary Specification (Section 6)

Conventions

The following conventions have been applied in this document:

- Security Functional Requirements – Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.
  - Iteration: allows a component to be used more than once with varying operations. In the ST, iteration is indicated by a parenthetical number placed at the end of the component. For example, FDP_ACC.1(1) and FDP_ACC.1(2) indicate that the ST includes two iterations of the FDP_ACC.1 requirement.
  - Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., [**assignment**]). Note that an assignment within a selection would be identified in italics and with embedded bold brackets (e.g., [***[selected-assignment]***]).
  - Selection: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., [***selection***]).
  - Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (e.g., "… **all** objects …" or "… ~~some~~ **big** things …").
- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.

1.1 Security Target Reference

ST Title – L3Harris Common Data Loader Version 02.01 Security Target

ST Version – Version 0.4

ST Date – 12/12/24

1.2 TOE Reference

TOE Identification – L3Harris Common Data Loader Version 02.01

TOE Developer – L3Harris

Evaluation Sponsor – L3Harris

1.3 TOE Overview

The Target of Evaluation (TOE) is L3Harris Common Data Loader Version 02.01.

The TOE provides software Full Drive Encryption of removable memory modules (RMM).

1.4 TOE Description

The L3Harris Common Data Loader Version 02.01 (CDL) is a full-disk encryption solution composed of the Common Data Loader Operational Flight Platform (CDL OFP) software and the Ground Stations Software (CDL GSS).

The CDL Operational Flight Platform is an embedded avionics module for onboard non-volatile, network-attached storage. The TOE provides a layer of software encryption on top of embedded removable memory modules (RMM) inserted into the CDL OFP, which themselves are intended to be external self-encrypting drives (separately evaluated by another vendor). The CDL OFP software is designed to operate on a physical hardware device featuring up to three RMMs, two independent ethernet ports, two RS-232 serial interfaces (one for maintenance and one for debugging), and six discrete inputs including one for zeroization.

The TOE also includes the CDL Ground Station Software, a full, stand-alone software suite capable of provisioning and managing the RMMs used with the CDL OFP. The CDL Ground Station Software includes both the Provisioning Tool used to initialize the Data at Rest solution and the Mission Planning Software used to load encrypted data for a given deployment of the RMMs for use in the CDL OFP. The CDL GSS Provisioning Tool and the CDL GSS Mission Planning Software are designed to run on separate machines as they require different OS support.

The TOE in total consists of three distinct software parts. The three parts in aggregate provide the security functionality laid forth in this document and are evaluated as one solution as the combined TOE solution covers all of the required functionality. The CDL GSS is used to initially provision the FDE solution, configuring any necessary keys and authentication factors, while the CDL OFP and CDL GSS Mission Planning Software are both used to load and retrieve encrypted data on to the protected drives, either from the ground or vehicle deployment respectfully.

The TOE is distributed as a software package designed to work on a specific set of hardware. The TOE has the following hardware/software requirements:

- The Operational Flight Platform requires a computer with a Human Computer Interface (HCI) and a NXP P3041 processor. The tested configuration included a L3Harris-bespoke OFP computer running Buildroot 2012.05 on Linux Kernel 3.5.3 on NxP QorlQ P3041 (e500mc).
- The CDL GSS Provisioning Tool requires a native Ubuntu 20.04 LTS computer with eSATA port (or equivalent USB adapter). The tested configuration ran the CDL GSS Provisioning Tool software on a laptop running native Ubuntu 20.04 and Linux Kernel 5.4.0-1091-fips with an Intel Haswell i7-4810MQ.
- The CDL GSS Mission Planning Software requires an Ubuntu 22.04 LTS computer with eSATA port (or equivalent USB adapter). The tested configuration ran the CDL GSS Mission Planning Software on a laptop running Ubuntu 22.04.2 LTS with Linux Kernel 5.15.90.1 through Windows Subsystem for Linux (WSL) 2 on an Intel Haswell i7-4810MQ.

Because the CDL GSS software (both the Provisioning Tool Mission Planning Software) do not rely on any hardware acceleration and the OS provides a hardware abstraction layer, the CDL GSS software executes identically irrespective of the underlying CPU. Additionally, the tested configuration utilized the separately evaluated Novachips Co., Ltd. Scalar and Express P-series SSD, version NV.R1900 RMM units to provide the second layer of hardware full-disk encryption.

The TOE features several network protocols such as Network File System (NFS), Real Time Protocol (RTP) and Real time Streaming Protocol (RTSP) which are not tested as a part of this evaluation. The FDEEEcPP20E and FDEAAcPP20E Protection Profiles did not consider nor include networking protocols as part of the security functional requirements, and thus did not include any such protocol requirements. Therefore, these protocols have not been examined as part of the required assurance activities and consequently the evaluation can make no claims about the TOE's networking protocols.

1.4.1 TOE Architecture

The TOE provides a software Full Drive Encryption solution that can accept up to three Removable Memory Module (RMM), each containing their own data drive.

1.4.1.1 Physical Boundaries

The Ground Station Software's physical boundary is the boundary of the software package distributed by the vendor. Similarly, the CDL OFP physical boundary is the software package distributed by the vendor, however the CDL OFP also utilizes a kernel cryptography library on the Operational Flight Platform that utilizes P3041-based hardware acceleration. Because of this, the TOE also includes the compatible NXP CPU hardware within its physical boundary. In total, the TOE's physical boundary is the boundary of the software packages for both the CDL OFP and GSS with the addition of the NXP processor hardware on the CDL OFP. Together, the components of the TOE provide a full solution to secure Data at Rest (DAR).

1.4.1.2 Logical Boundaries

This section summarizes the security functions provided by the L3Harris Common Data Loader Version 02.01:

- Cryptographic support
- User data protection
- Security management
- Protection of the TSF

1.4.1.2.1 Cryptographic support

The TOE includes cryptographic functionality for key management, user authentication, and block-based encryption including: symmetric key generation, encryption/decryption, cryptographic hashing, keyed-hash message authentication, and password-based key derivation. These functions are supported with suitable random bit generation, key derivation, salt generation, initialization vector generation, secure key storage, and key destruction. These primitive cryptographic functions are used to encrypt Data at Rest (including the generation and protection of keys and key encryption keys) used by the TOE.

1.4.1.2.2 User data protection

The TOE performs Full Drive Encryption on all partitions on the drive (so that no plaintext exists) and does so without user intervention.

1.4.1.2.3 Security management

The TOE provides each of the required management services necessary to manage the full drive encryption using a command line interface.

1.4.1.2.4 Protection of the TSF

The TOE implements a number of features to protect itself to ensure the reliability and integrity of its security features. It protects key and key material, and includes functions to perform self-tests and software/firmware integrity checking so that it might detect when it is failing or may be corrupt. If any of the self-tests fail, the TOE will not go into an operational mode.

1.4.2 TOE Documentation

Administrative Guide COMMON DATA LOADER (CDL) Data-At-Rest (DAR) Operational Flight Platform (OFP) and Ground Support Software (GSS), Initial Version 7 October 2024 [AGD]

2. Conformance Claims

This TOE is conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 5, April 2017.
  - Part 2 Extended
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1, Revision 5, April 2017.
  - Part 3 Conformant
- Package Claims:
  - collaborative Protection Profile for Full Drive Encryption - Authorization Acquisition, Version 2.0 + Errata 20190201, 01 February 2019
  - collaborative Protection Profile for Full Drive Encryption - Encryption Engine, Version 2.0 + Errata 20190201, 01 February 2019 (FDEAAcPP20E/FDEEEcPP20E)

Package	Technical Decision	Applied	Rationale
CPP_FDE_AA_V2.0E / CPP_FDE_EE_V2.0E	TD0769 - FIT Technical Decision for FPT_KYP_EXT.1.1	No	Relevant selections under FPT_KYP_EXT.1.1 not claimed
CPP_FDE_AA_V2.0E	TD0767 - FIT Technical Decision for FMT_SMF.1.1	Yes	SFR and Test AA updated for FMT_SMF.1.1
CPP_FDE_AA_V2.0E / CPP_FDE_EE_V2.0E	TD0766 - FIT Technical Decision for FCS_CKM.4(d) Test Notes	Yes	Test AA updated for FCS_CKM.4(d)
CPP_FDE_AA_V2.0E	TD0765 - FIT Technical Decision for FMT_MOF.1	Yes	Test AA updated for FMT_MOF.1
CPP_FDE_AA_V2.0E	TD0764 - FIT Technical Decision for FCS_PCC_EXT.1	No	TOE does not claim subsequent rounds of AES operations with device key and PBKDF2 output per FCS_COP.1(g) or FCS_COP.1(e)
CPP_FDE_AA_V2.0E	TD0760 - FIT Technical Decision for FCS_SNI_EXT.1.3, FCS_COP.1(f)	Yes	TOE uses IVs, no changes made to SFR, TSS information updated
CPP_FDE_AA_V2.0E	TD0759 - FIT Technical Decision for FCS_AFA_EXT.1.1	No	TOE does not make selection for external Smartcard under FCS_AFA_EXT.1.1
CPP_FDE_AA_V2.0E / CPP_FDE_EE_V2.0E	TD0606 - FIT Technical Recommendation for Evaluating a NAS against the FDE AA and FDEE	Yes	TOE is a NAS
CPP_FDE_EE_V2.0E	TD0464 - FIT Technical Decision for FPT_PWR_EXT.1 compliant power saving states	Yes	FPT_PWR_EXT.1 claimed, SFR updated
CPP_FDE_EE_V2.0E	TD0460 - FIT Technical Decision for FPT_PWR_EXT.1 non-compliant power saving states	Yes	FPT_PWR_EXT.1 claimed, AGD affected
CPP_FDE_AA_V2.0E / CPP_FDE_EE_V2.0E	TD0458 - FIT Technical Decision for FPT_KYP_EXT.1 evaluation activities	Yes	FPT_KYP_EXT.1 claimed, TSS includes required info

2.1 Conformance Rationale

The ST conforms to the FDEAAcPP20E/FDEEEcPP20E. As explained previously, the security problem definition, security objectives, and security requirements have been drawn from the PP.

3. Security Objectives

The Security Problem Definition may be found in the FDEAAcPP20E/FDEEEcPP20E and this section reproduces only the corresponding Security Objectives for operational environment for reader convenience. The FDEAAcPP20E/FDEEEcPP20E offers additional information about the identified security objectives, but that has not been reproduced here and the FDEAAcPP20E/FDEEEcPP20E should be consulted if there is interest in that material.

In general, the FDEAAcPP20E/FDEEEcPP20E has defined Security Objectives appropriate for Full Drive Encryption and as such are applicable to the L3HarrisCommon Data Loader Version 02.01 TOE.

3.1 Security Objectives for the Operational Environment

OE.INITIAL_DRIVE_STATE The OE provides a newly provisioned or initialized storage device free of protected data in areas not targeted for encryption.

OE.PASSPHRASE_STRENGTH An authorized administrator will be responsible for ensuring that the passphrase authorization factor conforms to guidance from the Enterprise using the TOE.

OE.PHYSICAL The Operational Environment will provide a secure physical computing space such than an adversary is not able to make modifications to the environment or to the TOE itself.

OE.PLATFORM_I&A The Operational Environment will provide individual user identification and authentication mechanisms that operate independently of the authorization factors used by the TOE.

OE.PLATFORM_STATE The platform in which the storage device resides (or an external storage device is connected) is free of malware that could interfere with the correct operation of the product.

OE.POWER_DOWN Volatile memory is cleared after power-off so memory remnant attacks are infeasible.

OE.SINGLE_USE_ET External tokens that contain authorization factors will be used for no other purpose than to store the external token authorization factor.

OE.STRONG_ENVIRONMENT_CRYPTO The Operating Environment will provide a cryptographic function capability that is commensurate with the requirements and capabilities of the TOE and Appendix A.

OE.TRAINED_USERS Authorized users will be properly trained and follow all guidance for securing the TOE and authorization factors.

OE.TRUSTED_CHANNEL Communication among and between product components (i.e., AA and EE) is sufficiently protected to prevent information disclosure.

4. Extended Components Definition

All of the extended requirements in this ST have been drawn from the FDEAAcPP20E/FDEEEcPP20E. The FDEAAcPP20E/FDEEEcPP20E defines the following extended requirements and since they are not redefined in this ST the FDEAAcPP20E/FDEEEcPP20E should be consulted for more information in regard to those CC extensions.

Extended SFRs:

- FDEAAcPP20E:FCS_AFA_EXT.1: Authorization Factor Acquisition - per TD0759
- FDEAAcPP20E:FCS_AFA_EXT.2: Timing of Authorization Factor Acquisition
- FDEAAcPP20E:FCS_CKM_EXT.4(a): Cryptographic Key and Key Material Destruction (Destruction Timing)
- FDEEEcPP20E:FCS_CKM_EXT.4(a): Cryptographic Key and Key Material Destruction (Destruction Timing)
- FDEAAcPP20E:FCS_CKM_EXT.4(b): Cryptographic Key and Key Material Destruction (Power Management)
- FDEEEcPP20E:FCS_CKM_EXT.4(b): Cryptographic Key and Key Material Destruction (Power Management)
- FDEEEcPP20E:FCS_CKM_EXT.6: Cryptographic Key Destruction Types
- FDEAAcPP20E:FCS_KDF_EXT.1: Cryptographic Key Derivation
- FDEEEcPP20E:FCS_KDF_EXT.1: Cryptographic Key Derivation
- FDEAAcPP20E:FCS_KYC_EXT.1: Key Chaining (Initiator)
- FDEEEcPP20E:FCS_KYC_EXT.2: Key Chaining (Recipient)
- FDEAAcPP20E:FCS_PCC_EXT.1: Cryptographic Password Construct and Conditioning
- FDEAAcPP20E:FCS_RBG_EXT.1: Extended: Cryptographic Operation (Random Bit Generation)
- FDEEEcPP20E:FCS_RBG_EXT.1: Random Bit Generation
- FDEAAcPP20E:FCS_SNI_EXT.1: Cryptographic Operation (Salt, Nonce, and Initialization Vector Generation) - per TD0760
- FDEEEcPP20E:FCS_SNI_EXT.1: Cryptographic Operation (Salt, Nonce, and Initialization Vector Generation)
- FDEEEcPP20E:FCS_VAL_EXT.1: Validation
- FDEEEcPP20E:FDP_DSK_EXT.1: Protection of Data on Disk
- FDEEEcPP20E:FPT_FUA_EXT.1: Firmware Update Authentication
- FDEAAcPP20E:FPT_KYP_EXT.1: Protection of Key and Key Material - per TD0769
- FDEEEcPP20E:FPT_KYP_EXT.1: Protection of Key and Key Material - per TD0769
- FDEAAcPP20E:FPT_PWR_EXT.1: Power Saving States
- FDEEEcPP20E:FPT_PWR_EXT.1: Power Saving States
- FDEAAcPP20E:FPT_PWR_EXT.2: Timing of Power Saving States
- FDEEEcPP20E:FPT_PWR_EXT.2: Timing of Power Saving States
- FDEAAcPP20E:FPT_TST_EXT.1: TSF Testing
- FDEEEcPP20E:FPT_TST_EXT.1: TSF Testing
- FDEAAcPP20E:FPT_TUD_EXT.1: Trusted Update
- FDEEEcPP20E:FPT_TUD_EXT.1: Trusted Update

5. Security Requirements

This section defines the Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs) that serve to represent the security functional claims for the Target of Evaluation (TOE) and to scope the evaluation effort.

The SFRs have all been drawn from the FDEAAcPP20E/FDEEEcPP20E. The refinements and operations already performed in the FDEAAcPP20E/FDEEEcPP20E are not identified (e.g., highlighted) here, rather the requirements have been copied from the FDEAAcPP20E/FDEEEcPP20E and any residual operations have been completed herein. Of particular note, the FDEAAcPP20E/FDEEEcPP20E made a number of refinements and completed some of the SFR operations defined in the Common Criteria (CC) and that PP should be consulted to identify those changes if necessary.

The SARs are also drawn from the FDEAAcPP20E/FDEEEcPP20E. The FDEAAcPP20E/FDEEEcPP20E should be consulted for the assurance activity definitions.

5.1 TOE Security Functional Requirements

The following table identifies the SFRs that are satisfied by L3Harris Common Data Loader Version 02.01 TOE.

Requirement Class	Requirement Component
FCS: Cryptographic support	FDEAAcPP20E:FCS_AFA_EXT.1: Authorization Factor Acquisition - per TD0759
	FDEAAcPP20E:FCS_AFA_EXT.2: Timing of Authorization Factor Acquisition
	FDEEEcPP20E:FCS_CKM.1(b): Cryptographic Key Generation (Symmetric Keys)
	FDEEEcPP20E:FCS_CKM.1(c): Cryptographic Key Generation (Data Encryption Key)
	FDEAAcPP20E:FCS_CKM.4(a): Cryptographic Key Destruction (Power Management)
	FDEEEcPP20E:FCS_CKM.4(a): Cryptographic Key Destruction (Power Management)
	FDEAAcPP20E:FCS_CKM.4(d): Cryptographic Key Destruction (Software TOE, 3rd Party Storage) - per TD0766
	FDEEEcPP20E:FCS_CKM.4(d): Cryptographic Key Destruction (Software TOE, 3rd Party Storage) - per TD0766
	FDEAAcPP20E:FCS_CKM_EXT.4(a): Cryptographic Key and Key Material Destruction (Destruction Timing)
	FDEEEcPP20E:FCS_CKM_EXT.4(a): Cryptographic Key and Key Material Destruction (Destruction Timing)
	FDEAAcPP20E:FCS_CKM_EXT.4(b): Cryptographic Key and Key Material Destruction (Power Management)
	FDEEEcPP20E:FCS_CKM_EXT.4(b): Cryptographic Key and Key Material Destruction (Power Management)
	FDEEEcPP20E:FCS_CKM_EXT.6: Cryptographic Key Destruction Types
	FDEAAcPP20E:FCS_COP.1(a): Cryptographic Operation (Signature Verification)
	FDEEEcPP20E:FCS_COP.1(a): Cryptographic Operation (Signature Verification)
	FDEAAcPP20E:FCS_COP.1(b): Cryptographic operation (Hash Algorithm)
	FDEEEcPP20E:FCS_COP.1(b): Cryptographic Operation (Hash Algorithm)
	FDEAAcPP20E:FCS_COP.1(c): Cryptographic operation (Keyed Hash Algorithm)
	FDEEEcPP20E:FCS_COP.1(c): Cryptographic Operation (Message Authentication)
	FDEAAcPP20E:FCS_COP.1(f): Cryptographic operation (AES Data Encryption/Decryption) - per TD0760
	FDEEEcPP20E:FCS_COP.1(f): Cryptographic Operation (AES Data Encryption/Decryption)
	FDEAAcPP20E:FCS_COP.1(g): Cryptographic operation (Key Encryption)
	FDEEEcPP20E:FCS_COP.1(g): Cryptographic Operation (Key Encryption)
	FDEAAcPP20E:FCS_KDF_EXT.1: Cryptographic Key Derivation

	FDEEEcPP20E:FCS_KDF_EXT.1: Cryptographic Key Derivation
	FDEAAcPP20E:FCS_KYC_EXT.1: Key Chaining (Initiator)
	FDEEEcPP20E:FCS_KYC_EXT.2: Key Chaining (Recipient)
	FDEAAcPP20E:FCS_PCC_EXT.1: Cryptographic Password Construct and Conditioning
	FDEAAcPP20E:FCS_RBG_EXT.1: Extended: Cryptographic Operation (Random Bit Generation)
	FDEEEcPP20E:FCS_RBG_EXT.1: Random Bit Generation
	FDEAAcPP20E:FCS_SNI_EXT.1: Cryptographic Operation (Salt, Nonce, and Initialization Vector Generation) - per TD0760
	FDEEEcPP20E:FCS_SNI_EXT.1: Cryptographic Operation (Salt, Nonce, and Initialization Vector Generation)
	FDEEEcPP20E:FCS_VAL_EXT.1: Validation
FDP: User data protection	FDEEEcPP20E:FDP_DSK_EXT.1: Protection of Data on Disk
FMT: Security management	FDEAAcPP20E:FMT_MOF.1: Management of Functions Behavior - per TD0765
	FDEAAcPP20E:FMT_SMF.1: Specification of Management Functions
	FDEEEcPP20E:FMT_SMF.1: Specification of Management Functions
	FDEAAcPP20E:FMT_SMR.1: Security Roles
FPT: Protection of the TSF	FDEAAcPP20E:FPT_KYP_EXT.1: Protection of Key and Key Material - per TD0769
	FDEEEcPP20E:FPT_KYP_EXT.1: Protection of Key and Key Material - per TD0769
	FDEAAcPP20E:FPT_PWR_EXT.1: Power Saving States
	FDEEEcPP20E:FPT_PWR_EXT.1: Power Saving States
	FDEAAcPP20E:FPT_PWR_EXT.2: Timing of Power Saving States
	FDEEEcPP20E:FPT_PWR_EXT.2: Timing of Power Saving States
	FDEAAcPP20E:FPT_TST_EXT.1: TSF Testing
	FDEEEcPP20E:FPT_TST_EXT.1: TSF Testing
	FDEAAcPP20E:FPT_TUD_EXT.1: Trusted Update
	FDEEEcPP20E:FPT_TUD_EXT.1: Trusted Update

Table 1 TOE Security Functional Components

5.1.1 Cryptographic support (FCS)

5.1.1.1 Authorization Factor Acquisition - per TD0759 (FDEAAcPP20E:FCS_AFA_EXT.1)

FDEAAcPP20E:FCS_AFA_EXT.1.1

The TSF shall accept the following authorization factors: [***a submask derived from a password authorization factor conditioned as defined in FCS_PCC_EXT.1***]. (TD0759 applied)

5.1.1.2 Timing of Authorization Factor Acquisition (FDEAAcPP20E:FCS_AFA_EXT.2)

FDEAAcPP20E:FCS_AFA_EXT.2.1

The TSF shall reacquire the authorization factor(s) specified in FCS_AFA_EXT.1 upon transition from any Compliant power saving state specified in FPT_PWR_EXT.1 prior to permitting access to plaintext data.

5.1.1.3 Cryptographic Key Generation (Data Encryption Key) (FDEEEcPP20E:FCS_CKM.1(b))

FDEEEcPP20E:FCS_CKM.1(b).1

Refinement: The TSF shall generate symmetric cryptographic keys using a Random Bit Generator as specified in FCS_RBG_EXT.1 and specified cryptographic key sizes [***256 bit***] that meet the following: no standard.

5.1.1.4 Cryptographic Key Generation (Data Encryption Key) (FDEEEcPP20E:FCS_CKM.1(c))

FDEEEcPP20E:FCS_CKM.1(c).1

Refinement: The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation method [***generate a DEK using the RBG as specified in FCS_RBG_EXT.1)]*** and specified cryptographic key sizes [***256 bits***].

5.1.1.5 Cryptographic Key Destruction (Power Management) (FDEAAcPP20E:FCS_CKM.4(a))

FDEAAcPP20E:FCS_CKM.4(a).1

Refinement: The TSF shall [***erase***] cryptographic keys and key material from volatile memory when transitioning to a Compliant power saving state as defined by FPT_PWR_EXT.1 that meets the following: a key destruction method specified in FCS_CKM.4(d).

5.1.1.6 Cryptographic Key Destruction (Power Management) (FDEEEcPP20E:FCS_CKM.4(a))

FDEEEcPP20E:FCS_CKM.4(a).1

The TSF shall [***erase***] cryptographic keys and key material from volatile memory when transitioning to a Compliant power saving state as defined by FPT_PWR_EXT.1 that meets the following: a key destruction method specified in FCS_CKM_EXT.6.

5.1.1.7 Cryptographic Key Destruction (Software TOE, 3rd Party Storage) - per TD0766 (FDEAAcPP20E:FCS_CKM.4(d))

FDEAAcPP20E:FCS_CKM.4(d).1

Refinement: The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [***For volatile memory, the destruction shall be executed by a [removal of power to the memory], For non-volatile storage that consists of the invocation of an interface provided by the underlying platform that [logically addresses the storage location of the key and performs a [single] overwrite consisting of [a pseudo-random pattern using the TSF's RBG]]***] that meets the following: no standard.

5.1.1.8 Cryptographic Key Destruction (Software TOE, 3rd Party Storage) - per TD0766 (FDEEEcPP20E:FCS_CKM.4(d))

FDEEEcPP20E:FCS_CKM.4(d).1

Refinement: The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [***For volatile memory, the destruction shall be executed by a [removal of power to the memory], For non-volatile storage that consists of the invocation of an interface provided by the underlying platform that [instructs the underlying platform to destroy the abstraction that represents the key]***]

that meets the following: no standard.

5.1.1.9 Cryptographic Key and Key Material Destruction (Destruction Timing) (FDEAAcPP20E:FCS_CKM_EXT.4(a))

FDEAAcPP20E:FCS_CKM_EXT.4(a).1

The TSF shall destroy all keys and key material when no longer needed.

5.1.1.10 Cryptographic Key and Key Material Destruction (Destruction Timing) (FDEEEcPP20E:FCS_CKM_EXT.4(a))

FDEEEcPP20E:FCS_CKM_EXT.4(a).1

The TSF shall destroy all keys and keying material when no longer needed.

5.1.1.11 Cryptographic Key and Key Material Destruction (Power Management) (FDEAAcPP20E:FCS_CKM_EXT.4(b))

FDEAAcPP20E:FCS_CKM_EXT.4(b).1

Refinement: The TSF shall destroy all key material, BEV, and authentication factors stored in plaintext when transitioning to a Compliant power saving state as defined by FPT_PWR_EXT.1.

5.1.1.12 Cryptographic Key and Key Material Destruction (Power Management) (FDEEEcPP20E:FCS_CKM_EXT.4(b))

FDEEEcPP20E:FCS_CKM_EXT.4(b).1

The TSF shall destroy all key material, BEV, and authentication factors stored in plaintext when transitioning to a Compliant power saving state as defined by FPT_PWR_EXT.1.

5.1.1.13 Cryptographic Key Destruction Types (FDEEEcPP20E:FCS_CKM_EXT.6)

FDEEEcPP20E:FCS_CKM_EXT.6.1

The TSF shall use [***FCS_CKM.4(d)***] key destruction methods.

5.1.1.14 Cryptographic Operation (Signature Verification) (FDEAAcPP20E:FCS_COP.1(a))

FDEAAcPP20E:FCS_COP.1(a).1

Refinement: The TSF shall perform cryptographic signature services (verification) in accordance with a [***Elliptic Curve Digital Signature Algorithm with a key size of 256 bits or greater***]
that meet the following:
[***FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Section 6 and Appendix D, Implementing 'NIST curves' [P-384]; ISO/IEC 14888-3, Section 6.4, for ECDSA schemes***].

5.1.1.15 Cryptographic Operation (Signature Verification) (FDEEEcPP20E:FCS_COP.1(a))

FDEEEcPP20E:FCS_COP.1(a).1

The TSF shall perform cryptographic signature services (verification) in accordance with a [***Elliptic Curve Digital Signature Algorithm with a key size of 256 bits or greater***] that meets the following: [***FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Section 6 and Appendix D, Implementing 'NIST curves' [P-384]; ISO/IEC 14888-3, Section 6.4, for ECDSA schemes***].

5.1.1.16 Cryptographic operation (Hash Algorithm) (FDEAAcPP20E:FCS_COP.1(b))

FDEAAcPP20E:FCS_COP.1(b).1

Refinement: The TSF shall perform cryptographic hashing services in accordance with a specified cryptographic algorithm [***SHA-256, SHA-384***] that meet the following: ISO/IEC 10118-3:2004.

5.1.1.17 Cryptographic Operation (Hash Algorithm) (FDEEEcPP20E:FCS_COP.1(b))

FDEEEcPP20E:FCS_COP.1(b).1

The TSF shall perform cryptographic hashing services in accordance with a specified cryptographic algorithm [***SHA-256, SHA-384***] that meet the following: ISO/IEC 10118-3:2004.

5.1.1.18 Cryptographic operation (Keyed Hash Algorithm) (FDEAAcPP20E:FCS_COP.1(c))

FDEAAcPP20E:FCS_COP.1(c).1

Refinement: The TSF shall perform cryptographic [keyed-hash message authentication] in accordance with a specified cryptographic algorithm [***HMAC-SHA-384***] and cryptographic key sizes [**384**] that meet the following: ISO/IEC 9797-2:2011, Section 7 'MAC Algorithm 2'.

5.1.1.19 Cryptographic Operation (Message Authentication) (FDEEEcPP20E:FCS_COP.1(c))

FDEEEcPP20E:FCS_COP.1(c).1

Refinement: The TSF shall perform message authentication in accordance with a specified cryptographic algorithm [***HMAC-SHA-384***] and cryptographic key sizes [**384 bits used in [*HMAC]***] that meet the following: [***ISO/IEC 9797-2:2011, Section 7 'MAC Algorithm 2'***].

5.1.1.20 Cryptographic operation (AES Data Encryption/Decryption) - per TD0760 (FDEAAcPP20E:FCS_COP.1(f))

FDEAAcPP20E:FCS_COP.1(f).1

The TSF shall perform data encryption and decryption in accordance with a specified cryptographic algorithm AES used in [***CBC***] mode and cryptographic key sizes [***256 bits***] that meet the following: AES as specified in ISO /IEC 18033-3, [***CBC as specified in ISO/IEC 10116***].

5.1.1.21 Cryptographic Operation (AES Data Encryption/Decryption) (FDEEEcPP20E:FCS_COP.1(f))

FDEEEcPP20E:FCS_COP.1(f).1

Refinement: The TSF shall perform data encryption and decryption in accordance with a specified cryptographic algorithm AES used in [***CBC***] mode and cryptographic key sizes [***256 bits***] that meet the following: AES as specified in ISO/IEC 18033-3, [***CBC as specified in ISO/IEC 10116,***].

5.1.1.22 Cryptographic operation (Key Encryption) (FDEAAcPP20E:FCS_COP.1(g))

FDEAAcPP20E:FCS_COP.1(g).1

Refinement: The TSF shall perform key encryption and decryption in accordance with a specified cryptographic algorithm AES used in [***CBC***] mode and cryptographic key sizes [***256 bits***] that meet the following: AES as specified in ISO /IEC 18033-3, [***CBC as specified in ISO/IEC 10116,***].

5.1.1.23 Cryptographic Operation (Key Encryption) (FDEEEcPP20E:FCS_COP.1(g))

FDEEEcPP20E:FCS_COP.1(g).1

Refinement: The TSF shall perform key encryption and decryption in accordance with a specified cryptographic algorithm AES used in [***CBC***] mode and cryptographic key sizes [***256 bits***] that meet the following: AES as specified in ISO/IEC 18033-3, [***CBC as specified in ISO/IEC 10116***].

5.1.1.24 Cryptographic Key Derivation (FDEAAcPP20E:FCS_KDF_EXT.1)

FDEAAcPP20E:FCS_KDF_EXT.1.1

The TSF shall accept [***a conditioned password submask***] to derive an intermediate key, as defined in [***NIST SP 800-132***] , using the keyed-hash functions specified in FCS_COP.1(c), such that the output is at least of equivalent security strength (in number of bits) to the BEV.

5.1.1.25 Cryptographic Key Derivation (FDEEEcPP20E:FCS_KDF_EXT.1)

FDEEEcPP20E:FCS_KDF_EXT.1.1

The TSF shall accept [***conditioned password submask***] to derive an intermediate key, as defined in [***NIST SP 800-132***], using the keyed-hash functions specified in FCS_COP.1(c), such that the output is at least of equivalent security strength (in number of bits) to the BEV.

5.1.1.26 Key Chaining (Initiator) (FDEAAcPP20E:FCS_KYC_EXT.1)

FDEAAcPP20E:FCS_KYC_EXT.1.1

The TSF shall maintain a key chain of: [***- intermediate keys originating from one or more submask(s) to the BEV using the following method(s): [key derivation as specified in FCS_KDF_EXT.1,]***] while maintaining an effective strength of [***256 bits***] for symmetric keys and an effective strength of [***not applicable***] for asymmetric keys.

FDEAAcPP20E:FCS_KYC_EXT.1.2

The TSF shall provide at least a [***256 bit***] BEV to [**the encryption engine**] [***- without validation taking place***].

5.1.1.27 Key Chaining (Recipient) (FDEEEcPP20E:FCS_KYC_EXT.2)

FDEEEcPP20E:FCS_KYC_EXT.2.1

The TSF shall accept a BEV of at least [***256 bits***] from the AA.

FDEEEcPP20E:FCS_KYC_EXT.2.2

The TSF shall maintain a chain of intermediary keys originating from the BEV to the DEK using the following method(s): [***- key encryption as specified in FCS_COP.1(g)***] while maintaining an effective strength of [***256 bits***] for symmetric keys and an effective strength of [***not applicable***] for asymmetric keys.

5.1.1.28 Cryptographic Password Construct and Conditioning (FDEAAcPP20E:FCS_PCC_EXT.1)

FDEAAcPP20E:FCS_PCC_EXT.1.1

A password used by the TSF to generate a password authorization factor shall enable up to [**128**] characters in the set of upper case characters, lower case characters, numbers, and [**'/', '*', '+', '=', '.', ' '**] and shall perform Password-based Key Derivation Functions in accordance with a specified cryptographic algorithm HMAC-[***SHA-384***], with [***1,001***] iterations, and output cryptographic key sizes [***256 bits***] that meet the following: NIST SP 800-132.

5.1.1.29 Extended: Cryptographic Operation (Random Bit Generation) (FDEAAcPP20E:FCS_RBG_EXT.1)

FDEAAcPP20E:FCS_RBG_EXT.1.1

The TSF shall perform all deterministic random bit generation services in accordance with [***ISO/IEC 18031:2011***] using [***CTR_DRBG (AES)***]].

FDEAAcPP20E:FCS_RBG_EXT.1.2

The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [***[one] software-based noise source(s),***] with a minimum of [***256 bits***] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 'Security Strength Table for Hash Functions', of the keys and hashes that it will generate.

5.1.1.30 Random Bit Generation (FDEEEcPP20E:FCS_RBG_EXT.1)

FDEEEcPP20E:FCS_RBG_EXT.1.1

The TSF shall perform all deterministic random bit generation services in accordance with [***ISO/IEC 18031:2011***] using [***CTR_DRBG (AES)***]].

FDEEEcPP20E:FCS_RBG_EXT.1.2

The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [***[one] software-based noise source(s),***] with a minimum of [***256 bits***] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 'Security Strength Table for Hash Functions', of the keys and hashes that it will generate.

5.1.1.31 Cryptographic Operation (Salt, Nonce, and Initialization Vector Generation) - per TD0760 (FDEAAcPP20E:FCS_SNI_EXT.1)

FDEAAcPP20E:FCS_SNI_EXT.1.1

The TSF shall [***use salts that are generated by a [DRBG as specified in FCS_RBG_EXT.1]***].

FDEAAcPP20E:FCS_SNI_EXT.1.2

The TSF shall use [***no nonces***].

FDEAAcPP20E:FCS_SNI_EXT.1.3

The TSF shall [***create IVs in the following matter [- CBC: IVs shall be non-repeating and unpredictable,]***]. (TD0760 applied)

5.1.1.32 Cryptographic Operation (Salt, Nonce, and Initialization Vector Generation) (FDEEEcPP20E:FCS_SNI_EXT.1)

FDEEEcPP20E:FCS_SNI_EXT.1.1
The TSF shall use [***use salts that are generated by a [DRBG as specified in FCS_RBG_EXT.1]***].

FDEEEcPP20E:FCS_SNI_EXT.1.2
The TSF shall use [***no nonces***].

FDEEEcPP20E:FCS_SNI_EXT.1.3
The TSF shall create IVs in the following manner [***- CBC: IVs shall be non-repeating and unpredictable***].

5.1.1.33 Validation (FDEEEcPP20E:FCS_VAL_EXT.1)

FDEEEcPP20E:FCS_VAL_EXT.1.1
The TSF shall perform validation of the BEV using the following method(s): [***- hash the BEV as specified in [FCS_COP.1(c)] and compare it to a stored hashed value***].

FDEEEcPP20E:FCS_VAL_EXT.1.2
The TSF shall require the validation of the BEV prior to allowing access to TSF data after exiting a Compliant power saving state.

FDEEEcPP20E:FCS_VAL_EXT.1.3
The TSF shall [***perform a key sanitization of the DEK upon a [configurable number] of consecutive failed validation attempts***]

5.1.2 User data protection (FDP)

5.1.2.1 Protection of Data on Disk (FDEEEcPP20E:FDP_DSK_EXT.1)

FDEEEcPP20E:FDP_DSK_EXT.1.1
The TSF shall perform Full Drive Encryption in accordance with FCS_COP.1(f), such that the drive contains no plaintext protected data.

FDEEEcPP20E:FDP_DSK_EXT.1.2
The TSF shall encrypt all protected data without user intervention.

5.1.3 Security management (FMT)

5.1.3.1 Management of Functions Behavior - per TD0765 (FDEAAcPP20E:FMT_MOF.1)

FDEAAcPP20E:FMT_MOF.1.1
The TSF shall restrict the ability to [modify the behaviour of] the functions [use of Compliant power saving state] to [authorized users].

5.1.3.2 Specification of Management Functions (FDEAAcPP20E:FMT_SMF.1)

FDEAAcPP20E:FMT_SMF.1.1
The TSF shall be capable of performing the following management functions: [
a) forwarding requests to change the DEK to the EE,
b) forwarding requests to cryptographically erase the DEK to the EE,
c) allowing authorized users to change authorization values or set of authorization values used within the supported authorization method,
d) initiate TOE firmware/software updates,
e) [***configure the number of failed validation attempts required to trigger corrective behavior***]

5.1.3.3 Specification of Management Functions (FDEEEcPP20E:FMT_SMF.1)

FDEEEcPP20E:FMT_SMF.1.1
The TSF shall be capable of performing the following management functions: a) change the DEK, as

specified in FCS_CKM.1, when reprovisioning or when commanded, b) erase the DEK, as specified in FCS_CKM.4(a), c) initiate TOE firmware/software updates, d) [***configure the number of failed validation attempts required to trigger corrective behavior***].

5.1.3.4 Security Roles (FDEAAcPP20E:FMT_SMR.1)

FDEAAcPP20E:FMT_SMR.1.1
The TSF shall maintain the roles [authorized user].

FDEAAcPP20E:FMT_SMR.1.2
The TSF shall be able to associate users with roles.

5.1.4 Protection of the TSF (FPT)

5.1.4.1 Protection of Key and Key Material - per TD0769 (FDEAAcPP20E:FPT_KYP_EXT.1)

FDEAAcPP20E:FPT_KYP_EXT.1.1
The TSF shall
[***only store keys in non-volatile memory when wrapped, as specified in FCS_COP.1(d), or encrypted, as specified in FCS_COP.1(g) or FCS_COP.1(e)]***].
(TD0769 applied)

5.1.4.2 Protection of Key and Key Material - per TD0769 (FDEEEcPP20E:FPT_KYP_EXT.1)

FDEEEcPP20E:FPT_KYP_EXT.1.1
The TSF shall
[***only store keys in non-volatile memory when wrapped, as specified in FCS_COP.1(d), or encrypted, as specified in FCS_COP.1(g) or FCS_COP.1(e)]***].
(TD0769 applied)

5.1.4.3 Power Saving States (FDEAAcPP20E:FPT_PWR_EXT.1)

FDEAAcPP20E:FPT_PWR_EXT.1.1
The TSF shall define the following Compliant power saving states: [***G3***].

5.1.4.4 Power Saving States (FDEEEcPP20E:FPT_PWR_EXT.1)

FDEEEcPP20E:FPT_PWR_EXT.1.1
The TSF shall define the following Compliant power saving states: [***G3***].
(TD0464 applied)

5.1.4.5 Timing of Power Saving States (FDEAAcPP20E:FPT_PWR_EXT.2)

FDEAAcPP20E:FPT_PWR_EXT.2.1
For each Compliant power saving state defined in FPT_PWR_EXT.1.1, the TSF shall enter the Compliant power saving state when the following conditions occur:
user-initiated request,
[***system shutdown***].

5.1.4.6 Timing of Power Saving States (FDEEEcPP20E:FPT_PWR_EXT.2)

FDEEEcPP20E:FPT_PWR_EXT.2.1
For each Compliant power saving state defined in FPT_PWR_EXT.1.1, the TSF shall enter the Compliant power saving state when the following conditions occur: user-initiated request, [***no other conditions].***

5.1.4.7 TSF Testing (FDEAAcPP20E:FPT_TST_EXT.1)

FDEAAcPP20E:FPT_TST_EXT.1.1
The TSF shall run a suite of the following self-tests [***during initial start-up (on power on)***] to demonstrate the correct operation of the TSF: [**Cryptographic Algorithm Self-tests**].

5.1.4.8 TSF Testing (FDEEEcPP20E:FPT_TST_EXT.1)

FDEEEcPP20E:FPT_TST_EXT.1.1
The TSF shall run a suite of the following self-tests [***during initial start-up (on power on)***] to demonstrate the correct operation of the TSF: [**Cryptographic Algorithm Self-tests**].

5.1.4.9 Trusted Update (FDEAAcPP20E:FPT_TUD_EXT.1)

FDEAAcPP20E:FPT_TUD_EXT.1.1
Refinement: The TSF shall provide authorized users the ability to query the current version of the TOE [***software***].
FDEAAcPP20E:FPT_TUD_EXT.1.2
Refinement: The TSF shall provide authorized users the ability to initiate updates to TOE [***software***].
FDEAAcPP20E:FPT_TUD_EXT.1.3
Refinement: The TSF shall verify updates to the TOE software using a digital signature as specified in FCS_COP.1(a) by the manufacturer prior to installing those updates.

5.1.4.10 Trusted Update (FDEEEcPP20E:FPT_TUD_EXT.1)

FDEEEcPP20E:FPT_TUD_EXT.1.1
Refinement: The TSF shall provide authorized users the ability to query the current version of the TOE [***software***].
FDEEEcPP20E:FPT_TUD_EXT.1.2
Refinement: The TSF shall provide authorized users the ability to initiate updates to TOE [***software***].
FDEEEcPP20E:FPT_TUD_EXT.1.3
Refinement: The TSF shall verify updates to the TOE [***software***] using a [***digital signature as specified in FCS_COP.1(a)***] by the manufacturer prior to installing those updates.

5.2 TOE Security Assurance Requirements

The SARs for the TOE are the components as specified in Part 3 of the Common Criteria. Note that the SARs have effectively been refined with the assurance activities explicitly defined in association with both the SFRs and SARs.

Requirement Class	Requirement Component
ADV: Development	ADV_FSP.1: Basic Functional Specification
AGD: Guidance documents	AGD_OPE.1: Operational User Guidance
	AGD_PRE.1: Preparative Procedures
ALC: Life-cycle support	ALC_CMC.1: Labelling of the TOE
	ALC_CMS.1: TOE CM Coverage
ATE: Tests	ATE_IND.1: Independent Testing - Conformance
AVA: Vulnerability assessment	AVA_VAN.1: Vulnerability Survey

Table 2 Assurance Components

5.2.1 Development (ADV)

5.2.1.1 Basic Functional Specification (ADV_FSP.1)

ADV_FSP.1.1d
The developer shall provide a functional specification.

ADV_FSP.1.2d
The developer shall provide a tracing from the functional specification to the SFRs.

ADV_FSP.1.1c
The functional specification shall describe the purpose and method of use for each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.2c
The functional specification shall identify all parameters associated with each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.3c
The functional specification shall provide rationale for the implicit categorization of interfaces as SFR-non-interfering.

ADV_FSP.1.4c
The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.

ADV_FSP.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_FSP.1.2e
The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

5.2.2 Guidance documents (AGD)

5.2.2.1 Operational User Guidance (AGD_OPE.1)

AGD_OPE.1.1d
The developer shall provide operational user guidance.

AGD_OPE.1.1c
The operational user guidance shall describe, for each user role, the useraccessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.

AGD_OPE.1.2c
The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.

AGD_OPE.1.3c
The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.

AGD_OPE.1.4c
The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_OPE.1.5c
The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences, and implications for maintaining secure operation.

AGD_OPE.1.6c
The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfill the security objectives for the operational environment as described in the ST.

AGD_OPE.1.7c
The operational user guidance shall be clear and reasonable.

AGD_OPE.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.2.2 Preparative Procedures (AGD_PRE.1)

AGD_PRE.1.1d
The developer shall provide the TOE, including its preparative procedures.

AGD_PRE.1.1c
The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.

AGD_PRE.1.2c
The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

AGD_PRE.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AGD_PRE.1.2e
The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

5.2.3 Life-cycle support (ALC)

5.2.3.1 Labelling of the TOE (ALC_CMC.1)

ALC_CMC.1.1d
The developer shall provide the TOE and a reference for the TOE.

ALC_CMC.1.1c
The TOE shall be labelled with its unique reference.

ALC_CMC.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.3.2 TOE CM Coverage (ALC_CMS.1)

ALC_CMS.1.1d
The developer shall provide a configuration list for the TOE.

ALC_CMS.1.1c
The configuration list shall include the following: the TOE itself; and the evaluation evidence required by the SARs.

ALC_CMS.1.2c
The configuration list shall uniquely identify the configuration items.

ALC_CMS.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.4 Tests (ATE)

5.2.4.1 Independent Testing - Conformance (ATE_IND.1)

ATE_IND.1.1d
The developer shall provide the TOE for testing.

ATE_IND.1.1c
The TOE shall be suitable for testing.

ATE_IND.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.1.2e
The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified.

5.2.5 Vulnerability assessment (AVA)

5.2.5.1 Vulnerability Survey (AVA_VAN.1)

AVA_VAN.1.1d
The developer shall provide the TOE for testing.

AVA_VAN.1.1c
The TOE shall be suitable for testing.

AVA_VAN.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VAN.1.2e
The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.1.3e
The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

6. TOE Summary Specification

This chapter describes the security functions:

- Cryptographic support
- User data protection
- Security management
- Protection of the TSF

6.1 Cryptographic support

The Cryptographic support function satisfies the following security functional requirements:

- FDEAAcPP20E:FCS_AFA_EXT.1: The TOE supports password authorization factor, and the password may be between 19 and (up to) 32 characters in length and can be composed of letters, numbers, and the following symbols: '/', '*', '+', '-', '.', and ' '.
- FDEAAcPP20E:FCS_AFA_EXT.2: The TOE components do not have any power-saving states beyond power-off. After transitioning from the power-off state, the user must authenticate before the TOE will allow data to be read from or written to the drive.
- FDEEEcPP20E:FCS_CKM.1(b)/FCS_CKM.1(c): The TOE generates 256-bit DEKs using its CTR_DRBG(AES). Because the DRBG has a security strength of 256 bits, the DEKs generated are sufficient for the TOE's 256-bit AES data encryption/decryption. The TOE stores these keys encrypted in dedicated headers on the drives (in the first few megabytes of an unpartitioned, drives or at the start of each partition). The TOE only generates a new DEK upon provisioning which is handled exclusively by the CDL GSS Provisioning Tool component of the TOE.
- FDEAAcPP20E/FDEEEcPP20E:FCS_CKM.4(a): When the TOE powers off (as the TOE has no other power states other than on and off (G3)), all values in memory drain to a zero state.
- FDEAAcPP20E/FDEEEcPP20E:FCS_CKM.4(d): The TOE is composed of the CDL OFP with 2GB of RAM and the CDL GSS which has 32 GB of RAM. This serves as the working memory in which the TOE temporarily stores working copies of key material (for example, the Derived Key [DerKey], which is derived from the user's password and salt using PBKDFv2 and the DEKs currently in use (if any). The TOE clears keys from memory by a removal of power.

  Additionally, the TOE stores encrypted DEKs in a header for the encrypted drive partitions. The TOE clears these keys through an internal call using the CRYPT_WIPE_RANDOM pattern, which draws random data from the TOE's CTR_DRBG(AES).
- FDEAAcPP20E/FDEEEcPP20E:FCS_CKM_EXT.4(a): The TOE clears the DerKey and DEKs from userspace memory immediately after the operation for which it is needed, while DEKs will be held in kernel memory while the drive is accessible. If the user logs out, then the TOE will clear any in-use DEKs from kernel memory.
- FDEAAcPP20E/FDEEEcPP20E:FCS_CKM_EXT.4(b): The TOE has no Compliant power saving states other than power off (G3).
- FDEEEcPP20E:FCS_CKM_EXT.6: The TOE clears its keys in accordance with FCS_CKM.4(d).
- FDEAAcPP20E/FDEEEcPP20E:FCS_COP.1: The TOE performs cryptographic algorithms in accordance with the following NIST standards and has received the following CAVP algorithm certificates.

The TOE uses its Cryptographic Components in L3Harris Common Data Loader OFP - Kernel 02.01.00 library (userspace) when doing AES-256 CBC ESSIV:SHA-256 data encryption/decryption on the CDL OFP.

SFR	Algorithm	NIST Standard	Cert#
FCS_COP.1(b) (Hash)	SHA-256 Hashing	FIPS 180-4	A6015

SFR	Algorithm	NIST Standard	Cert#
FCS_COP.1(f) (AES)	AES-256 CBC Encrypt/Decrypt	FIPS 197	A6015

Table 3 Cryptographic Components in L3Harris Common Data Loader OFP – Kernel 02.01.00 Cryptographic Algorithms

The TOE uses its Cryptographic Components in L3Harris Mission Planning Software – Kernel 02.01.00 library (userspace) when doing AES-256 CBC ESSIV:SHA-256 data encryption/decryption on the CDL GSS Mission Planning Software.

SFR	Algorithm	NIST Standard	Cert#
FCS_COP.1(b) (Hash)	SHA-256 Hashing	FIPS 180-4	A6017
FCS_COP.1(f) (AES)	AES-256 CBC Encrypt/Decrypt	FIPS 197	A6017

Table 4 Cryptographic Components in L3Harris Mission Planning Software – Kernel 02.01.00 Cryptographic Algorithms

The TOE uses its Cryptographic Components in L3Harris Provisioning Tool – Kernel 02.01.00 library (userspace) when doing AES-256 CBC ESSIV:SHA-256 data encryption/decryption on the CDL GSS Provisioning Tool.

SFR	Algorithm	NIST Standard	Cert#
FCS_COP.1(b) (Hash)	SHA-256 Hashing	FIPS 180-4	A6019
FCS_COP.1(f) (AES)	AES-256 CBC Encrypt/Decrypt	FIPS 197	A6019

Table 5 Cryptographic Components in L3Harris Provisioning Tool – Kernel 02.01.00 Cryptographic Algorithms

The TOE uses its Cryptographic Components in L3Harris Common Data Loader OFP 02.01.00 library when verifying ECDSA P-384 w/ SHA-384 trusted updated signatures as well as performing key management operations on the CDL OFP.

SFR	Algorithm	NIST Standard	Cert#
FCS_COP.1(a) (Verify)	ECDSA P-384 w/ SHA-384 Verify	FIPS 186-4, ECDSA	A6014
FCS_COP.1(b) (Hash)	SHA-256 Hashing SHA-384 Hashing	FIPS 180-4	A6014
FCS_COP.1(c) (Keyed Hash)	HMAC-SHA-384	FIPS 198-1 & 180-4	A6014
FCS_COP.1(f) (AES) FCS_COP.1(g) (AES)	AES-256 CBC Encrypt/Decrypt	FIPS 197	A6014

Table 6 Cryptographic Components in L3Harris Common Data Loader OFP 02.01.00 Cryptographic Algorithms

The TOE uses its Cryptographic Components in L3Harris Mission Planning Software 02.01.00 library when verifying ECDSA P-384 w/ SHA-384 trusted updated signatures as well as performing key management operations on the CDL GSS Mission Planning Software.

SFR	Algorithm	NIST Standard	Cert#
FCS_COP.1(a) (Verify)	ECDSA P-384 w/ SHA-384 Verify	FIPS 186-4, ECDSA	A6016
FCS_COP.1(b) (Hash)	SHA-256 Hashing SHA-384 Hashing	FIPS 180-4	A6016
FCS_COP.1(c) (Keyed Hash)	HMAC-SHA-384	FIPS 198-1 & 180-4	A6016
FCS_COP.1(f) (AES) FCS_COP.1(g) (AES)	AES-256 CBC Encrypt/Decrypt	FIPS 197	A6016

Table 7 Cryptographic Components in L3Harris Mission Planning Software 02.01.00 Cryptographic Algorithms

The TOE uses its Cryptographic Components in L3Harris Provisioning Tool 02.01.00 library when verifying ECDSA P-384 w/ SHA-384 trusted updated signatures as well as performing key generation and management operations on the CDL GSS Provisioning Tool.

SFR	Algorithm	NIST Standard	Cert#
FCS_COP.1(a) (Verify)	ECDSA P-384 w/ SHA-384 Verify	FIPS 186-4, ECDSA	A6018

SFR	Algorithm	NIST Standard	Cert#
FCS_COP.1(b) (Hash)	SHA-256 Hashing SHA-384 Hashing	FIPS 180-4	A6018
FCS_COP.1(c) (Keyed Hash)	HMAC-SHA-384	FIPS 198-1 & 180-4	A6018
FCS_COP.1(f) (AES) FCS_COP.1(g) (AES) FCS_CKM.1(b) FCS_CKM.1(c)	AES-256 CBC Encrypt/Decrypt	FIPS 197	A6018
FCS_RBG_EXT.1 (Random)	AES-256 CTR_DRBG	SP 800-90A	A6018

Table 8 Cryptographic Components in L3Harris Provisioning Tool 02.01.00 Cryptographic Algorithms

- FDEAAcPP20E/FDEEEcPP20E:FCS_COP.1(a): The TOE utilizes ECDSA P-384 w/ SHA-384 signatures to verify the authenticity of software updates. Upon receiving a candidate update and the accompanying signature file, the TOE uses an embedded public key (see FPT_TUD_EXT.1 below for the location) to verify the ECDSA signature against the received image. The verification uses SHA-384 and follows the FIPS 186-4 ECDSA format.
- FDEAAcPP20E/FDEEEcPP20E:FCS_COP.1(b): The TOE's userspace and kernel cryptographic libraries provide SHA-256 as a part of ESSIV:SHA-256 IV generation when unwrapping the keyslot. The TOE's userspace cryptographic libraries additionally provide SHA-384 to be used as a part of PBKDFv2 password-based key derivation as well as for ECDSA signature verification of trusted updates.
- FDEAAcPP20E/FDEEEcPP20E:FCS_COP.1(c): The TOE implements HMAC-SHA-384 using 384-bit keys, the SHA-384 hash algorithm, a 1024-bit block size, and an output MAC length of 384 bits. The TOE uses HMAC-SHA-384 as part of all PBKDF (NIST SP 800-132) key management operations.
- FDEAAcPP20E/FDEEEcPP20E:FCS_COP.1(f): The TOE uses an AES CBC kernel implementation dedicated to drive encryption/decryption. This implementation uses AES- 256 bit keys.
- FDEAAcPP20E/FDEEEcPP20E:FCS_COP.1(g): The TOE uses its userspace cryptographic library's AES CBC implementation for key managements operations (decryption of the encrypted DEKs). This implementation uses AES- 256 bit keys.
- FDEAAcPP20E/FDEEEcPP20E:FCS_KDF_EXT.1: The TOE uses 800-132 (PBKDFv2) with HMAC-SHA-384 and a number of iterations and a 256 bit salt to transform the operator's password into a 256-bit Derived Key for decrypting the encrypted DEKs. The number of iterations is fixed at a value of 1,001 iterations which meets the minimum imposed by this requirement. The TOE CDL OFP also features additional security mechanisms to limit the number of authentication attempts to that it is not solely relying on derivation speed as a method to prevent brute force attacks.
- FDEAAcPP20E/FDEEEcPP20E:FCS_KYC_EXT.1/2: The TOE uses PBKDFv2 to transform the operator's password into a 256-bit BEV, and then uses that BEV to AES decrypt the DEKs stored in the header(s) stored on the drive.
- FDEAAcPP20E:FCS_PCC_EXT.1: The TOE allows passwords up to 128 characters in length and will reject any attempts to use additional characters. The TOE allows uppercase/lowercase letters, numbers, and the following symbols: '/', '*', '+', '=', '.', and ' '. The TOE conditions passwords by combining them with a 256-bit salt using PBKDFv2.
- FDEAAcPP20E/FDEEEcPP20E:FCS_RBG_EXT.1: The TOE includes an AES-256 CTR_DRBG that it seeds with at least 256-bits of entropy from a software-based noise source. Random bit generation is only used for DEK generation which is exclusively performed by the CDL Provisioning Tool.
- FDEAAcPP20E/FDEEEcPP20E:FCS_SNI_EXT.1: The TOE generates its salts using its AES-256 CTR_DRBG. The TOE generates its AES-CBC IVs using ESSIV:SHA256. The TOE does not generate nonces nor tweaks (as the TOE doesn't support AES-XTS).
- FDEEEcPP20E:FCS_VAL_EXT.1: The TOE validates the operator's password by first subjecting the password and salt to PBKDFv2 to form the Derived Key (DerKey). The TOE uses the DerKey to decrypt the masterKey stripes and reconstitutes the masterKey; however, before using the masterKey, the TOE first performs iterative

HMAC-SHA-384 using the operator's password, the masterKey salt, masterKey iterations, and masterKey as inputs, and then compares the resulting value to the stored masterKey's digest stored in the header to ensure the two match.

The CDL GSS Provisioning Tool allows an administrator to configure a maximum number of authentication attempts during the provisioning process. If the TOE detects more than a configured number of incorrect passwords while attempting to unlock the CDL OFP, then the TOE will block all subsequent attempts to validate the operator's password (and not even attempt to validate the password) before performing a key sanitization of the DEK. The TOE stores this counter in the LUKS metadata which persists between reboots.

6.2 User data protection

The User data protection function satisfies the following security functional requirements:

- FDEEEcPP20E:FDP_DSK_EXT.1: The TOE provides FDE that encrypts the entirety of each drive partition through AES-CBC block-based encryption. The Admin Guide describes the TOE's initialization process and setup for the SW-layer. The TOE maintains a separate, unencrypted, internal Flash chip to house its Linux-based firmware that is beyond the FSM drive that the TOE encrypts. The date drive is used as a raw block device, thus the TOE encrypts the entire drive (with a small area reserved for the LUKS header).

6.3 Security management

The Security management function satisfies the following security functional requirements:

- FDEAAcPP20E:FMT_MOF.1: The TOE claims no Compliant power saving states beyond power off. Only the authorized administrator can issue the shutdown command. Note the only accounts on the TOE are authorized administrators.
- FDEAAcPP20E/FDEEEcPP20E:FMT_SMF.1: The TOE provides each of the required management services with the additional management function to configure the number of failed validation attempts required to trigger corrective behavior. Upon provisioning the TOE Data at Rest solution, the TOE Provisioning Tool will prompt for a configurable number for the maximum number of software unlock attempts between 1 and 15. Upon reaching this authentication limit after provisioning, the CDL OFP will trigger a sanitization of the DEK. Because the TOE fulfills the AA and EE requirements together, the TOE need not "forward" requests to change the DEK or cryptographically erase the DEK. The TOE supports commands for erasing the DEK, creating a new partition, and changing of the authorization factors through administrator commands. Changing the DEK and changing authorization factors is only possible when provisioning the TOE which is performed only by the TOE Provisioning Tool. The Admin Guide describes updating the TOE's software through the vendor-provided MLV (Memory Load and Verify) Windows Tool for updating the CDL OFP. Similarly, the Admin Guide contains detailed descriptions for how to update the CDL GSS's software through the platform's software repository. The TOE does not provide any manageable power-saving states.
- FDEAAcPP20E:FMT_SMR.1 – The TOE maintains an administer role that can administer the TOE.

6.4 Protection of the TSF

The Protection of the TSF function satisfies the following security functional requirements:

- FDEAAcPP20E/FDEEEcPP20E:FPT_KYP_EXT.1: The TOE stores encrypted DEKs in the header of each drive partition.
- FDEAAcPP20E/FDEEEcPP20E:FPT_PWR_EXT.1/2: The TOE provides the compliant power-saving state G3, mechanical off. The TOE enters this state when the user shuts off the device. The TOE must be fully rebooted from this state.
- FDEAAcPP20E/FDEEEcPP20E:FPT_TST_EXT.1: The TOE includes the following power-up Known Answer Tests (KATs) to ensure that each of its cryptographic algorithms operates correctly.

- kernel/hardware accelerator – SHA-384 hashing test
- kernel/hardware accelerator – AES-256 CBC encrypt/decrypt test
- kernel/hardware accelerator – integrity test
- userspace – SHA hashing tests
- userspace – HMAC-SHA tests
- userspace – AES-256 CBC encrypt/decrypt test
- userspace – AES-256 CTR_DRBG test
- userspace – ECDSA sign/verify test
- userspace – integrity test

- FDEAAcPP20E/FDEEEcPP20E:FPT_TUD_EXT.1: The TOE can display its current firmware version and has the ability to update its software using signed updates. The TOE will verify the signature on a firmware upgrade (using its userspace library in conjunction with the embedded public key to verify the ECDSA P-384 with SHA-384 signature) before installing it, and will reject any update with an invalid signature.

7. Key Management Description

The key management description explains each key, crypto module and overall encryption architecture. Each key is identified in the table below.

Key Identifier	Storage Location	How Key is Protected	How key is Derived	Strength of Key	When Key is Destroyed
User Passphrase	Memory - transient	N/A	N/A	N/A	Immediately after use
Derived Key	Memory - transient	N/A	The TOE uses 800-132 KDF in counter mode using HMAC-SHA-384 and a number of iterations and a 256-bit salt to transform the operator's password into a Derived Key	256 bits	Immediately after use
DEK (Master Key)	Memory - transient	N/A	Generated from approved DRBG	256 bits	Immediately after use
DEK (Master Key)	Partition Header - persistent	AES CBC Encrypted	Generated from approved DRBG	256 bits	When partition closed or when partition no longer encrypted

Table 9 Key Identification

Keys are destroyed through zeroization or overwrite when no longer needed. The data encryption engine is based on LUKS, and is composed of both a userspace component and a kernel-level component. The userspace component handles derivation of the Derived Key from the user's password and the subsequent decryption of the Master Key (DEK) with the Derived Key. The kernel-level component receives the DEK from the userspace component and then encrypts/decrypts data written to/read from the encrypted partition/drive. The TOE uses PBKDFv2 to transform the operator's password into a 256-bit BEV, and then uses that BEV to AES decrypt a Master Key (DEK) stored in the header stored on the drive. The data encryption software runs atop the Linux software (running on the vendor's Network Attached Storage (NAS) device hardware), where the software executes on the main processor, with the software stored in a dedicated OS/system drive. While the TOE does not encrypt its internal Flash Storage memory, it provides no access to it, and only exposes the encrypted Removable Memory Module (drive) to network-attached NAS clients and to applications running on the TOE. The TOE ensures that access to the RMM/drive is always encrypted, and does not permit plaintext access to protected partitions or drive. TOE only provides access to the RMM/drive once fully initialized and after receiving the administrator's password.

The three L3Harris Common Data Loader Version 02.01 software components rely on cryptography from two main cryptographic modules:

1. Userspace – used for all LUKS key management (but not encrypting/decrypting drive data) and trusted update verification
2. Kernel – used for encrypting data on the partition

Although the TOE components only each rely on the above two modules for all cryptographic functions, the TOE utilizes multiple versions of the cryptographic modules between the various components of the TOE. See Section 6.1 Cryptographic Support for more details on which version is used for each individual TOE component and information on the algorithm certificates.