Ref: AMAHSM-ST-1.0

SECIRON – Android Mobile Application Hardening Sandbox Module (AMAHSM) version: 7.0 Security Target

DOCUMENT VERSION	1.0
DOCUMENT DATE	28 DECEMBER 2023

SECIRON

Unit 704, Uptown One, No. 1, Jalan SS21/58,
Damansara Uptown, 47400
Petaling Jaya, Selangor
Email: business@msafe.co.jp
Tel: +86 18401318060
Website: https://www.seciron.com

Prepared by:

ACROSS VERTICALS

CYBERTRONICS LAB
SECURING 4IR

DOCUMENT REVISION HISTORY

Version No.	Published Date	Description of changes	Author
0.1	13 April 2023	First release	Reyes Foong
0.2	8 August 2023	CAR-C129 - <AMAHSM> Amendments	Reyes Foong
0.3	17 October 2023	CAR-C129 - <AMAHSM> Amendments	Reyes Foong
0.4	1 December 2023	Company name change	Reyes Foong
1.0	28 December 2023	Final Version Upgrade	Reyes Foong

TABLE OF CONTENTS

1 Security Target Introduction

1.1 Security Target Reference

Security Target Title:	SECIRON – Android Mobile Application Hardening Sandbox Module (AMAHSM) version: 7.0 Security Target
Security Target Version:	1.0
Security Target Date:	28 December 2023

Table 1 - ST Reference

1.2 TOE Reference

TOE Name & Version	**TOE NAME:**	**TOE VERSION:**
	SECIRON – ANDROID MOBILE APPLICATION HARDENING SANDBOX MODULE (AMAHSM)	7.0
TOE Initial:	AMAHSM	

Table 2 - TOE Reference

1.3 Terminology and Acronyms

Acronyms	Full Name
CC	Common Criteria
EAL	Evaluation Assurance Level
OSP	Organizational Security Policy
PP	Protection Profile
SAR	Security Assurance Requirements
SFR	Security Functional Requirements
ST	Security Target

TOE	Target of Evaluation
TSF	TOE Security Functionality
TSS	TOE Summary Specification
SaaS	Software as a Service
SO Library File	Shared Object Library File
XML	Extensible Markup Language
DEX	Dalvik executable
APK	Android Package Kit

1.4 Product Overview

IronWALL is a cutting-edge security solution that helps manage, prevent and protect mobile applications from security risks. IronWALL is designed to safeguard against a wide range of threats, including mobile application tampering, reverse engineering, debugging, jailbreaks, application cloning, malware, repackaging and other attacks on untrusted environment.

Furthermore, IronWALL also effectively mitigates potential risks by reducing attack surface exposure. Android Mobile Application Hardening Sandbox Module is a product designed by SECIRON and developed as part of IronWALL.

This product integrates protection technologies for various security flaws into the application client without changing the application code, providing customers with a full lifecycle management covering application development, packaging, distribution, and operation. The integrated security guarantee service effectively prevents malicious attacks against mobile applications such as de-compilation, repackaging, memory injection, dynamic debugging, data theft, transaction hijacking, and application phishing, and comprehensively protects application software security.

The hardening core technology includes:

1. Code Anti-Reverse
2. Application Tamper Protection
3. Memory Anti-Debug Protection
4. Data Leakage Protection
5. Operating Environmental Protection

1.5 TOE Overview

The TOE Overview summarizes the usage and major security features of the TOE. The TOE Overview provides a context for the TOE evaluation by identifying the TOE type, describing the product, and defining the specific evaluated configuration.

1.5.1 Usage and Major Security Feature of the TOE

The Android Mobile Application Hardening Sandbox Module (AMAHSM) is a product hosted on the cloud and packaged as SaaS service that provides mobile application security hardening. Users can upload their APK files to be hardened, select the desired hardening policy and download the hardened APK file with hash verification.

The major security features of the TOE included in the evaluation is:

- Cryptography Support
- Protection of the TSF
- Security Audit

1.5.2 TOE Type

AMAHSM is a mobile application hardening tools that allow users to protect their android mobile application (APK file).

1.5.3 Non-TOE hardware/firmware/software required by the TOE

The following figure shows the typical operational environment of the TOE.

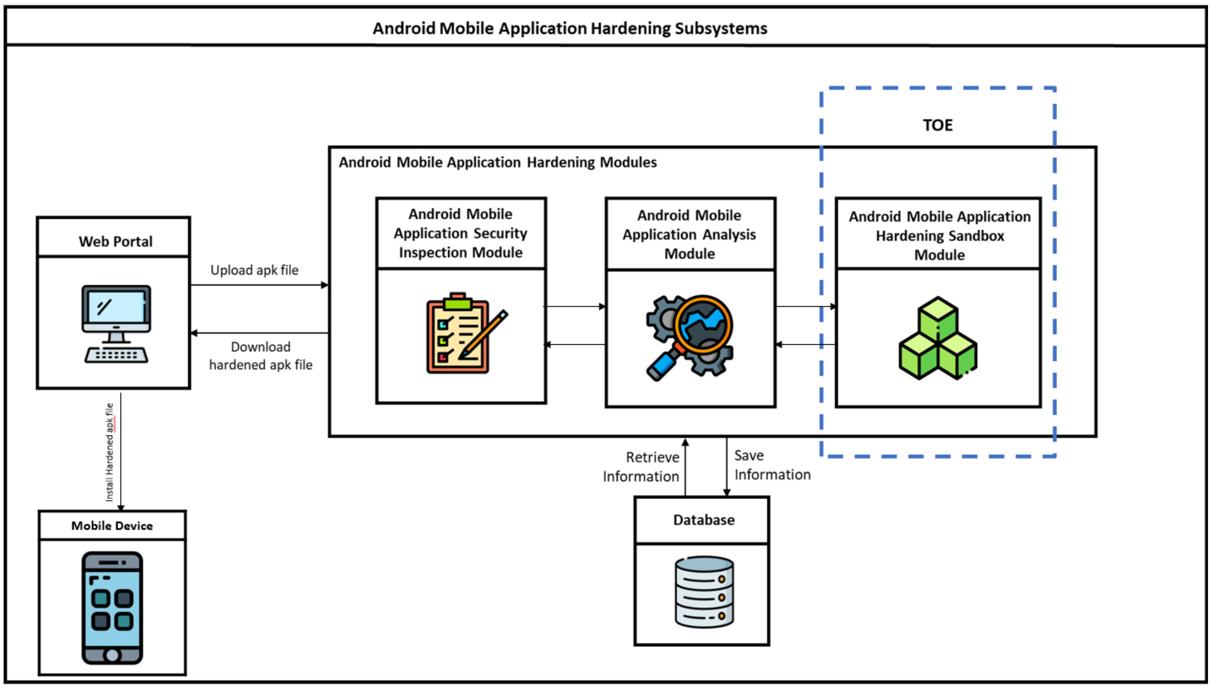
Android Mobile Application Hardening Subsystems
TOE
Android Mobile Application Hardening Modules
Web Portal
Upload apk file
Download hardened apk file
Android Mobile Application Security Inspection Module
Android Mobile Application Analysis Module
Android Mobile Application Hardening Sandbox Module
Install Hardened apk File
Mobile Device
Retrieve Information
Save Information
Database

Figure 1 SECIRON Android Mobile Application Hardening Subsystem High-Level Diagram

Mobile Application package will be uploaded by the User through Web Portal which allow User to create and define mobile application hardening rules that understand by the TOE. Once mobile application package is hardened, mobile application package will be installed in mobile device for testing to ensure the hardening in place.

The supporting components for the TOE are as follow:

a) **Web Portal**

Web portal allow user to setup mobile application hardening policy and upload mobile application package for hardening purpose.

Compatible Browser: -

- Compatible with IE8 browser and later.
- Compatible with Google Chrome browser.
- Compatible with Firefox browser.
- Compatible with Safari browser.
- Compatible with Edge browser.

Equipment	Components
Web Browser	Version 119.0.6045.160 (Official Build) (64-bit)

b) **Android Mobile Application Security Inspection Module**

Mobile Application Package Scanner to ensure uploaded mobile application package meets security requirements and free from malware or malicious codes.

c) **Android Mobile Application Analysis Module**

Mobile Application package analysis before hardening process to identify for mobile application package related information such as UI/UX and Framework Information.

d) **Database**

Storage for Application Files, System Configuration, Management Information and Security Policies. User activity and Hardening Activity will be logged by the system and stored it in database for troubleshooting and security audit purpose. Created hardening Policy will be stored in the database as well.

e) **Mobile Device**

Mobile device to perform testing on hardened mobile application package to ensure hardening policy is in place.

Equipment	Components
Mobile Device #1	Google Pixel 5
Mobile Device #2	Asus ROG Phone 2

1.6 TOE Description

This section primarily addresses the physical and logical components of the TOE included in the evaluation.

1.6.1 Physical Scope of the TOE

There is no physical scope of the TOE as the TOE is hosted on the cloud as a SaaS application.

1.6.2 Logical Scope of the TOE

The logical scope of TOE is described based on the following security functional requirement.

1.6.2.1 Cryptography Support

The TOE generates cryptographic keys (decryption keys) that are to be stored within the encrypted SO Library Files. The TOE performs several cryptographic operations including code encryptions, SO Library Files encryption, hash generation and RSA signature generation. These cryptographic operations are performed in accordance to strong encryption algorithm with adequate key length.

1.6.2.2 Protection of TSF

During the hardening process, the data from Classes.DEX, AndroidManifest.XML, and SO Library Files in the APK file are extracted separately to be hardened and modified accordingly. A hash is generated to ensure the integrity of the file throughout the process.

The hardening process will verify integrity of files to ensure there were no unauthorized tampering during runtime. Should the verification failed, the execution process will be terminated.

1.6.2.3 Security Audit

The TOE generates logs from the web portal operations. The logs shall include events from the web portal such as user login, hardening submission and hardening policy addition or modification. These logs are stored in the server and not on the user's local device to prevent unauthorized modifications.

2 Conformance Claims

The following conformance claims are made for the TOE and ST:

CCv3.1 conformant	The ST and the TOE are Common Criteria conformant to Common Criteria version 3.1 Revision 5.
Part 2 conformant	The ST is Common Criteria Part 2 conformant.
Part 3 conformant	The ST is Common Criteria Part 3 conformant.
Package conformant	EAL 2.
Protection Profile conformance	None.

3 TOE Security Problem Definition

3.1 Assumption

The assumptions are to ensure the security of the TOE and its deployed environment.

A.USER	The users are trusted; the users shall not maliciously compromise the security functionality of the TOE. The users are well-trained; the user shall comply to the operating procedures stipulated in the user guidance.
A.DATAFLOW	The data flow to the TOE must be between the subsystems and TOE as defined in the use case.

Table 3: Assumptions

3.2 Threats

This section describes the threats that are addressed by the TOE:

T.TAMPER	An unauthorized person may obtain access to the TOE during runtime and make unauthorized changes to the data.
T.DATA	Data may not be protected adequately during transmission within TOE.
T.LOGS	An unauthorized person may intentionally or unintentionally delete audit records to destroy evidence of adverse events executed.

Table 4: Threats

3.3 Organizational Security Policies

There are no Organizational Security Policies that the TOE must comply to.

4 Security Objectives

Security objectives are formed to address the security problem definition defined in earlier section. The security implementation in TOE and its environment will meet these objectives.

4.1 Security Objectives for the TOE

The security objectives for the TOE as following:

O.DATA	TOE shall ensure data are secured in transmission to and from TOE.
O.TAMPER	TOE shall ensure that data are protected from unauthorized tampering.
O.LOGS	TOE shall ensure the logs are protected from unauthorized access to prevent unauthorized modifications to the generated logs.

Table 5: Security Objectives for the TOE

4.2 Security Objectives for the Operational Environment

The security objectives for the TOE operational environment as following:

OE.USER	The users are trusted; the users shall not maliciously compromise the security functionality of the TOE. The users are well trained; the user shall comply with the operating procedures stipulated in the user guidance.

Table 6: Security Objectives for the Operational Environment

4.2.1 Security Objectives Rationale

Table 7 maps security objectives to threats and assumptions described in Section 4. The table illustrates that each threat is countered by at least one security objective, that each assumption is upheld by at least one security objective, and that each objective counters at least one threat or upholds at least one assumption.

Threats and Assumptions Security Objectives	T.TAMPER	T.DATA	T.LOGS	A.USER	A.DATAFLOW
O.TAMPER	✓				
O.DATA		✓			✓
O.LOGS			✓		
OE.USER				✓	

Table 7 - Security Objectives Rationale Mapping

5 Extended Components

This section defines the extended Security Functional Requirements (SFRs) and extended Security Assurance Requirements (SARs) applicable for the TOE.

5.1 Extended Security Functional Requirement (SFR)

There are no extended SFR components defined for this evaluation.

5.2 Extended Security Assurance Requirement (SAR)

There are no extended SAR components defined for this evaluation.

6 TOE Security Requirements

This section provides the security functional and assurance requirements that must be satisfied by a compliant TOE. These requirements consist of functional components from Part 2 of the CC, extended requirements, and an Evaluation Assurance Level (EAL) that contains assurance components from Part 3 of the CC.

6.1 Conventions

Part 2 of the Common Criteria defines an approved set of operations that may be applied to the statement of security functional requirements. Following are the operations and the document conventions as used within this ST to depict their application:

Assignment The assignment operation provides the ability to specify an identified parameter within a requirement. Assignments are depicted using bolded text and are surrounded by square brackets as follows [**assignment**].

Selection The selection operation allows the specification of one or more items from a list. Selections are depicted using bold italics text and are surrounded by square brackets as follows [***selection***].

Refinement The refinement operation allows the addition of extra detail to a requirement. Refinements are indicated using bolded text, for **additions**, and strike-through, for ~~deletions~~.

Iteration The iteration operation allows a component to be used more than once with varying operations. Iterations are depicted by placing an acronym at the end of the component identifier as follows: FCS_COP.1 (SWP).

6.2 Security Functional Requirements (SFR)

This section contains the security functional requirements (SFRs) for the TOE. The summary of SFRs is listed in following table.

Component	Component Name
Class FCS: Cryptographic support	
FCS_CKM.1	Cryptographic key generation
FCS_COP.1	Cryptographic operation
Class FPT: Protection of the TSF	
FPT_ITT.1	Basic internal TSF data transfer protection
FPT_ITT.3	TSF data integrity monitoring
Class FAU: Security Audit	
FAU_GEN.1	Audit data generation

Table 8: Security Functional Requirements List

6.2.1 Class FCS: Cryptographic support

FCS_CKM.1 Cryptographic key generation

Hierarchical No other components.

Dependencies [FCS_CKM.2 Cryptographic key distribution, or

FCS_COP.1 Cryptographic operation]

FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [**Advanced Encryption Standard (AES)**] and specified cryptographic key sizes [**256 Bit**] that meet the following: [**FIPS PUB 197**].

Cryptographic Algorithm	Cryptographic Key Sizes	Standards
AES-CBC	256 bits	FIPS 197, Advanced Encryption Standard (AES)

Table 9 Cryptographic Algorithm, Cryptographic Key Sizes and Standards for FCS_CKM.1.1

Application Notes Cipher Block Chaining Mode (CBC) is used as the mode of operation for AMAHSM AES 256 Encryption Algorithm.

6.2.2 Class FCS: Cryptographic support

FCS_COP.1 Cryptographic Operation

Hierarchical No other components.

Dependencies [FDP_ITC.1 Import of user data without security attributes, or

FDP_ITC.2 Import of user data with security attributes, or

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1 The TSF shall perform [**“Cryptographic Operations” in Table 10**] in accordance with a specified cryptographic algorithm [**"Cryptographic Algorithm” in Table 10**] and cryptographic key sizes [**“Cryptographic Key Sizes” in Table 10**] that meet the following: [**“Standards” in Table 10**].

Cryptographic Operations	Cryptographic Algorithm	Cryptographic Key Sizes	Standards
Code Encryption and SO Encryption Key Algorithm	AES-CBC	256 bits	FIPS 197, Advanced Encryption Standard (AES)
Generate the SHA256 hash summary information of each file in the application APK package	SHA256	256 bits	FIPS 180-4, Secure Hash Standard
RSA Signature generation	RSA 1024	1024 bits	FIPS 186-5, Digital Signature Standard (DSS)

Table 10 Cryptographic Operation, Cryptographic Algorithm, Encryption Key Length and Standard for FCS_COP.1.1

Application Notes Cipher Block Chaining Mode (CBC) is used as the mode of operation for AMAHSM AES 256 Encryption Algorithm.

6.2.3 Class FPT: Protection of the TSF

FPT_ITT.1 Basic internal TSF data transfer protection

Hierarchical No other components.

Dependencies No dependencies

FPT_ITT.1.1 The TSF shall protect TSF data from [***modification***] when it is transmitted between separate parts of the TOE.

Data	Protection Method
CLASS.DEX	All Data are extracted separately from the APK file, and encrypted separately.
AndroidManifest.XML	

SO Library Files	The integrity verification information file is generated under the assets\meta-data directory, including 1) save the manifest.mf of the HASH value , 2) rsa.pub that holds the RSA public key 3) Save the signed rsa.sig

Table 11 Data and Protection Method for FPT_ITT.1.1

6.2.4 Class FPT: Protection of the TSF

FPT_ITT.3 TSF data integrity monitoring

Hierarchical No other components.

Dependencies FPT_ITT.1 Basic internal TSF data transfer protection

FPT_ITT.3.1 The TSF shall be able to detect [***modification of data***] for TSF data transmitted between separate parts of the TOE.

FPT_ITT.3.2 Upon detection of a data integrity error, the TSF shall take the following actions: [**Terminate hardening process**].

Data Integrity Error	Actions to be Taken
File integrity hash verification failed	Terminate the hardening process due to unauthorized tampering during runtime.

Table 12 Data Integrity Error and Actions to be Taken for FPT_ITT.1.1

6.2.5 Class FAU: Security audit

FAU_GEN.1 Audit data generation

Hierarchical No other components.

Dependencies FPT_STM.1 Reliable time stamps

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

~~a) Start-up and shutdown of the audit functions;~~

b) All auditable events for the [***not specified***] level of audit; and

c) [

a. User login

b. Hardening submission

c. Hardening policy creation or modification

].

FAU_GEN.1.2	The TSF shall record within each audit record at least the following information: a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [***None***].
Application Notes	The start-up and shutdown of the audit function are not applicable and only can be turn off (not disable temporary) if the TOE is being turn off/power off

6.3 Security Assurance Requirements

This ST claims compliance to the assurance requirements from the CC EAL2 assurance package. This EAL was chosen based on the security problem definition and the security objectives for the TOE. The chosen assurance level is consistent with the claimed threat and environment.

The following table summarized the TOE assurance requirements drawn from CC Part 3.

Assurance Class	Assurance components
ADV: Development	ADV_ARC.1 Security architecture description
	ADV_FSP.2 Security-enforcing functional specification
	ADV_TDS.1 Basic design
AGD: Guidance Documents	AGD_OPE.1 Operational user guidance
	AGD_PRE.1 Preparative procedures
ALC: Lifecycle Support	ALC_CMC.2 Use of a CM system
	ALC_CMS.2 Parts of the TOE CM coverage
	ALC_DEL.1 Delivery procedures
ASE: Security Target evaluation	ASE_CCL.1 Conformance claims
	ASE_ECD.1 Extended components definition
	ASE_INT.1 ST introduction
	ASE_OBJ.2 Security objectives
	ASE_REQ.2 Derived security requirements
	ASE_SPD.1 Security problem definition
	ASE_TSS.1 TOE summary specification
ATE: Tests	ATE_COV.1 Evidence of coverage
	ATE_FUN.1 Functional testing
	ATE_IND.2 Independent testing - sample
AVA: Vulnerability assessment	AVA_VAN.2 Vulnerability analysis

Table 13: Security Assurance Requirements for EAL2

7 TOE Summary Specifications

TOE addressed the security functional requirements as following:

7.1 Cryptographic Support

TOE shall generate the cryptographic keys during the hardening process that is in accordance with specific cryptographic key generation algorithm and specific cryptographic key sizes. The key generation shall adhere to specified standards.

TOE shall ensure that encryption is implemented on application's code and SO Library Files. The hash for verification purposes shall also be generated. Both the encryption and hashing operation shall be performed in accordance with specific cryptographic algorithm and cryptographic key sizes, and shall adhere to specified standards.

Relevant SFR: FCS_CKM.1, FCS_COP.1

7.2 Protection of the TSF

TOE shall protect the application APK's data when extracting the components. The integrity verification information file shall be generated and stored appropriately.

TOE shall detect any modification of data during the hardening process while components are extracted. If the file integrity hash verification fails, the TOE shall terminate the hardening process.

Relevant SFR: FPT_ITT.1, FPT_ITT.3

7.3 Audit Data Generation

TOE shall generate an audit record of auditable events, where a log will be generated to document the events of web portal operations. The audit record shall include user login, hardening submission, and hardening policy creation or modification.

Relevant SFR: FAU_GEN.1

8 Rationale

8.1 Protection Profile Conformance Claim Rationale

ST does not claim conformance to any Protection Profile. Hence, there are no elements to be covered in the conformance claim rationale.

8.2 Security Objectives Rationale

This section explains how threat, assumptions and OSP are related to each other. The following tables show threat, assumptions and organizational policy being mapped to security objectives.

8.2.1 Rationale of Security Objectives Mapped to Threats

Threats	Security Objectives	Rationale
T.TAMPER An unauthorized person may obtain access to the TOE during runtime and make unauthorized changes to the data.	**O.TAMPER** TOE shall ensure that data are protected from unauthorized tampering.	This security objective counters threat because TOE shall prevent unauthorized changes with implementation of hash integrity verification.
T.DATA Data may not be protected adequately during transmission within TOE.	**O.DATA** TOE shall ensure data are secured in transmission to and from TOE.	This security objective counters threat because TOE shall prevent data leakage to occur while transmitting data to and from TOE with implementation of encryption.
T.LOGS An unauthorized person may intentionally or unintentionally delete audit records to destroy evidence of adverse events executed.	**O.LOGS** TOE shall ensure the logs are protected from unauthorized access to prevent unauthorized modifications to the generated logs.	This security objective counters threat because TOE shall protect unauthorized modifications to logs by preventing unauthorized access.

Table 14 - Rationale of Security Objectives Mapped to Threats

8.2.2 Rationale of Security Objectives Mapped to OSP

Not applicable since there is no OSP declared in ST.

8.2.3 Rationale of Security Objectives Mapped to Assumptions

Assumptions	Security Objectives	Rationale
A.USER The users are trusted; the users shall not maliciously compromise the security functionality of the TOE. The users are well-trained; the user shall comply to the operating procedures stipulated in the user guidance.	**OE.USER** The users are trusted; the users shall not maliciously compromise the security functionality of the TOE. The users are well trained; the user shall comply with the operating procedures stipulated in the user guidance.	This security objective upholds assumption because the users shall be non-hostile and follows guidance documentation accordingly; however, the user is not free from human error and mistakes.
A.DATAFLOW The data flow to the TOE must be between the subsystems and TOE as defined in the use case.	**O.DATA** TOE shall ensure data are secured in transmission to and from TOE.	This security objective upholds assumption because the TOE shall protect the data flow to and from the TOE with encryption.

Table 15 - Rationale of Security Objectives Mapped to Assumptions

8.3 Extended Security Functional Requirement Rationale

Not applicable since there is no Extended Security Functional Requirement (SFR) declared in ST.

8.4 Extended Security Assurance Requirement Rationale

Not applicable since there is no extended Security Assurance Requirement declared in ST.

8.5 Security Functional Requirements Rationale

This section provides the rationale of using SFRs to meet the security objectives for the TOE and justify the SFRs dependencies that have been satisfied or not satisfied.

8.5.1 Rationale for SFR Mapped to Security Objectives for TOE

Security Objectives	SFRs	Rationale
O.DATA TOE shall ensure data are secured in transmission to and from TOE.	FCS_CKM.1 FCS_COP.1	This SFR requires the TOE to encrypt the TSF data for secure transmission. It traces back to this objective.
O.TAMPER TOE shall ensure that data are protected from unauthorized tampering.	FPT_ITT.1 FPT_ITT.3	This SFR requires the TOE to validate the hash integrity of the file for any trace of unauthorized tampering. It traces back to this objective.
O.LOGS TOE shall ensure the logs are protected from unauthorized access to prevent unauthorized modifications to the generated logs.	FAU_GEN.1	This SFR requires the TOE to ensure the logs are not stored on users' local devices to prevent unauthorized modifications to the generated logs. It traces back to this objective.

Table 16 - Rationale for SFR Mapped to Security Objectives for TOE

8.5.2 SFR Dependency Rationale

The following table provides a demonstration that all SFRs dependencies included in the ST have been satisfied.

SFR	Dependency	Dependency Met?	Justification
FCS_CKM.1	FCS_CKM.2, or FCS_COP.1 FCS_CKM.4	Yes (FCS_COP.1) No	- FCS_CKM.4 is not applicable as key management is out of scope
FCS_COP.1	FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1 FCS_CKM.4	Yes (FCS_CKM.1) No	- FCS_CKM.4 is not applicable as key management is out of scope
FPT_ITT.1	-	-	-
FPT_ITT.3	FPT_ITT.1	Yes	-
FAU_GEN.1	FPT_STM.1	No	The timestamp is provided by the environment

Table 17 - SFR Dependencies

----------------------------------END OF DOCUMENT----------------------------------