Arxan Technologies

Arxan Cryptographic Key & Data Protection

FIPS 140-2 Cryptographic Module Non-Proprietary Security Policy

Version: 1.7

Date: February 10, 2017

1 Table of Contents

9 List of Tables

10 List of Figures

1 Introduction

This document defines the Security Policy for the Arxan Cryptographic Key & Data Protection cryptographic software module, hereafter denoted the Module. The Module is a sophisticated implementation of whitebox cryptography that takes keys for use in the cryptographic operations performed by methods in the Arxan Cryptographic Key & Data Protection libraries. The Arxan Cryptographic Key & Data Protection module allows you to use both obfuscation and encryption on sensitive data and chain together cryptographic operations to reduce or remove the possibility of a successful attack. The Module meets FIPS 140-2 overall Level 1 requirements.

Table 1 – Arxan Cryptographic Key & Data Protection Cryptographic Module Configurations

Module	SW Version	OE	Tested Platform
Arxan Cryptographic Key & Data Protection	1.0	Android KitKat 4.4.1	Samsung Galaxy Tablet 4

The Module is intended for use by US Federal agencies and other markets that require FIPS 140-2 validated software modules. From the point of view of Physical Security Area 5 of FIPS 140-2, the Module is a multi-chip standalone embodiment; the cryptographic boundary is the entire monolithic library file, libtfit_fips_module.so.

The FIPS 140-2 security levels for the Module are as follows:

Table 2 – Security Level of Security Requirements

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

1.1 Physical Cryptographic Boundary

The physical form of the Module is defined as a general purpose computer/device running the Module. The physical ports are defined as the ports of a general purpose computer.

1.2 Software and Logical Cryptographic Boundary

Figure 1 depicts the Module's operational environment. The logical boundary is the entire monolithic library file. The logical interface is defined as the API for the library.

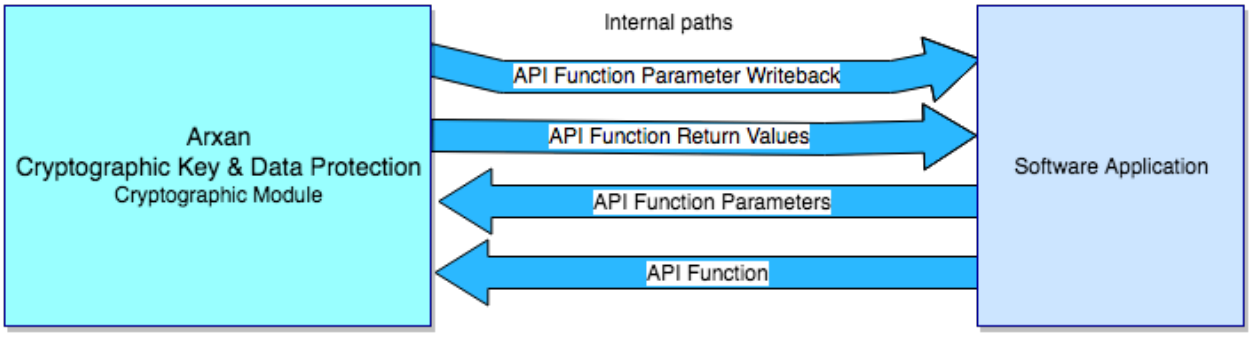

Figure 1 – Module Block Diagram

The Arxan Cryptographic Key & Data Protection module is loaded as a dynamic library as part of a host software application.

Table 3 – Ports and Interfaces

Port	Description	Logical Interface Type
API Functions	The externally visible Arxan Cryptographic Key & Data Protection library symbols.	Control in
API Function Parameter Writeback	The area where results of operations are placed.	Data out
API Function Return Values	The return values indicate the status of different operations.	Status out
API Function Parameters	The values that will control the operation of the Arxan Cryptographic Key & Data Protection module.	Control in, Data in

1.3 Mode of Operation

The Module only operates in FIPS mode. Access to the non-FIPS capabilities requires a new delivery from Arxan Technologies.

2 Cryptographic Functionality

The Module implements only the FIPS Approved cryptographic functions listed in the table below.

Table 4 – Approved and CAVP Validated Cryptographic Functions

Algorithm	Description	Cert #
AES [Arxan Cryptographic Key & Data Protection AES Component v1.0]	[FIPS 197, SP 800-38A] Functions: Encryption, Decryption Mode: ECB Key size: 256-bits	AES Cert. #4123
ECDSA [Arxan Cryptographic Key & Data Protection ECDSA Component v1.0]	[FIPS 186-4] Function: Signature Verification Curve/SHA: P-256, SHA-256	ECDSA Cert. #938
HMAC [Arxan Cryptographic Key & Data Protection HMAC Component v1.0]	[FIPS 198-1] Functions: Generation, Verification, Authenticated Integrity Check SHA size: SHA-256	HMAC Cert. #2694
KAS (ECC CDH) [Arxan Cryptographic Key & Data Protection KAS_Component v1.0]	[ECC CDH Primitive (SP800-56A Section 5.7.1.2)] Curve: P-256	CVL Cert. #930
SHA [Arxan Cryptographic Key & Data Protection SHA Component v1.0]	[FIPS 180-4] Functions: Digital Signature Verification, non-Digital Signature Applications SHA sizes: SHA-256, SHA-512	SHS Cert. #3392
Triple-DES [Arxan Cryptographic Key & Data Protection TDES Component v1.0]	[SP 800-67] Functions: Encryption, Decryption Mode: Three-key; TECB mode	Triple-DES Cert. #2253

2.1 Critical Security Parameters

All CSPs used by the Module are described in this section. All usage of these CSPs by the Module (including all CSP lifecycle states) is described in the services detailed in Section 4.

Table 5 – Critical Security Parameters (CSPs) used in Approved mode

CSP	Description / Usage
AES symmetric key	Symmetric key used with AES for encryption/decryption
Triple-DES symmetric key	Symmetric key used with Triple-DES for encryption/decryption
EC Diffie-Hellman private key	Secret EC Diffie-Hellman private key
EC Diffie-Hellman shared secret	Shared secret obtained through EC Diffie-Hellman
HMAC key	Key used during HMAC calculation verification

2.2 Public Keys

Table 6 – Public Keys

Key	Description / Usage
EC Diffie-Hellman public key	EC Diffie-Hellman static public key component
ECDSA public key	ECDSA public key

3 Roles, Authentication and Services

3.1 Assumption of Roles

The module supports two distinct operator roles, User and Cryptographic Officer (CO). Roles are assumed implicitly. Both roles have access to all services.

The Module does not support a maintenance role or bypass capability. The Module does not support concurrent operators.

Table 7 – Roles Description

Role ID	Role Description
CO	Cryptographic Officer – The CO is responsible for installing and removing the Arxan Cryptographic Key & Data Protection module from the system as necessary. He or she also has all the permissions a User has.
User	User – The User is allowed to interact with any of the exposed Arxan Cryptographic Key & Data Protection interfaces. He or she may provide various data and control inputs and will be able to receive status and data out information.

3.2 Services

All services implemented by the Module are listed in the table below. Each service description also describes all usage of CSPs by the service.

Table 8 – Approved Security Services

Service	Algo	CO	USER	Associated API	Description	Status Output
Encrypt	AES	X	X	tfit_aes_encrypt	Encrypt data with AES	"0" - SUCCESS Positive/negative value- FAILURE
	Triple-DES	X	X	tfit_3des_encrypt	Encrypt data with Triple-DES	"0" - SUCCESS Positive/negative value- FAILURE
Decrypt	AES	X	X	tfit_aes_decrypt	Decrypt data with AES	"0" – SUCCESS Positive/negative value- FAILURE
	Triple-DES	X	X	tfit_3des_decrypt	Decrypt data with Triple-DES	"0" – SUCCESS Positive/negative value- FAILURE

Verify Signature	ECDSA	X	X	tfit_ecdsa_verify	Verify ECDSA signature	"0" – SUCCESS Positive/negative value- FAILURE
Generate Shared Secret	ECDH	X	X	tfit_ecdh	Generates a shared secret using EC Diffie-Hellman	"0" – SUCCESS Positive/negative value- FAILURE
Zeroize		X	X	tfit_zeroize	Zeroize all keys	"0" - SUCCESS Positive/negative value- FAILURE
Hash Data	SHS	X	X	tfit_sha_digest	Hash message using SHS (SHA-256 or SHA-512)	"0" - SUCCESS Positive/negative value- FAILURE
HMAC Data	HMAC	X	X	tfit_hmac_verify tfit_hmac_generate	Verify keyed-hash message authentication code	"0" - SUCCESS Positive/negative value- FAILURE
Initialize AES Encryption		X	X	tfit_load_key_aes_encrypt	Initialize AES algorithm by setting the key used by AES Encrypt service	"0" - SUCCESS Positive/negative value- FAILURE
Initialize AES Decryption		X	X	tfit_load_key_aes_decrypt	Initialize AES algorithm by setting the key used by AES Decrypt service	"0" - SUCCESS Positive/negative value- FAILURE
Initialize Triple-DES Encryption		X	X	tfit_load_key_3des_encrypt	Initialize Triple-DES algorithm by setting the key used by Triple-DES Encrypt service	"0" - SUCCESS Positive/negative value- FAILURE
Initialize Triple-DES Decryption		X	X	tfit_load_key_3des_decrypt	Initialize Triple-DES algorithm by setting the key used by Triple-DES Decrypt service	"0" - SUCCESS Positive/negative value- FAILURE
Initialize EC Diffie-Hellman		X	X	tfit_load_key_ecdh	Initialize Diffie-Hellman algorithm by setting the key material and curve used by Generate Shared Secret Service	"0" - SUCCESS Positive/negative value- FAILURE
Initialize ECDSA		X	X	tfit_load_key_ecdsa_verify	Initialize ECDSA algorithm by setting the key material	"0" - SUCCESS Positive/negative

Verify					and curve used by Verify Signature Service	value- FAILURE
Initialize HMAC		X	X	tfit_load_key_hmac_verify tfit_load_key_hmac_generate	Initialize HMAC algorithm by setting the key material used by HMAC Data Service	"0" - SUCCESS Positive/negative value- FAILURE

Table 9 – Non-Crypto Services

Service	Description	CO	U
Get Status	This service (API: tfit_get_error) returns the status of the Module	X	X
Get Version	This service (API: tfit_get_version) returns the version of the Module	X	X
Self-Test	Self-Test Service is invoked by power cycling or reloading the Module.	X	X

Table 10 - CSP Access Rights

Service	CSPs				
	AES Symmetric key	Triple-DES Symmetric key	EC Diffie-Hellman private key	EC Diffie-Hellman Shared Secret	HMAC key
Encrypt	R	R	-	-	-
Decrypt	R	R	-	-	-
Verify Signature	-	-	-	-	-
Generate Shared Secret	-	-	R	W	-
HMAC Data	-	-	-	-	R
HASH Data	-	-	-	-	-
Initialize AES Encryption	W	-	-	-	-
Initialize AES Decryption	W	-	-	-	-
Initialize Triple-DES Encryption	-	W	-	-	-
Initialize Triple-DES Decryption	-	W	-	-	-
Initialize EC Diffie-Hellman	-	-	W	-	-
Initialize ECDSA Verify	-	-	-	-	-
Initialize HMAC	-	-	-	-	W
Self-Test	-	-	-	-	-
Get Status	-	-	-	-	-
Get Version	-	-	-	-	-
Zeroize	Z	Z	Z	Z	Z

Table 10 - CSP Access Rights describes the relationship between access to CSPs and the module's services. The modes of access shown in the table are defined as:

- G = Generate: The module generates the CSP.
- R = Read: The module reads the CSP. The read access is typically performed before the module uses the CSP.
- W = Write: The module writes the CSP. The write access is typically performed after a CSP is imported into the module, or when the module overwrites an existing CSP.
- Z = Zeroize: The module zeroizes the CSP.

4 Self-tests

Each time the Module is powered up it tests that the cryptographic algorithms still operate correctly and that sensitive data have not been damaged. Power up self–tests are available on demand by power cycling the Module.

On power up or reset, the Module performs the self-tests described in Table 11 below. All KATs must be completed successfully prior to any other use of cryptography by the Module. If one of the KATs fails, the Module enters the error state.

Table 11 – Power Up Self-tests

Test Target	Description
Software Integrity Test	KAT: HMAC-SHA-256, using a 32-byte HMAC key
AES	KATs: Encryption, Decryption Modes: ECB Key size: 256-bits (32-bytes)
Triple-DES	KATs: Encryption, Decryption Mode: ECB Key size: 192-bits
EC Diffie-Hellman	KATs: Per IG 9.6 – Primitive "Z" Computation Parameter Sets/Key sizes: EC
ECDSA	KAT: Signature Verification Curve/Key size: P-256 w/ SHA-256
SHA	KATs: SHA-256 and SHA-512

The initiation of self-testing performed at power up was designed according to the guidance IG 9.10 "Power-Up Tests for Software Module Libraries", http://csrc.nist.gov/groups/STM/cmvp/documents/fips140-2/FIPS1402IG.pdf - that is, upon creation of the module's address space the OS loader proceeds to automatically execute the entry point function (constructor), which automatically invokes all power up self-tests listed above.

5 Operational Environment

The Module is software only and operates on a general purpose computer. The Module was tested on the following operating environment:

- Android – ARM 32bit (Android KitKat 4.4.1) on a Samsung Galaxy Tablet 4

6 Mitigation of Other Attacks Policy

The Module does not claim any attack mitigation beyond FIPS 140-2 Level 1 requirements.

7 Security Rules and Guidance

The Module design corresponds to the Module security rules. This section documents the security rules enforced by the cryptographic module to implement the security requirements of this FIPS 140-2 Level 1 module.

1. The operator is capable of commanding the Module to perform the power up self-tests by power cycling the system or by re-loading the library into the application.
2. Power up self-tests do not require any operator action.
3. Data output is inhibited by self-tests and error states.
4. Status information does not contain CSPs or sensitive data that if misused could lead to a compromise of the module.
5. The module does not support concurrent operators.
6. The module does not support a maintenance interface or role.
7. The module does not support manual key entry.
8. The module does not have any external input/output devices used for entry/output of data.

8 References and Definitions

The following standards are referred to in this Security Policy.

Table 12 – References

Abbreviation	Full Specification Name
[FIPS140-2]	*Security Requirements for Cryptographic Modules*, May 25, 2001
[SP800-131A]	*Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths*, January 2011

Table 13 – Acronyms and Definitions

Acronym	Definition
AES	Advanced Encryption Standard
CO	Cryptographic Officer
CSP	Critical Security Parameter
EC Diffie-Hellman	Elliptic Curve Diffie-Hellman
ECDSA	Elliptic Curve Digital Signature Algorithm
FIPS	Federal Information Processing Standard
HMAC	Hash-based Message Authentication Code
KAS	Key Agreement Scheme
KAT	Known answer test; PCT = Pairwise consistency test)
NIST	National Institute of Standards and Technology
RSA	Rivest Shamir Adleman
SHA	Secure Hash Algorithm
Triple-DES	Triple Data Encryption Standard