COMMON CRITERIA RECOGNITION ARRANGEMENT
FOR COMPONENTS UP TO EAL 4

Certification Report

EAL 4+ (ALC_DVS.2, AVA_VAN.5)

Evaluation of

TÜBİTAK BİLGEM UEKAE
SMART CARD OPERATING SYSTEM (AKiS) v1.2.2I
AKILLI KART İŞLETİM SİSTEMİ (AKiS) v1.2.2I

issued by

Turkish Standards Institution
Common Criteria Certification Scheme

Date	**: 08.08.2011**
Pages	**: 37**
Certification Report Number	**: 14.10.01/11-248**

This page left blank on purpose.

----- o -----

TABLE OF CONTENTS:

LIST OF TABLES

LIST OF FIGURES

This page left blank on purpose.

----- o -----

CERTIFICATION REPORT

The Certification Report is drawn up to submit the Certification Committee the results and evaluation information upon the completion of a Common Criteria evaluation service performed under the Common Criteria Certification Scheme.

Certification Report covers all non-confidential security and technical information related with a Common Criteria evaluation which is made under the PCC Common Criteria Certification Scheme. This report is issued publicly to and made available to all relevant parties for reference and use.

1.INTRODUCTION

The Common Criteria Certification Scheme (CCCS) provides an evaluation and certification service to ensure the reliability of Information Security (IS) products. Evaluation and tests are conducted by a public or commercial Common Criteria Test Laboratory (CCTL) under CCCS' supervision.

CCTL is a facility, licensed as a result of inspections carried out by CCCS for performing tests and evaluations which will be the basis for Common Criteria certification. As a prerequisite for such certification, the CCTL has to fulfill the requirements of the standard ISO/IEC 17025 and should be accredited with respect to that standard by the Turkish Accreditation Agency (TÜRKAK), the national accreditation body in Turkey. The evaluation and tests related with the concerned product have been performed by TÜBİTAK-BİLGEM-UEKAE-OKTEM, which is a public CCTL.

A Common Criteria Certificate given to a product means that such product meets the security requirements defined in its security target document that has been approved by the CCCS. The Security Target document is where requirements defining the scope of evaluation and test activities are set forth. Along with this certification report, the user of the IT product should also review the security target document in order to understand any assumptions made in the course of evaluations, the environment where the IT product will run, security requirements of the IT product and the level of assurance provided by the product.

This certification report is associated with the Common Criteria Certificate issued by the CCCS for

AKILLI KART İŞLETİM SİSTEMİ(AKiS) v1.2.2I - SMART CARD OPERATING SYSTEM (AKiS) v1.2.2I whose evaluation was completed on 20.05.2011 and whose evaluation technical report was drawn up by OKTEM (as CCTL), and with the Security Target document with version no 05 of the relevant product.

2.GLOSSARY

CCCS:	Common Criteria Certification Scheme
CCTL:	Common Criteria Test Laboratory
CCMB:	Common Criteria Management Board
CEM:	Common Evaluation Methodology
AKiS:	Smart Card Operating System (**A**kıllı **K**art **İ**şletim **S**istemi)
ETR:	Evaluation Technical Report
IT:	Information Technology
OKTEM:	Common Criteria Test Center (as CCTL)
PCC:	Product Certification Center
ST:	Security Target
TOE:	Target of Evaluation
TSF:	TOE Security Function
TSFI:	TSF Interface
SFR:	Security Functional Requirement
TÜBİTAK:	Turkish Scientific and Technological Research Council
TÜRKAK:	Turkish Accreditation Agency
BİLGEM:	Center of Research For Advanced Technologies of Informatics and Information Security
UEKAE:	National Electronics and Cryptology Research Institute
EAL:	Evaluation Assurance Level
PP:	Protection Profile

Table 1 - Glossary

3.EXECUTIVE SUMMARY

Evaluated IT product name:

Smart Card Operating System (AKiS) v1.2.2I

Akıllı Kart İşletim Sistemi(AKiS) v1.2.2I

IT Product version:

v1.2.2I

Developer`s Name:

TÜBİTAK BİLGEM UEKAE AKIS Project Group

Name of CCTL :

TÜBİTAK BİLGEM UEKAE OKTEM Common Criteria Test Laboratory

Completion date of evaluation :

20.05.2011

Common Criteria Standard version :

- Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model, Version 3.1, Revision 3, July 2009
- Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components, Version 3.1, Revision 3, July 2009
- Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components, Version 3.1, Revision 3, July 2009

Common Criteria Evaluation Method version :

- Common Methodology for Information Technology Security Evaluation v3.1 rev3, July 2009

Short summary of the Report:

1) Assurance Package :

EAL 4+ (ALC_DVS.2, AVA_VAN.5)

2) Functionality :

AKiS v1.2.2i is a smart card operating system which can be used in personal identification, digital sign, health care system, smart logon, secure email.

TOE SECURITY FUNCTIONS

Cryptographic Operations	**1. Sign** In Sign security function, plain data sent by the user within the APDU command is signed (decrypted) with the key that is previously referenced with another command. Signed data is transmitted back to the user. The point here is not the secrecy of the data; it is the integrity of the data. RSA 2048 algorithm can be used for this operation, so the referenced key must be an RSA 2048 key and it must own all the parameters required for this operation.
	2. Verify Signature In Verify Signature security function, signed part of the data sent by the user within the APDU command is encrypted with the key that is previously referenced with another command and the encrypted data is compared with the plain part of the data sent at the end of signed data within the command. After the comparison, a response is transmitted back to the user indicating whether the signature is verified or not. The point here also is not the secrecy of the data; it is the integrity of the data. RSA 2048 algorithm can be used for this operation, so the referenced key must be an RSA 2048 key and it must own all the parameters required for this operation.

3. Encryption

In Encryption security function plain data sent by the user within the APDU command is encrypted with the key that is previously referenced with another command. Encrypted data is transmitted back to the user as a response. Here both the secrecy and the integrity of the data is of concern. For the encryption operation, any of the RSA 2048, 3DES (DDES) and DES algorithms can be used, so the referenced key can be any of these algorithms’ keys. But the key must own all the parameters required for this operation.

4. Decryption

In Decryption security function, cipher data sent within the APDU command is decrypted with the key that is previously referenced with another command. The plain text is transmitted back to the user as a response. Also here both the secrecy and the integrity of the data is of concern. For the decryption operation, any of the RSA 2048, DES3 and DES algorithms can be used, so the referenced key can be any of these algorithms’ keys. But the key must own all the parameters required for this operation.

For the correct operation of the security functions described above, the user should reference an appropriate key (application-DF key) before the cryptographic operation takes place. Here to reference a key means moving the key from the EEPROM memory area into the RAM memory area in order to use it. Also before this operation, the user must load the key into that application specific EEPROM memory area in a secure way. Loading more than 1 key to an application (DF) is possible (maximum 20 keys). The algorithms for these keys may be different (any of RSA, DES3 and DES).

5. Cryptographic Checksum Calculation

Cryptographic checksum is used in order to protect user data integrity. Cryptographic checksum calculation function calculates the checksum of the plain data and the initialization vector sent within the command according to the reference key sent prior to the command by the user.

The first part of the plain data sent within the command is XORed with the initialization vector and encrypted with the reference key. The data formed after this operation serves as the new initialization vector for the second part of the plain data. The operation is repeated until all parts of the data is encrypted. Calculation of cryptographic checksum is performed using DES or 3DES algorithms in the TOE. That’s why; the reference key must belong to one of these algorithms.

	6. Cryptographic Checksum Verification Cryptographic checksum verification is performed in two steps. Firstly, the checksum of the plain data and the initialization vector sent within the command is calculated according to the reference key. Secondly, the calculated checksum is compared with the checksum within the command. If they match, an operation successful response is returned. If they don't match, an error message is returned. A mismatch means that the data integrity has been corrupted. Calculation of cryptographic checksum is performed using DES or 3DES algorithms in the TOE. That's why, the reference key must belong to one of these algorithms.
Authentication and Authorization Functions	**1. Administrator Authentication (with System PIN)** Administration life cycle is a life cycle which allows only the administrator to run administration commands. In order to pass to the administration life cycle, System PIN must be verified. If a wrong System PIN is entered 3 times, the card goes to the death life cycle. Only the administrator can change the System PIN. System PIN must be minimum 4, maximum 16 digits. System PIN initial value is given at production state during constitution of MF. Administration commands such as CHANGE_KEY, ERASE_FILES are to be performed in this life cycle by the administrator (after personalization phase). Only administrator can change the life cycle from Operation to Administration and vice versa. **2. Authenticated User Authentication (with PIN)** On a directory (DF) created with PIN, in order to perform PIN verification in operation life cycle, PIN must be set first. During PIN change operation, if the operation is interrupted by taking out the card from the card reader, old PIN is valid. PIN must be minimum 4, maximum 16 digits. When the PIN is input maximum PIN error value times (if it is not set at configuration, default value is 3) incorrectly, that directory (DF) becomes INVALID and only the administrator can make that DF reusable by resetting PIN error counter. After the error counter is reset, authenticated user can use his DF with the PIN the administrator gave him. During PIN change, if the old PIN is input incorrectly, error counter is incremented by 1. After maximum PIN retry number incorrect entries, the DF becomes INVALID. For performing operations in operation life cycle on a DF created with PIN, VERIFY command must be performed successfully. Access to the EFs/DFs under that DF is dependent to their own access conditions.

Operation life cycle is a life cycle belonging to the user and the authenticated user usually. For this reason, in order to change the life cycle to administration, system PIN must be entered. Furthermore, a user/authenticated user cannot create directories (DFs).

3. Authorizing User to an Operation

Authorizing user to an operation function is used for making the decision if the user is authorized or not to perform the operation he wants. In this function, the user must transmit a secret data known both by the user and the TOE within the operation's command. The user is authorized to the operation only if he transmits that secret data accurately and completely. Otherwise, the user will not be allowed to perform that operation. Here, the secret data transmitted in the command can be a key encrypted by itself or a special data encrypted by a key depending on the involved command and the user type.

For being authorized, the user should either know the key or both the key and the special data according to the command and the user type. For this operation, one of RSA2048 and DES3 algorithms can be used depending on the command being used. So the referenced key may belong to one of these algorithms, but the key must own all the parameters required for this operation.

This function is concerned with the commands; Exchange Challenge, Change Key, Erase Files and External Authenticate (activation). There is an error counter for each of these commands separately. The secret data used for authorization may be common for some of these commands, but the error counter is not counted for each faulty usage of the secret data itself, it is counted for each faulty usage of the command.

4. Authentication of User to TOE

Authentication of user to TOE function is used to determine if the user is a secure user in order to use the active application. User is expected to transmit a secret data known both by the user and the application within the command. If the user transmits this secret data correctly and completely, he is defined as a secure user for the application. Otherwise, the user will not be allowed to perform any secure operation on that application. Here, the secret data transmitted within the command is a random number generated by the TOE and encrypted with a key belonging to that application. Each random generated by the TOE can only be used once. TOE guarantees a random number to be used for the authentication of user to TOE at most once. For this operation, one of DES3 and DES algorithms can be used, so the referenced key may belong to one of these algorithms, but the key must own all the parameters required for this operation.

	5. Authentication of TOE to User Authentication of TOE to user function is used to decide whether the TOE is secure and correct TOE or not. TOE is expected to encrypt the data within the incoming command with a key known both by the user and the TOE and transmit back the encrypted data. TOE is defined as a secure TOE for the user, only if transmits this secret data correctly and completely. Otherwise, it is not reliable for a user to use the TOE. Here, the secret data transmitted within the response is a random number generated by the user and encrypted with a key belonging to that application. For this operation, one of DES3 and DES algorithms can be used, so the referenced key may belong to one of these algorithms, but the key must own all the parameters required for this operation.
Cryptographic Keys	Proprietary key access function is used to write and erase application keys from EEPROM, RAM/XRAM. Each key has unique ID number per application. Two components of DES and 3DES keys are written with in the same APDU command whereas each component of RSA keys is written in a separate command with different parameters by TOE. While the keys are written into TOE, the algorithm of the key and type of the cryptographic operations will be used with this key are determined by APDU command. The key is not allowed to be written if algorithm of key is inconsistent with determined cryptographic operations for this key. DES and 3DES keys cannot be used with sign and verify signature operations. They can be used with Ext. Auth., Int. Auth., Encryption, Decryption, MAC and verify MAC operations. RSA keys cannot be used with Ext. Auth., Int. Auth., MAC, verify MAC operations. They can be used with Encryption, Decryption, verify signature operations. All key lengths are checked whether the length of the key is meaningful or not. The key is not allowed to be written with an invalid length. Content of all system keys and DF keys are checked if they are not entirely composed of 0xFF. All keys must include at least one byte different from 0xFF. Otherwise, the key is not allowed to be written. When a key is erased, all components belonging to that key are also erased, they are removed from the key table and their connections are deleted. Proprietary key access function reads the modulus and public components of RSA keys which are loaded to the application. Since these components are public, they can be read without any authentication. DES, 3DES keys and PDAT, QDAT, DPDAT, DQDAT, QINVDAT components of RSA keys are not given outside the card. An error response is produced if these components are tried to be read. Proprietary key access function (Reading from EEPROM to RAM) transports all of the parameters of the requested application key from

	EEPROM to RAM. An initialization function completely fills the buffer on RAM containing the application key with 0xFF.
Secure Messaging	With a bit in APDU command’s CLA byte, it is decided whether to use secure (encrypted) messaging or not. Secure messaging is mandotory according MF or DF access rights. If MF is created with secure messaging access right all commands under MF must be encrypted. If DF is created with secure messaging access right all commands under DF must be encrypted. EXCHANGE CHALLENGE command does not need to be encrypted. In secure messaging all the data transmitted within a command is encrypted with the session key according to 3DES algorithm. As both sides (users and TOE) know the session key, they decrypt the incoming commands with the session key to interpret them.
Integrity of the Objects	DF, DF keys, EF, System/DF PIN, System/DF PUK and life cycle integrity check is performed with checksum. In every write and erase operation the checksum is being updated, checksum is controlled in every read operation. If there is a corruption in DF, EF, System/DF PIN, and System/DF PUK, a warning or error message is returned as a response to the user. If there is a corruption in DF keys, System/DF PIN, System/DF PUK an error is returned and the corrupted data is no longer allowed to use. If there is a corruption in EF header, an error is returned and the corrupted EF is no longer allowed to use. But if the corruption occurred in EF body a warning returned and the corrupted EF is used. When DF, EF, DF PIN/PUK or DF keys are being deleted, the delete operation is performed by releasing the connections on tables. When a page is taken from the memory area, that page is being erased before the operation. During DF/EF creation and System/DF PIN/PUK change if the operation is interrupted by ejecting the card from the card reader, old data becomes valid in order to protect the data integrity, because it is not clear where the write operation is interrupted. When an uncontrolled access is detected to write to the special areas of EEPROM, an error code indicating a memory error is returned and the write operation is not permitted. Command integrity is provided with the checksum byte which is at the end of the command. If the checksum is wrong, the command sent from card to the reader or command sent from the reader to the card must be repeated. At each reset, card life cycle is tested whether it is one of the defined states or not, checksum is also controlled. If the card life cycle data is corrupted, TOE returns to the activation life cycle. Code memory checksum is calculated and returned to the user within the GET DATA command. User can check the code memory integrity by this way.

Access Conditions on the DFs and EFs	DF/EF access conditions are controlled according to the command to be performed. Access control is made: - Read/Write access: If the DF has a write access, new DFs and EFs can be created/deleted under that DF. If the EF has a write access, EF can be written/updated. In order to read an EF, the EF must have read access. - Security access with System/DF PIN: User can access DF and any EF under that DF if the DF is created without PIN. If the DF is created with PIN, access conditions of EFs under that DF, depends on the access conditions of EF in the operation life cycle - Security access with key authentication: If the DF is created with key authentication, the user uses the internal and external authenticate commands in order to get authenticated into that directory. This subject is explained in **Authentication of User to TOE**.
Function Countering Physical Attacks	**1. Countering System/DF PINs and System/DF PUK Attacks** In order to prevent unexpected jumps in critical code points which may be caused by external attacks, there is a double check in code lines controlling System/DF PIN and System/DF PUK. System/DF PIN/PUK control flag is a byte (8 bit) instead of one bit for the unexpected multiple changes on the flag. System/DF PIN and PUK error counters are incremented before to check it for the attacks against to System/DF PUK/PIN error counters by using power of the smart card. System/DF PUK/PIN is verified when all the digits is completed against the timing attacks. **2. Physical Sensors** SLE66CX680PE chip produces an NMI when code, data and IRAM areas are attacked. In this chip, there are different sensors for the physical attacks such as low/high frequency, low/high voltage, temperature, glitch and light detectors. Chip produces a HW reset signal or NMI interrupt when these sensors sense an abnormal situation. TOE goes to a reset (soft RESET) state if NMI is produced. Random number generator and Sanity of the Physical sensors are checked by calling the Infineon library functions in the main procedure of TOE.

Table 2 - TOE Security Functions

3) AKiS v1.2.2I Operating System Phases

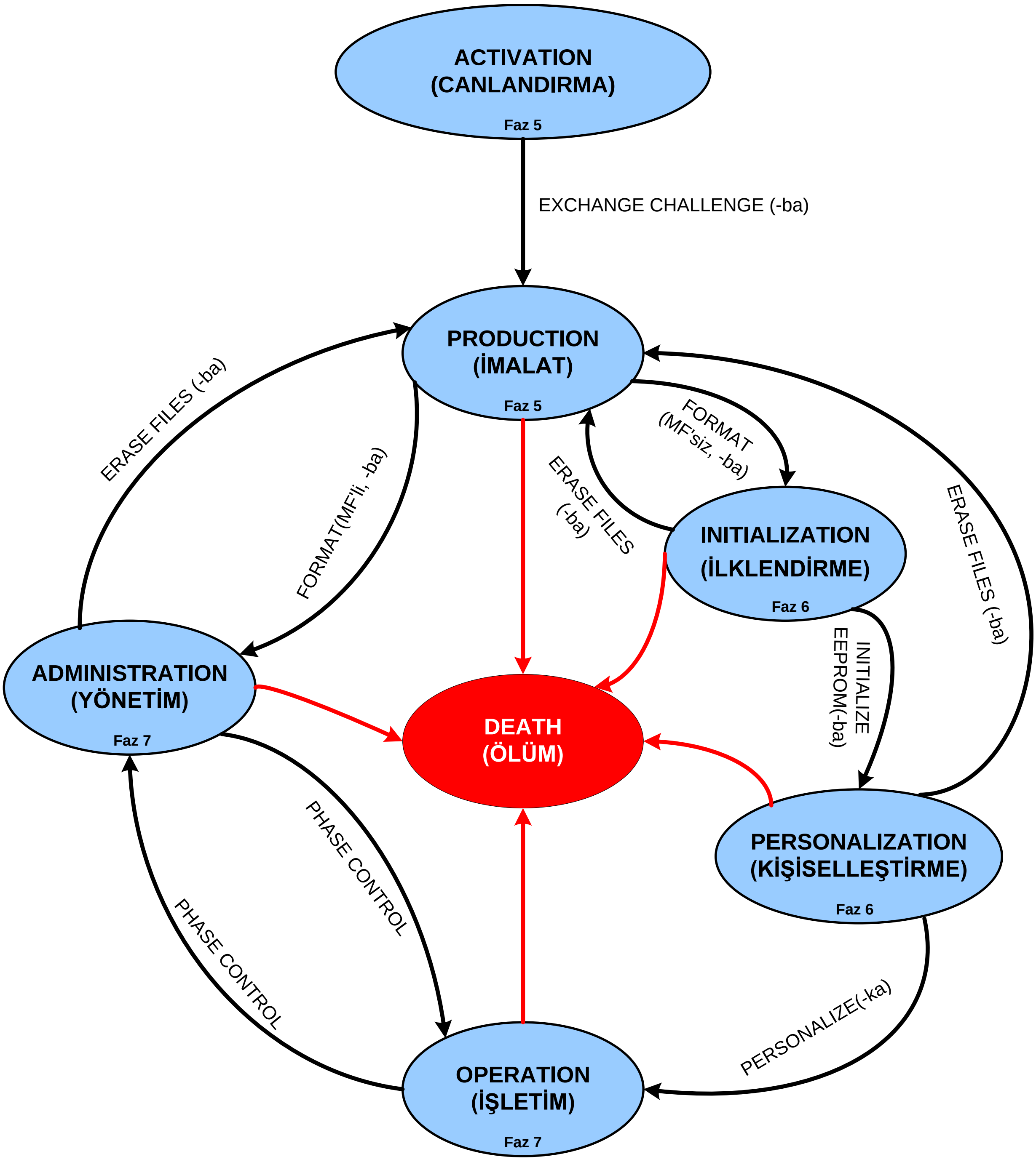
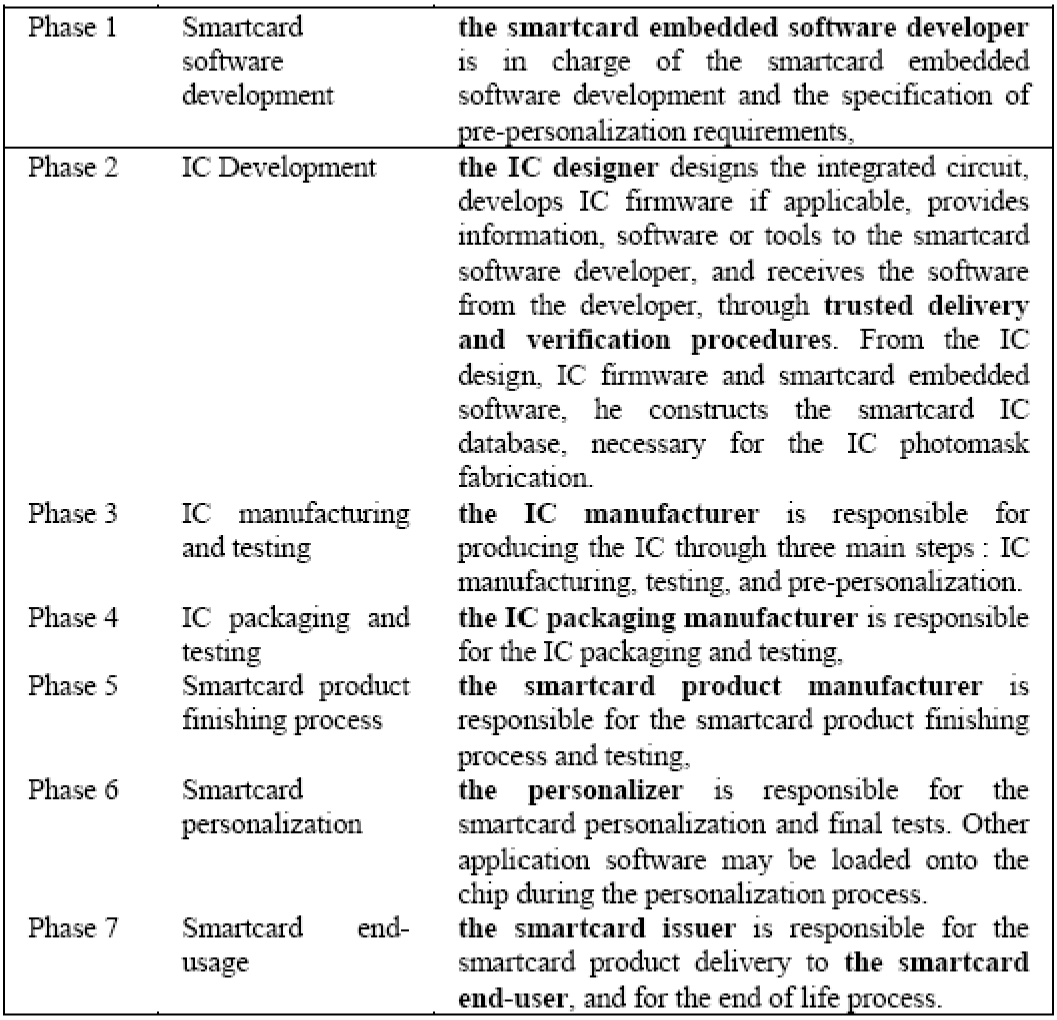


Figure 1 - TOE Life cycle phases



Phase 1	Smartcard software development	**the smartcard embedded software developer** is in charge of the smartcard embedded software development and the specification of pre-personalization requirements,
Phase 2	IC Development	**the IC designer** designs the integrated circuit, develops IC firmware if applicable, provides information, software or tools to the smartcard software developer, and receives the software from the developer, through **trusted delivery and verification procedures**. From the IC design, IC firmware and smartcard embedded software, he constructs the smartcard IC database, necessary for the IC photomask fabrication.
Phase 3	IC manufacturing and testing	**the IC manufacturer** is responsible for producing the IC through three main steps : IC manufacturing, testing, and pre-personalization.
Phase 4	IC packaging and testing	**the IC packaging manufacturer** is responsible for the IC packaging and testing,
Phase 5	Smartcard product finishing process	**the smartcard product manufacturer** is responsible for the smartcard product finishing process and testing,
Phase 6	Smartcard personalization	**the personalizer** is responsible for the smartcard personalization and final tests. Other application software may be loaded onto the chip during the personalization process.
Phase 7	Smartcard end-usage	**the smartcard issuer** is responsible for the smartcard product delivery to **the smartcard end-user**, and for the end of life process.

Figure 2 - Smart Card Product Life Cycle

A smart card life cycle process consists of some certain phases as shown in Figure 2. TOE takes place within Phase 6 and Phase 7 on this process. Just like the whole smart card, TOE also consists of some phases which will be called as “Life cycle phases” (Figure 1) in order to obstruct a confusion.

There are 7 different life cycle phases available on TOE. Relations and crossing between these life cycle phases are shown in Figure 1. Also there are some several keys available on TOE in order to be used within the execution of the secure commands. Command interpreter of TOE is designed to execute some special commands for the different life cycle phases.

These phases are;

Activation:

Main purposes of the activation life cycle phase is; check if the smart card includes correct TOE and load the initial values of the keys that will be used on the execution of the secure commands (initialization and personalization key).

Production:

Main purpose of the production phase is to format the EEPROM memory of the card and prepare the card for the next step. The next step would be the Initialization phase or the Administration phase depending on the format type. MF(Master File) is created in the Production phase.

Initialization:

Main purpose of the initialization phase is to load the initialization data into the card. Therefore the file system will begin to construct on the EEPROM on each command.

Personalization:

Main purpose of the personalization phase is to load the personalization data into the card. Henceforth the card will include unique data belonging to the end user.

Administration:

Administration phase is the management phase for the administrator and the authorized user. Changes in the file system or file system errors are handled in this phase. Smart cards with TOE have the reusability feature.

Operation:

In operation phase, TOE is also available for the end user.

Death:

When some security conditions are not satisfied or it is noticed that security is trying to be surpassed, TOE forces the card into death phase.

4) Summary of Threats addressed by the evaluated IT product:

The TOE counter such threats presented in the table below and provide functions for countermeasure to them.

Threats on all smartcard product life cycle phases (1 to 7)	**T.CLON** Functional cloning of the TOE (full or partial) appears to be relevant to any phase of the smart card product life-cycle, from phase 1 to phase 7. Generally, this threat is derived from specific threats combining unauthorized disclosure, modification or theft of assets at different phases. **T.DIS** Unauthorized disclosure of the smartcard embedded software, data or any related information.
	T.MOD Unauthorized modification of the smartcard embedded software and data.
Threats on smartcard product life cycle phase 1	**T.T_TOOLS** Theft or unauthorized use of the smartcard embedded software development tools (such as PC, databases). TOE system design, basic software, TOE hex code, activation key are subject to threats.
	T.FLAW Introduction of flaws in the TOE due to malicious intents or insufficient development. TOE system design, basic software, TOE hex

	code are subject to threats.
	T.T_SAMPLE Theft or unauthorized use of integrated circuit samples containing the embedded software (e. g. bound out, dil, evalOS). TOE hexcode is subject to threat.
	T.MOD_INFO Unauthorized modification of any information (technical or detailed specifications, implementation code, design technology, tools characteristics) used for developing software or loading data. TOE system design, basic software, TOE hex code are subject to threats.
	T.DIS_TEST Unauthorized disclosure of the smartcard embedded software test information including interpretations. Application data and activation key is subject to threat.
	T.DIS_INFO Unauthorized disclosure of any information (technical or detailed specifications, implementation code, design technology, tools characteristics) used for developing software or loading data. This includes sensitive information on IC specification, design and technology, software and tools. TOE system design, basic software, TOE hex code are subject to threats.
Threats on delivery of software and related information from smartcard product life cycle phases 1, 2, 3 and 6	**T.T_DEL** Theft or unauthorized use of the smartcard embedded software and any additional application data delivered to the IC designer, IC manufacturer or to the personalizer. TOE hex code and application data are subject to threats.
	T.MOD_DEL Unauthorized modification of the smartcard embedded software and any additional application data delivered to the IC designer, IC manufacturer or to the personalizer. TOE hex code and application data are subject to threats.
	T.DIS_DEL Unauthorized disclosure of the smartcard embedded software and any additional application data delivered to the IC designer, IC manufacturer or to the personalizer. TOE hex code and application data are subject to threats.
Threats on smartcard product life cycle phase 2	**T.DIS_TEST** Unauthorized disclosure of the smartcard embedded software test information including interpretations. TOE hex code is subject to threat.
	T.DESIGN_IC Poor IC design leading to IC security mechanisms not meeting state of the art level. Application data is subject to threat.
Threats on smartcard product life cycle phases 3 to 6	**T.T_PRODUCT** Theft or unauthorized use of the smartcard product or any related information. For example, unauthorized use of the embedded software application functions. TOE hex code and application data are subject to threats.
	T.DIS_TEST Unauthorized disclosure of the smartcard embedded software test information including interpretations. TOE hex code and application data are subject to threats.
Threat on smartcard product life cycle phase 7	**T.T_PRODUCT** Theft or unauthorized use of the smartcard product or any related information. For example, unauthorized use of the embedded software application functions. Application data is subject to threat.

Table 3 - Threats

5) Disclaimers:

This certification report and the IT product defined in the associated Common Criteria document has been evaluated at an accredited and licensed evaluation facility conformance to Common Criteria for IT Security Evaluation, version3.1 revision3, using Common Methodology for IT Products Evaluation, version3.1 revision 3. This certification report and the associated Common Criteria document apply only to the identified version and release of the product in its evaluated configuration. Evaluation has been conducted in accordance with the provisions of the CCCS, and the conclusions of the evaluation facility in the evaluation report are consistent with the evidence adduced. This report and its associated Common Criteria document are not an endorsement of the product by the Turkish Standardization Institution, or any other organization that recognizes or gives effect to this report and its associated Common Criteria document, and no warranty is given for the product by the Turkish Standardization Institution, or any other organization that recognizes or gives effect to this report and its associated Common Criteria document.

4.IDENTIFICATION

AKiS v1.2.2i is a smart card operating system which can be used in personal identification, digital sign, health care system, smart logon, secure email. TOE:

- Is loaded into ROM of the Infineon Smart Card (SLE66CX680PE/m1534a13) during the manufacturing phase. SLE66CX680PE/m1534a13 has EAL 5+ (ALC_DVS.2, AVA.MSU.3, AVA_VLA.4) certificate.
- Does not allow loading of executable files, communicates with the PC via card reader according to ISO/IEC 7816-4 T = 1 protocol,
- Implements user and interface authentication,
- Is capable of binary file operations (open, update, erase, read),
- Supports fixed length linear, variable length linear, fixed length cyclic file structures and file operations (open, append record, update record, read record),
- Follows the life cycles (activation, manufacturing, initialization, personalization, administration, operation and death) and operates functions according to the present life cycle,
- Encrypts, decrypts, digitally signs and verifies with RSA/DES/3DES cryptographic algorithms by using HW modules of the SLE66CX680PE,
- Calculates SHA-1 hash.

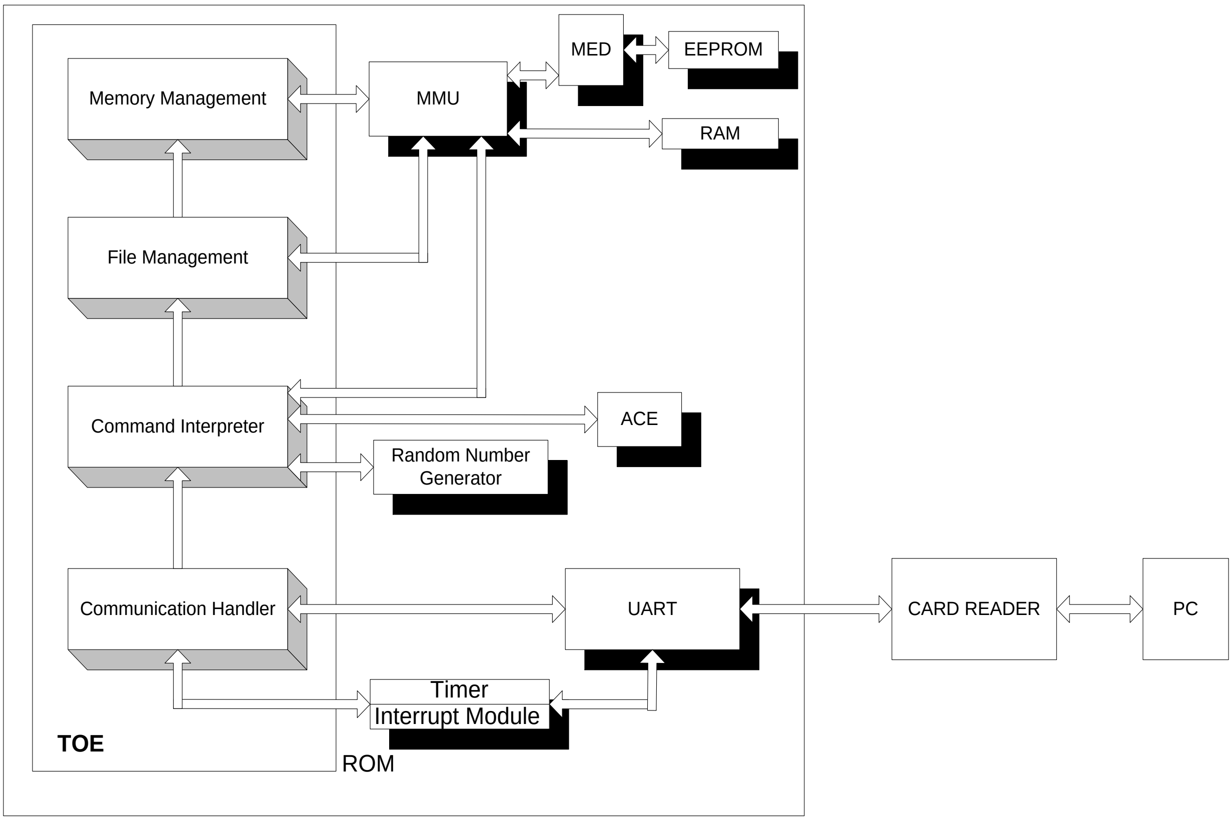


Figure 3 - TOE's components and environment

5.SECURITY POLICY

Organizational Security Policies

- **OSP.SECURE_DF** Creation of DFs with the secure messaging attributes.

6. ARCHITECTURAL INFORMATION

AKiS v1.2.2i Algorithms and crypto specifications are;

Authentication;

External Authenticate: DES/DES3/RSA (1024 bit)

Internal Authenticate: DES/DES3

Encryption;

DES-ECB: Plain data can be encrypted with a DES key.

DES3-ECB: Plain data can be encrypted with a DES3 (DDES) key (A-B-A key structure).

RSA2048: Plain data can be encrypted with an RSA2048 key.

Decryption;

DES-ECB: Encrypted data can be decrypted with a DES key.

DES3-ECB: Encrypted data can be decrypted with a DES3 (DDES) key (A-B-A key structure).

RSA2048: Encrypted data can be decrypted with an RSA2048 key.

Digital Sign;

RSA2048: Plain data can be signed with an RSA2048 key.

Digital Sign Verification;

RSA2048: Signed data with the length equal to the RSA2048 key modulus length can be verified with an RSA2048 key.

Data Integrity;

DES-MAC: Cryptographic checksum is calculated with a DES key.

Hash;

SHA-1: Data can be hashed with SHA-1 algorithm.

Smart cards are used as electronic authentication keys, digital signs, GSM cards and bank cards. Also, they are used as electronic passports and e-government cards such as personal identification and health care cards.

Basically smart card consists of 3 main parts:

- Metallic unit on plastic material which is called plastic module (physical plastic card)
- Silicon chip located in the metallic unit on the plastic module. This chip consists of microprocessor, ROM, RAM, EEPROM and some hardware units (decoders, advanced

crypto engine, RNG, MED)

- Operating system (written in ROM and enables the operation of card functions using hardware units)

From the 3 parts listed above, only the third one is developed by TÜBİTAK - BİLGEM - UEKAE. The first part is developed by a card manufacturer company (who provides the conditions that are presented in AKiS_TeslimveIsletim document) and the second part is developed by Infineon Company. The second part has EAL 5+ (compatible with BSI0002) certificate. TOE operates on Infineon's SLE66CX680PE chip. Chip consists of; 8051 based microprocessor, ROM, EEPROM, RAM, Advanced Crypto Engine (ACE), Random Number Generator, MMU, UART, Timers and MED.

TOE is embedded in ROM during chip manufacturing and can't be changed afterwards. However, data can be written into EEPROM under operating system's control.

TOE will be located in a smart chip planted to a plastic card. The interface of the card to the outside world is over a smart card reader or access device such as POS (Point of Sale) machine. PC (over the smart card reader) or access device transmits the commands to the smart card. Incoming commands are interpreted by TOE and the response is transmitted back to the access device or to the PC over smart card reader (Figure 4).

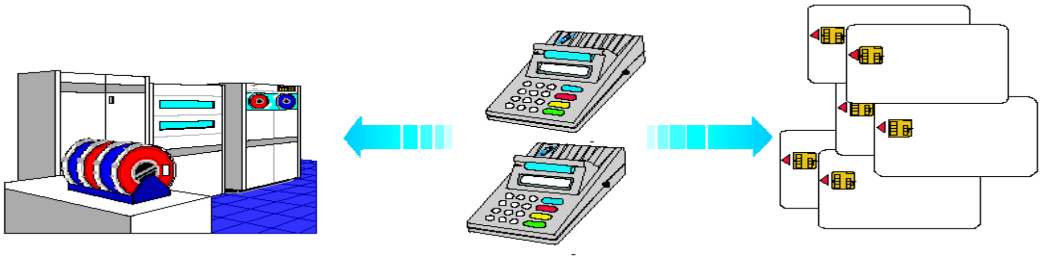


Figure 4 - TOE's environment

The smart card has 8 pins according to the IEC/ISO 7816-2 which is shown in Figure 5. Smart card communicates with reader via I/O pin. 2 pins are reserved for future use. In the past, smart card's EEPROM was being programmed by Vpp pin which is not used anymore. VCC, GND, RST and CLK pins are used to operate the smartcard.

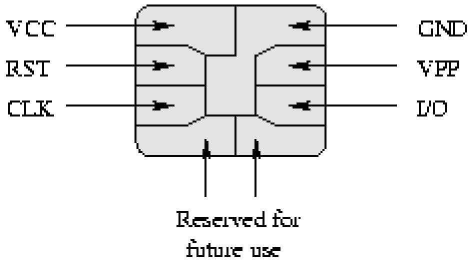


Figure 5 - Smart Card Connection Pins

TOE Components

TOE components are Figure 6;

- Memory Manager
- File Manager
- Command Interpreter
- Communication Handler

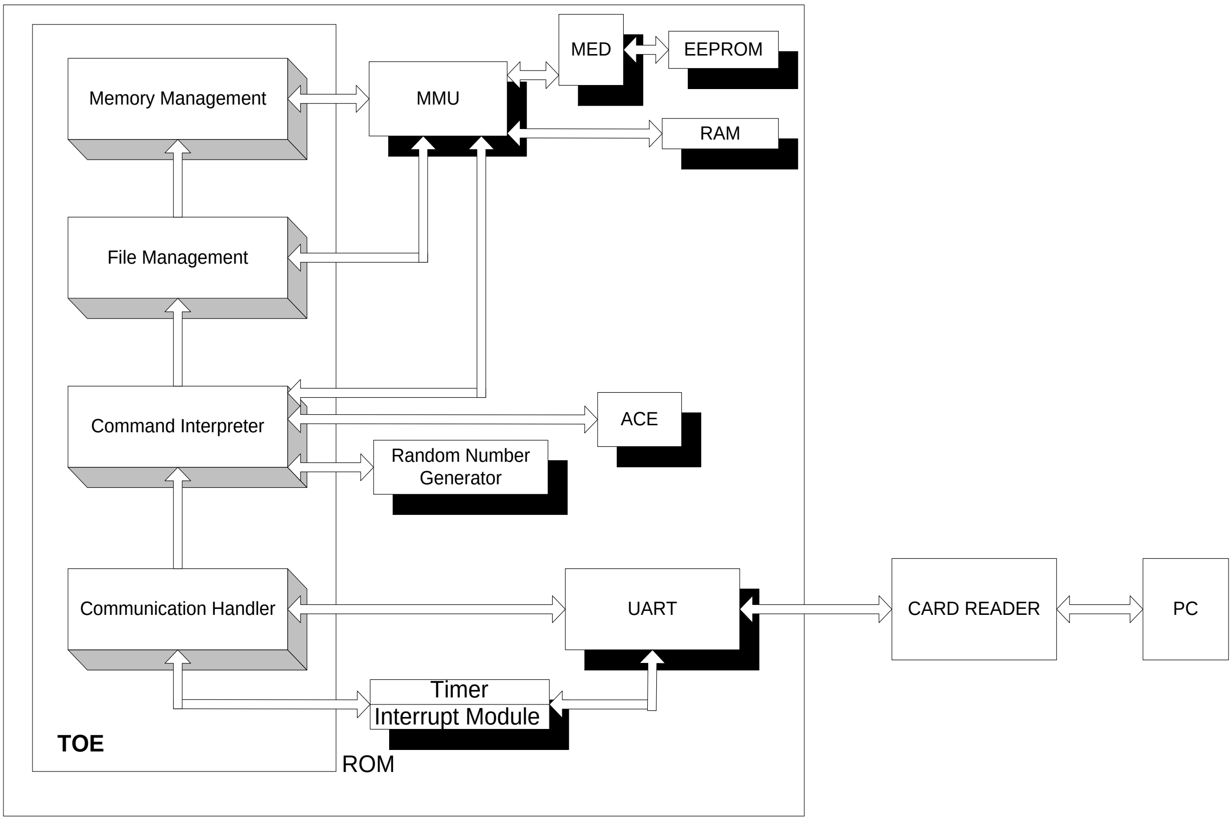


Figure 6 - TOE's components and environment

Message is received by UART which is managed by communication handler in TOE. The message comes in TPDU format which is mentioned above. Incoming TPDU packet is analysed and block type decision is made by the communication handler. TPDU can include 3 different types of block, named R, S and I block. R and S blocks are used to control the protocol. I block carries the command which is transmitted to the command interpreter and executed in TOE. When command execution is finished, communication handler sends the answer to the reader via UART. If the command is related with the file system, command interpreter calls the file manager. File manager is responsible for the operations in the file field which is in the EEPROM. Memory manager is used to open new file, close file, delete page and attach new page.

TOE Scope

TOE’s scope and boundaries are shown in Table 4. During TOE evaluation a PC/SC compatible smart card reader is needed. Smart card reader’s driver and smart card communication software must be installed to the computer for operation.

TOE	AKiS(Akıllı Kart İşletim Sistemi) v1.2.2i
Hardware	Infineon SLE66CX680PE/m1534a13 chip (ECO2000 CPU, 244K ROM , 68K EEPROM, 6K External RAM, MMU, UART, Timers, MED, ACE Advanced Crypto Engine, RNG (Random Number Generator)

Table 4 - TOE's Scope and Boundaries

7. ASSUMPTIONS AND CLARIFICATION OF SCOPE

TOE consists of the components which are defined in section 6 (Architectural information). Except these, Other components are not in the scope of Common Criteria Evaluation.

7.1 Assumptions

Assumptions on the TOE delivery process phase 1 to phase 7	**A.DLV_CONTROL** procedures must guarantee the control of the TOE delivery and storage process and conformance to its objectives as described in the following secure usage assumptions. Secure storage and handling procedures are applicable for all TOE's parts (programs, data, documents).
	A.DLV_CONF procedures must also prevent if applicable any non-conformance to the confidentiality convention and must have a corrective action system in case any non-conformance or misprocessed procedures are identified.
	A.DLV_PROTECT procedures shall ensure protection of material/information under delivery including the following objectives: • non-disclosure of any security relevant information, • identification of the elements under delivery, • meeting confidentiality rules (confidentiality level, transmittal form, reception acknowledgment), physical protection to prevent external damage.
	A.DLV_TRANS procedures shall ensure that material/information is delivered to the correct party.
	A.DLV_TRACE procedures shall ensure traceability of delivery including the following parameters: • origin and shipment details, • reception, reception acknowledgment, • location material/information.
	A.DLV_AUDIT procedures shall ensure that corrective actions are taken in case of improper operation in the delivery process and highlight all non-conformances to this process.
	A.DLV_RESP procedures shall ensure that people dealing with the procedures for delivery have got the required skill, training and knowledge to meet the procedure requirements and to act to be fully in accordance with the above expectations.
Assumptions on IC development (smartcard product life cycle phase 2)	**A.IC_PRODUCT** the Smartcard integrated circuit is designed and built using state of art technology with the aim of achieving security objectives.
	A.IC_ORG procedures dealing with physical, personnel, organizational, technical measures for the confidentiality and integrity of smartcard embedded software and data (e.g. source code and any associated documents) shall exist and be applied in the smartcard IC database construction.

Assumptions on smartcard product life cycle phases 3 to 6	**A.USE_TEST** it is assumed that appropriate functionality testing of the smartcard functions is used in phases 3 to 6.
	A.USE_PROD it is assumed that security procedures are used during all manufacturing and test operations through smartcard production phases to maintain the confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorized use).
Assumption on smartcard product life cycle phase 7 and on delivery to these phases	**A.USE_SYS** it is assumed that the security of sensitive data stored/handled by the system (terminals, communications ...) is maintained.
Assumption on the intended usage of the TOE, related with TOE Life Cycle Phases (Figure 1) except Activation phase	**A.USE_OPR** after giving a warning message from TSF for corrupted objects (DF-EF-DF PIN-DF PUK, System PIN, System PUK), it is assumed that the user knows which corrupted objects can be used or not without taking any risk for security and availability of the TOE.

Table 5 - Assumptions

7.3 Clarification of Scope

Under normal conditions; there are no threats which TOE must counter but did not; however Operational Environment and Organizational Policies have countered. Information about threats that are countered by TOE and Operational Environmental are stated in the Security Target document.

8. DOCUMENTATION

AKiS v1.2.2I Security Target

Version Number and Date: 05 – 19.04.2011

AKiS v1.2.2I Administrator’s And User’s Guide

Ver. Number and Date: 01 – 30.12.2011

9. IT PRODUCT TESTING

During the evaluation, all evaluation evidences of TOE were delivered and transferred completely to CCTL by the developers. All the delivered evaluation evidences which include software, documents, etc are mapped to the assurance families of Common Criteria and Common Methodology; so the connections between the assurance families and the evaluation evidences has been established. The evaluation results are available in the Evaluation Technical Report (ETR) of AKiS v1.2.2I.

It is concluded that the TOE supports EAL 4+ (ALC_DVS.2, AVA_VAN.5). There are 24 assurance families which are all evaluated with the methods detailed in the ETR.

IT Product Testing is mainly realized in two parts:

1) Developer Testing :

- **TOE Test Coverage**: Developer has prepared TOE System Test Document according to the TOE Functional Specification documentation.
- **TOE Test Depth:** Developer has prepared TOE System Test Document according to the TOE Design documentation which includes TSF subsystems and its interactions.
- **TOE Functional Testing:** Developer has made functional tests according to the test documentation. Test plans, test scenarios, expected test results and actual test results are in the test documentation.

2) Evaluator Testing :

- **Independent Testing:** Evaluator has done a total of 33 sample independent tests. 20 of them are selected from developer`s test plans. The other 13 tests are evaluator`s independent tests. All of them are related to TOE security functions.
- **Penetration Testing:** Evaluator has done 16 penetration tests to find out if TOE`s vulnerabilities can be used for malicious purposes. The potential vulnerabilities and the penetration tests are in “TOE Security Functions Penetration Tests Scope” which is in Annex-C of the ETR and the penetration tests and their results are available in detail in the ETR document as well.

The result of AVA_VAN.5 evaluation is given below:

- It is determined that TOE, in its operational environment, is resistant to an attacker possessing "**High**" attack potential.
- For the product AKiS v1.2.2I, **there are no exploitable vulnerabilities** in the scope of the assumptions in ST (Competent Administrators, Officers and Auditors will be assigned to manage the TOE and the information it contains and authorized users will not intentionally perform hostile actions).

10. EVALUATED CONFIGURATION

During the evaluation; the configuration of evaluation evidences which are composed of Software source code, Common Criteria documents, sustenance document and guides are shown below:

Evaluation Evidence: TOE – AKiS Smart Card Operating System (Akıllı kart işletim sistemi)

Version Number: 1.2.2i

Production Date : 29.03.2011

Evaluation Evidence: AKiS v1.2.2i Source Code (Kaynak Kodu)

Version Number and Date: 1.0 – 01.02.2011

Evaluation Evidence: AKiS v1.2.2i Detailed Design Document (Ayrıntılı Tasarım Dokümanı)

Version Number and Date: 03 – 19.04.2011

Evaluation Evidence: AKiS v1.2.2i Functional Specification Document (Fonksiyonel Belirtim Dokümanı)

Version Number and Date: 04 – 09.06.2011

Evaluation Evidence: AKiS v1.2.2i Security Architecture Document (Güvenlik Mimari Dokümanı)

Version Number and Date: 03 – 19.04.2011

Evaluation Evidence: AKiS v1.2.2i Delivery and Usage Document (Teslim ve İşletim Dokümanı)

Version Number and Date: 06 – 09.06.2011

Evaluation Evidence: AKiS v1.2.2i Configuration Management Document (Konfigürasyon Yönetim Planı)

Version Number and Date: 06 – 10.06.2011

Evaluation Evidence: AKiS v1.2.2i Development Environment Security and Development Tools (Geliştirme Ortam Güvenliği ve Geliştirme Aletleri Dokümanı)

Version Number and Date: 03 – 19.04.2011

Evaluation Evidence: AKiS v1.2.2i Life Cycle Document (Kullanım Ömrü (Yaşam Çevrimi) Dokümanı)

Version Number and Date: 03 – 19.04.2011

Evaluation Evidence: AKiS v1.2.2i Security Target Dokümanı

Version Number and Date: 05 – 19.04.2011

Evaluation Evidence: AKiS v1.2.2i System and Test Document (Sistem ve Test Dokümanı)

Version Number and Date: 05 – 09.06.2011

Evaluation Evidence: AKiS v1.2.2i Administrator and User Manual Document (Yönetici ve Kullanıcı Kılavuzu Dokümanı)

Version Number and Date: 07 – 09.06.2011

Evaluation Evidence: AKiS v1.2.2i Differences between Versions (Versiyonlar Arası Farklar Dokümanı)

Version Number and Date: 06 – 09.06.2011

11. RESULTS OF THE EVALUATION

Table 8 below provides a complete listing of the Security Assurance Requirements for the TOE. These requirements consists of the Evaluation Assurance Level 4 (EAL 4) components as specified in Part 3 of the Common Criteria, augmented with ALC_DVS.2 and AVA_VAN.5.

Component ID	Component Title
ASE_INT.1	ST Introduction
ASE_CCL.1	Conformance Claims
ASE_SPD.1	Security Problem Definition
ASE_OBJ.2	Security Objectives
ASE_ECD.1	Extended Components Definition
ASE_REQ.2	Security Requirements
ASE_TSS.1	TOE Summary Specification
ADV_ARC.1	Security Architecture
ADV_FSP.4	Functional Specification
ADV_IMP.1	Implementation Representation
ADV_TDS.3	TOE Design
AGD_OPE.1	Operational User Guidance
AGD_PRE.1	Preparative Procedures
ALC_CMC.4	Configuration Management Capabilities
ALC_CMS.4	Configuration Management Scope
ALC_DEL.1	Delivery
ALC_DVS.2	Development Security
ALC_LCD.1	Life-Cycle Definition
ALC_TAT.1	Tools and Techniques
ATE_COV.2	Coverage
ATE_DPT.1	Depth
ATE_FUN.1	Functional Tests
ATE_IND.2	Independent Testing
AVA_VAN.5	Vulnerability Analysis

Table 6 - Security Assurance Requirements for the TOE

The Evaluation Team assigned a Pass, Fail, or Inconclusive verdict to each work unit of each EAL 4 assurance component. For Fail or Inconclusive work unit verdicts, the Evaluation Team advised the developer about the issues requiring resolution or clarification within the evaluation evidence. In this way, the Evaluation Team assigned an overall Pass verdict to the assurance component only when all of the work units for that component had been assigned a Pass verdict. So for TOE AKiS v1.2.2I the result of the assessment of all evaluation tasks are “Pass”.

<u>Results of the evaluation:</u>

AKiS v1.2.2I product was found to fulfill the Common Criteria requirements for each of 24 assurance families and provide the assurance level EAL 4+ (ALC_DVS.2, AVA_VAN.5) .This result shows that TOE is resistant against the “HIGH’’ level attack potential and it countervails the claims of the functional and assurance requirements which are defined in ST document.

There is no residual vulnerability (vulnerabilities can be used as evil actions by the hostile entities who have BEYOND HIGH level attack potential), that they do not affect the evaluation result, found by CCTL(OKTEM) laboratory under the conditions defined by the evaluation evidences and developer claims.

12. EVALUATOR COMMENTS/ RECOMMENDATIONS

No recommendations or comments have been communicated to CCCS by the evaluators related to the evaluation process of AKiS v1.2.2I product, result of the evaluation, or the ETR.

13. CERTIFICATION AUTHORITY COMMENTS/ RECOMMENDATIONS

The certifier has no comments or recommendations related to the evaluation process of AKiS v1.2.2I product, result of the evaluation, or the ETR.

14. SECURITY TARGET

Information about the Security Target document associated with this certification report is as follows:

Name of Document	**:** AKiS V1.2.2I Security Target
Version No.	**:** 05
Date of Document	**:** 19.04.2011

This Security Target describes the TOE, intended IT environment, security objectives, security requirements (for the TOE and IT environment), TOE security functions and all necessary rationale.

15. BIBLIOGRAPHY

[1] Common Criteria for Information Technology Security Evaluation, Version 3.1 Revision 3, July 2009

[2] Common Methodology for Information Technology Security Evaluation, CEM, Version 3.1 Revision 3, July 2009

[3] AKIS v1.2.2I Security Target Version: 05 Date: 19.04.2011

[4] Evaluation Technical Report (Document Code: DTR 09 TR 02), June 13, 2011

[5] PCC-03-WI-04 CERTIFICATION REPORT PREPARATION INSTRUCTIONS, Version 2.0

[6] CC Supporting Document Guidance, Mandatory Technical Document, Application of Attack Potential to Smartcards, Version 2.7 Revision 1, March 2009, CCDB-2009-03-001

[7] CC Supporting Document Guidance, Mandatory Technical Document, Application of CC to Integrated Circuits, Version 3.0 Revision 1, March 2009, CCDB-2009-03-002

[8] Common Criteria Protection Profile Machine Readable Travel Document with “ICAO Application”, Basic Access Control, BSI-CC-PP-0055, Version 1.10, 25th March 2009

[9] Common Criteria Protection Profile Machine Readable Travel Document with “ICAO Application”, Extended Access Control, BSI-CC-PP-0026, Version 1.2, 19 November 2007, BSI

[10] Joint Interpretation Library, Attack Methods for Smartcards and Similar Devices, confidential Version 1.5, February 2009, BSI

16. APPENDICES

There is no additional information which is inappropriate for reference in other sections.