

INTERNATIONAL STANDARD

ISO
11166-1

First edition
1994-11-15

Banking — Key management by means of asymmetric algorithms —

Part 1:

Principles, procedures and formats

*Banque — Gestion des clés au moyen d'algorithmes asymétriques —
Partie 1. Principes, procédures et formats*



Reference number
ISO 11166-1 1994(E)

AVA Documents
Document Sharing Hub

Contents

	Page
Section 1 General	1
1.1 Scope.....	1
1.2 Normative references	1
1.3 Definitions.....	2
1.4 Abbreviations and notation.....	4
1.5 Key management facility.....	7
1.6 Key generation, storage, back-up and destruction.....	7
1.7 Keys	9
 Section 2 Key certification	 11
2.1 Minimum requirements for secure operation of a Key Certification Centre.....	11
 Section 3 Automatic distribution of keying material	 13
3.1 Requirements for the automated key management architecture.....	13
3.2 Automated key management architecture	13
3.3 Primitive processes.....	14
 Section 4 Cryptographic service messages	 22
4.1 Message formats and flows.....	22
4.2 Generation of Cryptographic Service Messages.....	33
4.3 Processing Cryptographic Service Messages	41

© ISO 1994

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the publisher.

International Organization for Standardization
Case Postale 56 • CH-1211 Genève 20 • Switzerland

Printed in Switzerland

Annexes

A	Overview of principles of the standard.....	5 3
B	Examples of the message flow and message format	5 4
C	Methods for manual handling of public keys which are not in the form of a certificate.....	5 9
D	An example of dual CKC operation.....	6 1
E	Recommended CKC security audit log contents and use ...	6 2
F	Bibliography.....	6 4

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 approval before their acceptance as International Standards by the ISO Council. They are approved in accordance with ISO procedures requiring at least 75% approval by the member bodies voting.

International Standard ISO 11166-1 was prepared by Technical Committee ISO/TC 68, *Banking and relating financial services*, Subcommittee 2, *Operations and procedures*.

This part of ISO 11166 is related to ISO 8732, *Banking — Key management (wholesale)* from which it was developed.

ISO 11166 consists of the following parts, under the general title *Banking — Key management by means of asymmetric algorithms*:

— *Part 1: Principles, procedures and formats*

— *Part 2: Approved algorithms using the RSA cryptosystem.*

Annexes A to F of this part of ISO 11166 are for information only.

Introduction

This part of ISO 11166 describes procedures for the secure management of keying material used to protect messages in a wholesale banking environment.

Key management is the process whereby cryptographic keys and initialisation vectors (keying material) are provided for use by two parties and continue to be subject to secure handling procedures until they have been destroyed. The security of the data enciphered or authenticated by means of keying material is dependent upon the prevention of unauthorised disclosure, modification, substitution, insertion or deletion of keys or initialisation vectors (IVs). If they are compromised, the security of the related data can no longer be ensured. Thus, key management is concerned with the generation, distribution, storage, custody, monitoring, destruction, and back-up procedures for keying material. Also, by the formalisation of such procedures provision is made for audit trails to be established.

Automated key distribution is the means by which two communicating parties are provided with identical symmetric keying material (cryptographic keys and, where needed, IVs) by using a defined procedure via a communication channel. Automated key distribution in this standard employs two types of keys :

- 1) Keys for asymmetric algorithms. These are used to encipher and decipher data keys and also to produce and verify digital signatures for the authentication of data keys or messages containing these keys.
- 2) Keys for symmetric algorithms. These are the data keys which are the subject of key management. They are also used to provide integrity check values (MACs) on some messages exchanged for key management.

The level of security to be achieved needs to be related to a number of factors, including the sensitivity of the data concerned and the likelihood that it will be intercepted, the practicality of any envisaged encipherment process, and the cost of providing, and breaking, a particular means of providing security. It is therefore necessary for each communicating pair to agree about the extent and detail of security and key management procedures. Absolute security is not practically achievable, so key management procedures need to aim to reduce the opportunity for a breach of security and also to aim for a "high" probability of detection of any illicit access or change to keying material that may occur despite any preventative measures. This applies at all stages of the generation, and use of keying material, including those processes that occur in cryptographic equipment and those related to communication of cryptographic keys and initialisation

vectors between communicating pairs or key centres. Thus, while wherever possible this International Standard has specified requirements in absolute terms, in some instances a level of judgement cannot be practically avoided. For instance, defining the frequency of key change is beyond the scope of this standard, and will be dependent upon the degree of risk associated with the factors listed above.

Part 1 of this International Standard has been divided into sections, as follows :

1. General
2. Key certification
3. Automatic distribution of keying material
4. Cryptographic Service Messages.

The final details of the key management procedures need to be agreed between the communicating pair(s) concerned and will thus remain the responsibility of the communicating pair(s). An aspect of the details to be agreed upon will be the identity and duties of particular individuals. This International Standard does not concern itself with allocation of individual responsibilities as this needs to be considered uniquely for each key management implementation.

Annex A gives an overview of the principles employed.

Banking — Key management by means of asymmetric algorithms —

Part 1:

Principles, procedures and formats

Section 1: General

1.1 Scope

This part of ISO 11166 specifies methods for the management of keying material used for the encipherment, decipherment and authentication of messages exchanged in the course of wholesale financial transactions only. It specifies requirements for:

- i) the control of keying material during its life to prevent unauthorised disclosure, modification, substitution, and replay;
- ii) the manual or automatic distribution of keying material to permit interoperability between cryptographic equipment or facilities using the same algorithms;
- iii) ensuring the integrity of keying material during all phases of its life, including its generation, distribution, storage, entry, use, archival and destruction;
- iv) recovery in the event of failure of the key distribution process.

This part of ISO 11166 provides a means whereby an audit log can be identified for all keying material.

This International Standard is designed for the management of keying material for use with symmetric algorithms. The key distribution is performed using an asymmetric algorithm or algorithms. It is designed for messages formatted and transmitted in coded character sets.

This International Standard defines two alternative methods for two communicating parties to obtain identical symmetric keying material which they can then use to protect communications.

One method is *key transport*, in which one party generates keying material and sends it to the other party, suitably protected by using asymmetric algorithms.

The second method is *key exchange*, in which each party of the communicating pair sends the other a block of data and using the received block each party derives keying material in such a way that the sets of keying material thus derived by the two parties are identical.

NOTE — A well known example of key exchange is the Diffie-Hellman key exchange method.

In this International Standard, the term *key distribution* will encompass both key transport and key exchange. This part of ISO 11166 includes key transport but does not preclude key exchange. Specific key exchange procedures may be the subject of subsequent parts of this International Standard.

The procedures specified are appropriate for use by financial institutions and by their corporate and government customers, and in other relationships where the interchange of information requires confidentiality, protection and authentication.

1.2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this part of ISO 11166. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this part of ISO 11166 are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO/IEC 646:1991, *Information technology — ISO 7-bit coded character set for information interchange*.

ISO 8730:1990, *Banking — Requirements for message authentication (wholesale)*.

ISO 8731-1:1987, *Banking — Approved algorithms for message authentication — Part 1: DEA*.

ISO 8731-2:1992, *Banking — Approved algorithms for message authentication: — part 2: Message authenticator algorithm*.

1.3 Definitions

For the purposes of this part of ISO 11166, the following definitions apply.

1.3.1 asymmetric algorithm: A cryptographic algorithm employing a *public key* and a *secret key*. Together these form an asymmetric key set.

1.3.2 audit log: See *security audit log*.

1.3.3 authentication: A process used, between a sender and a receiver, to ensure *data integrity* and to provide *data origin authentication*.

1.3.4 certificate: The public key of a user, together with some other information, rendered unforgeable by a signature with the secret key of the certification authority which issued it.

1.3.5 certification authority: An authority trusted by all users to create and sign certificates.

1.3.6 ciphertext: Enciphered information.

1.3.7 code: A symbol representing data, typically to facilitate automated processing.

1.3.8 communicating pair: Two *logical parties* who have previously agreed to exchange data.

NOTE — A party and a Key Certification Centre do not constitute a communicating pair.

1.3.9 Co-ordinated Universal Time: The time scale maintained by the Bureau International de l'Heure (International Time Bureau) that forms the basis of a co-ordinated dissemination of standard frequencies and time signals.

NOTE — May alternatively be described as Greenwich Mean Time (GMT).

1.3.10 counter: An incrementing count used between two parties to control their key distributions.

1.3.11 cryptographic equipment: Equipment in which cryptographic functions (e.g. *encipherment*, *authentication*, *key generation*) are performed.

1.3.12 cryptographic key: A parameter used in conjunction with an algorithm for the purpose of *validation*, *authentication*, *encipherment*, *decipherment*, *signature construction* or *deconstruction*.

1.3.13 cryptographic keying material: See *keying material*.

1.3.14 cryptography: The discipline which embodies principles, means, and methods for the transformation of data in order to hide its information content, prevent its undetected modification and/or prevent its unauthorised use.

NOTE — Cryptography determines the methods used in encipherment and decipherment. An attack on a cryptographic principle, means or method is cryptanalysis.

1.3.15 cryptoperiod: A defined period of time during which a specific *cryptographic key* is authorised for use, or during which time the *cryptographic keys* for a given system may remain in effect.

1.3.16 data integrity: The property that data has not been altered or destroyed in an unauthorised manner.

1.3.17 data key: A *cryptographic key* used for the *encipherment*, *decipherment* or *authentication* of data.

1.3.18 data origin authentication: The corroboration that the source of data received is as claimed.

1.3.19 decipherment: The reversal of a corresponding reversible *encipherment*.

1.3.20 decryption: See *decipherment*.

1.3.21 digital signature: A value used to validate the source and content of part or all of the text of a message. For the purposes of this

part of ISO 11166, it is generated and verified using an asymmetric algorithm. In some applications, it can also provide a non-repudiation service.

1.3.22 dual control: A process of utilising 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 materials, e.g. *cryptographic key*.

1.3.23 encipherment: The cryptographic transformation of data (see *cryptography*) to produce *ciphertext*.

1.3.24 encipherment public key: The public key which may be used to encipher data keys.

1.3.25 encryption: See *encipherment*.

1.3.26 exclusive-or: See *modulo-2 addition*.

1.3.27 field tag: A unique string of characters used in formatted messages that identifies the meaning and location of the associated data field.

1.3.28 hexadecimal digit: A single character in the range 0-9, A-F (upper case), representing a four bit string.

1.3.29 initialisation vector (IV): A number used as a starting point for *encipherment* of a data sequence. It increases security, by introducing additional cryptographic variance, and also facilitates the synchronisation of *cryptographic equipment*.

1.3.30 interoperability: The ability to exchange *cryptographic keys*, whether manually or in an automated environment, with any other party.

1.3.31 key: See *cryptographic key*.

1.3.32 key component: One of at least two parameters having the format of a *cryptographic key* that is combined with one or more like parameters by means of *modulo-2 addition* to form a *cryptographic key*.

1.3.33 Key Certification Centre: A facility operated by the certification authority which generates and returns certificates.

1.3.34 key generator: A type of *cryptographic equipment* used for generating *cryptographic keys* and, where needed, *initialisation vectors*.

1.3.35 key loader: An electronic, self-contained unit which is capable of storing at least one *cryptographic key* and transferring that *cryptographic key*, upon request, into *cryptographic equipment*.

1.3.36 key management facility: A protected enclosure (e.g. room or *cryptographic equipment*) and its contents where cryptographic elements reside.

1.3.37 keying material; cryptographic keying material: The data (e.g. keys and IVs) necessary to establish and maintain a *keying relationship*.

1.3.38 keying relationship: The state existing between a *communicating pair* during which time they share at least one *data key*.

1.3.39 logical party: One or more physical parties forming one member of a *communicating pair*.

1.3.40 Message Authentication Code (MAC): A field in a message between a sender and a receiver used to validate the source and content of part or all of the text of a message. A MAC is generated and verified using a symmetric algorithm.

1.3.41 modulo-2 addition; exclusive-or: A binary addition with no carry, giving the following values:

$$0 + 0 = 0$$

$$0 + 1 = 1$$

$$1 + 0 = 1$$

$$1 + 1 = 0$$

1.3.42 originator: The party (logical or other) that is responsible for originating a Cryptographic Service Message.

1.3.43 plaintext: Unenciphered information.

1.3.44 public key: That part of an asymmetric key set which is known to other parties than the originator of the key set.

1.3.45 recipient: The party (logical or other) that is responsible for receiving a Cryptographic Service Message.

1.3.46 secret key: (in an asymmetric or public key cryptosystem) That part of an asymmetric key set which is kept secret by the originator of the key set.

1.3.47 security audit: An independent review and examination of system records and activities in order to test for adequacy of system controls, to ensure compliance with established policy and operational procedures and to recommend any indicated changes in control, policy and procedures.

1.3.48 security audit log: Log containing data collected and potentially used to facilitate a *security audit*.

1.3.49 signature construction: The cryptographic transformation of data (See *cryptography*) to ensure data origin authentication.

1.3.50 signature deconstruction: The reversal of *signature construction* to produce data from which it is possible to determine whether the signature is valid or not.

1.3.51 signature public key: A public key which may be used to deconstruct signatures.

1.3.52 symmetric algorithm: a cryptographic algorithm employing the same value of key for both enciphering and deciphering or for both authentication and validation.

1.3.53 validation: The process of checking the *data integrity* of a message, or selected parts of a message.

1.3.54 zeroisation: A method of erasing or overwriting electronically stored data.

1.4 Abbreviations and notation

1.4.1 Abbreviations

The following abbreviations are used in this part of ISO 11166.

Abbreviation	Meaning	Description (see also table 2)
BC	Block	The elementary unit of data that an asymmetric algorithm can encipher.
BE	Enciphered Block	A field that holds keys enciphered by an asymmetric master key.
BES	Enciphered and Signed Block	A field that holds keys enciphered by an asymmetric master key and signed by another asymmetric master key.
CKC	Key Certification Centre	A facility operated by the certification authority which generates and returns certificates.
CSM	Cryptographic Service Message	A message for the establishment of keys or for carrying related information used to control a keying relationship.
CTP	Counter P	Counter used in a keying relationship.
CTR	Counter R	The received value of the counter found to be in error.
CV	Certificate Value	The value of a certificate (certified public key) exchanged between two users.

CVO	Certificate Value Only	Indicates that only a Certificate Value is requested by a Cryptographic Service Message.
DEA	Data Encryption Algorithm	The symmetric block encipher algorithm specified in ANSI X3.92.
DSM	Disconnect Service Message	A message type used to discontinue one or more keys or to terminate a keying relationship.
ECB	Electronic Code Book	A mode of implementing an encipherment algorithm.
EDC	Error Detection Code	A field in a Cryptographic Service Message used to validate the data integrity of the message.
EDCV	Effective Data of a Certificate Value	Date and Coordinated Universal Time before which a certificate is not valid.
EDK	Effective Date of Key(s)	Date and Co-ordinated Universal Time on which the data key(s) are activated.
ERF	Error Field	The identification of error conditions detected in a prior Cryptographic Service Message.
ESM	Error Service Message	A message type used to give a negative acknowledgement on receipt of any Cryptographic Service Message other than an ESM, and to give the recipient data with which to recover.
EXCV	Expiration date of a Certificate Value	Date and Co-ordinated Universal Time after which a certificate is not valid.
FFC	Function and Format Code	A code word in a certificate obtained by transforming text. It defines both the function of the certified public key and the content and format of the rest of the certificate text.
FKD	Function of Data Key	Indicates the functions for which a data key may be used.
FKP	Function of Public Key	Indicates whether the KAM to which a public key belongs can be used for encipherment, signature verification or both.
IABE	Identifier of block encipherment algorithm	-
IABS	Identifier of block signature algorithm	-
IACS	Identifier of certificate signature algorithm	-
IAPK	Identifier of algorithm used with the certified key	-
IDA	Identifier of Authentication Key	Identifies the data key used to authenticate a Cryptographic Service Message.
IDCC	Identifier of Key Certification Centre	-
IDCK	Identifier of certification centre key	Identifier (name) of the key used by the Key Certification Centre to produce a certificate.
IDCV	Identifier of the Public Key certified	Identifier (name) of a Public Key certificate.
IDD	Identifier of Key to be Discontinued	-

IDEK	Identifier of Enciphering Key Certificate	Identifier (name) of the certificate containing the public key used to encipher the Enciphered Block being transmitted in a Cryptographic Service Message.
IDK	Key Identifier	Identifier of the key being transmitted in a Cryptographic Service Message.
IDSK	Identifier of Signature Key Certificate	Identifier (name) of the certificate containing the public key to be used to deconstruct the signature of the Enciphered Block being transmitted in a Cryptographic Service Message or the signature of the whole message.
IV	Initialisation Vector	See 1.3.29
KAM	Asymmetric Master Key Set	Comprises a public and a secret key employed to protect the distribution of data keys by encipherment or digital signature.
KD	Data Key	A key used to operate directly on data.
KDX	Fixed Data Key	A key with fixed value used in the computation of an Error Detection Code.
KP	Public Key	The value of an asymmetric Public Key in a certificate.
KSM	Key Service Message	A message type used to transfer keys between communicating pairs. Optionally may transfer a certificate.
MAC	Message Authentication Code	See 1.3.40
MCL	Message Type	The tag for the field that defines the type of Cryptographic Service Message.
ND	Name of Data Key	Field containing the identification of a data key contained in an Enciphered Block.
ORG	Originator	Originator of CSM.
OWN	Owner	The name of the party to whom a certificate was originally issued.
P	Key Parity	Indicates that the plaintext key conforms to the specification for odd parity.
RCV	Recipient	Recipient of CSM.
RSI	Request Service Initiation Message	Used to request keys and/or a certificate from another party. Optionally may transfer a certificate.
RSM	Response Service Message	Used to provide an authenticated acknowledgement. Optionally may transfer a certificate.
SIG	Digital Signature field	Optional field of a Key Service Message containing a signature of the whole message.
SVR	Service Request	Specifies type of service requested.

1.4.2 Notation

1.4.2.1 *Operators* are represented by the following lower case letters :

- a for authentication
- e for encipherment
- d for decipherment
- s for signature construction by transforming the text
- v for signature deconstruction
- σ for signature construction by means of a separate signature

"[]" denotes representation of field contents

"{ }" denotes the subject of an operator. If the operator involves a key, the type of key is inserted between the operator and its subject.

For an asymmetric algorithm, e and v are operations performed with public keys and d and s are the operations performed with secret keys.

1.4.2.2 *Concatenation* is indicated by "||"

1.4.2.3 *Modulo-2* addition is indicated by "+"

1.4.2.4 *Fields* are separated by "b", space

1.4.2.5 *Sub-fields* within a field are separated by ".", full stop

1.4.2.6 *The types of key* are :

- KAM asymmetric master key set
- KD data key
- KDX data key, plaintext, for computation of the Error Detection Code, hexadecimal e.g. 0123456789ABCDEF (see 3.3.1.1.)

1.5 Key management facility

1.5.1 General

A key management facility shall provide a means of access control whereby its contents are protected from unauthorised disclosure, modification, substitution, replay, insertion or deletion.

NOTE — To achieve such control, action needs to be taken to either preclude access or to ensure that attempts to gain access have a high probability of being detected and reported.

1.5.2 Contents of key management facility

All cryptographic equipment, including key generation equipment, shall be located within a key management facility.

NOTE — Cryptographic equipment may itself act as the key management facility, and so provide all the required functions.

1.6 Key generation, storage, back-up and destruction

1.6.1 Generation of keys and initialisation vectors

1.6.1.1 General

The output from a key generator shall be automatically checked for generation failure (e.g. the repeated output of the same key). Operation of the key generator shall stop immediately if any failure is detected.

The design of this generation process shall be such that no cryptographic advantage is gained by attacking the key generation process rather than the encipherment process.

Secret keys shall not be available in plaintext form from cryptographic equipment, even upon failure of the equipment, other than for backup purposes under dual control.

A means shall be provided for the manual zeroisation of plaintext keys.

1.6.1.2 Generation of symmetric keys and initialisation vectors

A symmetric key set shall be generated only in a key management facility and under management control.

The generation of symmetric keys and initialisation vectors shall be by means of a process that ensures that all keys and initialisation vectors are random or pseudo-random.

1.6.1.3 Generation of asymmetric key sets

An asymmetric key set shall be generated only in a key management facility and under management control.

The generation process shall conform to a standard specified as part of an asymmetric algorithm approved for use with this standard.

1.6.2 Key storage

Secret key values in plaintext form shall be stored only as components for backup purposes (see 1.6.3.1) or within a key management facility.

Security features shall be built into the software and hardware for the purpose of preventing unauthorised access. Any attempt which threatens to gain unauthorised access to plaintext secret key values in the protected memory shall result in the stored plaintext secret key being erased or otherwise rendered unintelligible.

Where plaintext public keys are stored and are not in the form of a certificate, or where their certificate has been checked and they will be used without re-checking the certificate, access which can change the public key value shall be prevented by security features built into the software and hardware.

A short term power failure shall not result in the loss of a key.

Where keys are associated with counters (see 3.3.2), the counters shall be protected against erasure, loss or lowering of a counter by security features built into the software and hardware.

Custody of keys, whether in the form of key components, secure storage devices or backup key management facilities with keys loaded, shall be recorded on the security audit log. Lists of staff designated to hold or access keys shall be kept.

1.6.3 Backup of secret key values and recovery

1.6.3.1 Backup of secret keys

Obtaining of plaintext values of secret keys for backup purposes shall be under dual control and recorded in the security audit log.

Backup values may be held in a secure storage device such as a key loader or in a backup key management facility.

All backup copies of keys shall be subject to the same or a greater level of security control as keys in current use.

If readable backup copies are made, these shall be in the form of at least two key components, requiring all the components to be known in

order to recover the key values. Each component shall include a sum check of sufficient size that the probability of undetected error is less than one in 1000. Key components shall be in the custody of a different individual for each component. The identity of the person receiving the key component shall be recorded in the security log.

If backup secret key values are held in a secure storage device, access to the stored values shall be controlled by positive user authentication, e.g. access identifier and password or other methods, to prevent unauthorised access.

If secret key values are transferred to a backup key management facility, this shall be under dual control and be recorded in the security audit log.

1.6.3.2 Recovery

Recovery of key values from backup or the placement into operation of a backup key management facility shall be under dual control and be recorded in the security audit log.

Where backup keys are held as key components in individual custody, a means shall be provided for manual entry of key components. Components shall be entered by each individual with no others present. A means of correcting errors or re-entering the entire key component shall be provided. The sum check shall be performed at entry and an indication of error given to the operator. A set of key components with sum check error shall not be accepted by the system. If a plaintext key component is displayed during entry, it shall be visible only to authorised persons, and shall be cleared immediately after the key entry process is completed.

Each component entry attempt and its outcome shall be recorded in the security audit log, with the identity of the operator.

1.6.4 Key destruction

Plaintext secret key values or the components thereof shall be zeroised at the end of their life.

Any intermediate storage of plaintext keys shall be zeroised once the transfer of the key to another location is complete.

Key destruction that is triggered by an attempt at unauthorised access shall be by zeroisation. Key destruction shall be effective in the presence of power failure.

All copies of keys that are no longer required shall be destroyed under dual control.

A detailed record of withdrawal from service and destruction shall be retained for audit trail purposes.

1.6.5 Security audit log

A security audit log shall be required both for operation and to support the audit function. The security audit log shall provide sufficient detail to reconstruct events, provide "due care," meet legal requirements and allocate liability. The contents of a security audit log shall be date and time stamped.

The security audit log shall be maintained in a form which prevents unauthorised modification or destruction of the record.

The security audit log shall record all key management operations, such as key generation, backup, recovery and destruction, together with the identity of the person authorising the operation and persons handling any key material (such as key components or keys stored in portable devices or media). All actions resulting from the suspected compromise of any keying material shall be recorded in the security audit log.

Custody of keys and devices or media holding keys shall be recorded in the security audit log.

The security audit log shall not contain any record of plaintext key values. It may hold check values derived from keys by means of a one-way function, as a means of identifying keys and verifying their correctness.

The use to be made of the security audit log is not specified in this part of ISO 11166.

1.7 Keys

1.7.1 Custody of secret keys

Secret keys stored outside the key management facility shall be enciphered or otherwise not be capable of being disclosed.

Lists of the staff authorised to manage or access keys shall be kept. These lists shall not contain any plaintext key values.

1.7.2 Validity of keys

Keys should normally be allocated a unique identifier and an effective date. The communicating pair shall agree upon the cryptoperiod for each key.

Data keys may be distributed on the basis that they are for immediate or for future use (see 1.7.4). The sender of a key shall not employ that key until an authenticated acknowledgement has been received from the receiving party, checked and found to be valid. Where a key has not been specifically identified (e.g. by number or effective date), it shall be the only such key and shall be put into service by the receiver after the acknowledgement has been sent and by the sender after the acknowledgement has been received, checked and found to be valid.

Where it is suspected or known that a key has been compromised, it shall no longer be considered to be valid and shall be withdrawn from current use.

1.7.3 Key changes

Possession of another party's certified public key potentially enables a data key to be automatically established with that party. Management controls shall be applied such that any new keying relationship shall be positively authorised before any data key established by such automatic means becomes valid.

Keys shall be changed :

- a) at the end of the cryptoperiod; or
- b) at the request of either member of the communicating pair; or
- c) immediately after it is known or suspected that a key has been compromised.

All key changes shall be acknowledged. Where the cryptoperiods of an existing and a new key overlap, the effective date of the replacement key (or other implicit time reference) shall be specified whereupon the old key is no longer active. During this changeover period, both keys shall be held under the same level of security.

Keys withdrawn from use shall not be knowingly or intentionally re-used, except for the purpose of reconstructing a key/message pair (see 1.7.5).

1.7.4 Reserve keys

Where keys are stored in reserve, to facilitate planned or unexpected key changes, they shall be subject to the same level of security control as keys in current use.

1.7.5 Archiving of keys

Where the continued storage (archiving) of a key after the expiration of its cryptoperiod, or compromise, is required each such key shall be uniquely identified, or converted into a different form or format so that there is no ambiguity that it is archived and obsolete. All archived keys shall be enciphered under a key designated

for that purpose. It shall not be possible to use archived keying material other than for the reconstruction of a key/message pair.

All restrictions and security measures that apply to operational keys also apply to archived keys.

NOTE — The detailed procedures for the archiving of keys are application dependent and are not defined in this part of ISO 11166.

Section 2: Key certification

2.1 Minimum requirements for secure operation of a Key Certification Centre

2.1.1 General

The authentication of public keys is an essential security requirement. This is ensured by exchanging them in the form of certificates. A certificate contains the public key and its identifying data, with a digital signature provided by the Key Certification Centre (CKC).

Public keys contained in a certificate from the CKC may be exchanged by automated or other means. In no instance shall a public key before its certification be automatically handled by a process defined in section 3 of this part of ISO 11166. A recipient of a certificate shall verify the signature of the certificate before using the public key contained in the certificate. After checking the certificate, the public key may be used, provided it has been stored in a manner which prevents modification or substitution of the stored value or of its identification data and that it has not been cancelled (see 2.1.5).

The signature of the certificate shall be verified with the aid of the public key of the CKC.

The Key Certification Centre is "trusted" by its subscribers. Such trust is based on cryptographic equipment designed to meet requirements for financial institution use, sound management and control practices and is confirmed by an independent audit function (internal, external or both) which shall report audit results to the subscribers.

The security audit log shall be maintained as described in 1.6.5.

The CKC security audit log shall be protected from modification or substitution, for example by the use of a digital signature. In this case, the public/secret key pair used for signing the security audit log shall not be used for any other purpose.

An example of the contents of a typical CKC security audit log contents is contained in annex E.

2.1.2 Generation and storage of the asymmetric key set of the CKC

The asymmetric key set of the CKC shall be generated in a key management facility under

dual control, in addition to the criteria specified in 1.6.1.3. for key generation.

The secret key of the CKC shall be stored in the key management facility in which it is used. Backup of the CKC secret key shall be provided as specified in 1.6.3.1 with appropriate means of recovery as specified in 1.6.3.2.

2.1.3 Distribution of the CKC public key

The CKC public key shall be entered into each key management facility. The authenticity of this key is essential. It is the recipient's responsibility to ensure that the CKC public key is authentic. Methods of accomplishing this are described below.

Initial loading of the CKC public key may be performed in a secure environment under the control of the CKC before the key management facilities are distributed.

Other circumstances may require the distribution of a CKC public key to key management facilities that are already in use. If a trustworthy keying relationship exists between the CKC and each party, an existing shared key may be used to protect the integrity of the transmitted CKC public key and to provide authentication.

If no such keying relationship exists, the distribution of the CKC public key shall be protected by manual methods, such as distribution by a number of different and independent channels, with cross-checking of the received values (examples are : letter, fax, E-mail, recorded messages accessed by phone, newspaper).

Because the CKC public key is a large number, manual distribution and cross-checking is inconvenient. The key may be transmitted to the key management facility automatically and without protection, provided that its authenticity is then verified by means of a cryptographic check value which is authenticated. The check value shall be formed by a public one-way function giving at least a 32 bit result.

The check value for a planned CKC public key distribution may be distributed well in advance to allow adequate verification of its authenticity, without exposing the CKC public key during the period before it comes into use.

NOTE — If an authentic CKC public key is loaded by a number of parties, their exchange of certificates with other parties and the verification of the signatures will reveal the existence of false CKC public key values if such exist.

2.1.4 Submission of a user public key for certification

To preserve the authenticity of public keys, the CKC must ensure that public keys for which it provides certificates are genuine.

The method used to submit a public key and receive a certificate in return shall meet these criteria :

- a) The CKC shall verify the authentic origin of the submitted key
- b) The CKC shall authenticate the submitted public key value.

The trust in the Certification Authority is in part based on the methods employed to identify the authenticity of its members prior to the certification of their public keys. The method used shall be defined and enforced by the centre for its membership.

This part of ISO 11166 does not specify the methods to be used.

NOTE — Authentication of a submitted public key by the CKC is the first step in providing secure automatic key distribution. This first step cannot be entirely automatic but it can be based on authentication data which were exchanged when a business relationship was established (examples of such data are the manual signature of an authorised signatory or a cryptographic key held in a smart card). The nature of this pre-existing data varies among applications of this part of ISO 11166.

Examples are given in annex C.

2.1.5 Action on compromise of a user's secret key

If it becomes evident that a user's secret key has been compromised, the key shall immediately be replaced and the compromised key cancelled. All keying material ever sent and protected using that compromised key (without regard to type) shall be replaced at the earliest opportunity.

The owner of the key shall take immediate action to inform all parties in the system that all certificates of the public key relating to the compromised key have been cancelled. This may take the form of a message authenticated by the CKC and sent to all parties, or the maintenance of an on-line list of revoked certificates (Hot List) by the CKC.

The party with the compromised key shall take immediate action to generate a new asymmetric key set and obtain a certificate for the public key. By sending this certificate to other communicating parties, they can be informed of the change of key.

2.1.6 Lifetime of a certified public key

A certificate shall contain an expiry date in the identifying data. A new certificate shall be generated and distributed before that date.

It shall be possible for the CKC or the party identified in the certificate to withdraw a certificate prior to its expiry date.

After the expiry date, any message requiring the use of the expired certificate shall be rejected.

When a certificate expires, keying material sent and protected by that certificate (without regard to type) should be replaced as soon as it is possible without disrupting service.

2.1.7 Compromise of the CKC secret key

If the CKC secret key is suspected of being compromised, the CKC shall take immediate action to generate a new key set and distribute the new CKC public key.

The most cautious reaction of users in this event is to distrust all keying material, since the start of the compromise is not known. At some risk, users may choose to trust public keys of communicating parties for which the certificates were received well before the known compromise, and data keys subsequently exchanged using these public keys.

Design of the physical and operational security of the CKC (such as software integrity) shall minimise the possibility of the compromise of its secret key and maximise the probability that a compromise will be detected.

NOTE — Optionally, more than one CKC may be provided, each CKC operating independently of the others. A user's public key may then be contained in certificates from more than one CKC. The compromise of the secret key of one CKC does not invalidate certificates produced by other CKCs.

This optional method of operation increases immunity to the compromise of the secret key of one CKC. A user's public key may continue to be used, employing a certificate which is still valid.

Annex D describes a different method of operating using two CKCs.

Section 3: Automatic distribution of keying material

3.1 Requirements for the automated key management architecture

This part of ISO 11166 is designed to meet the following requirements for automated key management. It is assumed that:

- 1) the number of parties in the network is expandable;
- 2) each party of a communicating pair has an asymmetric key set with a certificate of the public key produced by a certification authority;
- 3) each party has an authentic and up-to-date copy of the certification authority's public key.

3.1.1 The architecture shall support the ability to have at least one data key between communicating pairs.

3.1.2 The architecture shall support the ability to change data keys automatically between communicating pairs.

3.1.3 A particular data key shall be used for either encipherment/decipherment or for authentication, but not for both, except when authenticating a Cryptographic Service Message.

3.1.4 A data key shared between a communicating pair shall not be disclosed to a third party.

3.1.5 An asymmetric secret key shall not be disclosed to any other party.

3.1.6 The same data key shall not be knowingly or intentionally used by more than one communicating pair.

3.1.7 The compromise of any key shared between any communicating pair shall not compromise any third party.

3.1.8 The architecture shall support any party initiating a secure connection with any other party, subject to management controls.

3.2 Automated key management architecture

3.2.1 General

The architecture shall consist of two layers of keys (see figure 1).

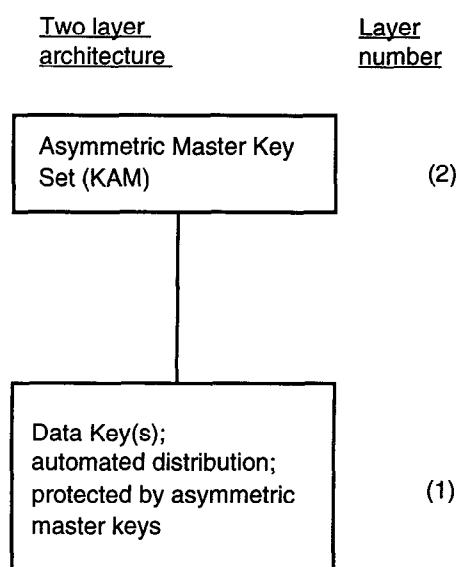


Figure 1 — Key distribution architecture

The upper layer shall comprise one or two asymmetric master key sets (KAM). The lower layer shall comprise a data key or keys (KD).

3.2.1.1 Protection of keys for key transport

The public key of a KAM shall be used to encipher the data key(s) such that it/they can only be deciphered by the intended recipient.

The secret key of a KAM shall be used to authenticate the source of the enciphered data key and, optionally, to provide a digital signature on the message transporting it.

Depending on the properties of the algorithm(s) used, these processes may share a KAM or alternatively separate encipherment and signature KAMs may be employed. The corresponding public keys of these KAMs are called *encipherment public key* and *signature public key* respectively. When a single KAM is used for both purposes, the qualifiers *encipherment* and *signature* may be ignored.

A KAM shall comprise an asymmetric pair of keys. The KAM shall be generated in a key management facility. If the secret key is released from a facility for purposes of backup and recovery, this shall be performed under management control, according to the principles described in 1.6.3.1. The secret key shall be used for constructing the digital signature of originated messages and/or decipherment of received blocks of enciphered data. The public key(s) shall be certified by the key certification centre according to the principles described in clause 2.1. Only when a certificate has been received and verified shall the public key be used for encipherment of originated messages and/or the deconstruction of digital signatures in received messages.

3.2.2 Distribution of keying material

Public keys shall be distributed in the form of a certificate created by the key certification centre. The requirements for the operation of the key certification centre are given in clause 2.1.

For the transport of data keys, they shall be enciphered prior to distribution, using the public key of the KAM of the recipient, provided that the certificate of the public key has been checked and its signature verified.

NOTE — Two data keys may be sent, enciphered, in a single Cryptographic Service Message (see clause 4.1).

The encipherment of data or initialisation vectors (IV) prior to distribution shall be carried out using data keys. Alternatively, the IV may be included in the block carrying enciphered data keys.

When a new key is received, all stored keys with the same identifier (name) shall be replaced at the effective date of the new key. In addition, when a key is discontinued, all keys of the same identifier (name) shall be discontinued.

The recipient of cryptographically protected data from which keying material is to be extracted shall be able to identify the relevant KAM(s).

3.2.3 Environments

Key management by means of asymmetric algorithms employs only one environment. In the terminology of ISO 8732, it is a point-to-point environment in which the communicating pair share (in the form of certificates) the public key parts of their KAMs.

In this part of ISO 11166, the environment is implicit and it will not be referred to elsewhere in the standard.

Each member of a communicating pair has its own KAM(s). In order to send data keys to another party, the sending party requires the receiving party's encipherment public key. The receiving party requires the sending party's signature public key in order to verify the received data keys.

All public keys shall be certified by a common key certification centre. A method of operating with dual key certification centres is illustrated in annex D. The resulting certificates may be exchanged in advance of the communication of data keys and then stored by the parties. Alternatively, they can be exchanged as part of the sequence of messages used to communicate data keys in which case, parties do not need to maintain a store of their correspondent's certificates.

Each member of the communicating pair shall verify the validity of the certificate of the other member before using it.

3.3 Primitive processes

NOTE — The notation used in this clause is described in clause 1.4.

3.3.1 Encipherment, decipherment, authentication, error detection, signature construction and signature deconstruction

This subclause specifies primitive processes for use in key distribution by the method of key transport. Primitive processes for use in the key exchange method may be the subject of subsequent parts of this International Standard.

3.3.1.1 Approved algorithms and notations for encipherment, decipherment, authentication and error detection

Approved algorithms for asymmetric encipherment will be given in subsequent parts of this International Standard and the approved algorithms for authentication (also used for error detection) are given in ISO 8731. The notations for encipherment, decipherment, authentication and error detection, respectively, are as follows (both symmetric and asymmetric encipherment use these notations):

enciphered text = eK{plaintext}

plaintext = dK{enciphered text}

authentication code = MAC = aK{data}

error detection code = EDC = aKDX{data}

where key KDX, if of 64 bit length, shall be :

KDX = 0123456789ABCDEF

Where the key of the authentication algorithm is not of 64 bit length, the value of KDX shall be the quoted value appropriately reiterated or truncated.

The processes for asymmetric encipherment and decipherment as applied in this standard are specified in 3.3.1.5. and 3.3.1.6. and in figures 2 and 3.

3.3.1.2 Approved algorithms and notations for digital signature construction and deconstruction

Two forms of signature are employed in this part of ISO 11166:

- a) Signature construction by transforming the text.

Only the transformed text need be sent in the message but, optionally, the untransformed text may also be sent.

Approved processes for digital signature by transforming the text as applied in this part of ISO 11166 to the first sub-field of the field BES, will be specified in subsequent parts of this International Standard. This method of digital signature is also known as "digital signature with message/text recovery".

The notations for signature construction and deconstruction are as follows :

signed text = sKS{text}

text = vKP{signed text}

where KS and KP are the respective secret and public keys of an appropriate asymmetric key set.

Signature construction operates directly on the text to be protected.

Signature deconstruction recovers the original text. Verification of the correctness of signature shall employ redundancy. In this part of ISO 11166 redundancy is provided in the form of a MAC and a counter (CTP).

The processes for signature construction and deconstruction of type a), as applied to an enciphered block, are specified in 3.3.1.7 and 3.3.1.8 and in figures 4 and 5.

This form of digital signature may optionally be used in the formation of a certificate (see 3.3.4.1) and in this case, the redundancy shall be contained within the signed text.

- b) Signature construction by means of a separate signature.

Both the signature and the text are sent in the message.

When Cryptographic Service Messages are to be signed, the signature shall be computed using an algorithm approved for use with this part of ISO 11166. The signature is a function of the whole text to be signed and a secret key. The notation used is:

$SIG = \sigma_{KS}\{text\}$

where KS is the appropriate secret key of an asymmetric key set.

Signature deconstruction consists of extracting from the message the text which has been signed and verifying the value of SIG in relation to this text. The verification process is specified in the part of this International Standard which contains the applicable digital signature algorithm.

This form of digital signature may optionally be used in the formation of a certificate (see 3.3.4.3).

3.3.1.3 Encipherment and decipherment of initialisation vectors

When IVs are enciphered and deciphered with a symmetric algorithm, a data key (KD) shall be employed with the following notations :

enciphered IV = eKD {IV}

IV = dKD {enciphered IV}

When DEA is used, it shall be in ECB mode and IVs shall have a maximum length of 64 bits.

See 3.3.3 for the encipherment of IVs with asymmetric algorithms.

3.3.1.4 Encipherment and decipherment of a block using an asymmetric algorithm — General

The block to be enciphered is denoted BC and is formed as specified in 3.3.3. BC is not a field of any cryptographic service message. When BC is enciphered it becomes the first sub-field of BE, which is a field of the cryptographic service message of type KSM. When BC is both enciphered and signed it becomes the first sub-field of BES, which is a field of the cryptographic service message of type KSM.

Optional extra sub-fields of BE and BES identify the asymmetric key sets (KAMs)

employed for encipherment and signature construction. In the case of BE, the key set employed for signature of the whole message in the SIG field of the KSM may be identified in the third of its optional sub-fields (see 4.1.5.1).

3.3.1.5 Encipherment of a block by asymmetric keys

A block BC shall be enciphered with the asymmetric cryptographic algorithm using the following formula :

$$\text{enciphered BC} = eKP2\{BC\}$$

where :

BC is a block as described in 3.3.3.

KP2 is the (encipherment) public key of the receiver.

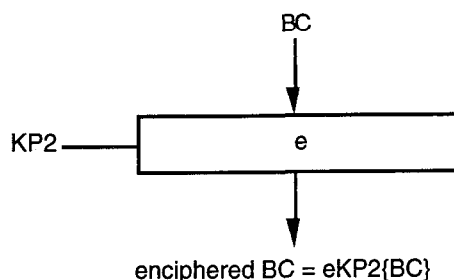


Figure 2 — Encipherment

3.3.1.6 Decipherment of an enciphered block by asymmetric keys

An enciphered block BC shall be deciphered using the following formula :

$$BC = dKS2\{\text{enciphered BC}\}$$

where :

enciphered BC is a block as described in 3.3.1.5.

KS2 is the (decipherment) secret key of the receiver.

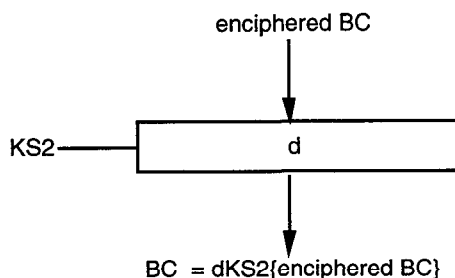


Figure 3 — Decipherment

3.3.1.7 Signature construction of an enciphered block by asymmetric keys

An enciphered block shall be signed by transforming the text with the asymmetric cryptographic algorithm using the following formula :

$$\text{Signed enciphered BC} = sKS1\{\text{enciphered BC}\}$$

where :

enciphered BC is specified in 3.3.1.5.

KS1 is the (signature) secret key of the sender

The combined processes of encipherment and signature are :

$$\text{signed enciphered BC} = sKS1\{eKP2\{BC\}\}$$

This combined process shall be performed as part of a single call to the key management facility.

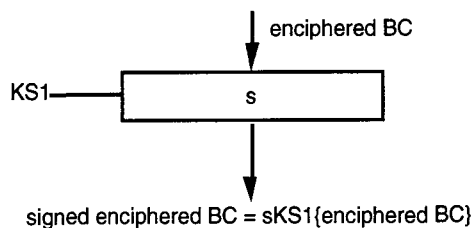


Figure 4 — Signature construction, as applied to enciphered BC

3.3.1.8 Signature deconstruction and decipherment of a signed enciphered block by asymmetric keys

The signature of a signed and enciphered block BC shall be deconstructed using the following formula :

$$\text{enciphered BC} = vKP1\{\text{signed enciphered BC}\}$$

where :

enciphered BC is specified in 3.3.1.5.

KP1 is the (signature) public key of the sender

The verification of the correctness of the signature shall employ redundancy in the form of a MAC and a counter (CTP) as specified in 4.3.4 (table 11, entries for BES and MAC).

The combined processes of signature deconstruction and decipherment are :

$$BC = dKS2\{vKP1\{\text{signed enciphered BC}\}\}$$

This combined process shall be performed as part of a single call to the key management facility.