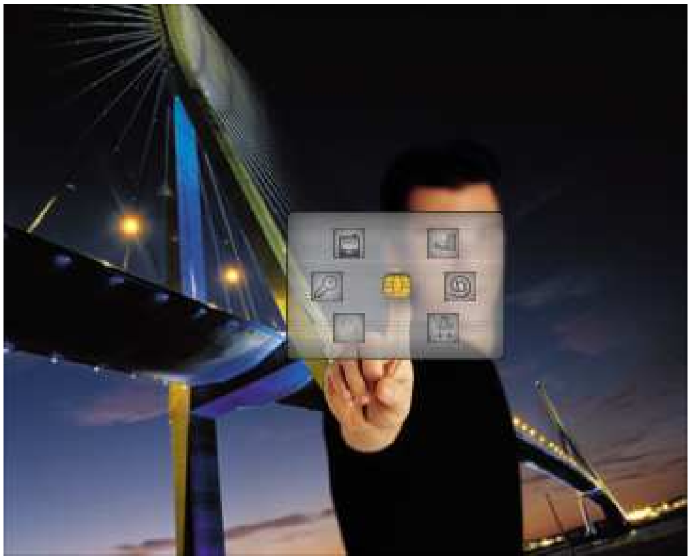
# SafesITe FIPS 201 Applet on

# SafesITe PIV TPC DM Card

# Security Policy



| | |
|---|---|
| TITLE | SafesITe FIPS 201 Applet on SafesITe PIV TPC DM Card<br>**- Security Policy** |
| REF. | **SP01R10610 – 08** |
| DATE: | **22/05/08** |

# TABLE OF CONTENTS

## Table of figures:

# 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)

**[13]** NIST Special Publication 800-73-1
Interfaces for Personal Identity Verification - Information Security – March 2006

**[14]** FIPS PUB 201 - Federal Information Processing Standards Publication
Personal Identity Verification of Federal Employees and Contractors - February 25, 2005

# 1 Scope

This Security Policy specifies the security rules under which the SafesITe FIPS 201 Applet on SafesITe PIV TPC DM Card, herein identified as the **"GCX4-PIV II – FIPS"** product, 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).

The commercial name of the product is:

SafesITe FIPS 201 Applet on SafesITe PIV TPC DM Card

Where:

- SafesITe PIV TPC DM is a Java platform available in two versions. This Java Card platform may also be referred as "GCX4" or "GXP4" in this document.
- SafesITe FIPS201 applet is an applet loaded on the Java Card platform. This applet may also be referred as "PIV applet" or "PIV-II Applet" in this document.

# 2 Introduction

## 2.1 GEMALTO Smart Card Overview

GEMALTO aims to provide **FIPS140-2 Level 2** cryptographic smart cards. Together, the card and applets provide authentication, encryption, and digital signature cryptographic services. This **whole product**, made up of the GEMALTO platform and the PIVII applet is aimed to reach FIPS 140-2 L2 compliance. The present document is dedicated and focused on both the GEMALTO GCX4 platform and the GEMALTO PIVII applet.

This security policy specifies the security rules under which our Java Card **GCX4 platform and our PIVII applet** operate.

## 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.

The PIVII applet doesn't implement any cryptographic services. But when needed the applet uses cryptographic services provided by the card platform.
The platform 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 standard[8]**.

The card life cycle is managed according to the **Global Platform (GP) specification**. Issued cards have been loaded with a set of applets, cryptographic keys, and a PIN, and are moreover in the "SECURED" state. The security implementation is fully compliant with the **Global Platform (GP) 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, on-board RSA Key generation, Triple-DES CBC and ECB and AES ECB and CBC algorithms.

## 2.3 Security Level

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

| Security Requirements Section | Security Level |
|---|---|
| Cryptographic Module Specification | 2 |
| Cryptographic Module Ports and Interfaces | 2 |
| Roles, Services and Authentication | 3 |
| Finite State Model | 2 |
| Physical Security | 3 |
| Operational Environment | N/A |
| Cryptographic Key Management | 2 |
| EMI/EMC | 3 |
| Self-Tests | 2 |
| Design Assurance | 3 |
| Mitigation of Other Attacks | 2 |

**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 'ICC micro-module edge' of the **GCX4-PIVII – FIPS** a set of "embedded" hardware and firmware that implements cryptographic functions and processes, including cryptographic algorithms, key generation and applications services. **GCX4-PIVII – FIPS** 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 dual interfaces (i.e. contact and contact-less) where the same security level is achieved. The card is designed in following configurations:

***GCX4:*** This is a dual interface card providing both contact and contactless interfaces. This card has hardware version GCX4-M2569420, GCX4-M2569422, or GCX4-A1004155 (hardware module depending) and firmware versions **GCX4-FIPS EI07 (MPH051)** and **GCX4-FIPS EI08**. It is identified by three historical bytes that are present in ATS (TH8, TH9, TH10) and ATR (T6, T7, T8) having same respective values. These three bytes should be:

- 83h 11h 11h : for the configuration where RSA is supported in contactless mode
- 83h 11h 10h : for the configuration where RSA is not supported in contactless mode

***GXP4:*** This is a contact only card. This card has hardware version GXP4-M2569430 and firmware versions **GXP4-FIPS EI07 (MPH052)** and **GXP4-FIPS EI08**. It is identified by historical bytes T6 T7 T8 in the ATR that should be:

- 83h 11h 50h

Depending on the market and the end-customer requirements, either contact or contact-less interfaces can be disabled during the manufacturing. Moreover, for the contact-less interface, Public Key (PK) support (i.e. PK enbabled or PK disabled) can be also configured during the manufacturing depending on market and the end-customer requirements. This results in three configurations described below. All three configurations were FIPS 140-2 tested.

- **CONFIGURATION 1**: The product is initialized in dual interface mode; it means that both contact and contact-less mode are operated, with FIPS PK self-tests and PK service enabled.
- **CONFIGURATION 2**: The product is initialized in dual interface mode; it means that both contact and contact-less mode are operated, with FIPS PK self-tests and PK service enabled in contact mode and without FIPS PK self-tests and so no PK services in contactless mode.
- **CONFIGURATION 3**: The product is initialized in contact mode only, with FIPS PK self-tests and PK service enabled.

The following table gives an overview of those 3 different configurations regarding contact-less and PK support.

| | CONTACT-LESS | PK SUPPORT IN CONTACTLESS MODE |
|---|---|---|
| CONFIGURATION 1 | Yes | Yes |
| CONFIGURATION 2 | Yes | No |
| CONFIGURATION 3 | No | Yes |

**Table 2 – Contact-less and PK support configurations**

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.**

All the components of **GCX4-PIVII – FIPS** that are included in the cryptographic module boundaries, are those as shown in the following figure:

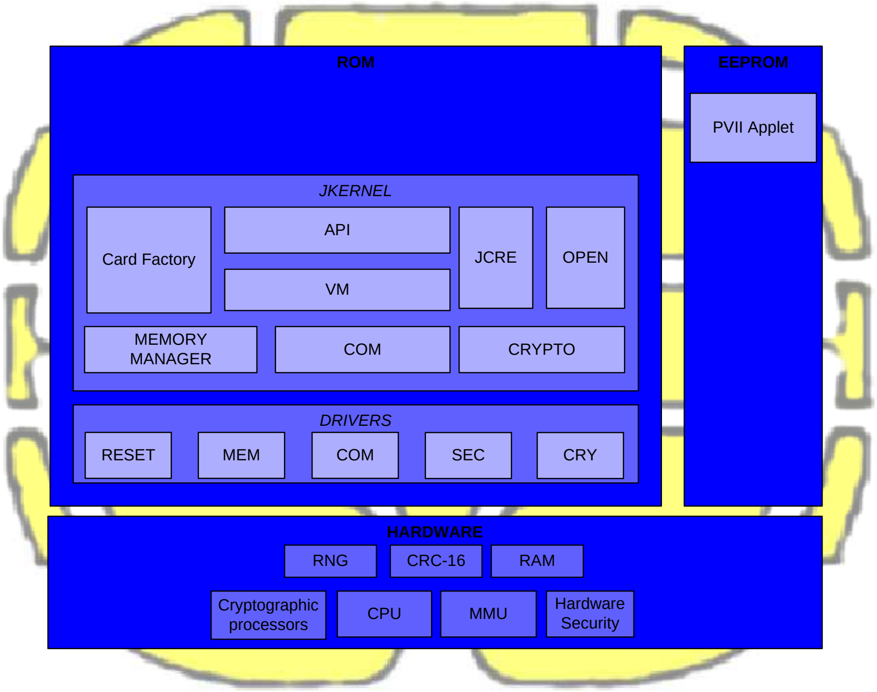


**Figure 1- Cryptographic Module Boundary**

## 3.2 Language level

The scope of this security policy is focused both on the Java Card Platform and on the PIVII applet (in eeprom).
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.
The application code "Applet" is designed in Java Card language that is a high level language. The applet code is complying with Java card code verifier that insures compliance with language rules.

## 3.3 FIPS Approved Security Functions

The following table gives the list of FIPS approved security functions that are provided by the **GCX4-PIVII – 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 following X9.31 | Yes |
| | Signature following PKCS#1with SHA-1 hashing | Yes |
| | Verification following PKCS#1with 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 |
| **Triple-DES MAC** | ECB and CBC modes | Yes |

**Table 3 – FIPS Approved Security Functions**

FIPS approved security functions used specifically by the **PIVII applet** are:

- **Triple-DES**
- **SHA-1**
- **RSA**
- **P-RNG**

# 4 Cryptographic Module Ports and Interfaces

The **GCX4-PIVII – FIPS** 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:

**GCX4-PIVII – FIPS** follows the standards **"ISO 7816-1 Physical characteristics" [10]** and **"ISO 7816-2 Dimensions and contact location" [10]**.

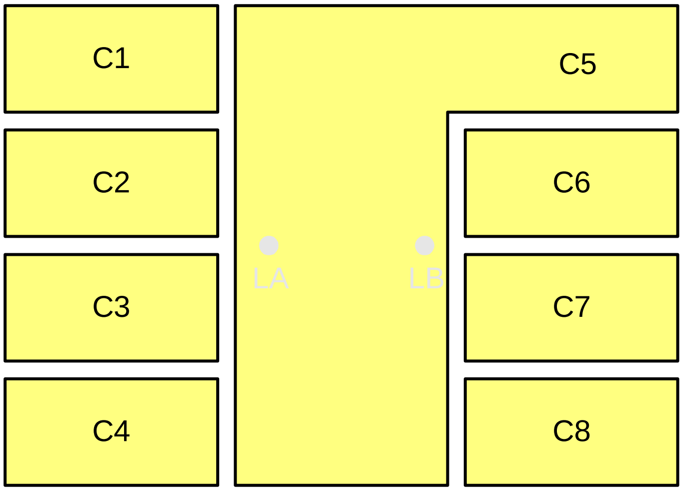


**Figure 2 - Contact plate example – Contact physical interface**

| 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**

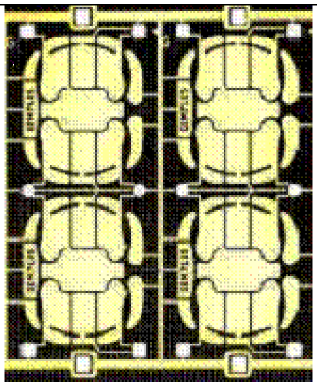
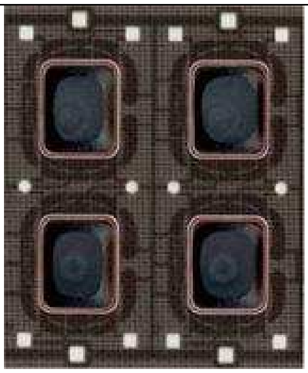
### 4.1.2 Conditions of use

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

| Conditions | Range |
|---|---|
| Voltage | 3 V and 5.5 V |
| Frequency | 1MHz to 10MHz |

**Table 5 - Voltage and frequency ranges**

Pictures – Contact Mode

| **MIND-L Thermal black resin technology**<br>**Hardware version : GXP4-M2569430** | |
|---|---|
|  |  |
| **MIND-L** design[1] | **Thermal** black resin Technology |

## 4.2 Physical Port – Contact-less mode

### 4.2.1 Contacts assignments

In the contact-less mode the GCX4-PIVII FIPS cryptographic module follows the standard **"ISO 14443 RF Interface" [12]** and only uses two connections that are physically different and distinct from the connections used in the contact mode. Those electrical connections, LA and LB, are placed on the module backside and are used to connect an external **antenna loop that is not within the cryptographic boundaries of the module.**

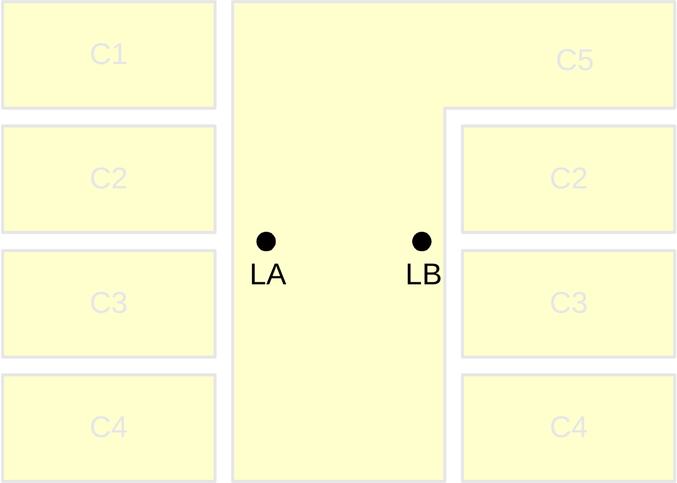


**Figure 3 - Contact plate example - Contact-less antenna contacts**

[1] The contact plate of the module may not be marked "GEMALTO". This cosmetic feature is not security relevant.

| Contact No. | Assignments | Contact No. | Assignments |
|---|---|---|---|
| LA | Antenna coil connection | LB | Antenna coil connection |

**Table 6- Contact plate pin list – Contact-less mode**

### 4.2.2 Condition of uses

The radiofrequences and transmission protocols follow the **"ISO 14443 RF Interface" [12].** The conditions of use are the following:

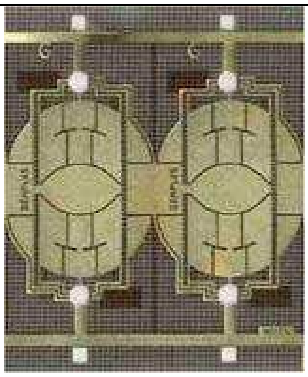
| Conditions | Range |
|---|---|
| Supported bitrate | 106 Kbits/s, 212 Kbits/s and 424 Kbits/s |
| Operating field | Between 1.5 A/m and 7.5 A/m rms |
| Frequency | 13.56 MHz +- 7kHz |

**Table 7 - Voltage and frequency ranges**

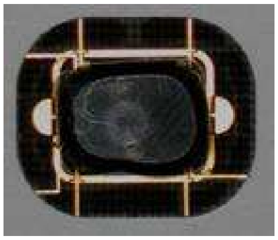
## Pictures – Dual Mode

Two types of modules are used :

1/ Hardware version GCX4-M2569420

| **GEM Combi Thermal black resin process, contact and contactless technology**<br>**Hardware version : GCX4-M2569420** | |
|---|---|
|  | |
| Gem combi design[2]<br>**Hardware version : GCX4-M2569420** | **Thermal** black resin Technology<br>**Hardware version : GCX4-M2569420** |

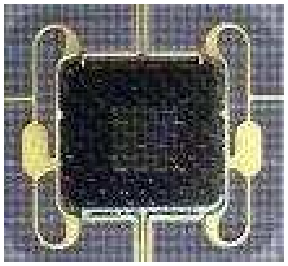
2/ Hardware version GCX4-M2569422

| **GEM Combi Thermal black resin process, contact and contactless technology**<br>**Hardware version : GCX4-M2569422** | |
|---|---|
| |  |
| Gem combi design<br>**Hardware version : GCX4-M2569422** | **Thermal** black resin Technology<br>**Hardware version : GCX4-M2569422** |

3/ Hardware version GCX4-A1004155

| **GEM Combi Thermal black resin process, contact and contactless technology**<br>**Hardware version : GCX4-A1004155** |
|---|

[2] The contact plate of the module may not be marked "GEMPLUS". This cosmetic feature is not security relevant.

| | |
|---|---|
| |  |
| Gem combi design<br>**Hardware version : GCX4-A1004155** | **Thermal** black resin Technology<br>**Hardware version : GCX4-A1004155** |

## 4.3 Logical Interface

**GCX4-PIV II – FIPS** provides services to both external devices and internal applets as the PIV II and Card Manager applets.
External devices have access to services by sending APDU commands while internal applets as PIV II applet have access to services through internal API entry points.
The cryptographic module provides an execution **sandbox for the PIV II applet** and performs the requested services according to its roles and services security policy.

For security reasons, **GCX4-PIVII – FIPS** inhibits all data output via the data output interface when an error state is reached and during self-tests.

# 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.

### 5.1.2 Identity based authentication policy

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.
The following table describes the roles associated to the Cryptographic Module:

| **Cryptographic Officer Role** | **Description** |
|---|---|
| **Cryptographic Officer (CO)** | This role is responsible for managing the security configuration of the card manager and security domains. The CO role authenticates to the cryptographic module by demonstrating to the Card Manager or PIV II application knowledge of a GP secure channel TRIPLE-DES key set stored within the Card Manager. By successfully executing the GP secure channel mutual authentication protocol, the CO role establishes a secure channel to the Card Manager and execute services allowed to the CO role in a secure manner. |
| **PIV Card Application Administrator** | The PIV Card Application Administrator role represents an external application requesting the services offered by the PIV II applet. An applet authenticates the Application Operator role by verifying possession of the Application External Authenticate (XAUT) TRIPLE-DES key |
| **User Role** | **Description** |
| **Card Holder role** | The Card Holder role is responsible for ensuring the ownership of his cryptographic module, and for not communicating his PIN to other parties. The PIV II applet authenticates the Card Holder by verifying the PIN value. |
| **Card Holder II role** | The Card Holder II role is responsible for unblocking and/or changing the Card Holder PIN. The PIV II authenticates the Card Holder II by verifying the PIN value. |
| **Maintenance Role** | **Description** |
| **None** | |

**Table 8 - Role profile definitions**

### 5.1.3 Mechanism Interfaces

The following tables describes the mechanisms for authentication of the roles:

| Interface | Description |
|---|---|
| **INITIALIZE UPDATE**<br>*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**<br>*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 9 - Mechanism interfaces in personalization and applicative phase**

| Interface | Description |
|---|---|
| **GENERAL AUTHENTICATE**<br>*APDU* | The APDU command is used to perform a cryptographic operation such as an authentication protocol using the data provided in the data field of the command and returns the result of the cryptographic operation in the response data field.<br>The GENERAL AUTHENTICATE command shall be used to authenticate the card or a card application to the client-application (INTERNAL AUTHENTICATE), to authenticate an entity to the card (EXTERNAL AUTHENTICATE), and to perform a mutual authentication between the card and an entity external to the card (MUTUAL AUTHENTICATE).<br>The GENERAL AUTHENTICATE command shall be used to realize the signing functionality on the PIV client-application programming interface. |
| **VERIFY**<br>*APDU* | This APDU command initiates the comparison in the card of the reference data with data field of the command.<br>The referenced PIN must be successfully verified |

**Table 10 - Mechanism interfaces in applicative phase**

### 5.1.4 Security rules

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

| Rule Identifier | Description |
|---|---|
| IA_PIN_RULE.1 | It is not possible to get authenticated through the PIN authentication mechanism if the authorized number of attempts is reached. |
| IA_PIN_RULE.2 | It is not possible to get authenticated through the PIN authentication mechanism if the referenced PIN is not found |
| IA_PIN_RULE.3 | It is not possible to get authenticated through the PIN authentication mechanism if the submitted PIN is incorrect |
| IA_PIN_RULE.4 | The pin must be re-authenticated if the card is reset |
| IA_PIN_RULE.5 | The pin must be re-authenticated if a new application is selected on the same channel |
| IA_PIN_RULE.6 | The pin remains active if another application is selected on another channel |
| IA_PIN_RULE.7 | The PIN length must be 8 characters. |
| | |
| ia_co_rule.2 | The Cryptographic Officer must be re-authenticated if the card is reset. |
| ia_co_rule.3 | The Cryptographic Officer must be re-authenticated if the cryptographic module detects a secure messaging corruption. |

**Table 11 - Security rules**

### 5.1.5 Strengths of Authentication Mechanisms:

| **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). |
| PIN verification | $\left(\frac{1}{62^{8}}\right)$ |
| | Pin verification is the responsibility of the PIVII applet (alpha-numeric characters are allowed) that defines and maintains its own security policy regarding PIN but uses the PIN management services provided by the platform. |
| Card Application Administrator authentication | $1/2^{168}$ |
| | CAA authentication is the responsibility of the PIVII applet using External Authenticate option of the GENERAL AUTHENTICATE command that involves verifying decryption of an 8-byte challenge using the secret key. |

**Table 12 - 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.

List of the security related process or mechanisms specified for the PIVII applet during the applicative life cycle :

- **Secure messaging** : It is possible to open a secure channel during the personalization phase of the applet (between the personalization device and the card, when the applet is in the SELECTABLE state) by using the security domain of the java platform. Opening of this secure channel is necessary to perform the initial personalization (pre-personalization) of the PIV Applet. Once this initial PIV Applet pre-personalization is completed, the applet is in Application mode.
In Application mode opening of a secure channel is optional. A secure channel may be part of access conditions to a particular object in which case it becomes necessary to access that object.

- **Access Conditions :** Each object stored in the card embeds its own access conditions. These conditions defines the minimum security required to access to the object. As the access to the object is done through a command, a security condition is defined for each command accessing the object.

An **Access Rule** is encoded with an **Access Mode byte**, followed by one or more **Security Condition bytes**

The PIV Data objects Access management rules:

* **Free (always)**: No access condition.
* **Never**: No execution possible.
* **PIN**: The referenced PIN must be successfully verified. This flag is set until an incorrect PIN verification or an application selection or a reset.
* **PIN Always**: The referenced PIN must be successfully verified by the previous command.
* **Authentication**: The external authentication (using general authenticate command) must have been successfully performed with the referenced key. The authentication flag is set until a new successful authentication, an application selection or a reset.
* **Secure Channel (SM)**: A Secure Channel in MAC+ Encrypt mode must be opened.

Secure Messaging During Personalization phase :

- The Card Manager through API used by PIV II personalization provides the secure messaging.
  In a GP 2.1.1 card, the secure messaging is initiated after a mutual authentication. It means that INITIALIZE UPDATE **and** EXTERNAL AUTHENTICATE **commands have been successfully executed.**
  Secure channel can have four following modes:
  - Mutual Authentication required before attempting any command: **AUTHENTICATION**.
  - All commands require a previous Mutual authentication and must be sent with Integrity (and/or Authentication): **MAC** mode.
  - All commands require a previous Mutual authentication and must be in **MAC** & **ENCRYPTION** mode.
- When in application mode only MAC & Encrypted  mode is possible.

## 5.3 Services

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

**When the Card Manager applet is selected the following commands are available :**

| 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:<br>1. Replace a single or multiple keys within an existing key set version;<br>2. Replace an existing key set version with a new key version;<br>3. Add a new key set version containing a single or multiple keys<br>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. |

**Table 13 – System applet  Interfaces and services**

**When PIVII applet is selected the following commands are available :**

* APDU not available in contactless mode

| Interface | Service Description |
|---|---|
| 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. |
| 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. |
| MANAGE CHANNEL - *APDU* | This command is used to open and close supplementary logical channels. |
| END PERSONALIZATION – *APDU* | The APDU command is used to end the personalization step. |
| VERIFY*– *APDU* | The APDU is used to initiate the comparison in the card of the reference data indicated with authentication data in the data field of the command. |
| GET DATA – *APDU* | This APDU command retrieves the data content of the single data object whose tag is given in the data field. The entire object is returned. |
| GENERAL AUTHENTICATE – *APDU* | The APDU command performs a cryptographic operation such as INTERNAL AUTHENTICATE, EXTERNAL AUTHENTICATE, MUTUAL AUTHENTICATE<br>The GENERAL AUTHENTICATE command shall be used to realize the signing functionality on the PIV client-application programming interface. |
| GENERATE ASYMETRIC KEY PAIR* – *APDU* | The APDU command initiates the generation and storing in the card of the reference data of an asymmetric key pair, i.e., a public key and a private key. The public key of the generated key pair is returned as the response to the command. |
| CHANGE REFERENCE DATA* – *APDU* | The APDU command initiates the comparison of the verification data with the current value of the reference data and if this comparison is successful replaces the reference data with new reference data. |
| RESET RETRY COUNTER* – *APDU* | The APDU command resets the retry counter of the key reference to its initial value and changes the reference data associated with the key reference. The command enables recovery of the PIN card application in the case that the cardholder has forgotten a PIV Card Application PIN. |
| PUT DATA* – *APDU* | During the personalization the APDU command is used to create and/or update Data Objects, PIN, Triple-DES secret keys, RSA private keys & property template. |
| SELECT – *APDU* | |
| | The ADPU command is used to select an application |

**Table 14 – PIV II applet Interfaces and services**

| | Cryptographic officer role | Card Application Administrator role | Card Holder role | Card Holder II role | Unauthenticated role |
|---|---|---|---|---|---|
| DELETE | X | | | | |
| EXTERNAL AUTHENTICATE | X | | | | |
| GET DATA (Card Manager) | X | X | X | X | X |
| GET STATUS | X | | | | |
| INITIALIZE UPDATE | X | | | | |
| INSTALL | X | | | | |
| LOAD | X | | | | |
| MANAGE CHANNEL | X | X | X | X | X |
| PUT DATA (Card manager) | X | | | | |
| PUT KEY | X | | | | |
| SELECT | X | X | X | X | X |
| SET STATUS | X | | | | |
| STORE DATA | X | | | | |
| GET DATA (PIV Applet) | X | X | X | X | X |
| PUT DATA (PIV applet) | | X | | | |
| CHANGE REFERENCE DATA | | | X | | |
| END PERSONALIZATION | X | | | | |
| GENERAL AUTHENTICATE | | X | X | | |
| GENERATE ASYMETRIC KEY PAIR | | X | | | |
| RESET RETRY COUNTER | | | | X | |
| VERIFY | | | X | | |

**Table 15 – Authenticated and unauthenticated role accorded interfaces and services**

### 5.3.1 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.2 | An **applet** is responsible for managing its own life cycle state, in accordance with the GP specification. |
| 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 16 - Security rules**

## 5.4 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 support a multiple concurrent operators. |
| 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 17 - GEMALTO additional security rules**

## 5.5 Security Relevant Data Item

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

- **GP key set of the Card Manager**
- **Secure channel session key**
- **Card Holder PIN**
- **Card Holder II PIN**
- **The PIV authentication key**
- **The PIV card application authentication key**
- **The PIV card application digital signature key**
- **The PIV card application key management key**
- **PRNG Seed and seed key**

The following table defines 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 | GP key set of the Card Manager<br>Secure channel session leys | U<br>U |
| GET STATUS | Secure channel session keys | U |
| INITIALIZE UPDATE | Secure channel session keys<br>PRNG seed and seed key | U<br>U |
| INSTALL | Secure channel session keys | U |
| LOAD | Secure channel session keys | U |
| PUT DATA | Secure channel session keys<br>PIV card application authentication key<br>PIV card application key management key | U |
| PUT KEY | GP key set of the Card Manager<br>Secure channel session leys | W<br>U |
| SET STATUS | Secure channel session keys | U |
| STORE DATA | Secure channel session keys | U |
| GENERAL AUTHENTICATE | PIVII keys | U |
| VERIFY | Card Holder PIN | U |
| RESET RETRY COUNTER | unblocking PIN (Card Holder II PIN)<br>Card Holder PIN | U<br>W |
| CHANGE REFERENCE DATA | Card Holder PIN | W<br>U |
| GENERATE ASYMMETRIC KEY PAIR | PIV II keys<br>Card Holder PIN | W<br>U |

**Table 18 - Security Relevant Data Items**

# 6 Finite State Model

The **GCX4-PIVII – 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 for both platform and PIVII applet.

# 7 Physical Security

The **GCX4-PIV II – FIPS** single chip module is designed to meet the **FIPS140-2 level 3 Physical Security requirements**.

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.

# 8 Operational Environment

This section does not apply to **GCX4 – PIV II– 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 Keys

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 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 (before 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 Triple-DES keys are double length keys (16 bytes),
- All Triple-DES operations are performed using Triple-DES encryption or decryption.
- All Triple-DES 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 keys within a key set version have the following different functionality:

- Secure Channel Encryption (K-Enc) is used for generation of keys used for secure channel encryption.
- Secure Channel Message Authentication Code Key (K-Mac) is used for generation of keys used for secure channel MAC verification.
- Data Encryption Key (DEK) is used for sensitive data encryption.

**Secure Channel session keys (each key is 16-bytes):**
The Secure Channel session keys are generated as per the GP specifications using random challenge values and Card Manager Key Set.

- $S_{enc}$: used to encrypt command and response APDU data encrypted mode of the secure channel to provide message confidentiality.
- $S_{mac}$: used to MAC command and response APDU data in MAC mode of the secure channel to provide message integrity.

**DAP Public key:** The 1024-bit DAP public key used for verifying loading of applets is also managed by the Card Manager applet.

**PRNG Seed and seed key:** These are CSPs used in the ANSI X9.31 RNG. They are stored in EEPROM across power-cycles and in RAM during module execution.

## 9.2 PIV II Application Keys

**PIV II applet** use keys of the following key types through the cryptographic services of the module :
Triple-DES Keys, RSA public and private keys

Appendix C gives a table representation of PIV-II Applet's Keys and associated access conditions as installed in the product. Is also given for information list of data objects managed by the applet and their respective access conditions.

### 9.2.1 PIV II Applet Key management:

The PIV II applet manages five types of keys through the platform cryptographic services:

- The ***PIV authentication key*** : This key (asymmetric RSA) is generated on the card. This key is used to support card authentication for an interoperable environment, and it is a **mandatory non exportable key.**
  This key shall be generated on the PIV Card. The PIV Card shall not permit exportation of the PIV authentication key. The PIV authentication key must be available only through the contact interface of the PIV Card. Private key operations may be performed using an activated PIV Card without explicit user action (e.g., the PIN need not be supplied for each operation).

- The ***PIV card application administration key*** *:* This key is a symmetric Triple DES key. It may be used for personalization and post-issuance activities. The PIV Card shall not permit exportation of the card authentication key. This key shall be imported to the card and allows authentication of the Card Application Administrator..

- The ***PIV card application digital signature key*** : This key (asymmetric RSA) may support document signing.
  The PIV digital signature key shall be generated on the PIV Card. The PIV Card shall not permit exportation of the digital signature key. If present, cryptographic operations using the digital signature key may only be performed using the contact interface of the PIV Card. Private key operations may not be performed without explicit user action.

- The ***PIV card application key management key*** : This key (asymmetric RSA)may support key establishment and transport. This Key may be used as an encryption key. This key may be generated on the PIV Card or imported to the card. If present, the key management key must only be accessible using the contact interface of the PIV Card. This key is sometimes called an encryption key or an encipherment key.

- The ***PIV card authentication key*** : This key (asymmetric RSA) may be used for physical access control. The PIV card authentication key shall be generated on the PIV Card. The PIV Card shall not permit exportation of the card authentication key.

### 9.2.2 PIV II Applet security domain

It is possible to open a secure channel during the personalization phase and also application mode of the PIV II applet by using the security domain of the java platform. During the personalization, the applet restricts the use of authentication mechanism, defined in GP Only the mode 3 is allowed when the Card Manager state is "SECURED", and modes 1, 2 and 3 if Card Manager state is "INITIALIZED" or "OP_READY".
During Application mode only mode 3 is allowed. In mode 3 the Secure Channel must be MAC+ ENCRYPT.

## 9.3 Key Generation

The cryptographic module on-board key generation is able to generate RSA key and RSA Chinese Remainder Keys. Strong prime numbers are generated in compliance with X9.31 standard.

For the **PIV II applet asymmetric keys** , the card stores a corresponding X.509 certificate.
The PIV Card imports and stores a corresponding X.509 certificate to support validation of the corresponding private key.

Keys are generated in the cryptographic module using the GENERATE ASSYMETRIC KEY PAIR command.

## 9.4 PIV-II Application Key Entry

Keys are entered in the cryptographic module using the PUT DATA APDU command of the PIV II applet and with the authentication of Card Holder, Card Application Administrator or Crypto Officer. The PIV-II applet ensures that Secure Channel is MAC+ENCRYPT so that keys are entered in encrypted form.

### 9.4.1 Input Data

The PIVII applets key set structure are presented to the card in plaintext. The key set structure includes a check value for each key in order to ensure their integrity.

## 9.5 Card Manager Key Entry

The Card manager applet provides the Put key APDU to replace the Card Manager keyset. This service is only available to the Crypto Officer. The Card Manager enforces entering cryptographic Triple-DES keys securely within a secure channel. The Card Manager key set already present within the cryptographic module is the default key set. If this key set version is replaced, the replacement becomes the default.

## 9.6 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.
Triple-DES keys are stored in the 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 keys are checksumed and masked . All keys are stored in plaintext in the module.

The Java inheritance mechanism ensures that a created Java object such as a key belongs to its owner, that is 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 |
| RSA Keys pair | Private portion in CRT (Chinese remainder theorem):<br>Chinese Remainder **P** component<br>Chinese Remainder **Q** component<br>Chinese Remainder **PQ** component<br>Chinese Remainder **DP1** component<br>Chinese Remainder **DQ1** component<br>Public portion<br>Public exponent e component<br>Modulus N component |

**Table 19 - Key types and components mapping table**

The PIN is a critical security parameter that implements the JavaCard OwnerPin class.

## 10 EMI/EMC

The **GCX4 – PIV II – FIPS** cryptographic module has been tested to meet the EMI/EMC requirements specified in FCC Part 15 Subpart J, Class B.

# 11 Self Tests

The **GCX4 – FIPS** platform performs the following self-tests to ensure that the module works properly. All the self tests are done by the platform.

| SELF-TESTS | EXECUTION |
|---|---|
| Cryptographic algorithm test<br>(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.<br>(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 20 - Self-tests list**

## 11.1 Self-Test Execution

After **GCX4-PIV II– 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 after the first APDU command is issued. 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's 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 enters a state where no more command can be performed. The behavior of the card depends on error:

- **Severity level 1 error:**
  - integrity test, internal error counter is incremented, the card returns an error status before becoming mute.
- **Severity level 2 error:**
  - cryptographic algorithms tests, 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, the card returns an error status before becoming mute.

When the internal error counter reaches a certain value the card becomes 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.

# 12 Design Assurance

The **GCX4-PIV II – FIPS** meets the Level 3 Design Assurance section requirements.

## 12.1 Configuration Management

The **GCX4-PIV II – 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 **GCX4-PIV II – FIPS** is designed and developed using a configuration management system that is clearly ruled and operated.

Some additional documents ('Delivery and Operation', 'Reference Manual', 'Card Initialization Specification' and 'Applet Initialization Specification' documents) define and describe the step necessary to deliver and operate securely the **GCX4-PIV II– FIPS**.

## 12.3 Guidance Documents

Guidance document to be provided with **GCX4-PIV II – FIPS** is intended to be the 'Reference Manual'. Such a document is designed to in order to allow a secure operation of **GCX4-PIV II – FIPS** by its users as defined in the '**Roles, Services and Authentication'** chapter

# 13 Mitigation of Other Attacks

The GCX4-PIV II – FIPS has been designed to mitigate the following attacks:

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

A separate and proprietary document describes the mitigation of attacks policy provided by the GCX4-PIV II FIPS platform.

## 13.1 Hardware Security Mechanisms

Additionally, the embedded **P5CD072/P5CC072 chip from Philips** 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.

### 13.1.1 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 reset is generated.

### 13.1.2 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 reset is generated.

### 13.1.3 High/Low Temperature Sensor

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

### 13.1.4 Shields

Shields cover different chip areas

### 13.1.5 Fault injection detection

Fault injection mechanisms are implemented such as redundancy checking (parity, duplication) on internal data and transmissions. When an error is detected a reset is generated.
Light sensors are implemented to detect light attacks commonly used when trying to inject faults.

### 13.1.6 Light sensor

Light sensors are spread in different parts of the chip. When light attack is detected a reset is generated.

### 13.1.7 Glitch sensor

Glitch sensor is present and monitors Vcc and Vss. When the sensor is triggered a reset is generated.

### 13.1.8 Filters

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

### 13.1.9 BUS Scrambling

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

### 13.1.10 Memory Ciphering

Some dedicated and Philips proprietary ciphering algorithms are implemented in order to protect data in the different memory areas such as EEPROM, ROM and RAM.

# 14 Appendix A –PIV-II applet data model.

The product when delivered is preset with an instance of the PIV-II applet pre-personalized in a way to create a data model complying with NIST specification SP800-73-1.
This data model can be represented with the following two tables, one for data objects and one for security objects. These tables contain additional definition of access condition compared with SP800-73-1 to indicate the possibility of personalization and its access condition as personalization definition is not part of SP800-73-1.

### PIV-II Data Model set in GEMALTO Standard GemPIV cards

All PIV-II data objects are initialized (created with their access condition) before card delivery. They are left empty.

| | | | | | | | CONTACT | CONTACT | CONTACT | CONTACTLESS | CONTACTLESS | CONTACTLESS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Data Object for Interoperable Use** | **Container ID** | **BER-TLV Tag** | **Key ID** | **Key Algo** | **M/O** | **Max Size** (Bytes) | **AC Read In contact** | **AC Update In Contact** | **AC for Associated Key Use** | **AC Read Contactless** | **AC Update Contactless** | **AC for Associated Key Use** |
| **CCC: Card Capability Container** | 'DB00' | '5FC107' | N/A | N/A | Mandatory, In gemPIV | ***400*** | Always | Admin. Auth. ('9B') | N/A | Never | Never | N/A |
| **CHUID :**<br>**Card Holder Unique Identifier** | '3000' | '5FC102' | N/A | N/A | Mandatory, In gemPIV | ***3400*** | Always | Admin. Auth. ('9B') | N/A | Always | Never | N/A |
| **X.509 Certificate for PIV Authentication** | '0101' | '5FC105' | **'9A'** | 06 | Mandatory, In gemPIV | ***1910*** | Always | Admin. Auth. ('9B') | PIN | Never | Never | Never |
| **Card Holder Fingerprint I** | '6010' | '5FC103' | N/A | N/A | Mandatory, In gemPIV | ***4050*** | PIN | Admin. Auth. ('9B') | N/A | Never | Never | N/A |
| **Printed Information** | '3001' | '5FC109' | N/A | N/A | Optional, In GemPIV | ***120*** | PIN | Admin. Auth. ('9B') | N/A | Never | Never | N/A |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Card Holder Facial Image** | '6030' | '5FC108' | N/A | N/A | Optional, In GemPIV | ***12800*** | PIN | Admin. Auth. ('9B') | N/A | Never | Never | N/A |
| **X.509 Certificate for Digital Signature** | '0100' | '5FC10A' | **'9C'** | 06 | Optional, In GemPIV | ***1910*** | Always | Admin. Auth. ('9B') | PIN Always | Never | Never | Never |
| **X.509 Certificate for Key Management** | '0102' | '5FC10B' | **'9D'** | 06 | Optional, In GemPIV | ***1910*** | Always | Admin. Auth. ('9B') | PIN | Never | Never | Never |
| **X.509 Certificate for Card Authentication** | '0500' | '5FC101' | **'9E'** | **06** | Optional, In GemPIV | ***1910*** | Always | Admin. Auth. ('9B') | Always | Always | Never | Always |
| **Security Object** | '9000' | '5FC106' | N/A | N/A | Mandatory, In gemPIV | ***1050*** | Always | Admin. Auth. ('9B') | N/A | Never | Never | Never |

Underlined information are those not specified in SP800-73 and defined by GEMALTO.

## Authentication Algorithms and Key References set in GemPIV cards

All PIV-II PINs and Keys are initialized (created with their access condition) before card delivery.
The two PINs and Key '9B' are preset. Other keys are left empty.

| | | | | | | | | CONTACT | CONTACT | CONTACTLESS | CONTACTLESS |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Algo. Id** | **Key Ref** | **BER-TLV Tag** | **Key Reference Name** | **Authenticated Role** | **Retry Reset Value** | **Pre-set value at G+** | **Command to Personalize** | **Update Access condition** | **Usage Access condition** | **Update Access condition** | **Usage Access condition** |
| N/A | **'80'** | N/A | Card Holder PIV Card Application PIN | Card Holder | **<u>3</u>** | Fixed value | Change Reference Data (CRD) | CRD: Always<br>Note: PIN value in data field checked by the command. | Verify: Always | CRD: Never | Verify: Never |
| N/A | **'81'** | N/A | Card Holder II PIN Unblocking Key (PUK) | Card Holder II | **<u>3</u>** | Fixed value | **<u>Change Reference Data (CRD)</u>** | CRD: Always<br>Note: PIN value in data field checked by the command. | Verify: Always | CRD: Never | Verify: Never |
| '06' | **'9A'** | 'FF900A' | PIV Authentication Key | PIV Card Application Provider | N/A | | **<u>Put Data (PD)</u><br>Gen Asymmetric Key Pair (GAKP)** | PD: Never<br>GAKP: Admin. Auth. ('9B') | General Auth.: PIN | PD: Never<br>GAKP: Never | General Auth.: Never |
| '00'<br>(default = '03' in SP800-73) | **'9B'** | 'FF840B' | PIV Card Application Administration Key | PIV Card Application Administrator | N/A | **Fixed or Diversified** | **<u>Put Data</u>** | <u>PD: Admin. Auth. ('9B') **+ SM***</u> | General Auth.: Always | PD: Never<br>GAKP: Never | General Auth.: Always |
| '06' | **'9C'** | 'FF900C' | PIV Card Application Digital Signature Key | | N/A | | **<u>Put Data (PD)</u><br>Gen Asymmetric Key Pair (GAKP)** | PD: Never<br>GAKP: Admin. Auth. ('9B') | General Auth.: PIN Always | PD: Never<br>GAKP: Never | General Auth.: Never |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| '06' | **'9D'** | 'FF900D' | PIV Card Application Key Management Key | | N/A | | **Put Data (PD)**<br>**Gen Asymmetric Key Pair (GAKP)** | PD: Admin. Auth. ('9B') **+ SM***<br>GAKP: Admin. Auth. ('9B') | General Auth.: PIN | PD: Never<br>GAKP: Never | General Auth.: Never |
| **'06'** | **'9E'** | 'FF900E' | PIV Card Authentication Key | | N/A | | **Put Data (PD)**<br>**Gen Asymmetric Key Pair (GAKP)** | PD: Never<br>GAKP: Admin. Auth. ('9B') | General Auth.: Always | PD: Never<br>GAKP: Never | General Auth.: Always |

"+" : stands for AND: both conditions need to be fulfilled to allow the operation.

**- END OF DOCUMENT -**