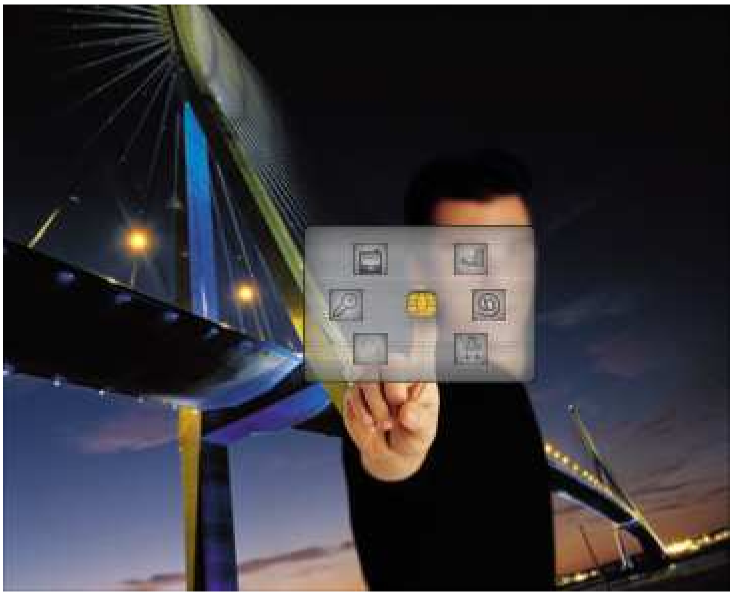
GemXpresso R4 E36/E72 PK

Security Policy



TITLE	**GemXpresso R4 E36/E72 PK - Security Policy**
REF.	**SP01R10630 - _05**
DATE:	**06/10/06**

TABLE OF CONTENTS

4.1.1 PIN assignments and contact dimensions: 10
4.1.2 Conditions of use........ 10
4.2.1 APDU commands........ 11
4.2.2 API interface........ 12
5.1.1 Introduction........ 13
5.1.2 Identity based authentication policy........ 13
5.1.3 Mechanism interfaces........ 14
5.1.4 Security rules........ 15
5.1.5 Mechanism strengths........ 15
5.2.1 Introduction........ 15
5.2.2 Services........ 16
5.2.3 Security rules........ 18
7.2.1 High/Low Frequency Sensor 21
7.2.2 High/Low Voltage Sensor........ 21
7.2.3 High/Low Temperature Sensor........ 21
7.2.4 Active Shields........ 21
7.2.5 Light sensor........ 21
7.2.6 Glitch sensor........ 22
7.2.7 Filters........ 22
7.2.8 BUS Scrambling........ 22
9.6.1 Input Data........ 25
12.3.1 Security Rules and Design 30

Table of figures:

Figure 1- Cryptographic Module Boundary 8
Figure 2 - Contact plate example – Contact physical interface........ 10

References

[1] FIPS PUB 140-2 – Federal Information Processing Standard Publication – Security requirements for cryptographic modules – 2001, May the 25th, with change notice (12-03-2002).

[2] Derived Tests Requirements for FIPS PUB 140-2 - Federal Information Processing Standard Publication – Security requirements for cryptographic modules – 2004, March the 24th.

[3] NIST Web site, http://www.nist.gov

[4] Global Platform – Release 2.1.1

[5] Visa Global Platform – Release 2.1.1

[6] Java Card API Specification – (SUN) – Release 2.2.1

[7] Java Card Runtime Environment (JCRE) Specification (SUN) – 2.2.1

[8] Java Card Virtual Machine (VM) Specification – SUN – Release 2.2.1

[9] RSA PKCS#1: RSA Cryptographic Standard (RSA Laboratories) – 2.1

[10] ISO 7816 parts 1-6 (ISO / IEC)

[11] ISO X9.31

[12] ISO 14443 RF Interface (ISO / IEC)

1 Scope

This Security Policy specifies the security rules under which the GemXpresso R4 E36/E72 PK, herein identified as the **"GX4-S – FIPS"** platform, must operate. Some of these rules are derived from the security requirements of **FIPS140-2' standard [1]**, others are derived from the Gemalto' experience in embedded security software.

These rules define the interrelationships between the:

- Module users and administrators,
- Module services,
- Security Relevant Data Items (SRDIs).

2 Introduction

2.1 Gemalto Smart Card Overview

Gemalto aims to provide **FIPS140-2 Level 3** cryptographic smart cards. The cards are based on a Gemalto Open OS Platform and on which FIPS 140-2 Level 3 validated platform-independent applets may be loaded and instantiated at post issuance. The card provides authentication, encryption and digital signature cryptographic services to applets. It is under the charge of the applets to be loaded and instantiated within the card to use, in conformance with specifications, the different services offered by the platform.

Moreover, FIPS 140-2 Level 3 validation is required for the applets to be loaded and instantiated within the card in order to reach the FIPS 140-2 Level 3 compliance for the **whole and composite product (i.e. platform plus post issuance applets).** Applet or package loading and installation are outside the FIPS mode operation and thus are outside the scope of the FIPS validation of the module.

This security policy specifies the security rules under which the module- **GX4-S - FIPS (cryptographic module)** operates.

2.2 Gemalto Smart Card Open Platform

The cryptographic module is a state of the art Java Open Platform-based smart card. This highly secure platform benefits from all the Gemalto expertise in Java Card security, from the latest developments in cryptographic resistance against known attacks, and provides FIPS approved cryptographic algorithms and self-tests. Additional software countermeasures have also been added by Gemalto.

This cryptographic module uses a state of the art manufacturing flow in terms of security and provides applets with memory, cryptographic and I/O services.
The cryptographic module ensures on-card applets safe coexistence thanks to its secure Virtual Machine (VM) and firewall. The Java VM is fully compliant with the **Java Card 2.2.1 standard [8]**.

The card life cycle is managed according to the **Global Platform (GP) specification**. Issued cards have been loaded with a set of cryptographic keys, and are moreover in the "SECURED" state. The security implementation is fully compliant with the **Global Platform (GP) 2.1.1 specification**.
The cryptographic module integrates symmetric and asymmetric cryptographic algorithms as specified in the **JavaCard specification [6]** and offers RSA for Signature/Verification, SHA-1, hashing functions, on-board RSA Key generation, Triple-DES CBC and ECB and AES ECB and CBC algorithms.

The module has hardware version GXP4-M2612410 and firmware version GX4-S_E005 (MSA029).

2.3 Security Level

The cryptographic module meets the overall requirements applicable to **FIPS140-2 Level 3**. The individual security requirements meet the level specifications as follows.

Security Requirements Section	Security Level
Cryptographic Module Specification	3
Cryptographic Module Ports and Interfaces	3
Roles, Services and Authentication	3
Finite State Model	3
Physical Security	3
Operational Environment	N/A
Cryptographic Key Management	3
EMI/EMC	3
Self-Tests	3
Design Assurance	3
Mitigation of Other Attacks	3

Table 1 – FIPS 140-2 Security Levels

3 Cryptographic Module Specification

3.1 Gemalto Crypto-Module Cryptographic Boundary

The Cryptographic Boundary is defined to be the 'module edge' of the **GX4-S – FIPS** referred to hereafter as the Micro Module, a set of "embedded" hardware and software that implements cryptographic functions and processes, including cryptographic algorithms and key generation. **GX4-S – FIPS Micro-Module** is a single chip implementation of a cryptographic module. The micro-module is designed to be embedded in a plastic card body to provide an **ISO-7816 [10]** compliant smart card.

The Cryptographic Module provides only a single contact interface.

During the Gemalto manufacturing process, the chip (ICC) is wire-bonded on the inner side of a contact plate, then globe-topped with resin. **The resulting Micro-Module meets the physical security requirements of FIPS140-2 Level 3.**

Additionally, from the customer point of view two EEPROM configurations are available: 36 or 72KB EEPROM This customer configuration is purely functional one and does not impact in any way the security aspect of the product. This memory configuration is set up during initialization process.

All the components of the **GX4-S – FIPS Micro-Module** that are included in the cryptographic module boundary are those as shown in the following figure:

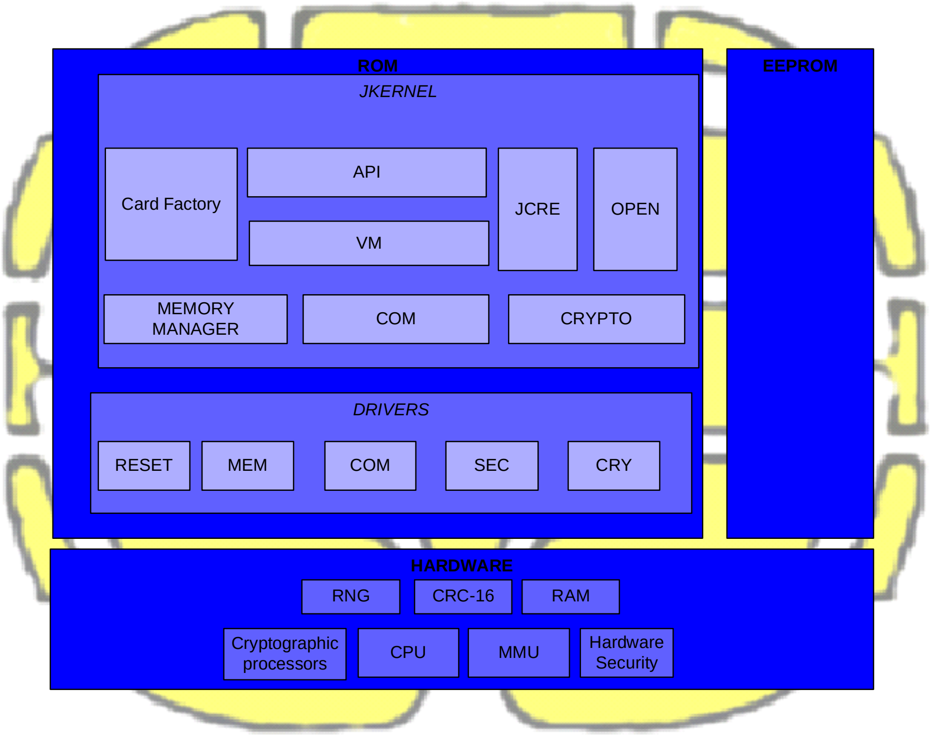


Figure 1- Cryptographic Module Boundary

The following sections provide a description of the different entities presented in this scheme.

3.2 ROM

The chip's ROM includes the **GX4-S – FIPS** Operating System (OS) meaning that the OS is protected against disclosure and modification. The cryptographic module is implemented using a high level language, a limited number of software modules that require fast processing have been written in a low-level language. This OS includes the following design entities:

<table>
<tr><th>Design item</th><th>Functionality</th></tr>
<tr><td colspan="2"><b>JKERNEL layer</b></td></tr>
<tr><td colspan="2">The Jkernel sub-system provides a Java Card-oriented environment for the Applications sub-system, including:
- (<b>JCRE</b>) the Java Card 2.2.1 runtime environment including OS dispatcher, registry, loader, logical channel management, RMI.
- (<b>API</b>) the public APIs: Java Card 2.2.1 and other optional APIs (e.g. proprietary…).
- (<b>VM</b>) the Java Card 2.2.1 virtual machine including bytecode interpreter, firewall, exception management, VM extension for bytecode optimizer.
- (<b>OPEN</b>) the GP2.1.1 Open Platform environment including card content management, key management, card and applet life cycle management, security policy.
- (<b>Card Factory</b>) the Card Factory, including OS bootstrap, OS initialization, self-tests, industrialization command.
- (<b>MEMORY MANAGER</b>) the Memory Manager including services such as memory access, allocation, delete, GC.
- (<b>COM</b>) the Communication handler including services, such as ATR, PSS, T=0, T=1, T=CL.
- (<b>CRYPTO</b>) the Cryptography engines including services such as Triple-DES (ECB, CBC), hash functions (SHA), RSA (including padding such as PKCS), on-board key generation, AES (ECB, CBC).</td></tr>
<tr><td colspan="2"><b>DRIVERS layer</b></td></tr>
<tr><td colspan="2">The drivers layer provides the following services:
- (<b>RESET</b>) OS startup and chip initialization, IT/exception vectors, MMU/banking configuration…
- (<b>MEM</b>) memory module including NVM access, atomicity and transaction management…
- (<b>COM</b>) communication module including IO exchanges, timing control…
- (<b>SEC</b>) security module including counter-measures and fault attack management, CRC, RNG…
- (<b>CRY</b>) Cryptography module including basic algorithms such as Triple-DES, SHA, RSA</td></tr>
</table>

Table 2 – ROM – content description

3.3 EEPROM – Applets

The chip's EEPROM can store applets. However, in order to remain FIPS 140-2 L3 validated for the resulting GX4-S with subsequently instantiated applets, those applets would have to be FIPS 140-2 Level 3 validated independently.

3.4 Hardware Chip

The cryptographic module includes the **S3CC9TC** ICC. It includes:

- CPU 16 bit CalmRISC16 core
- EEPROM 72 KB or 36KB depending of initialization configuration
- ROM 384 KB
- RAM
  - 8KB Data memory
  - 2KB Crypto memory
- Shields
  - Active shield
- Passivation detector
- Hardware and enhanced security sensors
  - Low/high voltage sensor
  - Low/high clock frequency sensor
  - Low/high temperature sensor
  - Light sensor
  - Glitch sensor
- Bus scramble for EEPROM, RAM
- Memory Management and Protection Unit through MMU,
- Random Generator,
- Cryptographic Co-Processors
  - High speed Triple-DES co-processor – prevention SPA and DPA
  - PKI co-processor (Tornado)
- Contact interfaces.

3.5 FIPS Approved Security Functions

The following table gives the list of FIPS approved security functions that are provided by the **GX4-S – FIPS** Java Card API.

SECURITY FUNCTION	DETAILS	FIPS APPROVED
Triple-DES	ECB mode in encryption	Yes
	ECB mode in decryption	Yes
	CBC mode in encryption	Yes
	CBC mode in decryption	Yes
SHA-1	Hashing operation	Yes
RSA	Key generation	Yes
	Signature following PKCS#1 with SHA-1 hashing	Yes
	Verification following PKCS#1 with SHA-1 hashing	Yes
P-RNG	Pseudo Random Number Generation	Yes
AES	ECB mode in encryption	Yes
	ECB mode in decryption	Yes
	CBC mode in encryption	Yes
	CBC mode in decryption	Yes

Table 3 – FIPS Approved Security Functions

4 Cryptographic Module Ports and Interfaces

The **GX4-S – FIPS Micro-Module** restricts all information flow and physical access.
Physical and logical interfaces define all entry and exit points to and from the micro module.

4.1 Physical Port – Contact mode

4.1.1 PIN assignments and contact dimensions:

GX4-S – FIPS Micro-Module follows the standards **"ISO 7816-1 Physical characteristics" [10]** and **"ISO 7816-2 Dimensions and contact location" [10]**.

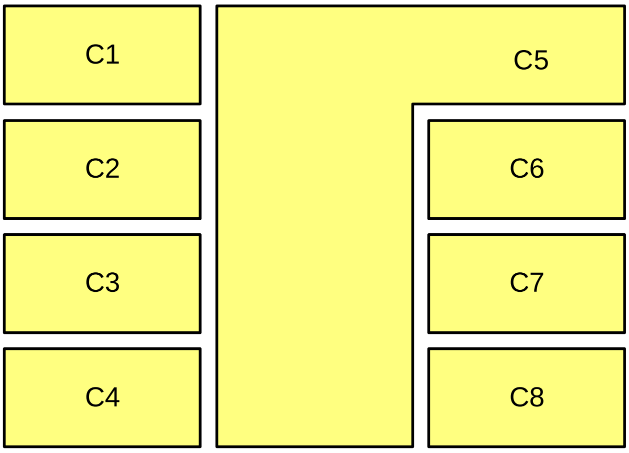


Figure 2 - Contact plate example – Contact physical interface

Contact No.	Assignments	Contact No.	Assignments
C1	VCC (Supply voltage)	C5	GND (Ground)
C2	RST (Reset signal)	C6	Not connected
C3	CLK (Clock signal)	C7	I/O (Data Input/Output)
C4	Not connected	C8	Not connected

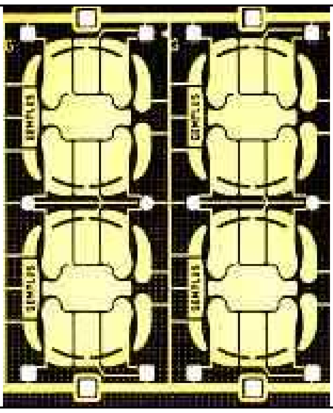
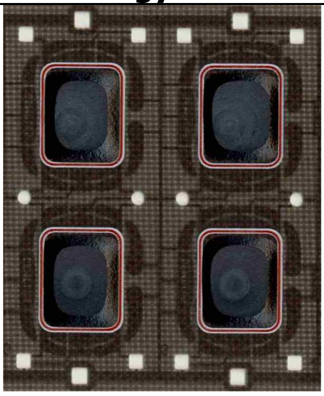
Table 4 - Contact plate pin list – Contact mode

4.1.2 Conditions of use

The electrical signals and transmission protocols follow the **ISO 7816-3 [10]**. The conditions of use are the following:

Conditions	Range
Voltage	1,62 V and 5.5 V
Frequency	1MHz to 5MHz

Table 5 - Voltage and frequency ranges

MIND-L Thermal black resin technology	
	
MIND-L design	**Thermal** black resin Technology

The thermal black resin technology consists in a resin that is applied on top of the chip at the back side of the module after the connection of the chip to the back side of the contact plates has been completed. This resin is applied in a semi liquid form and is polymerized by temperature. This resin is characterized by its black color and opacity that makes the silicon chip invisible when the module is finished. Its hardness provides efficient mechanical protection and tamper evidence; attempt to mechanically open the module will result in visible damage and/or loss of functionality by braking of either silicon chip and/or wires.

4.2 Logical Interface

GX4-S – FIPS Micro-Module provides services to external devices and internal applets with services. External devices have access to services by sending APDU commands while internal applets have access to services through internal API entry points.

For security reasons, **GX4-S – FIPS Micro-Module** inhibits all data output via the data output interface when an error state is reached and during self-tests.

4.2.1 APDU commands

The data exchange protocol between the cryptographic module and an outside device follows the **ISO 7816-4 [10] standard**. The cryptographic module acts as a slave device, receiving and executing APDU commands from outside devices. The cryptographic module receives APDU commands, performs the related internal processes according to its security policy, and then answers with APDU responses.

An APDU command consists of a mandatory command header of four bytes conditionally followed by a command body (Input Data). The response APDU consists of a conditional response body followed by a mandatory response trailer of two bytes. ISO APDU Types 1, 2, 3 and 4 are supported.

ISO Command Type	**Description**
Type 1 – ISO command	No input data, no response data
Type 2 – ISO "Out" command	No input data, response data
Type 3 – ISO "In" command	Input data, no response data
Type 4 – ISO "In" and "Out" command	Input data, response data

Table 6 - Accepted ISO APDU types

The cryptographic module enforces the establishment and use of a secure path for exchanging sensitive data with an external device.

4.2.2 API interface

GX4-S – FIPS Micro-Module provides trusted applets with internal services through its **JavaCard [6]** and **GP [5] APIs**.
The cryptographic module provides an execution sandbox for the applets and performs the requested services according to its roles and services security policy.
The available API services are defined in the following section.

5 Roles, Services and Authentication

This section specifies the roles, security rules, services, and Security Relevant Data Items (SRDI) of the cryptographic module. The Identification and Authentication Policy, and the Access Control Policy define the interrelationships between roles, identities, through the services and security rules.

The services that are provided by the cryptographic module are listed in the subsection labeled "SERVICES" in the Access Control Policy description.

5.1 Identification and Authentication Policy

5.1.1 Introduction

This section is dedicated to our identity-based authentication policy, and the related security rules of the mechanism interfaces and SRDI. The module performs identity-based authentication using PIN and cryptographic keys. A unique index value is associated with the PIN or the cryptographic key to uniquely identify the off-card entity performing the authentication.

5.1.2 Identity based authentication policy

The module supports two roles: The Crypto-Officer and User roles that are the followings:

Crypto-Officer role:

o Cryptographic Officer is considered as the smart card administrator and must authenticate via a secure channel, to the card manager. The CO has possession of the Card Manager Secure Channel keyset.

User role:

o User is considered as an entity who has possession of the Security Domain keyset and can request the services provided by the Security Domain on the card. User authenticates in the same way as a CO.

Each role is assumed implicitly as the module does not provide for explicit role selection. The services provided by the module to each role is specified in the table below

Roles/Services	Crypto Officer role Authenticated	User role Authenticated	No role Unauthenticated
INSTALL	X	X	
LOAD	X	X	
DELETE	X	X	
EXTERNAL AUTHENTICATE	X	X	X
GET DATA	X	X	X
GET STATUS	X	X	
INITIALIZE UPDATE	X	X	X

Roles/Services	Crypto Officer role Authenticated	User role Authenticated	No role Unauthenticated
PUT DATA	x	x	
PUT KEY	x	x	
SELECT	x	x	x
SET STATUS	x	x	
STORE DATA	x	x	
SET ATR	x	x	
GET MEMORY SPACE	x	x	
MANAGE CHANNEL	x	x	x

Table 7- Card Manager services Vs Roles

An operator can initiate module self-tests by issuing a card reset and issuing an APDU command. A user can also retrieve the module ATR on card power-up

5.1.3 Mechanism interfaces

The following table describes the mechanisms for identity authentication:

Interface	Description
INITIALIZE UPDATE *APDU*	This APDU command initiates the setting up of a secure channel. The card generates the session keys and exchanges data with the host.
EXTERNAL AUTHENTICATE *APDU*	This APDU command is used by the card to authenticate the host and to determine the level of security required for all subsequent commands. A previous and successful execution of the INITIALIZE UPDATE command is necessary prior to processing this command.

Table 8 - Mechanism interfaces

5.1.4 Security rules

The following table presents the security rules applied to these mechanisms:

Rule Identifier	Description
ia_co_rule.1	The Cryptographic Officer & User cannot get authenticated if the authorized number of attempts is reached.
ia_co_rule.2	The Cryptographic Officer & User must be re-authenticated if the card is reset or another applet is selected.
ia_co_rule.3	The Cryptographic Officer & User must be re-authenticated if the cryptographic module detects APDU communication corruption.

Table 9 - Security rules

5.1.5 Mechanism strengths

The strength of the mechanisms is the following:

Authentication Mechanism	Strength of Mechanism
GP mutual authentication	$\left(\frac{1}{2^{112}}\right)$
	The cryptogram sent is 8 bytes long and Triple-DES 2keys is used (i.e. 2 x 56 relevant bits key length).

Table 10 - Mechanism strengths

5.2 Access Control Policy

5.2.1 Introduction

This chapter is dedicated to access control security rules. Some services provided by the cryptographic module are subject to privileges. Privileges can be obtained by construction (for example at applet initialization) or by being identified as a privileged user.

- The **administrative commands** are restricted: these APDU commands can be used only in a secure channel (**session**). A secure channel is open when the card user has been authenticated through the GP mechanism as being the owner of the **Cryptographic Officer or User** keyset. The secure channel is closed if the card is reset or if the system closes it.
- The Java objects created by the applets are protected by the **Firewall** mechanism of the JCRE. The rules that are applied to **Java object accesses** are specified in the **JCRE specification [5]**. The firewall is a means of protecting applet information.
- **The loaded applets life cycle state** (Card Manager applet included) can be managed by the Cryptographic Officer. Proposed transitions must be coherent with the **GP specification**. An applet can manage its own life cycle state under the same conditions. An additional condition is imposed to applets that attempt to change the Card Manager life cycle state: they must have the privilege for **Card life cycle management**.

5.2.2 Services

The rules are applied to all the following service interfaces. (The service interfaces have been grouped according to the role to which they provide a service.)

Interface	Service Description
DELETE – *APDU*	This APDU is used to delete a uniquely identifiable object such as an Executable Load File, an application, optionally an Executable Load File and its related Applications or a key.
EXTERNAL AUTHENTICATE – *APDU*	This APDU command is used by the card to authenticate the host and to determine the level of security required for all subsequent commands. A previous and successful execution of the INITIALIZE UPDATE command is necessary prior to processing this command.
GET DATA – *APDU*	This APDU command is used to retrieve a single data object.
GET STATUS – *APDU*	This APDU command is used to retrieve the Card Manager, load file (package), and application life cycle data specific to the GP specification.
INITIALIZE UPDATE – *APDU*	This APDU command initiates the setting up of a secure channel. The card generates the session keys and exchanges data with the host.
INSTALL – *APDU*	This APDU command informs the card of the various steps required to load, install and make an applet selectable within the card.
LOAD – *APDU*	One or more LOAD commands are used to load the bytecode of the load file (package) defined in the previously issued INSTALL command to the card.
MANAGE CHANNEL – *APDU*	This command is used to open and close supplementary logical channels.
PUT DATA – *APDU*	This APDU command is used to set the value of the various data elements utilized and managed by the Card Manager (deprecated OP command)
PUT KEY – *APDU*	This APDU is used to: 1. Replace a single or multiple keys within an existing key set version; 2. Replace an existing key set version with a new key version; 3. Add a new key set version containing a single or multiple keys Key value is encrypted.
SELECT – *APDU*	This APDU command is used for selecting an application.
SET STATUS – *APDU*	This APDU command is used to change the state of the Card Manager or to change the life cycle state of an application.
STORE DATA – *APDU*	This APDU command is used to transfer data to an application or the security domain (card manager) processing the command.
SET ATR – *APDU*	

Interface	**Service Description**
	This APDU command is used to change atomically the current card ATR Warm and cold ATRs can be changed independently. Up to 3 different ATRs may be stored in EEPROM, in addition to the default ROM ATR. Each of them can be configured for cold reset, warm reset or both.
GET MEMORY SPACE – *APDU*	
	This APDU command is used to get the available memory space in the card

Table 11 - Cryptographic officer accorded interfaces and services

	Cryptographic officer role	**User role**	**User role**
	Authenticated	**Authenticated**	**Unauthenticated**
DELETE	**X**	**X**	
EXTERNAL AUTHENTICATE	**X**	**X**	**X**
GET DATA	**X**	**X**	**X**
GET STATUS	**X**	**X**	
INITIALIZE UPDATE	**X**	**X**	**X**
INSTALL	**X**	**X**	
LOAD	**X**	**X**	
MANAGE CHANNEL			**X**
PUT DATA	**X**	**X**	
PUT KEY	**X**	**X**	
SELECT	**X**	**X**	**X**
SET STATUS	**X**	**X**	
STORE DATA	**X**	**X**	
SET ATR	**X**	**X**	
GET MEMORY SPACE	**X**	**X**	

Table 12 – Authenticated and unauthenticated role accorded interfaces and services

Regarding the applet allowed interfaces; those interfaces are this API defined in Java Card API Specification – (SUN) – Release 2.2.1 and Global Platform – Release 2.1.1.

5.2.3 Security rules

The following table presents the security rules applied:

Rule Identifier	Description
ac_co_rule.1	Administrative commands can only be used by the **Cryptographic Officer.**
ac_java_rule.1	**JCRE firewall** checks are enforced by the cryptographic module to ensure Java object protection.
ac_life_rule.1	The **Cryptographic Officer** is responsible for locking and terminating the Card Manager life cycle state.
ac_life_rule.3	The **Cryptographic Officer** is responsible for managing the life cycle state of any applet (including system applets), in accordance with the GP specification.

Table 13 - Security rules

5.3 Additional Gemalto Security Rules

The following rules apply in addition to the FIPS140-2 requirements. The cryptographic module:

Rule Identifier	Description
AD_RULE.1	Does not input/output plain-text private/secret keys or other critical security parameters.
AD_RULE.2	Does not support a bypass mode.
AD_RULE.3	Does not provide a maintenance role/interface.
AD_RULE.4	Requires re-authentication when changing roles.
AD_RULE.5	Does not allow the loading of Software/Firmware - only applets.

Table 14 - Gemalto additional security rules

5.4 Security Relevant Data Item

The Security Relevant Data Items (SRDIs) of the cryptographic module are the following:

- **Secure Channel key set**
- **Secure channel session keys**
- **PRNG seed and seed key**

The following table proposes an association between the services or authentication mechanisms (the interface name is provided) and the SRDI they access. The access types are labeled as follows:

-
- W: write access
- U: the value is not explicitly read, but used within the scope of a comparison or computation process

Interface	SRDI	Access type
DELETE	Secure channel session keys	U
EXTERNAL AUTHENTICATE	Secure Channel key set Secure channel session leys	U U
GET STATUS	Secure channel session keys	U
INITIALIZE UPDATE	Secure Channel key set Secure channel session keys PRNG seed and seed key	U W U
INSTALL	Secure channel session keys	U
LOAD	Secure channel session keys	U
PUT DATA	Secure channel session keys	U
PUT KEY	Secure Channel key set Secure channel session leys	W U
SET STATUS	Secure channel session keys PRNG seed and seed key Secure Channel key set	U W W
STORE DATA	Secure channel session keys Secure Channel key set	U W
GET DATA	Secure channel session keys	U
SELECT	Secure channel session keys	W
SET ATR	Secure channel session keys	U
GET MEMORY SPACE	Secure channel session keys	U

Table 15 - Security Relevant Data Items

This document may be reproduced only in its original entirely (without revision).

6 Finite State Model

The **GX4-S – FIPS** is designed using a finite state machine model that explicitly specifies every operational and error state.

The cryptographic module includes Power on/off states, Cryptographic Officer states, User services states, applet loading states, Key/PIN loading states, Self-test states, Error states, and the GP life cycle states.

An additional document (Finite State Machine document) identifies and describes all the states of the module including all corresponding state transitions.

7 Physical Security

The **GX4-S – FIPS** single chip module is designed to meet the **FIPS140-2 level 3 Physical Security requirements**. Specifically, the module micro-module is enclosed within a hard opaque epoxy.

7.1 Manufacturing Process

The manufacturing process consist of wire bonding the ICC over printed circuit plate providing ISO contacts and sealing the chip and wires in a 'glue globe':

- Opaque black epoxy coating polymerized with temperature

Any mechanical attack attempting to extract the chip from the micro-module results in damaging the chip so that it cannot work anymore. Furthermore, attempts to attack the chip or micro-module will result in signs of tampering such as scratches and deformation.

The module is designed for embedding in a plastic card body for Smart Card manufacturing.

Note: the chip is designed in such a way that no data can be collected by visual inspection.

7.2 Hardware Security Mechanisms

The embedded **S3CC9TC chip from Samsung** provides the cryptographic module with hardware security mechanisms such as probing detection, low frequency and supply voltage monitoring. The chip reacts to a low/high clock frequency, and low/high power supply voltage by resetting the cryptographic module. Any unprotected sensitive data are lost.

7.2.1 High/Low Frequency Sensor

The external clock frequency is monitored. If it is higher than the maximum value or lower than the minimum value, a security flag is raised.

7.2.2 High/Low Voltage Sensor

The supply voltage is monitored. If it is higher than the maximum value or lower than the minimum value, a security flag is raised.

7.2.3 High/Low Temperature Sensor

The temperature is monitored. If it is higher than the maximum value or lower than the minimum value, a security flag is raised.

7.2.4 Active Shields

Shields cover some different chip areas:

The active shield covers the ROM, the EEPROM and the analog blocks such as voltage regulator, oscillator and sensors

7.2.5 Light sensor

Light sensor is in the analogic part. The light sensor is hidden by the top metal layers of the circuit and cannot be distinguished by simple observation.

7.2.6 Glitch sensor

Glitch sensor is present and monitors Vcc and Vss. When the sensor is triggered a flag is raised.

7.2.7 Filters

Filter is present on the RST (reset signal) and CLK (clock signal) lines.

7.2.8 BUS Scrambling

Logical addresses have no correlation thanks to the use of 'address scrambling' at the BUS level.

8 Operational Environment

This section does not apply to **GX4-S – FIPS**. No code modifying the behavior of the cryptographic module operating system can be added after its manufacturing process.

Only authorized applets can be loaded at post-issuance under control of the Cryptographic Officer. Their execution is controlled by the cryptographic module operating system following its security policy rules.

9 Cryptographic Key Management

9.1 Card Manager Key Set

The cryptographic module implements **GP[4]** specifications. The card issuer security domain includes key sets for card administration purposes. These key sets are used to establish a secure communication between the Card Manager applet and the Cryptographic Officer.

When the Card Manager is the selected applet, all commands besides those required to set up the secure channel must be performed within a secure channel. The one exception to this rule relates to the GET DATA APDU command that can be issued to the Card Manager without first setting up a secure channel. The SELECT and MANAGE CHANNEL APDUs are part of the platform and can be invoked by anyone.

The card life cycle state determines which modes are available for the secure channel. In the SECURED card life cycle state, all command data must be **secured by at least a MAC**. As specified in the GP specification, there exist earlier states (prior to card issuance) in which a MAC might not be necessary to send Card Manager commands. The key set associated with the secure channel is such that:

- All DES keys are double length keys (16 bytes),
- All DES operations are performed using triple DES encryption or decryption in ECB mode.
- All MAC generations result in an 8-byte field. These 8 bytes constitute the MAC.

Key sets are identified by Key Version Numbers (‘01’ to ‘7F’). The three keys within a key set version have the following different functionality:

- Secure Channel Encryption (Kenc) is used for secure channel authentication and derivation of session encryption key.
- Secure Channel Message Authentication Code Key (Kmac) is used for secure channel authentication and derivation of session MAC verification key secure channel MAC verification.
- KEY Encryption Key (KEK) is used for encryption of secret and private keys input into the module.

9.2 Secure channel session keys

After a secure channel is opened, two Triple-DES 16-byte session keys are stored in RAM. They are:

- Secure Channel Session Encryption Key (Senc) is used to decrypt incoming command APDU data and encrypt outgoing response APDU data within the secure channel.
- Secure Channel Session MAC Key (Smac) is used to verify MAC over incoming command and APDU data and compute MAC over outgoing response APDU data within the secure channel.

9.3 PRNG seed and seed key

The seed & seed key values used by the ANSI X9.31 Approved RNG are stored in EEPROM. The seed is 8-bytes while seed-key is 16-bytes

9.4 DAP public key

This 1024-bit RSA public key is stored in EEPROM and used for verifying DAP signature during firmware loads in cases where a Security Domain with DAP privilege is installed in the card

9.5 Key Generation

The cryptographic module on-board key generation is able to generate RSA key and RSA Chinese Remainder Keys. This functionality is available to on-board applets only.

9.6 Key Entry

Keys are entered in the cryptographic module using the PUT KEY or STORE DATA APDU command and under the responsibility of the applets. Non-system applets are out of the scope of this Security Policy.

The Card Manager & Security Domain applets enforce entering cryptographic Triple-DES keys securely within a secure channel.

The Cryptographic Officer or User sends the PUT KEY APDU command to:

- Replace multiple keys within an existing key set version.
- Replace an existing key set version with a new key set version.
- Add a new key set version containing multiple key(s).

9.6.1 Input Data

While the key set structure can be presented to the card in encrypted form or in plaintext, **the key values are always encrypted with the Key Encryption Key**. The key set structure includes a check value for each key in order to ensure their integrity.

9.7 Key Storage

Keys are protected against unauthorized disclosure, unauthorized modification, and unauthorized substitution.

Secret and private keys are Java objects. As a consequence, they are protected by the firewall from illegal access. An applet that owns a key is responsible for not sharing it.

RSA, Triple-DES and AES keys are stored encrypted in EEPROM (physical security of the Philips chip) and are under the protection of the firewall that prevents key from being accessed by non-authorized applets. Moreover, RSA keys are checksumed, Triple-DES and AES keys are checksumed and masked.

The Java inheritance mechanism ensures that a created Java object such as a key belongs to its owner, i.e. an applet and its execution context.

The cryptographic module stores key components according to the key type.

KEY TYPE	KEY COMPONENT
Triple-DES keys	Key value component
AES keys	Key value component
RSA keys	Public exponent **e** component Modulus **N** component Private exponent **d** component
RSA Keys CRT	Chinese Remainder **P** component Chinese Remainder **Q** component Chinese Remainder **PQ** component Chinese Remainder **DP1** component Chinese Remainder **DQ1** component

Table 16 - Key types and components mapping table

9.8 Key Zeroization

The cryptographic module provides applets with the capability to set all plaintext cryptographic keys and other unprotected critical security parameters within the module to zero. This can be done by setting the card state to TERMINATED.

10 EMI/EMC

The **GX4-S – FIPS** cryptographic module has been tested to meet the EMI/EMC requirements specified in FCC Part 15 Subpart B.

11 Self Tests

The **GX4-S – FIPS** performs the following self-tests to ensure that the module works properly.

SELF-TESTS	EXECUTION
Cryptographic algorithm test (Known-answer tests for Triple-DES, AES, SHA-1, RSA)	At Power-Up
Software/firmware integrity test.	At Power-Up
Pseudo Random Number Generator test. (Known-Answer Test for P-RNG output)	At Power-Up
Security error test	At Power-UP
Sensors test	At Power-Up
Pair-wise consistency test.	Conditional
Software load test.	Conditional
Continuous random number generator test.	Conditional

Table 17 - Self-tests list

11.1 Self-Test Execution

After **GX4-S– FIPS** is powered up and before executing any APDU commands, the module enters the self-test state and performs all of the cryptographic algorithm and software integrity self-tests as specified in FIPS 140-2 standard **[1]**. In addition to those tests, it also performs chip sensors verification and security status verification:

- **Sensors test:** at startup, the card detects if a hardware security error has been held during the previous session. If so, the card enters a mute state.
- **Security errors test:** at startup, if a pre-defined number of security errors is reached, the card is terminated as per Global Platform specifications. The Get Data command is the only command that remains available.

These tests are conducted automatically as part of the normal functions of the cryptographic module. They do not require any additional operator intervention, nor applet specific functions.

Power-up self-tests are executed upon reset. The cryptographic module start-up process has been designed in such a way that it cannot be bypassed. This enforces the execution of the self-tests before allowing any use and administration of the module, thus guaranteeing a secure execution of the module cryptographic services.

If these self-tests are passed successfully, the cryptographic module returns the status words relating to the requested APDU command via the status interface and incoming APDUs are processed.

All data output via the output interface are inhibited while any power-up and conditional self-test is running.

Resetting the cryptographic module, provides a means by which the operator can repeat the full sequence of power-up operating tests.

11.2 Self-Test Failure

No cryptographic operations can be processed and no data can be output via the data output interface, while in the error state.

If an error occurs during the **SW load self-test**, an error code is returned via the status interface and the secure channel is closed (loading is aborted).

If an error occurs during another self-test, the card outputs an error code & then enters a mute state where no more command can be performed. The behavior of the card depends on error:

- Cryptographic algorithms tests, integrity test, internal error counter is incremented, the card returns an error status before becoming mute.
- Conditional self-tests (PRNG continuous test and pair wise consistency test), internal error counter is incremented, and the card returns an error status before becoming mute.

An error while loading an applet closes the secure channel with the Card Manager. It shall be re-opened, to retry applet loading: the Cryptographic Officer has to be re-authenticated. If the internal error counter reaches maximum allowed value the card is TERMINATED.

To recover from mute state, the card must be reset.

12 Design Assurance

12.1 Configuration Management

The **GX4-S – FIPS** is designed and developed using a configuration management system that is clearly ruled and operated.

An additional document (Configuration Management Plan document) defines the methods, mechanisms and tools that allow to identify and place under control all the data and information concerning specification, design, implementation, generation, test and validation of the card software all along the development and validation cycle.

12.2 Delivery and Operation

The **GX4-S – FIPS** is designed and developed using a configuration management system that is clearly ruled and operated.

Some additional documents ('Delivery and Operation', 'Reference Manual' and 'Card Initialization Specification' documents) define and describe the step necessary to deliver and operate securely the **GX4-S– FIPS**.

12.3 Development

12.3.1 Security Rules and Design

The following table indicates the design documents implementing the cryptography module security rules.

RULE IDENTIFIER	DESIGN DOCUMENT
IA_PIN_RULE.1 IA_PIN_RULE.2	**SDD2 – JKernel / API for JLEP 2** – javacard.framework component
IA_CO_RULE.1 IA_CO_RULE.2 IA_CO_RULE.3	**Visa Global Platform – Release 2.1.1** – chapter "Secure Communication" **User Guidance**
AC_CO_RULE.1	**Global Platform – Release 2.1.1** – Chapter "Card Manager" **User Guidance** **SDD2 – JKernel / OPEN for JLEP 2** – Chapter "SD/ISD component"
AC_JAVA_RULE.1	**SDD2 – JKernel / VM for JLEP 2** **SDD2 – JKernel / JCRE for JLEP 2**
AC_LIFE_RULE.1	**Global Platform – Release 2.1.1** – Chapter "Card Manager"
AC_LIFE_RULE.2	**Global Platform – Release 2.1.1** – Chapter "Life cycle models"
AC_LIFE_RULE.3	**Global Platform – Release 2.1.1** – Chapter "Life cycle models"
AD_RULE.1 AD_RULE.2 AD_RULE.3 AD_RULE.4 AD_RULE.5 AD_RULE.6	**User Guidance**

Table 18 - Security rules and development documentation mapping

12.4 Guidance Documents

Guidance document to be provided with **GX4-S – FIPS** is intended to be the 'Reference Manual'. Such a document is designed to allow a secure operation of **GX4-S – FIPS** by its users as defined in the '**Roles, Services and Authentication'** chapter and in the scope of the SP boundaries:

- Cryptographic Officer,
- User

To determine the Approved mode of operation the CO & User must follow the rules in Appendix B of this document.

13 Mitigation of Other Attacks

The GX4-S – FIPS has been designed to mitigate the following attacks:

- Timing Attacks,
- Differential Power Analysis,
- Simple Power Analysis,
- Electromagnetic Analysis,
- Fault Attack.
- Card Tearing

A separate and proprietary document describes the mitigation of attacks policy provided by the GX4-S FIPS platform.

14 Appendix A – GP Specification

This chapter provides correspondences between the cryptographic module APDU commands and the GP specifications.

APDU COMMAND	DOCUMENTATION: GP SPECIFICATION 2.1.1	
DELETE	CHAPTER 9	SECTION 2
EXTERNAL AUTHENTICATE	APPENDIX D	SECTION 4 (SCP 01)
EXTERNAL AUTHENTICATE	APPENDIX E	SECTION 5 (SCP 02)
GET DATA	CHAPTER 9	SECTION 3
GET STATUS	CHAPTER 9	SECTION 4
INITIALIZE UPDATE	APPENDIX D	SECTION 4 (SCP 01)
INITIALIZE UPDATE	APPENDIX E	SECTION 5 (SCP 02)
INSTALL	CHAPTER 9	SECTION 5
LOAD	CHAPTER 9	SECTION 6
MANAGE CHANNEL	CHAPTER 9	SECTION 7
PUT DATA	CHAPTER 4	SECTION 12 (OP DEPRECATED COMMAND – SEE OP 2.0.1' SPECIFICATION)
PUT KEY	CHAPTER 9	SECTION 8
SELECT	CHAPTER 9	SECTION 9
SET STATUS	CHAPTER 9	SECTION 10
STORE DATA	CHAPTER 9	SECTION 11

Table 19 - Correspondence between APDU commands and GP Specifications

The **constraints of use** for each APDU commands are described in subsection 1 "Definition and scope".

The correct values of the **APDU parameters** (P1, P2, LC, and LE) are described in subsection 2 "Command message".

The **conditions of use** of the APDU commands correspond to the authorized sequences of APDU commands.

15 Appendix B – Identification and FIPS mode :

1. CPLC data element can be read with a Get Data command (tag 9F7Fh):
   In the FIPS mode, the first 6 bytes of the CPLC data (tag 9F 7Fh) must be :
   IC Fabricator – 42 50h
   IC Type – 30 72h
   Operating System Identifier: 12 91h
   Operating System release level: 05 00h
   These values identify clearly the **version GXP4-M2612410 and firmware version GX4-S_E005 (MSA029)** of the validated module**.**

2. The flow Identification byte must be **Dx** value to indicate **FIPS configuration**.
   This byte can be retrieving issuing a Get Data using tag 01 01h.
   The tag 01 01h can be broken down as follows:
   Card serial number: 8 bytes
   Reserved bytes: 3 bytes
   Flow identification: 1 byte
   Reserved bytes: 4 bytes

3. A card in **FIPS configuration** must have following historical bytes T5-T9 in the ATR :

T5 = FMN	**B0h**	Gemalto Family Name – ***JavaCard financial/e-business***
T6 = PRN	**83 h**	Gemalto Product Name – ***GX4***
T7 = OSV	**11 h**	Gemalto OS Version – ***JLEP 2 Mask Number 05***
T8 = PRV	**D0 h**	Gemalto Program Version – ***0***
T9 = CID	**A9h**	Gemalto Chip Identifier – ***Samsung S3CC9 TC***

4. "SECURED" must be the required state for card delivery outside Gemalto.
   The CO must check the state to ensure it is OP_SECURED to be in **FIPS mode**. If card state is not OP_SECURED both CO and User must open secure channel in atleast MAC mode.

- END OF DOCUMENT -