INTERNATIONAL STANDARD

ISO 10202-7

First edition
1998-07-15

Financial transaction cards — Security architecture of financial transaction systems using integrated circuit cards —

Part 7:
Key management

Cartes de transactions financières — Architecture de sécurité des systèmes de transactions financières utilisant des cartes à circuit intégré —

Partie 7: Gestion de clé

Reference number
ISO 10202-7:1998(E)

Contents

International Organization for Standardization
Case postale 56 • CH-1211 Genève 20 • Switzerland
Internet iso@iso.ch

Printed in Switzerland

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

International Standard ISO 10202-7 was prepared by Technical Committee ISO/TC 68, *Banking, securities and other financial services*, SC 6, *Retail financial services*.

ISO 10202 consists of the following parts, under the general title *Financial transaction cards — Security architecture of financial transaction systems using integrated circuit cards*:

— *Part 1: Card life cycle*

— *Part 2: Transaction process*

— *Part 3: Cryptographic key relationships*

— *Part 4: Secure application modules*

— *Part 5: Use of algorithms*

— *Part 6: Cardholder verification*

— *Part 7: Key management*

— *Part 8: General principles and overview*

Annexes A to E of this part of ISO 10202 are for information only.

Financial transaction cards — Security architecture of financial transaction systems using integrated circuit cards —

Part 7:
Key management

1 Scope

This part of ISO 10202 specifies key management requirements for financial transaction systems using integrated circuit cards. It defines procedures and processes for the secure management of cryptographic keys used during the card life cycle and transaction processing in an integrated circuit card environment. Both symmetric and asymmetric key management schemes are addressed. Minimum key management requirements are specified.

Key management is the process whereby cryptographic keys are provided for use between authorized communicating parties and those keys continue to be subject to secure procedures until they are destroyed. The security of the enciphered data is dependent upon the prevention of disclosure and unauthorized modification, substitution, insertion, or deletion of keys. Thus, key management is concerned with the generation, storage, distribution, use and destruction procedures for keys. Also, by the formalization of such procedures, provision is made for audit trails to be established.

This part of ISO 10202 is applicable between the ICC and the SAM in both on-line and off-line transaction processing environments, and between the ICC and the SAM or host security module in an on-line (end-to-end) environment.

2 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this part of ISO 10202. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this part of ISO 10202 are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards.

ISO 7812 (all parts), *Identification cards — Identification of issuers*.

ISO 7816-3, *Information technology — Identification cards — Integrated circuit(s) cards with contacts — Part 3: Electronic signals and transmission protocols*.

ISO 7816-4, *Information technology — Identification cards — Integrated circuit(s) cards with contacts — Part 4: Interindustry commands for interchange*.

ISO 7816-5, *Identification cards — Integrated circuit(s) cards with contacts — Part 5: Numbering system and registration procedure for application identifiers*.

ISO 8732, *Banking — Key management (wholesale)*.

ISO 8908, *Banking and related financial services — Vocabulary and data elements*.

ISO 9796, *Information technology — Security techniques — Digital signature schemes giving message recovery*.

ISO 9992-1, *Financial transaction cards — Messages between the integrated circuit card and the card accepting device — Part 1: Concepts and structures*.

ISO 9992-2, *Financial transaction cards — Messages between the Integrated Circuit Card and the Card Accepting Device — Part 2: Functions, messages (commands and responses), data elements and structures*.

ISO 10202-1, *Financial transaction cards — Security architecture of financial transaction systems using integrated circuit cards — Part 1: Card life cycle.*

ISO 10202-2, *Financial transaction cards — Security architecture of financial transaction systems using integrated circuit cards — Part 2: Transaction process*.

ISO 10202-3, *Financial transaction cards — Security architecture of financial transaction systems using integrated circuit cards — Part 3: Cryptographic key relationships*.

ISO 10202-4, *Financial transaction cards — Security architecture of financial transaction systems using integrated circuit cards — Part 4: Secure application modules*.

ISO 10202-5, *Financial transaction cards — Security architecture of financial transaction systems using integrated circuit cards — Part 5: Use of algorithms*.

ISO 10202-6, *Financial transaction cards — Security architecture of financial transaction systems using integrated circuit cards — Part 6: Cardholder verification*.

ISO 10202-8, *Financial transaction cards — Security architecture of financial transaction systems using integrated circuit cards — Part 8: General principles and overview*.

ISO 11568 (all parts), *Banking — Key management (retail)*.

ISO 13491 (all parts), *Banking — Secure cryptographic devices (retail)*.

3 Definitions and abbreviations

3.1 Definitions

For the purposes of this part of ISO 10202, the following definitions apply.

3.1.1
application data file
a file that supports one or more services

3.1.2
asymmetric algorithm
an algorithm for which the encipherment and decipherment keys are different and where it is computationally infeasible to deduce one from the other

3.1.3
authentication
a process used to ensure data integrity and data origin authentication

3.1.4
certificate
(See transaction certification code and public key certificate.)

3.1.5
certificate identifier
certificate information which enables proper verification of a key certificate

3.1.6
certification authority
an authority trusted by all users to create and assign certificates

3.1.7
common data file
a mandatory file which contains the common data elements used to describe the card, the card issuer and the cardholder

3.1.8
cryptographic function
a process performed (e.g. encryption, authentication, certification) using a cryptographic algorithm

3.1.9
cryptographic key (key)
a parameter used in conjunction with a cryptographic algorithm for executing cryptographic transformations

3.1.10
cryptoperiod
a defined period of time within which a cryptographic key is authorized for use, or during which time the cryptographic keys for a given system may remain in effect

3.1.11
data key
a cryptographic key used for the encipherment, decipherment or authentication of data

3.1.12
decipherment
the process of transforming ciphertext into plaintext

3.1.13
derivation key
a key used to generate a derived key

3.1.14
derived key
a symmetric key generated from a derivation key and non-secret variable data

NOTE The derivation key is used to generate a large number of keys (derived keys).

3.1.15
diversified key
(See derived key.)

3.1.16
dual control
a process of utilizing two or more separate entities (usually persons), operating in concert, to protect sensitive functions or information whereby no single entity is able to access or utilise the material (e.g. cryptographic key)

3.1.17
elementary file
a file which may contain data and/or file control information

3.1.18
explicit key identifier
(See key identifier.)

3.1.19
encipherment
the process of transforming plaintext into ciphertext

3.1.20
host/SAM derivation key
a derivation key used to derive ICC or SAM keys

3.1.21
host security module
a physically secure device used to support cryptographic functions and perform SAM functionality on a host system

3.1.22
ICC derivation key
an ICC (CDF or ADF) derivation key used to derive unique message data keys

3.1.23
key enciphering key
a key used to encipher another key

3.1.24
key generation module
a type of cryptographic equipment used for generating and deriving cryptographic keys

3.1.25
key identifier
specifies basic security requirements for the ICC

3.1.26
key loading module
an electronic, self-contained unit which is capable of storing at least one cryptographic key and transferring that cryptographic key, upon request, into a cryptographic device such as an ICC or a SAM

3.1.27
key synchronization
the process whereby two nodes verify that they are communicating with each other using an identical key

3.1.28
keying material
the data necessary to establish and maintain a keying relationship

3.1.29
master derivation key
a derivation key used by a bank card company or another organization to derive unique issuer or application supplier keys

3.1.30
physically secure device
(See ISO 13491.)

3.1.31
physically secure environment
(See ISO 11568.)

3.1.32
public key
that part of an asymmetric key set which is known to other parties than the generator of the key set

3.1.33
public key certificate
a set consisting of user credentials (including the public key) together with the trusted third party's digital signature of these credentials

3.1.34
secure cryptographic device
a device that provides secure storage for secret information such as keys and provides security services based on this secret information

3.1.35
secure application module
a physical module (or logical functionality in the CAD) intended to contain algorithm(s), related keys, security procedures and information to protect an application in such a way that unauthorized access is not possible

NOTE In order to achieve this the module shall be physically and logically protected.

3.1.36
symmetric algorithm
a cryptographic method using the same secret cryptographic key for encipherment and decipherment

3.1.37
tamper resistance
provision of physical protection for sensitive data, for the purpose of preventing successful attacks

3.1.38
transaction certification code
result of the transformation certification process producing an electronic signature, which could be either a MAC (based on a symmetric algorithm) or a digital signature (based on an asymmetric algorithm)

3.2 Abbreviations

ADF Application data file

CAD Card accepting device

CDF Common data file

CID Certificate identifier

e(..) Encipherment

EF Elementary file

IC Integrated circuit

ICC Integrated circuit card

KCD ICC (CDF or ADF) derivation key

KD Data key

Kx x is either I or A

KEK Key enciphering key

KHD Host or SAM derivation key

KID Key identifier

KMD Master derivation key

KSN Key sequence number

KVC Key verification code

S(..) Sign

SAM Secure application module

4 General security principles

Key management in financial transaction systems using integrated circuit cards shall conform to the following basic principles.

a) The key management adopted for one ICC system, which may include SAMs, shall not compromise the security of any other such system.

b) The key management adopted for one application in one ADF shall not compromise the security of any other application in any other ADF.

c) The ICC and SAM shall afford tamper resistance based on the principles described in ISO 10202-2 and ISO 10202-4.

d) The keying relationship shall be in accordance with ISO 10202-3.

e) The use of cryptographic algorithms to perform cryptographic functions shall be in accordance with ISO 10202-5.

f) Controls and audits shall be in force for key management of ICC, SAM, key generation and loading modules, host security modules, and other cryptographic devices used in financial transaction systems using integrated circuit cards.

Annex A (informative) provides an example of card life cycle key management.

Annexes B and C (informative) provide examples of symmetric key management techniques for transaction processing.

Annexes D and E (informative) provide examples of asymmetric key management.

5 ICC systems key management requirements

5.1 ICC and SAM life cycle

During the life cycle of the ICC and SAM, manual and automated key management processes shall provide the ability to load, update and disable cryptographic keys under the control of the party performing these key management functions. The key management processes used shall meet the cryptographic key relationship requirements defined in ISO 10202-3.

Protection of secret cryptographic keys of symmetric and asymmetric key management schemes shall be provided during all steps of the ICC and SAM life cycle when cryptographic keys are used. The manual procedures and automated processes used to protect cryptographic keys during the card life cycle shall meet the protection requirements defined in this part of ISO 10202.

5.2 Key life cycle protection

The key life cycle and protection requirements for key generation, storage, backup, distribution and loading, use, replacement, destruction, deletion, archive and termination, shall comply with those defined in this part of ISO 10202.

5.3 Key separation

In ICCs, SAMs and host security modules different key names shall be cryptographically separated from each other to ensure that cryptographic processing can operate only with the specific functional key names described in this part of ISO 10202.

Key separation shall be achieved by using keys which are separately generated or derived for each function. An ICC or SAM key of a certain name shall not be a variant, transformation, or derived from a key of another name.

5.4 Key management services

The key management services used shall implement techniques which ensure key separation, substitution protection, identification, integrity, and confidentiality, as described in this part of ISO 10202.

5.5 Key relationships

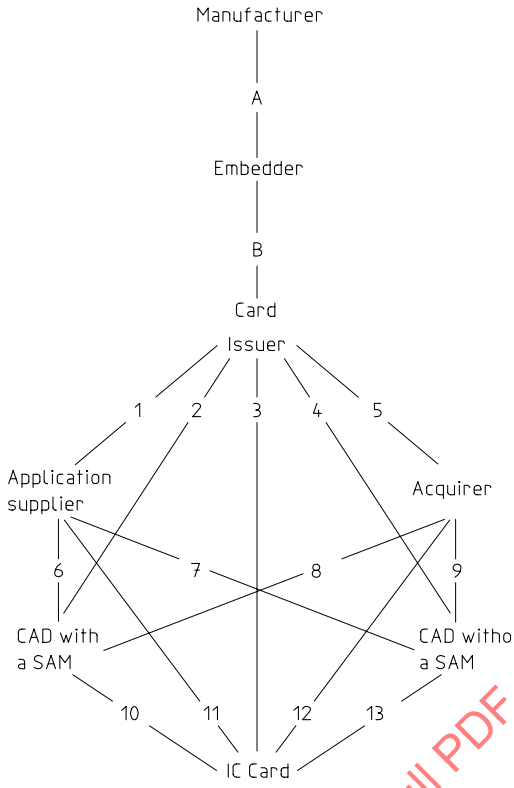
A key relationship shall exist when two parties share at least one cryptographic key. Figure 1 describes cryptographic key relationships in a financial transaction system using ICC and indicates where this part of ISO 10202 applies.

Key management procedures and processes for key relationships shall be agreed by the communicating parties. Contractual agreement defining the liabilities of each party responsible for the protection of cryptographic keys and of information protected using cryptographic keys, is outside the scope of this part of ISO 10202.

Public key management functions performed in a CAD without SAM shall be selected based on the security available in the CAD and meet the requirements of this part of ISO 10202.

The key management procedures and processes used in cryptographic key relationships not covered by this part of ISO 10202 (Figure 1, relationships 1 and 5) which may be part of a financial transaction system using ICCs, shall comply to ISO 11568. In these relationships the key numbering shall use the application identifier (AID) as defined by ISO 7812.

NOTE The AID may be limited to the registration identifier (RID) defined by ISO 7816-5.



Cryptographic key relationships covered by this part of ISO 10202: A, B, 2, 3, 6, 8, 10, 11, 12, 13

Cryptographic key relationships not covered by this part of ISO 10202: 1 and 5

Cryptographic key relationships using only asymmetric key management: 4, 7, 9, 13

Figure 1 — Cryptographic key relationships in a financial transaction system using ICCs

5.6 On-line transaction processing

During on-line transaction processing the automated key management process shall ensure protection during creation, transportation, and use of keys between the ICC, SAM or host security module(s).

Cryptographic keys transmitted to an ICC shall be enciphered end-to-end between the SAM or host security module and the ICC, under the control of the responsible parties.

5.7 Off-line transaction processing using a SAM

During off-line processing using a SAM the automated key management process shall have the ability to establish and maintain secure cryptographic key relationship(s) between the ICC and the SAM where secret keys are used.

5.8 CDF and ADF keys

No cryptographic key used in a CDF or an ADF shall intentionally be the same as a cryptographic key used in another ADF. CDF and ADF shall be cryptographically separated at ADF allocation as specified in ISO 10202-3 except when they belong to the same issuer/application supplier.

5.9 Physical security

The tamper resistance afforded by the ICC and SAM shall be based on the security principles described in ISO 10202-2 and ISO 10202-4. Although the ICC and SAM are not intended to be physically secure devices, they must provide a high level of tamper resistance.

5.10 CADs without a SAM

ICC authentication and transaction certificate verification may be implemented in a CAD without a SAM. Public keys are used by the CAD for authentication or certificate verification. The integrity of these public keys may be verified using a public key belonging to a higher authority such as the card issuer or the application supplier key certification centre.

When a CAD is being used as a communicating device for on-line authentication with a card issuer, an application supplier, or an acquirer, cryptographic key relationships 3, 11 or 12 shall apply.

6 ICC systems cryptographic keys

This clause defines the cryptographic keys which can be used in ICC systems which may include SAMs.

6.1 Definition of cryptographic keys

The following cryptographic key names are defined:

Cryptographic key name	Purpose
kMprd, kEprd	Control transfer of an IC and protect against substitution of an IC (Manufacturer, Embedder).
kIctl, kActl	Load CDF or ADF cryptographic keys.
kIaut, kAaut	Authenticate CDF or ADF.
kImac, kAmac	Authenticate CDF or ADF transaction commands and data.
kIenc, kAenc	Encipher CDF or ADF transaction data.
kIcer, kAcer	Generate CDF or ADF transactions certificates.
kI(i)kex	Load a *kA(i,j)ctl* for an ADF.
kA(i,j)ctl	Load ADF cryptographic keys.
kA(i,j)aut	Authenticate an ADF.
kA(i,j)mac	Authenticate ADF transaction commands and data.
kA(i,j)enc	Encipher ADF transaction data.
kA(i,j)cer	Generate ADF transaction certificates.

k is a generic notation for a key (*K*, *P* or *S*) which can be either a key for a symmetric or asymmetric algorithm. Symmetric keys are denoted by *K* (eg. *KIctl*). Public/secret key pairs for the above key names are prefixed by *P* and *S* (e.g. *PIctl*/*SIctl*).

Index i denotes a specific ADF, and index j denotes a specific key or set of keys related to one ADF.

The key relationship for each of the above keys is described in ISO 10202-3.

6.2 Key hierarchy

ICC and SAM cryptographic keys shall be loaded according to ISO 10202-3. The following hierarchy may be applied:

a) Production keys: Production keys are key enciphering keys used to load control keys and to protect against IC substitution. *kMprd* and *kEprd* shall be the only keys used to load control keys.

b) Control keys: Control keys are key enciphering keys used to load other cryptographic keys and parameters in an ICC or SAM. *kIctl* and *kActl* shall be the only keys used as control keys.

c) Key exchange keys: A key exchange key is a key used to provide cryptographic key separation in loading an ADF. *kIkex* shall be the only key used to load *kActl* in an ADF.

d) Data keys: Data keys are keys used for encipherment and decipherment (*kIenc,kAenc*), authentication of SAM, host security module, ICC and ADF (*kIaut,kAuat*), authentication of data (*kImac,kAmac*), and certification (*kIcer,kAcer*).

SAM key names may be:

a) Derivation keys: A key used to derive ICC keys.

b) Derived derivation keys: Keys derived from a master derivation key used to derive ICC Keys. Derived keys are denoted as *K'* (e.g. *KI'aut*).

c) An asymmetric key or key pair.

ICC key names may be:

a) Key enciphering keys: A key derived from a SAM derivation key or SAM derived derivation key (*K'*) used in an ICC as a key enciphering key to exchange session keys.

b) Derived keys: Keys derived from a SAM derivation key. Derived keys are denoted as *K'* (e.g. *KI'aut*).

c) Doubly derived keys: Keys derived from a SAM derived derivation key (*K'*) used to derive data keys. Doubly derived keys are denoted as *K''* (e.g. *KI''aut*).

d) An asymmetric key or key pair.

Data key names may be:

a) Derived keys: Keys derived from a SAM derivation key. Derived keys are denoted as *K'* (e.g. *KI'aut*).

b) Doubly derived keys: Keys derived from a derived SAM keys (*K'*). Doubly derived keys are denoted as *K''* (e.g. *KI''aut*).

c) Triply derived key: Keys derived from a doubly derived SAM and ICC key (*k''*). Triply derived keys are denoted as *K'''* (e.g. *KI'''aut*).

d) An asymmetric key or key pair.

7 Key life cycle

Key management involves the generation of suitable keys, their storage, their distribution to and by authorized recipients, their use, and their termination once they are no longer required. To protect keys during their lifetime keys are processed through a series of stages called key life cycle (ISO 10202-1 and ISO 10202-3). This clause describes the key life cycle protection requirements for ICC and SAM systems.

7.1 Key generation

Cryptographic keys used in ICCs, SAMs, and host security modules shall be randomly or pseudo-randomly generated, or cryptographically derived from other keys as defined in this part of ISO 10202.

7.2 Key storage

ICC, SAM, and host security module keys shall be protected during storage. Cryptographic keys residing outside the confines of ICCs, SAMs, host security modules, and other secure cryptographic devices, shall be stored in parts under dual control and split knowledge, or stored as key cryptograms under a key intended for storage purposes.

7.3 Key backup

Key backup occurs when a protected copy of a key is kept in storage during its operational use. The security protection requirements over cryptographic keys defined in this part of ISO 10202 apply to backup keys.

7.4 Key distribution and loading

Key distribution and loading is the process by which a key is manually or electronically transferred into a secure cryptographic device.

The key distribution process used for ICCs, SAMs and host security modules keys shall not disclose any secret keys, and shall protect public keys against substitution.

Plaintext secret keys shall be loaded in ICCs, SAMs, host security modules, and other secure cryptographic devices used in ICC systems, only when it has been assured that such devices have not been subject to prior tampering which might lead to disclosure or substitution of keys or sensitive data.

ICC and SAM key loading procedures shall meet the requirements defined in this part of ISO 10202.

7.5 Key use

Key use occurs when a key is employed for the cryptographic purpose for which it was intended. Unintended key use shall be prevented; therefore,

1) A key shall only be used for one purpose (see 6.1).

2) A key shall only be used for its intended purpose (see 6.1).

3) SAM and host security module keys shall exist in locations consistent with system operation.

The key management services defined in this part of ISO 10202 shall be used to provide the required security.

7.6 Key replacement

Key replacement occurs when a key is substituted for another when the original key is known or suspected to be compromised, or the end of its operational life is reached. Key replacement shall be irreversible.

Cryptographic keys in ICCs, SAMs, and host security modules used in ICC systems shall be replaced within the time deemed feasible to perform a dictionary attack upon the data enciphered under this key or within the time deemed necessary to determine the key by exhaustive attack. Alternately the life of an ICC key shall be longer than the life of the card.

The cryptographic period of a key should be set according to the potential misuse of a key, the level of security required, by considering the sensitivity of the information, the cost of breaking the cryptographic keys in use, and the degree of tamper resistance provided by the ICC, the SAM and/or host security module.

Replacement of a key known or suspected to be compromised shall only be performed by distributing a new key.

Replacement of a key which has reached the end of its operational life may be performed by distributing a new key, or replacing the key by a new key derived as specified in this part of ISO 10202.

If it is believed that unauthorized substitution of a secret key has occurred, then all associated keys shall be replaced once the associated cryptographic devices have been secured.

If it is believed that a public key has been added or substituted without authorization, then the bogus key shall be replaced by the original public key and its certificate once the associated ICCs, SAMs, host security modules, and CAD have been secured. The public key of the higher authority shall be used to verify the certificate of the reloaded key.

Replaced secret keys shall not be returned to operational use.

7.7 Key destruction

Key destruction ensures that an instance of a key no longer exists at a specific location. Information may still exist at the location from which the key can be feasibly reconstructed for subsequent use.

An instance of a key in SAMs and host security modules shall be destroyed when it is no longer required for use. For ICCs key destruction is not required.

Measures such as verification of key identifiers and maintenance of negative files of lost and stolen ICCs and SAMs should be in place to prevent the use of keys which are no longer active.

7.8 Key deletion

Key deletion is the process by which an unwanted key, and information from which the key may be reconstructed, is destroyed at its operational/use location. A key may be deleted from one location and continue to exist at another.

Host security module keys used in ICC systems which are no longer required shall be deleted.

Key deletion should be performed whenever possible for ICC and SAM keys which are no longer required. All data in the ICC should be reset to its required state. Measures such as verification of key identifiers and maintenance of negative files of lost and stolen ICCs and SAMs should be in place to prevent the use of keys which are no longer required.

7.9 Key archive

Key archive is the process by which a key which is no longer in operational use at any location is stored.

An archived key shall only be used to verify the legitimacy of transactions that occurred prior to archive. After such verification the instance of the key necessary to perform the verification shall be destroyed.

An archived key shall not be returned to operational use. Archived keys shall be securely stored for the life of all data or keys enciphered under such keys.

7.10 Key termination

Key termination occurs when a key is no longer required for any purpose and all copies of the key and information required to regenerate or reconstruct the key have been deleted from all locations where they ever existed.

Key termination shall be performed for all keys upon which the security or integrity of an ICC system, which may include SAMs, relies (refer to ISO 10202-1 and ISO 10202-4).

7.11 Reserve keys

Cryptographic keys that may be kept in reserve in an ICC or a SAM to facilitate planned or unexpected key changes shall be subject to the same level of protection as keys in current use.

8 Key management services

Key management services shall be used in financial transaction systems using integrated circuit cards to ensure key separation, prevent key substitution or addition, provide key identification, ensure key synchronization, ensure key integrity and confidentiality.

This clause describes techniques which may be used to provide key management services in ICCs and SAMs.

8.1 Key encipherment

Key encipherment is a technique whereby one key is enciphered using another. A key used to perform such encipherment is called a key enciphering key (KEK).

Symmetric encipherment of a single length key using a single or double length key, and encipherment of a double length key using another double length key, if used, shall be performed as described in ISO 11568.

A secret key shall be enciphered for transmission over a non-secure channel, or stored outside an ICC, SAM, host security module, and other secure cryptographic devices.

8.2 Key derivation

Key derivation is a technique for generating a large number of symmetric keys from a single key called a derivation key. Key derivation, when used, shall be performed as described in 10.1.

8.3 Key offsetting

Key offsetting is a technique applicable to symmetric algorithms for calculating a new KEK from an initial KEK, each time the KEK is used to encipher a key for transmission or storage.

8.4 Key notarization

Key notarization may be used to identify communicating parties and protect against key substitution when keys are transmitted.

8.5 Key tagging

Cryptographic keys in ICCs and SAMs shall only be used for their intended purposes. This shall be implemented through key separation and key identification such that keys cannot be accidentally or intentionally misused.

Key tagging may be used in SAMs and host security modules to prevent key misuse when keys are stored enciphered outside these devices.

When key tagging of explicitly identified keys is performed in a SAM, key tagging shall be accomplished by performing a modulo-2 addition of the effective key with its key identifier before it is enciphered for storage.

In host security modules key tagging shall be performed according to ISO 11568.

8.6 Key verification

A key verification code (KVC) is a value cryptographically related to the key and some non-secret information. The KVC is used to detect that: a key has been properly entered into an ICC, SAM, host security module or other cryptographic devices; a key has been correctly received; or has not been changed.

A KVC, if used, shall be calculated by enciphering binary zeros, and truncating the resulting ciphertext, e.g. taking the leftmost 24 bits (6 hexadecimal digits).

8.7 Key identification

ICC cryptographic keys may be implicitly or explicitly identified.

8.7.1 Implicit key identification

With implicit key identification there is no key identification exchanged. The key(s) associated with a transaction is (are) determined from other transaction information.

When implicit key identification is used, the design of the ICC, SAM, and host security modules used shall ensure that protection is provided against key substitution and key misuse.

8.7.2 Explicit key identification

Explicit key identification allows the recipient of a transaction to determine the appropriate key(s) associated with the transaction using a key identifier.

An ICC cryptographic key which is explicitly identified shall have a key identifier. The key identifier (KID) contains all the necessary information about the key function, cryptographic algorithm, and key to allow the correct usage of the key.

A KID used in ICCs and SAMs may be defined as follows:

<KID> = <format>| <purpose>| <algorithm>| <technique>| <number>

<format> : ISO, non - ISO

<purpose> : e.g. aut, mac, enc, cer, prd, ctl, kex

<algorithm> : <cipher>|<mode>|<parameter>

<cipher> : e.g. DES,MAA, RSA

<mode> : e.g. ECB, CBC, CFB

<parameters> : e.g. key length, rounds

<technique> : e.g. asymmetric, combined asymmetric-symmetric, static data key, session key, unique data key, master key, KEK, derivation key

<number> : key set number

NOTE | means a separation of fields.

8.8 Controls and audits

Audits and controls as described shall be implemented by the parties responsible for ICC SAM life cycle, and transaction processing, to prevent the disclosure, substitution, modification, deletion, and insertion of cryptographic keys.

Control and audit shall be implemented by the issuer for ICCs, SAMs, key generation and loading modules, host security modules and other cryptographic devices used in financial transaction systems using integrated circuit cards.

SAMs should be controlled and audited in a more stringent manner than ICCs as their compromise may have a greater impact than the compromise of an ICC.

Necessary procedures shall be in place to discontinue the use of ICCs, SAMs and key generation and loading modules whose cryptographic keys are considered as having been disclosed.

Inventories of stolen or lost ICCs and SAMs should be maintained under the responsibility of the respective issuers.

Controls and audit should also be imposed on individuals who manage cryptographic keys and/or devices.

9 ICC and SAM key loading processes

ICC and SAM keys shall be loaded according to the requirements of this part of ISO 10202. An example of key loading in ICCs using symmetric key management is shown in Annex A.

9.1 Loading of initial symmetric keys

Initial symmetric cryptographic keys used in ICCs and SAMs (production keys, 9.2; or issuer control key, 9.3) may be loaded as plaintext parts under at least dual custody or split knowledge, or using a key loading module or a key generation module. The use of these modules must be controlled under at least dual custody.

The procedures for initial key loading shall be performed in a physically secure environment. Every effort shall be made to ensure that initial keys are loaded in the ICC or SAM without the possibility of capture.

9.2 Loading of production keys

Production keys for symmetric algorithms (*kMprd*, *kEprd*) shall be loaded according to 9.1.

kMprd and *kEprd* unique to each ICC may be derived as shown in 10.1 when using a symmetric algorithm. The IC serial number shall then used to derive the ICC's *kMprd* and *kEprd* (Annex A).

For asymmetric algorithms, a unique public/secret key pair (*PMprd*/*SMprd* or *PEprd*/*SEprd*) may be loaded in the ICC, or generated by the ICC.

9.3 Loading of issuer keys

Unless the control key can be loaded in a physically protected manner as agreed by the card issuer and card personalizer, the issuer *kIctl* key shall be loaded under the control of the issuer in an ICC enciphered using the *kMprd* or *kEprd* key of the party which has previously accessed the ICC (see ISO 10202-3). *kMprd* or *kEprd* shall be replaced irreversibly by *kIctl* after successful authentication of the ICC and loading of *kIctl*. All other CDF keys shall be loaded enciphered using the CDF *kIctl* key.

The *kIctl* shall be unique to each ICC. For symmetric algorithms, key management technique 1 or 2 (see clause 10) may be used to derive the unique ICCs *kIctl*.

For asymmetric algorithms, unique public/secret key pair (*PIctl* and *SIctl*) must be loaded in the ICC, or generated by the ICC.

9.4 Loading of ADF keys

As described in ISO 10202-3, *kIctl* shall be used to load *kIkex*. *kActl* key shall then be loaded under the control of the application supplier enciphered under *kIkex*, and *kIkex* shall be replaced by *kActl* after successful authentication of the ICC and loading of *kActl*. All other ADF keys shall be loaded enciphered using the ADF *kActl* key.

For a SAM, *kActl* shall be loaded enciphered under the previous *kMprd* or *kEprd* key as described in ISO 10202-3. All other keys shall be loaded enciphered using *kActl*.

The *kIkex*, *kActl*, and all other keys loaded using *kActl* shall be unique to each ICC. For symmetric algorithms, key management technique 1 or 2 may be used to derive these keys.

For asymmetric algorithms, unique public/secret key pairs must be used for the above keys. If generated by a key generation module, either or both keys shall be loaded into an ICC as described above. Similarly, public/secret key pairs loaded into a SAM shall be loaded as described above.

9.5 Loading of public keys

Public keys used by asymmetric algorithms for key certification, such as the public key of a key certification centre, which are loaded in an ICC or SAM shall be loaded using a process which guarantees the integrity and authenticity of the public keys.

The integrity and authenticity of other public keys loaded into the SAM shall be cryptographically assured by verifying the certificate of the key.

9.6 Loading of secret keys of asymmetric algorithms

Secret keys of asymmetric cryptographic algorithms which are loaded into an ICC or SAM shall be loaded in the ICC or SAM according to the requirements of this part of ISO 10202. The public/secret key pairs shall be generated using a secure key generation module.

9.7 Generation of asymmetric public/secret key pairs

Public/secret key pairs may be generated by ICCs, SAMs and secure key generation modules as described in ISO 9796.

SAM and ICC generating public/secret key pairs shall provide in a secure manner the public key for certification by the higher certification authority to ensure the authenticity of the key.

Public/secret key pairs generated by a key generation module and transferred to an ICC or SAM shall be loaded according to the requirements of this part of ISO 10202.

9.8 Test keys

Test keys generation and management are outside the scope of this part of ISO 10202. Test keys shall only be used for testing purposes.

10 Symmetric key management techniques

Symmetric key management techniques used in ICC systems shall ensure that the requirement for unique ICC keys described in ISO 10202-3 is met. SAM or host derivation keys shall be unique for each key name used.

Separate secret SAM or host derivation keys shall be used to derive each different cryptographic key names (subclause 6.1) loaded into the ICC. These keys may be data keys (KD), key enciphering keys (KEK) or ICC derivation keys (KCD) used to implement one of the following key management techniques for transaction processing:

Technique 1: Static data keys.

Technique 2: Session keys.

Technique 3: Unique message keys.

Each of the above techniques may be implemented using explicit or implicit key identification.

The cryptoperiod of keys maintained in the ICC (CDF and ADF), or SAM shall be considered when selecting one of the above key management techniques.

Examples of each key management technique can be found in Annex B.

10.1 Derivation of ICC and SAM keys

ICC data keys (KD), ICC key enciphering keys (KEK), and ICC derivation keys (KCD) shall be derived from different SAM or host derivation keys (KHD) for each key names maintained in the SAM or the host security module. This approach allows the calculation of a large number of ICC keys from a single SAM or host security module key.

The derivation procedure for ICC and SAM keys shall be performed as in Figure 2.

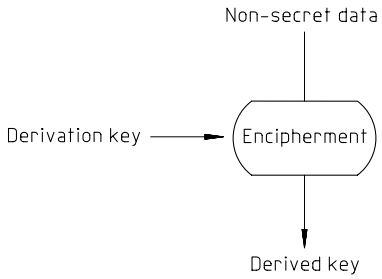


Figure 2 — Key Derivation

A single encipherment (e) process, or an encipherment-decipherment-encipherment (ede) process may be used to derive keys.

The non-secret data used in derivation of ICC (CDF and ADF) keys shall ensure that unique ICC keys are generated. For financial transaction systems using integrated circuit cards, the non-secret data shall be:

1) Unique cardholder related data (card data) such as PAN and expiry date, and card serial number.

and if unique message data keys are to be derived,

2) A cryptographic key sequence number maintained by the ICC.

SAM and host derivation keys (KHD) belonging to card issuers or application suppliers may optionally be derived from a master derivation key (KMD). The non-secret data shall be:

1) Unique bank identification data (bank data), or

2) Other issuer related data to derive optional issuer derivation keys.

A derivation key shall only be used to derive either:

— other derivation keys, or

— key enciphering keys, or

— data keys,

that will be used for the same purpose (name) as denoted by the derivation key. A key used for different purposes or functions shall therefore not be derived from a commom key. This does not prevent the possibility of a system key from which derivation keys for different purposes may be generated.

10.2 Key Management technique 1: Static data keys

Static ICC (CDF and ADF) keys may be used as data keys provided that the cryptoperiod of these keys exceeds the life of the ICC. The static keys are derived from a SAM or host security module derivation key belonging to the issuer or the application supplier.

The derivation of static ICC data keys (KD) shall be performed as follows:

KDx = eKHD(card data), or KDx = edeKHD(card data)

e.g. *KI'mac* = e*KImac*(card data)

KI'mac is the derived data key (KD)

KImac is the derivation key (KHD)

If protection against message replay is required, a message sequence number or data unique to each message shall be used when authenticating transactions.

10.3 Key management technique 2: Session keys

Session data keys may be randomly generated for transaction processing and transmitted to and from the ICC enciphered under a key enciphering key which has been initially loaded in the ICC. New session keys are exchanged every time the ICC is reset.

The derivation of ICC key enciphering keys (KEK) shall be performed as follows:

KEKx = eKHD(card data), or KEK = edeKHD(card data)

e.g. *KI'mac* = e*KImac*(card data)

If protection against message replay is required, a message sequence number or data unique to each message shall be used when authenticating transactions.

Randomly generated session keys shall be transmitted enciphered as specified in ISO 10202-5 under the proper key name. Either key exchange (KE) symmetric-symmetric or symmetric-symmetric-mutual-timeliness as described in ISO 10202-5 may be used.

NOTE Key exchange with ICC/SAM/Host authentication may be performed with this technique by having both communicating nodes participate in the generation of the data key (not described in this part of ISO 10202).

10.4 Key management technique 3: Unique message keys

Unique message data keys may be derived from an ICC derivation key. The data keys are derived from a cryptographic key sequence number (KSN) maintained by the ICC. A unique message key is valid only for a single ICC command/response.

The derivation of ICC derivation keys (KCD) shall be performed as follows:

KCD = eKHD(card data), or KCD = edeKHD(card data)

e.g. *KI'mac* = e*KImac*(card data)

The derivation of unique message keys (ICC command-response pair) shall be performed as follows:

KDx = eKCD(KSN), then KSN = KSN + 1

e.g. *KI''mac* = e*KI'mac*(KSN)

The KSN shall be maintained by the ICC and increased each time after a data key is derived.

The KSN is transmitted to all SAMs or host security modules involved in a session in order to derive the unique message keys.

The use of unique message data keys provides protection against message replay and protection against exhaustive key search, thus no additional unique data need be included in the message.

An example of key management technique 3 with implicit key identification is shown in Annex C.

10.5 Length of keys

Keys used in SAMs and host security modules as derivation keys should have a longer key length than ICC static keys, ICC key enciphering keys, and ICC derivation keys. Such keys may be, for example, a key of double length or two single length keys operating in an encipher-decipher-encipher (ede) mode.

Longer length data keys should be considered when a higher level of protection is required when using the static data keys technique (technique 1).

11 Asymmetric key management techniques

Asymmetric key management techniques used in ICC systems shall ensure that the requirement for unique ICC keys described in ISO 10202-3 is met. Each ICC and SAM shall have either or both a unique public/secret key pair for each key name used in CDF and ADF (see 6.1).

11.1 Use of asymmetric key management in a CAD with a SAM

When asymmetric key management is used in ICC systems, the public key corresponding to a secret key of an ICC is normally maintained by the ICC and supplied to the SAM when needed. The SAM then verifies the certificate of the ICC public key before using it.

Similarly the public key corresponding to a SAM secret key is normally maintained by the SAM and provided to the ICC when required. The ICC then verifies the certificate of the public key before using it.

The above approach to the management of public keys is preferred since it eliminates the requirement to maintain a large number of public keys in ICCs, SAMs, and host security modules.

An example of using asymmetric key management in a CAD with a SAM is shown in Annex D.

11.2 Use of asymmetric key management in a CAD without a SAM

Asymmetric key management may be used in a CAD without a SAM provided that the CAD provides physical or logical protection of the authenticity of public keys as defined in this part of ISO 10202 and to prevent the unauthorized addition of keys.

ICC authentication and transaction certificate verification may be implemented in a CAD without a SAM. Public keys are used by the CAD for authentication and certificate verification. The integrity of these public keys may be verified using a public key belonging to a higher authority such as the card issuer or the application supplier key certification centre.

An example of using asymmetric key management in a CAD without a SAM is shown in Annex E.

11.3 Public key certification requirements

A certification authority, such as the key certification centre of the issuer, shall be used to certify public keys stored in ICCs and SAMs. The certification authority shall be accessed to verify exchanged key certificates when the public key necessary to verify the certificates is not present in the ICC or the SAM.

A public key certificate shall contain, at a minimum, the identity of the trusted certification authority, the public key certified and its key identifier.

A public key certificate identifier (CID) may be used to identify a public key certificate. The certificate identifier contains the name of the trusted certification authority and the key identifier of the public key certified.

Public key certification shall be performed as described in ISO 10202-5.

11.4 Secure storage of secret keys

Secret keys, and associated secret data, of asymmetric key management schemes stored in ICC, SAM, or host security modules shall be protected against disclosure according to the principles of secret key protection defined in this part of ISO 10202.

11.5 Secure storage of public keys

The integrity and authenticity of public keys, stored in ICC, SAM, or host security modules shall be protected physically or logically within these devices. Public keys stored outside ICC, SAM, or host security modules shall be kept in certified form.

11.6 Exchange of certified public keys

A public key, corresponding to an ICC secret key, stored in an ICC shall be presented in certified form to the SAM or host security module for transaction processing. The SAM or host security module shall verify the certificate before transaction processing.

A public key, corresponding to a SAM or host security module secret key, stored in a SAM shall be presented in certified form to the ICC for transaction processing. The ICC shall verify the certificate before transaction processing.

11.7 Key length

Public/secret key pairs length shall be selected such that the cryptoperiod of these keys exceeds the life of the ICC or SAM.

11.8 Secure protocols

Secure protocols shall be used for key exchange, authentication and certificate verification processes performed in asymmetric key management environments. The processes described in ISO 10202-5 shall be used.

12 Combined asymmetric/symmetric key management

A combination of both symmetric and asymmetric key management schemes may be used in a financial transaction system using integrated circuit cards. For example, secret keys of a symmetric algorithm may be generated or distributed using an asymmetric key management scheme.

An asymmetric algorithm may also be used to complement the functionality of a symmetric algorithm, such as the implementation of a signature function, or the implementation of an authentication function in a CAD without a SAM.

12.1 Basic requirement

The use of a combined symmetric/asymmetric key management scheme shall meet the security requirements of both symmetric and asymmetric key management.

12.2 Exchange of symmetric keys

Symmetric keys exchanged using an asymmetric algorithm shall be exchanged using one of the techniques defined in ISO 10202-5.

Annex A
(informative)

Example of card life cycle using symmetric key management