Gemalto .NET Smart Card Security Policy

TITLE	.NET Smart Card - Security Policy
REV :	1.1
DATE OF CREATION:	15/01/08
LAST UPDATE :	12/02/09
AUTHORS:	Florence Defrance, Karen LU

TABLE OF CONTENTS

Contents

Table of Figures:

Table of Tables:

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] ISO 7816

[5] Infineon SLE88CFX4000P datasheet.

[6] Gemalto .NET smartcard framework SDK documentation.

[7] K. Sachdeva and S. Prevost, "Data Access Security Implementation Using the Public Key Mechanism," United States Patent Application, August 20, 2004. Omniworks secure server in Austin, scnet2\docs\public_key_token_76.0938.pdf.

[8] S.S. Keller, "NIST-Recommended Random Number Generator Based on ANSI X9.31 Appendix A.2.4 Using the 3-Key Triple DES and AES Algorithms," NIST, January 31, 2005.

1 Scope

This Security Policy specifies the security rules under which the Gemalto .NET Smart Card must operate. Some of these rules are derived from the security requirements of the **FIPS140-2' standard [1]**, others are derived from the Gemalto' experience in embedded security firmware.

These rules define the interrelationships between the:

- Module users and administrators,
- Module services,
- Critical Security Parameters (CSP)

2 Introduction

2.1 Gemalto Smart Card Overview

Gemalto aims to provide **FIPS140-2 Level 3** cryptographic smart cards. The Gemalto *.NET Smart Card* is a post-issuance programmable smart card. The .NET Smart Card provides cryptographic services to applications running on a local host computer or a remote computer. It is under the charge of applications to use the .NET Smart Card cryptographic module, in conformance with specifications.

2.2 Gemalto Smart Card Open Platform

The core technology implemented in the .NET Smart Card Framework is based on the HiveMinded Smartcard .NET reference implementation of the CLI, which conforms to the European Computer Manufacturers Association (ECMA) Common Language Infrastructure standard, ECMA-335.

The Gemalto .NET Smart Card was introduced by Gemalto in 2002 as an on-card application programming solution for .NET infrastructures. The technology offers support for multi-language application programming using an appropriate subset of the .NET class libraries. This subset has been tailored for smart card applications, and provides an optimized runtime environment on the smart card to enable communication between card and terminal using .NET remoting, ensure secure simultaneous execution of multiple applications, and exploit many other .NET framework features.

The .NET Smart Card is a cryptographic module that consists of hardware and firmware

This highly secure product benefits from Gemalto's expertise in security and provides FIPS Approved cryptographic algorithms and self-tests.

The cryptographic module integrates symmetric and asymmetric cryptographic algorithms, provides hashing algorithms, and a deterministic random number generator.

The module is referenced through PDM data model as S1001738 (chip platform) and A1002827 (mask). This tool includes a clear mapping to the hardware and firmware on the card.

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 **.NET Smart Card– FIPS** referred to hereafter as the Micro Module, a set of "embedded" hardware and firmware that implements cryptographic functions and processes, including cryptographic algorithms and key generation. **.NET Smart Card – 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** compliant smart card.

The cryptographic boundary of the Gemalto .NET Smart Card is the integrated circuit chip (ICC) module of the card. The ICC module includes hardware and firmware.

All the components of the **.NET Smart Card – FIPS Micro-Module** that are included in the cryptographic module boundary are those as shown in the following figure:
Gemalto .NET Smart Card firmware is strictly layered as illustrated in the following figure.

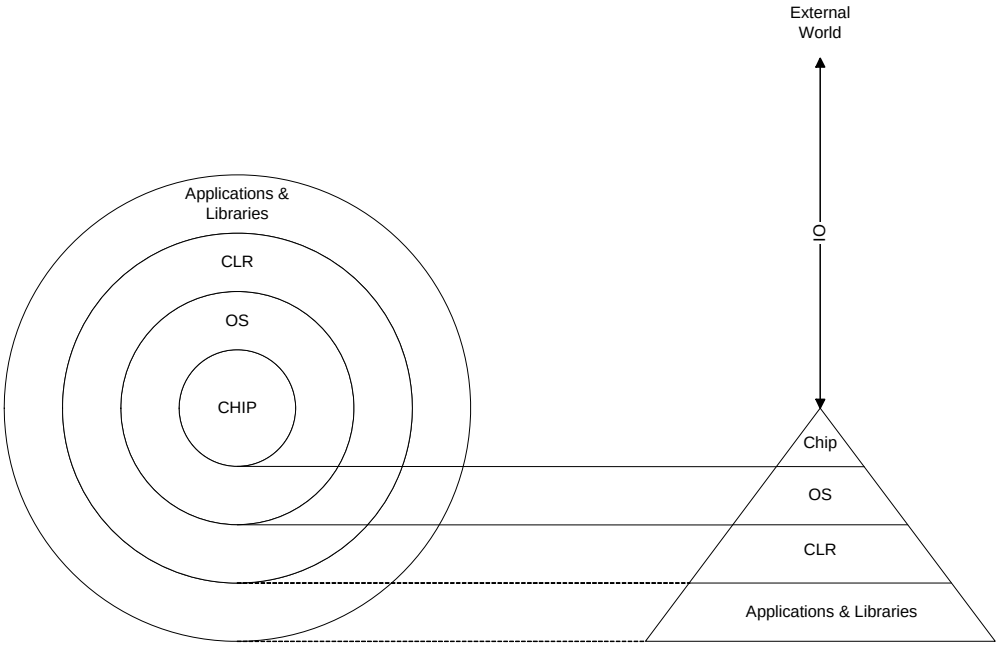

Figure 1 - Gemalto .NET Smart Card architecture.

3.2 .NET Smart Card applications

The Gemalto .NET Smart Card, FIPS 140-2 Level 3 validated, contains three on-card services loaded at personalization time: Content Manager, FIPS Assembly, and FIPS Access Manager.

3.2.1 Content manager

Content Manager Service, a part of the .NET platform (FW: v2.2) allows managing the life cycle of the card by managing the file system, setting card properties, and loading/unloading assemblies [6].

3.2.2 FIPS Assembly

The FIPS Assembly (FW: v1.1) is designed as a .NET on-card application so that the same .NET Smart Card mask can produce both FIPS cards and non-FIPS cards. The .NET Smart Card enables FIPS features when the FIPS assembly is loaded into the card. The assembly contains power-up tests of cryptographic algorithms. The rest of the power-up tests and conditional tests are encoded in the .NET Smart Card framework.

3.2.3 FIPS Access Manager

The FIPS Access Manager (FW: v1.0) application is an implementation of the .NET Smart Card Access Manager. It controls the access to the card, such as user login and administrator login.

3.3 Hardware Chip

- The cryptographic module includes the Infineon SLE 88CFX4000P secure IC chip.

3.4 FIPS Approved Mode of Operation

The module, as configured, only operates in the FIPS Approved mode of operation.

The module supports the loading of applications; however, once an application is loaded the module ceases to be a FIPS validated module. There is no assurance of operation unless the modified module has been FIPS 140 validated, per CMVP requirements.

The following tables provide a list of supported FIPS Approved and non-Approved security functions allowed in the Approved mode of operation. Of the security functions supported the module, as configured, employs the following security functions:

- RSA signature verification using SHA-1 hash.
- AES 192 bit key wrapping
- DRNG and NDRNG for random number generation.

SECURITY FUNCTION	Cert #	DETAILS
Triple-DES	719	ECB mode encryption / decryption
		CBC mode encryption / decryption
SHA-1	869	Hashing operation
SHA-256	869	Hashing operation
HMAC-SHA1	491	Keyed hash algorithm
RSA (1024 and 2048)	424	Signature - PKCS#1 with SHA-1 hashing
		Verification - PKCS#1 with SHA-1 hashing
RNG	503	Deterministic Random Number Generation (ANSI x9.31)
AES (128, 192, 256)	877	ECB mode encryption / decryption
		CBC mode encryption / decryption

Table 2 - FIPS Approved Security Functions

SECURITY FUNCTION	DETAILS
RSA (1024 and 2048)	Encryption / Decryption (provides 80 or 112 bits of encryption strength)
NDRNG	Infineon SLE 88CFX4000P TRNG

Table 3 - FIPS non-Approved Security Functions

4 Cryptographic Module Ports and Interfaces

The **.NET Smart Card – 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:

.NET Smart Card– FIPS Micro-Module follows the standards **"ISO 7816-1 Physical characteristics"** and **"ISO 7816-2 Dimensions and contact location" [4]**

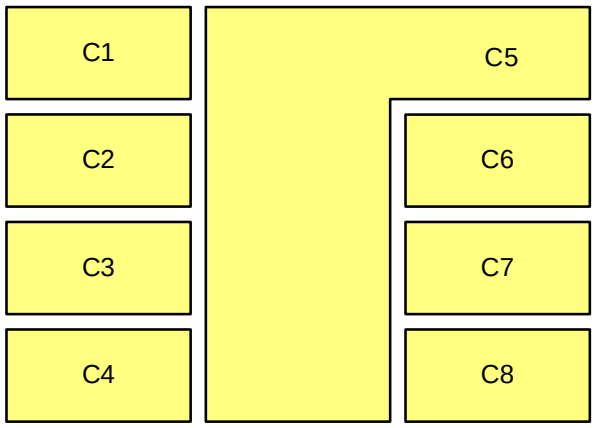

Figure 2 - Contact plate physical interface

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

Table 4 - Contact plate pin list – Contact mode

5 Roles, Services, and Authentication

5.1 Authentication

The Gemalto .NET Smart Card enforces the authentication of roles using operator authentication. The following table lists the roles, their authentication mechanisms and the strength of mechanism.

The module performs identity-based authentication using a unique PIN and challenge/response mechanism. A unique index value is associated with the PIN or the key to uniquely identify the off-card entity performing the authentication.

Role	Mechanism	Strength of Mechanism[1]
User	Challenge/response	The probability that a random attempt will succeed or a false acceptance will occur is $1/94^n$, where n the number of characters in the PIN. When n=4, the minimum PIN length, $1/94^4$ = 1/78,074,896, which is less than 1/1,000,000. With the .NET card (Infineon SLE88), the user authentication takes more than 2.5s. The maximum number of tries in one minute is 60/2.5 = 24. For n = 4, the probability of successfully authenticating to the module through random attempts within one minute is less than $24/94^4 = 0.0307 \times 10^{-5}$, which is less than 1/100,000.
Administrator (Crypto-Officer)	Challenge/response	The probability that a random attempt will succeed or a false acceptance will occur is $1/2^{64} = 1/1.84 \times 10^{19}$, which is less than 1/1,000,000. With the .NET card (Infineon SLE88), the admin authentication takes about 0.125s. The maximum number of tries in one minute is 60/0.125 = 480. The probability of successfully authenticating to the module through random attempts within one minute is $480/2^{64} = 2.61 \times 10^{-17}$, which is less than 1/100,000.

Table 5 - Roles, mechanism and strengths of authentication mechanisms

[1] The strength of an authentication mechanism is the probability that a random attempt succeeds

5.1.1 Critical Security Parameters (CSP)

The Gemalto .NET Smart Card has the following critical security parameters:

Table 6 - Table of CSP

CSP	Purpose
Administrator key	TDES key used to authenticate the administrator role. Note: The admin key must be updated at least once.
Admin Key encryption key (KEK)	AES key used by the administrator role. (Change the administrator key, Unblock user PIN and update Admin or User KEK.)
Seed and seed key	Initializes DRNG. The seed key is a TDES key.
User PIN	Authenticate the user role. Note: The user pin must be changed at least once.
User Key encryption key (KEK)	AES key used by the User role (Change the user PIN).

Public Key	Purpose
Gemalto FW public key	Authentication and integrity for FW loading.

5.1.2 Authentication Security Rules

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

Table 7 - Authentication Security Rules

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 remains active if another application is selected
IA_PIN_RULE.6	The PIN length must have a minimum length equal to 4
IA_PIN_RULE.7	The user PIN must be changed at least once
IA_CHAL/RESP_RULE.1	The administrator must be re-authenticated if the card is reset.
IA_CHAL/RESP_RULE.2	The administrator must be re-authenticated if the cryptographic module detects a corrupted response.
IA_ADM_KEY.1	The admin key must be changed at least once

5.2 Roles and Services

The Gemalto .NET Smart Card product supports two operator roles: User and Administrator (Crypto-Officer). The two roles are authenticated differently and have different access rights for resources and services on the card. The authorized services for each role are product dependent.

The follow table lists common authorized services for the user role and the administrator role for the .NET Smart Card.

Table 8 - Services Authorized for Roles

Role	Authorized Services	CSP	Access
User:	User Login: log into the card.	User PIN	Utilize
		DRNG State	Utilize
	Change PIN: allows authenticated user to modify PIN value used for authentication and settings.	User PIN	Utilize, Replace
		User KEK	Utilize
		DRNG State	Utilize
	Get Status: get module status information	N/A	N/A
	Logout: logout of the card	N/A	N/A
Administrator: (Crypto-Officer)	Administrator Login: log into the card.	Administrator Key	Utilize
		DRNG State	Utilize
	Logout: logout of the card	N/A	N/A
	Change Administrator key: allows authenticated administrator to update Admin Key value used for authentication and settings.	Administrator Key	Utilize, Replace
		Administrator KEK	Utilize
	Change KEK: update the Administrator KEK or User KEK	Administrator Key	Utilize
		Administrator KEK	Utilize, Replace
		User KEK	Utilize, Replace
	Un-Block user PIN	Administrator Key	Utilize

Role	Authorized Services	CSP	Access
		Aministrator KEK	Utilize
		User PIN	Replace
		DRNG State	Utilize
	Manage Card Content: View and manage content of files on card.	N/A	N/A
	Zeroize: Overwrites memory content with Zero's and resets the module.	Administrator Key	Destroy
		Administrator KEK	Destroy
		User PIN	Destroy
		User KEK	Destroy
		DRNG State	Destroy
	Get Status: Get module status information	N/A	N/A
Unauthenticated:	Get Status: Get module status information	N/A	N/A

5.2.1 Security rules

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

Table 9 - Security Rules

Rule Identifier	Description
AC_AMS_RULE.1	Only authorized entities can use Content Manager Service.
AC_PKT_RULE.2	Application loaded must be cryptographically signed.
AC_APD_RULE.3	Application Domain provides security and isolation among multiple applications On the card.
AC_FSA_RULE.4	Data files are protected using public and private privileges
AC_ACL_RULE.5	The role is checked before allowing actions on files.
AC_KEA_RULE.6	To key access the privileges must be checked.

AC_TMT_RULE.7	In case of interrupted transaction the transaction consistency is ensured.

6 Physical Security

The **.NET** Smart Card 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.

6.1 Manufacturing Process

The manufacturing process consists 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

Attempting to extract the chip from the micro-module has a high probability of causing serious damage to the chip. Furthermore, attempts to attack the chip or micro-module will result in signs of tampering such as scratches and deformation.

7 Operational Environment

The Gemalto .NET Smart Card provides a limited operational environment. The micro-module allows loading of digitally signed firmware. If any firmware is loaded into the micro-module, the module ceases to be FIPS 140-2 validated. The FIPS validated operational environment is as defined in Section 3 of this document.

8 EMI/EMC

The .NET Smart Card cryptographic module has been tested to meet the EMI/EMC requirements specified in FCC Part 15 Subpart B, Class B.

9 Self Tests

The .NET Smart Card performs the following self-tests to ensure that the module works properly.

Table 10 - Self-tests list

SELF-TESTS	EXECUTION
CRYPTOGRAPHIC ALGORITHM TEST (KNOWN-ANSWER TESTS FOR TRIPLE-DES, AES, SHA-1, SHA-256, RSA, HMAC-SHA1)	AT POWER-UP
FIRMWARE INTEGRITY TEST.	AT POWER-UP

SELF-TESTS	EXECUTION
RANDOM NUMBER GENERATOR TEST (KNOWN-ANSWER DRNG)	AT POWER-UP
PAIR-WISE CONSISTENCY TESTS.	CONDITIONAL
CONTINUOUS RANDOM NUMBER GENERATOR TEST FOR DRNG AND NDRNG.	CONDITIONAL
FIRMWARE LOAD TEST.	CONDITIONAL

The Gemalto .NET Smart Card does self-tests including power-up tests at start up and conditional tests during the runtime.

9.1 Self-Test Execution

After .NET Smart Card is powered up and before executing any APDU commands, the module enters the self-test state and performs all of the cryptographic algorithm and firmware integrity self-tests as specified in FIPS 140-2 standard **[1]**.

In addition to the FIPS 140-2 required self-tests, at each power-up (reset), the microprocessor chip of the cryptographic module performs security tests for chip sensors and built-in self-test to detect physical defects. The chip is muted if any of the tests fails.

These tests are conducted automatically in FIPS mode as part of the normal functions of the cryptographic module. They do not require any additional operator intervention.

Power-up self-tests are executed upon reset. The cryptographic module start-up process in FIPS mode 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.

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.

9.2 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 Firmware Load test, the smart card enters an error state and an error code is returned via the status interface and the loading is aborted.

When a FIPS test error occurs, the .Net Smart Card enters an error state where no commands will be performed and no data (except error status) will be sent out. The card sends out an error status and then mutes itself until the next reset.

10 Mitigation of Other Attacks

The .NET Smart Card has been designed to mitigate the following attacks:

- Fault Attack,
- Probing attacks.

A separate and proprietary document describes the mitigation of attacks policy provided by the .NET Smart Card.