

INTERNATIONAL STANDARD

**ISO
8732**

First edition
1988-11-15



INTERNATIONAL ORGANIZATION FOR STANDARDIZATION
ORGANISATION INTERNATIONALE DE NORMALISATION
МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ

Banking — Key management (wholesale)

Banque — Gestion de clés

Reference number
ISO 8732:1988 (E)

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 8732 was prepared by Technical Committee ISO/TC 68, *Banking and related financial services*.

Users should note that all International Standards undergo revision from time to time and that any reference made herein to any other International Standard implies its latest edition, unless otherwise stated.

Contents

	Page
Introduction	v
Section 1 : General	
1 Scope and field of application.....	1
2 References	1
3 Definitions	1
4 Abbreviations	3
5 Key management facility.....	5
6 Requirements of cryptographic equipment	5
7 Keying material	6
Section 2 : Manual distribution of keying material	
8 Despatch of manually distributed keying material	7
9 Receipt of manually distributed keying material	7
Section 3 : Automatic distribution of keying material	
10 Requirements for the automated key management architecture	9
11 Automated key management architecture.....	9
12 Encipherment and decipherment of keys and initialisation vectors	11
13 Cryptographic Service Messages	15
14 Generation of Cryptographic Service Messages	30
15 Processing Cryptographic Service Messages.....	49
Annexes	
A An example of manual key distribution and control procedures.....	71
B Notation.....	73
C Pseudo-random key and IV generator	75
D Windows and window management	77
E Dual Key Translation Centre application.....	79
F Keying material. Guidance on clearing and destruction procedures	81

ISO 8732 : 1988 (E)**Figures**

1	Key distribution architecture	9
2	Encipherment and decipherment of a single key by a single key	11
3	Encipherment and decipherment of a single key by a key pair	12
4	Encipherment and decipherment of a key pair by a key pair	12
5	Point-to-Point environment (normal message flow in sequence)	21
6	Point-to-Point environment (message flow with error messages)	21
7	Key Distribution Centre environment (normal message flow)	24
8	Key Distribution Centre environment (message flow with Error Service Messages)	24
9	Key Translation Centre environment (normal message flow)	27
10	Key Translation Centre environment (message flow with error messages)	27
11	Dual Key Translation Centre application (normal message flow)	80
12	Dual Key Translation Centre application (message flow with errors)	80

Tables

1	Processing counters (message authenticated)	14
2	Cryptographic Service Message: Fields and subfields	17
3	Fields used with each message type: Point-to-Point environment	23
4	Fields used with each message type: Key Distribution Centre environment	26
5	Fields used with each message type: Key Translation Centre environment	29
6	Contents of fields in Disconnect Service Message	30
7	Contents of fields in Error Recovery Service message	31
8	Contents of fields in Error Service Message	34
9	Contents of fields in Key Service Message	36
10	Contents of fields in Request For Service message	41
11	Contents of fields in Request Service Initiation message	43
12	Contents of fields in Response Service Message	44
13	Contents of fields in Response To Request message	46
14	Processing of Disconnect Service Message	49
15	Processing of Error Recovery Service message	51
16	Processing of Error Service Message	54
17	Processing of Key Service Message	56
18	Processing of Request For Service message	62
19	Processing of Request Service Initiation message	65
20	Processing of Response Service Message	66
21	Processing of Response To Request message	68
22	Processing counters with windows (message authenticated)	77

Introduction

This International Standard describes procedures for the secure management of the secret cryptographic keys used to protect messages in a wholesale banking environment, for instance messages between banks, or between a bank and a corporate customer, or a bank and a government.

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 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 these 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 electronic transmission of cryptographic keys (and, where needed, IVs) via a communication channel. Automated key distribution utilises two types of keys:

- 1) Key Enciphering Keys: used to encipher and decipher other keys.
- 2) Data keys: used to encipher and decipher initialisation vectors (IVs), to authenticate Cryptographic Service Messages, and to encipher/decipher or authenticate data.

Since key management facility(s) can be designed to replace electronically distributed Key Enciphering Keys and data keys automatically, manual intervention is kept to a minimum. Key Enciphering Keys generally have longer cryptoperiods than data keys.

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 the extent and detail of security and key management procedures. Absolute security is not practically achievable so key management procedures need not only to aim to reduce the opportunity for a breach of security but 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, exchange 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, whilst wherever possible this International Standard has specified requirements in absolute terms, in some instances a level of subjectivity 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.

This International Standard has been divided into sections, as follows:

- One: General
- Two: Manual distribution of keying material
- Three: Automatic distribution of keying material

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 detail to be agreed 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 example of the implementation of the requirements for manual distribution of keying material.

Banking — Key management (wholesale)

Section 1 : General

1 Scope and field of application

This International Standard 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. It specifies requirements for

- i) the control during its life of keying material 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 algorithm;
- 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 management process or when the integrity of the keying material is questioned.

It thus provides a means whereby an audit trail can be identified for all keying material.

This International Standard is designed for the use of symmetric algorithms for key distribution, where originator and recipient use the same key. It is designed for messages formatted and transmitted in coded character sets. It is intended that provision will, in due course, be made to cover the use of asymmetric algorithms for key distribution.

This standard does not provide a means to distinguish cryptographically between two physical parties when they share a common key.

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.

2 References

ISO 646, *Information processing — ISO 7-bit coded character set for information processing interchange*.

ISO 7982-1, *Bank telecommunications — Funds transfer messages — Part 1: Vocabulary and data elements*.

ISO 8372, *Information processing — Modes of operation for a 64-bit block cipher algorithm*.

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

ISO 8731, *Banking — Approved algorithms for message authentication*.

ANSI X3.92, *1981 Data Encryption Algorithm*.

3 Definitions

For the purpose of this International Standard the following definitions apply.

3.1 audit trail: see *security audit trail*.

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

3.3 bias: The condition where, during the generation of random or pseudo-random numbers, the occurrence of some numbers is more likely than others.

3.4 ciphertext: Enciphered information.

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

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

NOTE — A party and a Key Distribution Centre or Key Translation Centre exchanging Cryptographic Service Messages do not constitute a communicating pair.

3.7 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).

3.8 counter: An incrementing count used between two parties to control successive key distributions under a particular *Key Enciphering Key*.

3.9 cryptographic equipment: Equipment in which cryptographic functions (eg *encipherment*, *authentication*, key generation) are performed.

3.10 cryptographic key; key: A parameter used in conjunction with an algorithm for the purpose of *validation*, *authentication*, *encipherment* or *decipherment*.

3.11 cryptographic keying material: see *keying material*.

3.12 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.

3.13 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.

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

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

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

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

3.18 decryption: see *decipherment*.

3.19 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, eg *cryptographic key*.

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

3.21 encryption: see *encipherment*.

3.22 exclusive-or: see *modulo-2 addition*.

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

3.24 financial message: A message containing information which has financial implications.

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

3.26 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*.

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

3.28 key: see *cryptographic key*.

3.29 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*.

3.30 Key Distribution Centre: A facility which generates and returns *cryptographic keys* for distribution.

3.31 Key Enciphering Key: A *cryptographic key* used for the *encipherment* and *decipherment* of *cryptographic keys*.

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

3.33 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*.

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

3.35 key offset; offset: The result of adding a *counter* to a *cryptographic key* using *modulo-2 addition*.

3.36 Key Translation Centre: A facility which transforms and returns *cryptographic keys* for distribution.

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

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

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

3.40 Message Authentication Code (MAC): A *code* in a message between a sender and a receiver used to validate the source and part or all of the text of a message. The *code* is the result of an agreed calculation.

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

$$\begin{aligned} 0 + 0 &= 0 \\ 0 + 1 &= 1 \\ 1 + 0 &= 1 \\ 1 + 1 &= 0 \end{aligned}$$

3.42 notarisation: A method of modifying a *Key Enciphering Key* in order to authenticate the identities of the *originator* and the ultimate *recipient*.

3.43 notarising key: A *cryptographic key* used for *notarisation*.

3.44 notary seal: A value created from the identities of the *logical parties* of a *communicating pair*, and used in the creation of a *notarising key* (pair).

3.45 offset: See *key offset*.

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

3.47 plaintext: Unenciphered information.

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

3.49 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.

3.50 security audit trail: Data collected and potentially used to facilitate a *security audit*.

3.51 security life: The time span over which cryptographically protected data has value.

3.52 split knowledge: A condition under which two or more parties separately and confidentially have custody of the

constituent parts of a single key that, individually, convey no knowledge of the resultant *cryptographic key*.

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

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

4 Abbreviations

The following abbreviations are used in this International Standard:

The notation used in clauses 12 to 15 is described in annex B.

Abbreviation	Meaning	Description (see also table 2)
CKD	Key Distribution Centre	A facility which generates and returns cryptographic keys for distribution.
CKT	Key Translation Centre	A facility which transforms and returns keys for distribution.
CSM	Cryptographic Service Message	A message for transporting keys or related information used to control a keying relationship.
CTA	Counter A	Counter used between a CKD or CKT and party "A".
CTB	Counter B	Counter used between a CKD or CKT and party "B".
CTP	Counter P	Counter used in a Point-to-Point keying relationship.
CTR	Counter R	The value of the counter found to be in error.
DEA	Data Encryption Algorithm	—
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 the encipherment algorithm.
EDC	Error Detection Code	A code in a Cryptographic Service Message used to validate the data integrity of the message.
EDK	Effective Date of Key	Date and Co-ordinated Universal Time on which the data key is activated.
ERF	Error Field	The identification of error conditions detected in a prior Cryptographic Service Message.
ERS	Error Recovery Service	A message type used to recover from count or other errors in a Key Distribution Centre or Key Translation Centre environment.
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.
IDA	Identifier of Authentication Key	Identifies the key to be used to authenticate a Disconnect Service Message. The identified key is discontinued.
IDC	Identifier of Key Distribution Centre or Key Translation Centre	—
IDD	Identifier of Key to be Discontinued	—
IDK1	Key Identifier	Identifier of the key being transmitted in a Cryptographic Service Message.

ISO 8732 : 1988 (E)

Abbreviation	Meaning	Description
IDK2	Key Enciphering Key Identifier	Identifier (name) of the Key Enciphering Key or key pair used to encipher the key being transmitted in a Cryptographic Service Message.
IDU	Identity of Ultimate Recipient	The identity of the intended final recipient of a Cryptographic Service Message sent within a Key Distribution Centre or a Key Translation Centre environment.
IV	Initialisation Vector	—
KD	Data Key	A key used to encipher/decipher, or authenticate data.
KDU	Notarised Data Key	A data key enciphered under a notarising key (pair).
KDX	Fixed Data Key	A data key with fixed value used in the computation of an Error Detection Code.
KK	Key Enciphering Key	A cryptographic key used for the encipherment and decipherment of cryptographic keys.
*KK ¹⁾	Key Enciphering Key Pair	A pair of keys used for the encipherment and decipherment of keys.
KKM	Master Key Enciphering Key	The highest level Key Enciphering Key in a multi-layer key management architecture.
KKU	Notarised Key Enciphering Key	A Key Enciphering Key enciphered under a notarising key.
*KKU ¹⁾	Notarised Key Enciphering Key Pair	A Key Enciphering Key Pair enciphered under a notarising key pair.
KN	Notarising Key	A cryptographic key used for notarisation.
KSM	Key Service Message	A message type used to transfer keys between communicating pairs.
MAC	Message Authentication Code	—
MCL	Message Type	The tag for the field that defines the type of Cryptographic Service Message.
NOS	Notarisation Indicator	A tag that, when present, indicates that notarisation was used.
NS	Notary Seal	A value used for notarisation purposes.
ORG	Originator	Originator of CSM.
P	Key Parity	Indicates that the plaintext key conforms to the specification for odd parity.
RCV	Recipient	Recipient of CSM.
RFS	Request For Service Message	Used to request translation of keys by a Key Translation Centre for retransmission to another party.
RSI	Request Service Initiation Message	Used to request keys from another party.
RSM	Response Service Message	Used to provide an authenticated acknowledgement.
RTR	Response To Request Service Message	Used to send keys from a Key Distribution Centre or from a Key Translation Centre.
SVR	Service Request	Specifies type of service requested.

1) The asterisk* indicates that a pair of keys is involved. Where the use of a pair of keys is an option, in the main text the asterisk is enclosed in parentheses.

5 Key management facility

5.1 General

A key management facility shall provide 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.

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.

6 Requirements of cryptographic equipment

6.1 Generation of keys and initialisation vectors (IVs)

6.1.1 General

Key and IV generation procedures shall be under dual control.

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

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

Keys shall not be available in plaintext form from cryptographic equipment, even upon failure of the equipment, other than at the time of initial generation of a key.

A means shall be provided for the manual zeroisation of plaintext keys (see annex F).

6.1.2 Keys and IVs for manual distribution

All key generation, distribution and storage resources (eg copies, ribbons, etc) shall be protected from unauthorised use, alteration, replacement, destruction or exposure. Waste products shall be destroyed under dual control. The key generation process shall take place in an area where unauthorised viewing is prevented.

Where keys or IVs are printed, provision shall be made to protect them from unauthorised disclosure or replacement.

NOTE — Such protection may include uniquely identified key books with numbered pages protected by tamper resistant packaging so that page substitution is not possible.

Where a specially designed device containing electronically protected memory is used, safety devices shall be built into the software procedures or hardware to prevent unauthorised

access. Any attempts to gain unauthorised access into the protected memory shall result in the stored plaintext key being automatically erased, or otherwise rendered unintelligible. There shall be no external display, control or means of extracting the stored key without the linking or insertion of the device containing electronically protected memory into a secure receiver.

Where distribution of a key involves split knowledge, to ensure security, each key component shall be produced on a separate printed form or storage medium.

6.1.3 Keys and IVs for automated distribution

Where cryptographic equipment is used to generate keys and IVs automatically it shall be physically protected to prevent:

- 1) the disclosure, modification and replacement of the keys
- 2) the modification or replacement of the IVs
- 3) the modification or replacement of the key generation algorithm, or device.

6.2 Entry of keys

6.2.1 General

Cryptographic equipment shall permit, at either the system level or the device level, the entry of keys having a format complying with this standard. Access to key entry controls or systems shall be limited by physical or logical means, or both.

6.2.2 Manual entry of keys

A means shall be provided for the manual entry of keys or key components. A means of correcting individual errors or of re-entering the entire key shall be provided. If any plaintext key component is displayed it shall be visible only to authorised personnel and shall be cleared immediately after the key entry process is completed.

NOTE — Re-entry of an entire key may also be used as a means of verifying a previously entered key.

6.2.3 Automated entry of keys

Where a means is provided for automated entry of keys there shall be no display of the key during key entry. Keys retained on special devices such as key loaders shall be entered under dual control.

6.2.4 Parity checking

Where parity checking is available, the parity of plaintext keys or key components shall be verified during entry in order to preclude unintentional single bit modification of the key.

6.2.5 Storage of plaintext keys

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

6.2.6 Retention of electronically stored keys

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

ISO 8732: 1988 (E)

6.2.7 Electromagnetic interference

Protection shall be provided against compromise of keys as a result of radiation or conduction of electromagnetic interference from cryptographic equipment or key loaders.

6.2.8 Functional test

Immediately prior to manual key entry and system initialisation, the cryptographic equipment shall be subject to a test to check that it is operating correctly. This test shall include the operation of all control functions.

6.2.9 Operational error or failure

A means shall be provided to indicate the failure or incorrect operation of the cryptographic equipment (see also 6.1.1). A manual or automatic process shall be provided for the reporting and documentation of all such errors or failures.

6.3 Counter checking

Where keys are associated with counters (see 12.2) the cryptographic equipment shall provide a means for detecting and reporting the erasure, loss or lowering of a counter.

7 Keying material

7.1 Transportation and storage of keying material

Keying material shall be transported and stored in such a manner as to protect it against modification or substitution, and to prevent disclosure of plaintext keys before, during or after the period in which the keys are active.

Access to storage, including the movement of any keying material to or from storage, shall be under dual control. When keying material is entered or removed, the physical access shall be specifically authorised, physically or logically constrained, and fully documented.

7.2 Keys

7.2.1 Custody of keys

Dual control shall be maintained over keys at all times. Keys stored on a computer shall be enciphered or otherwise not be capable of being disclosed.

Lists of staff designated to hold or access keys shall be kept. These lists shall not contain any details of the content of keys.

7.2.2 Validity of keys

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

Data keys may be exchanged on the basis that they are for immediate or for future use (see 7.2.4). No key shall be operational until an authenticated acknowledgement has been received from the recipient. Where a key has not been specifically identified (eg by number or effective date) it shall be the only such key and shall be put into service by the communicating pair immediately after the recipient's acknowledgement is received by the originator.

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.

7.2.3 Key changes

Keys shall be changed:

- a) at the end of the cryptoperiod; or
- b) with the agreement of both members 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, an explicit date (or other implicit time reference) shall be specified whereupon the old key is no longer current. 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 7.2.5).

7.2.4 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.

7.2.5 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.

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

7.2.6 Back-up of keys

When a printed key is exchanged, the original printed form shall be retained for back-up. Where keys are exchanged automatically a protected copy shall be kept in storage. All back-up copies of active keys shall be subject to the same level of security control as keys in current use.

7.2.7 Destruction of keys (see also annex F)

All copies of keys that are no longer required shall be destroyed under dual control. Printed keys shall be destroyed by means of incineration, cross-cut shredding, or pulping, or other secure method.

Keys stored on magnetic media shall either be zeroised, under password control, or the magnetic media shall be destroyed as for printed keys.

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

Section 2 : Manual distribution of keying material

An example of manual key distribution and control procedures appears in annex A.

8 Despatch of manually distributed keying material

All documents accompanying manually distributed keying material shall be prepared prior to the generation of the keying material. This documentation shall include:

- a) A receipt for the keying material for signature by the recipient.
- b) Details of the recipient.
- c) Details of any passwords required for access to material distributed on magnetic storage media or other secure storage devices (eg key loaders).
- d) Where a courier service is used, a receipt for signature by the courier.
- e) Details of the date of generation of keying material, together with details of the issuer and the issue date.

All such documentation shall be signed by authorised signatories.

Once keying material has been generated (see 6.1), access to key components shall be controlled by the processes of dual control and split knowledge. Each key component shall be placed in a separate envelope which is sealed in such a manner that any subsequent unauthorised interference can be detected. Each envelope shall be marked to indicate its contents and the address of the appropriate function, and then placed in

a second, separate, envelope that is sealed, and addressed to the recipient. The second envelope shall give no indication of its contents.

NOTE — Each package thus consists of an outer envelope with a single inner envelope containing a single key component.

The individual components of a key shall be despatched, together with a receipt using a method to ensure separate despatch, for example, on different days. Any passwords required for access to magnetic storage media or other storage devices, eg key loaders, shall be despatched separately from the medium or device.

Where keying material is transported by mail then a secure method shall be used. Where delivery is by means of courier a receipt shall be obtained from the courier by the sender. The courier shall not be aware of the nature of the contents of an envelope.

9 Receipt of manually distributed keying material

Upon receipt of a package containing a key component the recipient shall examine the innermost envelope in order to check, so far as is possible, that access to its contents has not been attempted or achieved. If it is suspected that the security of the inner envelope has been compromised the sender shall be advised immediately. The signatures on the accompanying documentation shall be checked by the recipient for authenticity. The identity of the key components, eg sequence number or effective date, shall be recorded. When the recipient of the key component is satisfied with the authenticity of the key component the receipt that accompanied it shall be signed and returned (see also 7.2.2). Keys shall be placed in secure storage immediately upon receipt. Inner envelopes (ie those containing the key components) shall be retained under appropriate control (see clause 7).

Section 3 : Automatic distribution of keying material

10 Requirements for the automated key management architecture

This International Standard is designed to meet the following requirements for automated key management. It is assumed that:

- 1) the data network is expandable.
- 2) either a communicating pair has a Key Enciphering Key in common or each has a Key Enciphering Key Pair in common with a Key Distribution Centre or a Key Translation Centre.

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

10.2 Any communicating pair may share more than one Key Enciphering Key.

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

10.4 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.

10.5 A data key or Key Enciphering Key shared between a communicating pair shall not be disclosed to a third party (except for a Key Translation Centre (CKT) or a Key Distribution Centre (CKD)).

10.6 A key used between any communicating pair shall not intentionally be used between any other communicating pair.

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

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

10.9 The architecture shall support communicating parties that do not have a key generation capability.

10.10 The architecture shall support any party initiating a secure connection with any other party.

10.11 In a three layer architecture (see 11.1) the ability to exchange Key Enciphering Keys automatically between a communicating pair shall be provided.

11 Automated key management architecture

11.1 General

The architecture shall consist of either two or three layers of keys (see figure 1). All implementations shall have the capability of functioning in a two layer architecture.

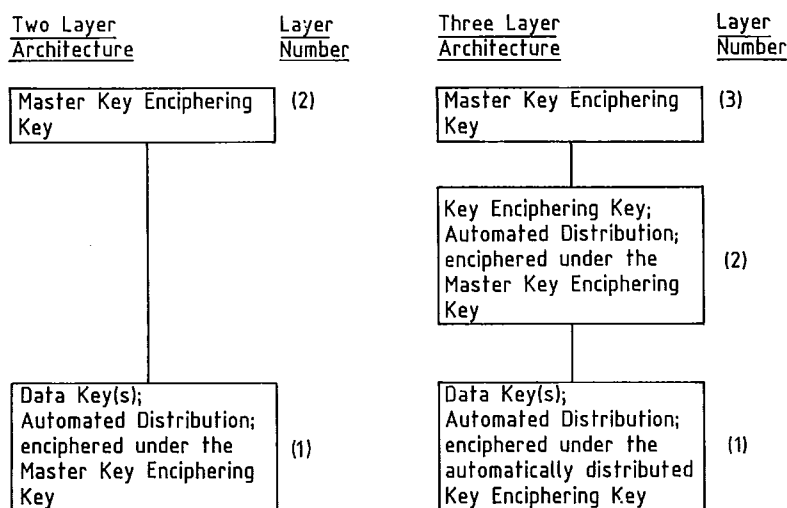


Figure 1 — Key distribution architecture

ISO 8732: 1988 (E)

In a two layer architecture the upper layer shall comprise a Master Key Enciphering Key or keys (KKM). The lower layer shall comprise a data key or keys (KD). These keys are enciphered using the KKM (see 11.2).

In a three layer architecture the uppermost layer shall comprise a Master Key Enciphering Key or keys (KKM). The second layer shall comprise one or more Key Enciphering Keys (KK) or key pairs (*KK). (These keys are enciphered using the KKM (see 11.2) prior to being automatically distributed). The lowest layer shall comprise a data key or keys (KD) enciphered using the KK or *KK (see 11.2). One or two data keys shall be transmitted with the automatically distributed Key Enciphering Key. These data keys shall be enciphered under the automatically distributed Key Enciphering Key. Subsequent key distribution messages need not include a Key Enciphering Key. When no Key Enciphering Key is transmitted, one or two data keys enciphered under a previously exchanged and automatically distributed Key Enciphering Key shall be sent.

11.2 Distribution of keying material

Master Key Enciphering Keys shall be manually distributed using the procedure specified in section two of this standard, or other secure means by mutual agreement. In no instance shall a master key be automatically distributed by a process defined in section three of this standard. Other keying material shall be distributed using Cryptographic Service Messages (see clause 13).

Automatically distributed Key Enciphering Keys or key pairs shall be enciphered prior to distribution using a Master Key Enciphering Key.

Data keys shall be enciphered prior to distribution using a Key Encipherment Key or key pair.

NOTE — Two data keys with the same identifier may be sent, enciphered, in a single Cryptographic Service Message (see clause 13).

The encipherment of data or initialisation vectors (IV) prior to distribution shall be carried out using data keys. Key Enciphering Keys shall only be used to encipher/decipher Key Enciphering Keys or data keys, and for no other purpose.

When a new key (keys) is received, all stored keys of the same type (Key Enciphering Keys or data keys) with the same name shall be replaced. In addition, when a key is discontinued, all keys of the same name (without regard to type) shall be discontinued.

Where keys are sent along with other protected data to the intended recipient, the recipient shall be able to recognise the key so that it can be deciphered and loaded before any cryptographic process can begin. If the recipient has multiple Key Enciphering Keys, information shall be sent identifying the key to be used.

11.3 Environments

11.3.1 Classification of key distribution environments

This sub-clause specifies requirements for the following key distribution environments:

- a) Point-to-Point;
- b) Key Distribution Centre (CKD);
- c) Key Translation Centre (CKT).

A Key Distribution Centre or Key Translation Centre may be used to reduce the number of Master Key Enciphering Keys in large networks. A mutually trusted party is designated as the centre. A *KK shared between any party and a Key Distribution Centre or Key Translation Centre permits secure communications to be established between that party and any other party that has a *KK relationship with the centre.

A Key Distribution Centre has the capability to generate and send data keys for distribution, and may send keys unsolicited or upon request.

A Key Translation Centre has the capability to transform and return keys for distribution by the originating party (eg, a Key Translation Centre does not require key generation capability; parties in the network have a peer relationship).

For compliance with this standard all key management systems providing automated distribution of keys shall provide a Point-to-Point implementation.

NOTE — The accommodation of a Key Distribution Centre or Key Translation Centre key management system is not a requirement of this standard.

11.3.2 Point-to-Point

A Point-to-Point environment exists when a communicating pair shares a Key Enciphering Key, either a single key or a key pair, so that further keys (whether Key Enciphering Keys or data keys) may be exchanged. At least one member of the communicating pair shall have the capability of generating or otherwise acquiring keys.

11.3.3 Key Distribution Centre

A Key Distribution Centre exists for the purpose of distributing generated or acquired data keys to a communicating pair, each member of which:

- a) wishes to communicate with the other but does not currently share keys; and
- b) shares a Master Key Enciphering Key Pair with the same Key Distribution Centre; but
- c) may not have the ability to generate keys

One member of the communicating pair (the originator) requests data key(s) from the Key Distribution Centre for later communication to the other member (the ultimate recipient).

The Key Distribution Centre shall generate or acquire data key(s) and shall send two sets to the originator using the Master Key Enciphering Key Pair shared with the originator. One set shall be notarised using the Key Enciphering Key shared between Party A and the centre; the other set shall be notarised using the Key Enciphering Key shared between Party B and the centre.

The originator shall then send the second set to the ultimate recipient.

11.3.4 Key Translation Centre

A Key Translation Centre is used to translate keys for future communication between a communicating pair, each member of which:

- a) wishes to communicate with the other but does not currently share keys;

b) shares a Master Key Enciphering Key Pair with the same Key Translation Centre; and

c) has the ability (through the originator) to generate or otherwise acquire keys

Key Enciphering Keys and data keys may be translated and exchanged, though only one of the two types (Key Enciphering Key or data key) shall be processed by the centre at one time.

A Key Enciphering Key or data key for use in future communication with the other member of the communicating pair (the ultimate recipient) shall be sent by the originator to the Key Translation Centre, enciphered under the offset Key Enciphering Key Pair shared between the originator and the Key Translation Centre.

The Key Translation Centre shall decipher this key, and then re-encipher it using notarisation and the Master Key Enciphering Key Pair shared with the ultimate recipient. The re-enciphered key shall be returned to the originator.

The originator shall redirect the re-enciphered key on to the ultimate recipient.

Dual Key Translation Centres may be implemented by the method described in annex E.

12 Encipherment and decipherment of keys and initialisation vectors

NOTE — The notation used in this clause is described in clause 4 and annex B.

12.1 Encipherment, decipherment, authentication and error detection

12.1.1 General

a) Master Key Enciphering Keys shared with a Key Distribution Centre (CKD) or a Key Translation Centre (CKT) shall comprise Key Enciphering Key Pairs (see 11.3.3 and 11.3.4).

b) Other communicating pairs shall share either single Master Key Enciphering Keys or Master Key Enciphering Key Pairs.

NOTE — Key pairs may be used where additional security is needed (eg, the Key Enciphering Key has a long security life).

c) Master Key Enciphering Keys shall not be superseded except by other Master Key Enciphering Keys.

d) Other Key Enciphering Keys may be single keys or key pairs. A Key Enciphering Key Pair shall not be enciphered/deciphered using a single key.

e) Data keys shall be single keys.

f) Key pairs shall not be used to encipher data.

The general formats for encipherment, decipherment, authentication and error detection, respectively are as follows:

$$(\text{enciphered quantity}) = eK(\text{plaintext quantity})$$

$$(\text{deciphered quantity}) = dK(\text{enciphered quantity})$$

$$(\text{authentication code}) = \text{MAC} = aKD(\text{data})$$

$$(\text{Error Detection Code}) = \text{EDC} = aKDX(\text{data}); \text{ where a 64 bit key is used KDX shall be}$$

$$KDX = 0123456789ABCDEF$$

NOTE — Where the key is not 64 bits in value, the value of KDX shall be this value appropriately reiterated or truncated.

The processes for encipherment and decipherment are specified in 12.1.2 to 12.1.6 and in figures 2-4. Authentication and error detection are described in 12.1.7 and 12.1.8.

Where DEA (ISO 8731-1) is used for encipherment/decipherment, it shall be in ECB mode.

12.1.2 Encipherment and decipherment of a single key by a single key

Where a single Key Enciphering Key is used, a single KK or KD shall be enciphered and deciphered with the cryptographic algorithm using the following formulae:

$$K(I)Z \text{ encipherment} = eKKY(K(I)Z)$$

$$\text{enciphered } K(I)Z \text{ decipherment} = dKKY(\text{enciphered } K(I)Z)$$

where:

$K(I)Z$ is a single key and may be a Key Enciphering Key, KKZ, or a data key, KDZ

KKY is a single Key Enciphering Key

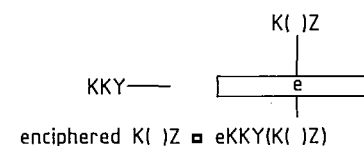


Figure 2(a) — encipherment

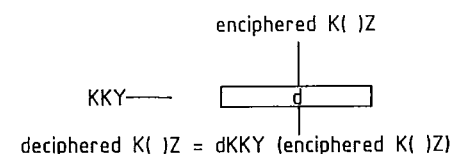


Figure 2(b) — decipherment

Figure 2 — Encipherment and decipherment of a single key by a single key

12.1.3 Encipherment and decipherment of a single key by a key pair

Where a Key Enciphering Key Pair is used, single KK or KD shall be enciphered and deciphered with the cryptographic algorithm using the following formulae:

ISO 8732: 1988 (E)

$$\begin{aligned} K()Z \text{ encipherment} &= \text{ede}^*KKY(K()Z) \\ &= \text{eKKIY}(\text{dKKrY}(\text{eKKIY}(K()Z))) \end{aligned}$$

$$\begin{aligned} K()Z \text{ decipherment} &= \text{ded}^*KKY(K()Z) \\ &= \text{dKKIY}(\text{eKKrY}(\text{dKKIY}(K()Z))) \end{aligned}$$

where:

$K()Z$ is a single key and may be a Key Enciphering Key, KKZ, or a data key, KDZ

*KKY is a Key Enciphering Key Pair; and

KKIY is the left member of the key pair,
KKrY is the right member of the key pair.

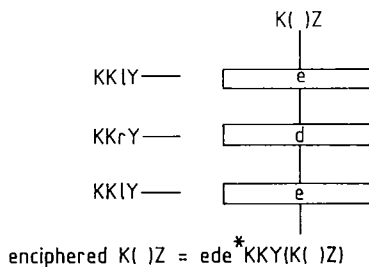


Figure 3(a) — encipherment

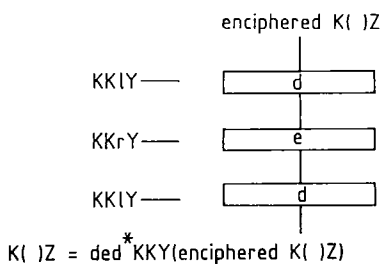


Figure 3(b) — decipherment

Figure 3 — Encipherment and decipherment of a single key by a key pair

$$\begin{aligned} *KKZ \text{ decipherment} &= \text{ded}^*KKY(*KKZ) \\ &= \text{dKKIY}(\text{eKKrY}(\text{dKKIY}(KKIZ))) \\ &\quad || \text{dKKIY}(\text{eKKrY}(\text{dKKIY}(KKrZ))) \end{aligned}$$

where:

*KKZ is the Key Enciphering Key Pair being enciphered; and

KKIZ is one member of the key pair,
KKrZ is the other member of the key pair

*KKY is a Key Enciphering Key Pair; and

KKIY is one member of the key pair,
KKrY is the other member of the key pair

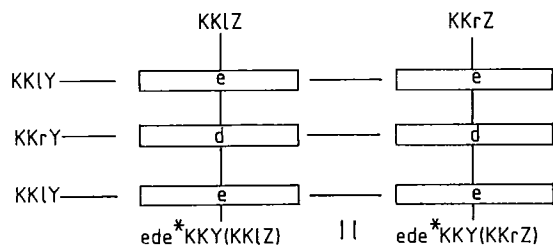


Figure 4(a) — encipherment

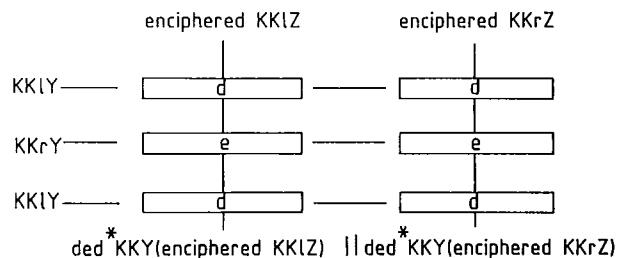


Figure 4(b) — decipherment

Figure 4 — Encipherment and decipherment of a key pair by a key pair

12.1.5 Encipherment and decipherment of a single key in systems using Key Enciphering Key Pairs

Facilities designed for use with key pairs can be made downward compatible to those using single keys by setting $KKIY = KKrY = KKY$.

12.1.6 Encipherment and decipherment of initialisation vectors

Where IVs are to be enciphered and deciphered, they shall be represented as hexadecimal characters. When DEA is used, IVs shall have a maximum length of 64 bits (represented as 16 hexadecimal characters).

$$\begin{aligned} *KKZ \text{ encipherment} &= \text{ede}^*KKY(*KKZ) \\ &= \text{eKKIY}(\text{dKKrY}(\text{eKKIY}(KKIZ))) \\ &\quad || \text{eKKIY}(\text{dKKrY}(\text{eKKIY}(KKrZ))) \end{aligned}$$

When IVs are enciphered and deciphered, they shall be enciphered and deciphered using a data key (KD) and with the cryptographic algorithm using the following formulae:

$$\text{enciphered IV} = \text{eKD(IV)}$$

$$\text{IV} = \text{dKD(enciphered IV)}$$

12.1.7 Authentication of Cryptographic Service Messages

When Cryptographic Service Messages are to be authenticated, the MAC shall be computed using a data key and the technique defined in ISO 8730 using the entire message text with no editing, and an authentication algorithm from ISO 8731. The following formula shall be used:

$$\text{MAC} = \text{aKD(data)}$$

A Cryptographic Service Message containing a single KD shall be authenticated using that KD. If two KDs are sent in a Cryptographic Service Message, the key used for authentication shall be the result of calculating the modulo-2 sum of the two data keys sent in the message.

12.1.8 Error detection

When it is desired to detect transmission or process errors and other means are not available, the authentication process specified in 12.1.7 shall be used, as in the following formula, to derive an Error Detection Code (EDC):

$$\text{EDC} = \text{aKDX(data)}$$

where KDX is a key with the fixed value 0123456789ABCDEF (see NOTE in 12.1.1).

12.2 Counters

12.2.1 Purpose

Counters are used to:

- detect duplicate Cryptographic Service Messages.
- detect a message received out of sequence.
- indicate the number of messages enciphered under a particular KK or *KK.
- resynchronise a service when counters indicate a loss of synchronisation.

Counters shall only increment, and never decrement, and shall never repeat during the use of the associated (*)KK.

Counters for key management purposes (as defined in this standard) shall be independent of the means used to differentiate between individual data messages. (eg. sequence numbers).

Counters shall be transmitted as fields in selected Cryptographic Service Messages and shall be authenticated by an appended MAC. Error status messages are an exception in that they are transmitted without a MAC.

NOTE — Counter integrity may be provided by an appended Error Detection Code (see 12.1.8).

Messages which use a Master Key Enciphering Key to encipher either KD (two layer architecture) or automatically distributed (*)KK (three layer architecture), shall use a counter which reflects the number of messages which have been transmitted using the particular Master Key Enciphering Key. In a three layer architecture where only KD(s) enciphered under an automatically distributed (*)KK appear in a Cryptographic Service Message, the counter shall reflect the number of messages which have been transmitted using the particular (*)KK for encipherment.

NOTE — A (*)KK may be key offset with a counter; the resulting key is used to encipher another key for transmission (see 12.3 and 12.4).

12.2.2 Management of counters

The contents of a counter shall be defined and manipulated as a binary number. Counters shall always be set to one upon successfully loading the associated (*)KK, except in the case where a (*)KK is being sent in a Request Service Message to a Key Translation Centre. In that case alone, the counter associated with the (*)KK being sent to the centre for translation shall be set to zero.

In a Point-to-Point and Key Translation Centre environment separate counters shall be maintained for each (*)KK shared between communicating pairs, ie an origination count and a reception count.

In a Key Distribution Centre (CKD) environment (where keys are never sent to a CKD; only from it) only one counter shall be maintained for each *KK shared between a party and a CKD, ie the CKD maintains an origination count and each communicating pair maintains its own reception count.

Any difference between the originator's counter and the recipient's counter shall be detected and, unless a 'window' has been agreed (see other considerations in 12.2.4 and annex D), the difference shall be immediately reported and the reason for its occurrence ascertained.

Duplicate messages shall be rejected and an error shall be reported to the originator. The recipient prepares an Error Service Message (see clause 14.4) in which the nature of the error being reported (duplicate message), the value of the reception counter (the recipient's expected count) and the value of the origination count as received in the related message, are included.

Where a counter resets to zero, is lost or lowered, the situation shall be interpreted as a catastrophic error and shall invalidate the associated (*)KK, which shall be immediately withdrawn from use. Detection of such a condition requires the enabling of an associated alarm. Recovery from such an event requires the use of a new (*)KK (either from storage or newly distributed), with the associated counters being reset to one.

12.2.3 Management of Cryptographic Service Messages (see table 1)

When a recipient receives a Cryptographic Service Message whose count equals the expected (stored) count, the message shall be accepted. Both the originator's counter and recipient's reception counter shall be incremented by one prior to the next message.

Where the counts do not match a log record shall be made.

ISO 8732: 1988 (E)

When the recipient receives a Cryptographic Service Message whose count is greater than the expected (stored) count, the message shall be accepted. The recipient's reception count shall be set to the received count plus one. This will be the new expected count.

When a recipient receives a Cryptographic Service Message whose count is less than the expected (stored) count, the message shall be rejected and an error shall be reported to the originator. The recipient shall return an Error Service Message in which the nature of the error being reported (count error), the value of the reception counter (the recipient's expected (stored) count) and the value of the count as received in the related

Cryptographic Service Message, shall be included.

NOTE — The count as received may be used to identify the Cryptographic Service Message which contained the counter in error.

The originator, upon receipt of the Error Service Message shall either:

- adjust his origination counter up to the expected count value returned in the Error Service Message, or
- establish a new (*)KK with the recipient and also reset the value of the associated counters to one.

**Table 1 — Processing counters
(message authenticated)**

Received count (CTA, CTB or CTP) equal to expected (stored) count	Received count (CTA, CTB, or CTP) greater than expected count	Received count (CTA, CTB, or CTP) less than expected (stored) count	Action to be taken on receipt of an ESM or ERS	
			Received count (CTA, CTB, or CTP) greater than origination count	Received count (CTA, CTB, or CTP) less than or equal to origination (stored) count
- o Accept message - o Generate RSM - o Log (optional) - o Increment expected count (+1)	- o Accept message - o Generate RSM - o Log (mandatory) - o Set expected count equal to received count + 1	- o Reject message - o Send ESM with expected count - o Log (mandatory)	- o Set origination (stored) count equal to expected count (CTA, CTB, or CTP) OR - o Change (*)KK associated with the CTA, CTB, or CTP - o Log (mandatory) - o Send new KSM or RTR with corrected count OR - o Wait for new key request (RSI or RFS)	- o Log (mandatory) - o Send new KSM or RTR with current origination (stored) count OR - o Wait for new key request (RSI or RFS)

12.2.4 Other considerations

In a CKD or CKT environment, it is possible for a recipient to receive Cryptographic Service Messages whose counts are out of sequence, yet the Cryptographic Service Messages are authenticated. In these two environments a party assuming the role of recipient may establish a window, representing a range of reception counter values such that the corresponding Cryptographic Service Messages may arrive out of sequence, and are accepted without declaring an error.

NOTE — A method of defining and managing a window is described in annex D.

12.3 Key offsetting

Key offsetting shall be used to transform a (*)KK *prior* to encipherment of a key by that (*)KK.

The modulo-2 sum of KB and CB shall be derived as follows:

$$KB + CB = (k_1 + c_1, k_2 + c_2, \dots, k_7 + c_7, p)$$

where:

KB is an eight bit key byte in which individual bits are designated as $k_1, k_2, \dots, k_7$ and p (see below).

CB is an eight bit counter byte in which individual bits are designated as $c_1, c_2, \dots, c_7, 0$.

p may be set to give the key byte, KB, odd parity.

The modulo-2 sum of a 64 bit key, KK, with a 56-bit counter is derived by modulo-2 adding the first byte of KK with the counter byte formed from the first seven high order bits of the counter (with a zero bit added) to derive the first byte of the result. Then the second byte of KK is modulo-2 added with the counter byte formed from the second seven bits of the counter (with a zero bit added). The process continues until the eighth byte of KK is modulo-2 added with the counter byte formed from the last seven bits of the counter (with a zero bit added).

The operation of key offsetting a single key is indicated by the expression:

$$KKo = (KK + CT)$$

where:

KKo is the offset Key Enciphering Key
 KK is the original Key Enciphering Key
 CT is the counter

Where a key pair is to be key offset, then the following equation is used:

$$*KKo = (KKI + CT) \parallel (KKr + CT)$$

where:

*KKo is the offset Key Enciphering Key Pair

KKI and KKr are the two members of the original key pair.

12.4 Notarisation of keys

Notarisation is a method for sealing keys with the identities of the communicating pair. It is achieved by creating a notary seal (NS). This sub-clause describes the derivation of the notary seal and how it is used. Once notarised, using the method described, keys can only be recovered with knowledge of the key used to perform notarisation and the identities of the communicating pair. A KD or a KK may be notarised before transmission by encipherment using a notarising key, (*)KN. (*)KN is formed by taking the modulo-2 sum of (*)KK with the notary seal (NS).

When notarisation is used the notarisation field "NOS" shall be present in the Cryptographic Service Message, unless the use of notarisation of a field is made explicit by the field tag (eg, K KU, K DU).

Where notarisation is to be achieved by combination with an ISO 646 character the following formula shall be used:

$$KB + AC = (k1 + b7, k2 + b6, \dots k7 + b1, p)$$

where:

KB is an eight bit key byte in which individual bits are designated as k1, k2 . . . k7 and p (see below)

AC is a eight bit character byte in which individual bits are designated as b1, b2 . . . b7, p (see below).

p may be set to give the byte odd parity.

Where notarisation is to be achieved by combination with an output byte of an encipherment algorithm the following formula shall be used:

$$KB + OB = (k1 + 01, k2 + 02, \dots k7 + 07, p)$$

where:

OB is any eight bit output byte of an encipherment algorithm, in which individual bits are designated 01, 02 . . . 08.

p may be set to give the byte odd parity.

The complete notarised key is formed by taking the modulo-2 sum of the eight pairs of bytes of a key and eight characters, or output bytes of an algorithm.

EXAMPLE

Suppose that Party A wishes to send a key to Party B. Let FM1 be the first eight characters of Party A's identity and let FM2 be the second eight characters of Party A's identity. Similarly let T01 and T02 represent the first and second halves of Party B's identity. If necessary, the identities are replicated to form

sixteen character identifiers. For example, if Party A's identity is 'CITYB' and Party B's identity is 'MANHAN', then FM1 is 'CITYBCIT'; FM2 is 'YBCITYBC'; T01 is 'MANHANMA'; and T02 is 'NHANMANH'.

Case 1: Computing a notarising key using a KK

Let KK be the key which is to be used to compute the notarising key. Then:

$$KKR = KK + FM1$$

$$KKL = KK + T01$$

$$NSI = eKKR(T02)$$

$$NSr = eKKL(FM2)$$

$$NS = [(leftmost\ 32\ bits\ of\ NSI) \parallel (rightmost\ 32\ bits\ of\ NSr)] + CT()$$

$$KN = KK + NS$$

KN is then used to notarise (by encipherment) either a KD or a KK.

Case 2: Computing a notarising key using a *KK

Let *KK be the key which is to be used to compute the notarising key. Then:

$$*KK = KKI \parallel KKr$$

$$KKR = KKr + FM1$$

$$KKL = KKI + T01$$

Then:

$$NSI = eKKR(T02) + CT()$$

$$NSr = eKKL(FM2) + CT()$$

NOTE — Where CT() is the counter used to key offset NSI and NSr, and the process for key offsetting is defined in section 12.3, above.

$$*KN = (KKI + NSI) \parallel (KKr + NSr)$$

KN is then used to notarise (by encipherment) either a KD or a ()KK.

13 Cryptographic Service Messages

13.1 General

Cryptographic Service Messages (CSM) shall be used for the automatic distribution and control of cryptographic keys and, where required, IVs, in a Point-to-Point environment and may be used to support a CKD or a CKT environment.

NOTE — For audit and control purposes, Cryptographic Service Messages may be journalised.

13.2 Message types

The following Cryptographic Service Message types are described in this standard:

ISO 8732: 1988 (E)

NOTE — The formats of these messages are specified in 13.4. The content is specified in clause 14.

a) Point-to-Point key management messages

1. Request Service Initiation message (RSI) (optional)

A message that requests that a new keying relationship be initiated. The originator of an RSI might not have a key generation capability.

2. Key Service Message (KSM) (mandatory)

A message that transfers a key from an originator to a recipient.

3. Response Service Message (RSM) (mandatory)

A message that provides an authenticated response to a KSM.

4. Error Service Message (ESM) (mandatory)

A message that reports an error in a previous Cryptographic Service Message.

5. Disconnect Service Message (DSM) (optional)

A message that is used to discontinue one or more keys.

b) Additional Cryptographic Service Message types for use with a Key Distribution Centre or Key Translation Centre

1. Request For Service message (RFS) (mandatory)

A message in a Key Translation Centre environment that sends keys to a Key Translation Centre to be translated for the ultimate recipient named in the IDU field.

2. Response To Requestor message (RTR) (mandatory)

A message in a centre environment that responds to an RFS, an ERS or to an RSI to the centre. An RTR may be initiated by a Key Distribution Centre.

3. Error Recovery Service message (ERS) (mandatory)

A message that reports count and key errors to the Key Distribution Centre or Key Translation Centre and requests resynchronisation of the count fields and re-initiation of service.

13.3 Character set and representation

The character set for Cryptographic Service Messages shall comprise the following:

0-9, A-Z (capital letter), comma (,), fullstop (.), space (b), solidus (/), hyphen (-), asterisk (*), left parenthesis ((), right parenthesis ()), carriage return and line feed.

The characters "." and "b" shall not be used in a subfield; "b" shall not be used within a field (except for the MAC field). The character "." shall only be used to separate subfields within a field.

All characters shall be represented as eight-bit characters (0, b7,

..., b1), where (b7, b6, ..., b1) are defined in ISO 646. Where this necessitates a code translation, the translation shall be for internal processing and computational purposes only.

Hexadecimal characters shall be represented by the characters 0-9 and A-F.

13.4 Message formats

a) The presence of a Cryptographic Service Message shall be denoted by "CSM".

b) A Cryptographic Service Message shall begin with a left parenthesis "(" and end with a right parenthesis ")"

c) Field tags (see table 2) shall be separated from field contents by a solidus "/"

d) Fields shall be separated by a space "b" and, if desired for readability, a carriage return and line feed.

e) Subfields within a field shall be separated by a full stop (or period) "." and, if desired for readability, a carriage return and line feed.

13.5 Fields and subfields

Cryptographic Service Messages shall comprise a tag (specifying that the message is a Cryptographic Service Message) and a sequence of fields, subfields and associated parameters.

Fields containing keys may consist of up to four subfields. Each subfield (whether present or not) shall be terminated by a full stop unless no subsequent subfield is present (see examples in NOTE). The ordering and content of the sub-field shall be as follows:

a) A key that has been enciphered or notarised shall always be in the first subfield and is the only subfield required.

b) The second subfield shall, if present, be a "P" to indicate that the plaintext key conforms to the specification for odd parity.

c) The third subfield shall, if present, contain the identity of the key sent in the first subfield.

d) The fourth subfield shall, if present, contain the identity of the key used to encipher the key sent in the first subfield.

NOTE — Examples of Key-Field Formats

1 KD/key . . . IDK1, is a field with two subfields containing data. IDK1 is the identity assigned to the KD. The key used to encipher the KD has not been explicitly defined. Hence its identity is assumed (is implicitly defined in the relationship).

2 KK/key.P.IDK1.IDK2, is a field that contains a key, KK, that conforms to the specification for odd parity and subsequently has been enciphered under the key whose identity is IDK2; the identity of KK is IDK1.

3 KDU/key, is a field with only one subfield that contains data. In this case, the identity of the new KD received in notarised form as KDU and the identity of the key used to decipher KDU are implicitly defined.

4 KD/key . . . IDK2, is a field with two subfields present. IDK2 is the identity of the key used to encipher the KD. No name is assigned to the KD.

Unless otherwise determined by prior agreement, if two KDs are sent, the first KD shall be used by the ultimate recipient for authentication; the second shall be used for encipherment.

13.6 Message flow

13.6.1 General

The fields and subfields shall be as defined in table 2. The fields

used with each message class, and their sequence, shall be as defined in tables 3 to 5.

When a message is received with an optional field(s) that is not implemented, an ESM with an "O" in the ERF field shall be returned except that, in the case of the EDC field, an ESM with an "O" in the ERF field may be returned or the field may be disregarded and processing may continue.

Table 2 — Cryptographic Service Message : Fields and subfields

Field tag	Name	Definition/remarks	Specification
CTA	Counter A	An incrementing binary counter. Associated with an *KK used to encipher either an (*)KK or KD(s) sent in a Cryptographic Service Message. Used between a CKD or a CKT and another party designated as "A". Set to one upon installation of this new *KK.	Up to 14 hex characters; leading zeros may be suppressed for transmission.
CTB	Counter B	An incrementing binary counter. Associated with an *KK used to encipher either an (*)KKU or KDU(s) sent in a Cryptographic Service Message. Used between a CKD or a CKT and another party designated as "B". Set to one upon installation of this new *KK.	Up to 14 hex characters; leading zeros may be suppressed for transmission
CTP	Counter P	An incrementing binary counter. Associated with an (*)KK or KD(s) sent in a Cryptographic Service Message. Used between communicating pairs, but not between a CKD or a CKT and another party. Set to one upon installation of this new (*)KK.	Up to 14 hex characters; leading zeros may be suppressed for transmission
CTR	Counter R	The value of the counter found to be in error.	Up to 14 hex characters; leading zeros may be suppressed for transmission.
EDC	Error Detection Code	The Error Detection Code when used shall be generated on all components of the associated service message (ERS for the CKD environment, ESM or RSI) using the editing, computation and formatting requirements for	9 characters (4 hex) <u>b</u> (4 hex)

Table 2 — Cryptographic Service Message : Fields and subfields (continued)

Field tag	Name	Definition/remarks	Specification
EDC (continued)		a MAC in ISO 8730. The hexadecimal key for EDC computation shall be 0123456789ABCDEF. (See NOTE in 12.1.1)	
EDK	Effective Date of Key	Date and Coordinated Universal Time of KD activation.	12 characters YYMMDDHHMMSS
ERF	Error Field	Error Codes are defined as: Up to 16 characters A : CTA error B : CTB error C : Cannot Process (Optional. May be used as a general error code where a more specific error code is not appropriate) D : CKD unknown E : Facility inoperative F : Format (syntax) error G : Reserved H : User defined I : Key Identifier not known to recipient K : Parity error in received key M : MAC error (failure to authenticate) O : Option not implemented P : CTP error T : CKT unknown U : IDU not known to the CKT or CKD X : EDC error (probable transmission error)	
IDA	Identifier of Key for Authentication	Identifies the key to be used to authenticate DSM. This key shall be discontinued.	Up to 16 characters
IDC	Identity of CKD or CKT	Identity of CKD or CKT to be used.	4 to 16 characters
IDD	Identifier of Key to be Discontinued	Identifies key to be discontinued.	Up to 16 characters
IDK1	Key Identifier	Identifies (names) the received key.	Up to 16 characters (subfield)
IDK2	Key Enciphering Key Identifier	Identifies (names) the (*)KK used to encipher/decipher a (*)KK or a KD.	Up to 16 characters (subfield)
IDU	Identity of Ultimate Recipient	This field is only used with a CKT or a CKD.	4 to 16 characters

Table 2 — Cryptographic Service Message : Fields and subfields (continued)

Field tag	Name	Definition/remarks	Specification
IV	Initialisation Vector	Starting point for encipherment/decipherment process. If enciphered, is always enciphered under a data key (KD). If only one data key is transmitted in a message, it shall be enciphered under that KD. If two KDs are transmitted, the IV shall be enciphered under the second KD. The first character shall be "E" (indicating that the IV is to be enciphered) or "P" (indicating that the IV is to be sent in plaintext form). This first character shall not form part of the IV for encipherment purposes'.	1 character followed by up to 32 characters (16 when DEA is used). Leading zeros of an enciphered IV shall not be suppressed for transmission.
KD	Data Key	Enciphered KD. May be used for encipherment or authentication. A maximum of two KD fields may be sent per message. If two KDs are sent, the first shall be used for authentication and the second shall be used for encipherment. In a Point-to-Point environment, the KD(s) sent in this field shall be notarised if and only if no KK field is present and the NOS field is present.	When DEA is used, 16 hex characters
KDU	Data Enciphering Key, Notarised	A KD enciphered under the *KN generated by the CKT or CKD for the ultimate recipient specified in the IDU field of a RTR or specified by the RCV field of a KSM that retransmits this key to the ultimate recipient. Up to two fields can be sent per message. If two KDUs are sent, the first shall be used for authentication and the second shall be used for encipherment.	When DEA is used, 16 hex characters
KK	Key Enciphering Key	Enciphered KK. In a Point-to-Point environment, the KK sent in this field shall be notarised if and only if the NOS field is present.	When DEA is used, 16 hex characters
*KK	Key Enciphering Key Pair	Enciphered KK Pair. In a Point-to-Point environment, the *KK sent in this field shall be notarised if and only if the NOS field is present.	When DEA is used, 32 hex characters

Table 2 — Cryptographic Service Message : Fields and subfields (continued)

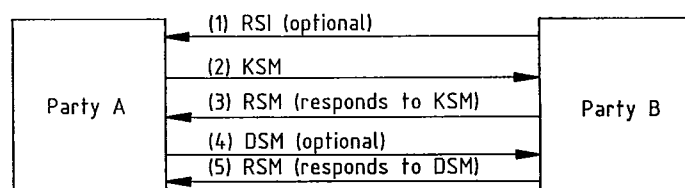
Field tag	Name	Definition/remarks	Specification
KKU	Key Enciphering Key, Notarised	A KK enciphered under the *KN generated by the CKT or CKD for the ultimate recipient, as specified in the IDU field.	When DEA is used, 16 hex characters
*KKU	Key Enciphering Key Pair, Notarised	A KK pair enciphered under the *KN generated by the CKT or CKD for the ultimate recipient, as specified in the IDU field.	When DEA is used, 32 hex characters
MAC	Message Authentication Code	The MAC shall be generated on all components of the associated Cryptographic Service Message using the editing, computation and format requirements of ISO 8730.	9 characters (4 hex) <u>b</u> (4 hex)
MCL	Message Type	Type of Cryptographic Service Message.	3 characters:- DSM, ERS, ESM, KSM, RFS, RSI, RSM or RTR.
NOS	Notarisation Indicator	Indicates use of notarisation process for (*)KK if (*)KK is present in a message. If no (*)KK is present, the KD(s) is (are) notarised.	Zero length field
ORG	Originator	Cryptographic Service Message originator.	4 to 16 characters
P	Key Parity	Used to indicate that the plaintext key conforms to the specification for odd parity.	1 character (subfield)
RCV	Recipient	Cryptographic Service Message recipient.	4 to 16 characters
SVR	Service Request	Specifies type of service requested. SVR requests one data key implicitly. KD requests two data keys. KK requests a Key Enciphering Key. *KK requests a Key Enciphering Key Pair (a KK or a *KK may be requested, but not both).	0 to 9 characters

Table 2 — Cryptographic Service Message : Fields and subfields (concluded)

Field tag	Name	Definition/remarks	Specification
SVR (continued)	Service Request	IV requests an IV. The IV shall be enciphered unless, by prior agreement, a plaintext IV is to be sent. A minimum of one and a maximum of three keys and an IV may be requested in a single ERS or RSI. Types requested shall be separated by periods.	0-9 characters

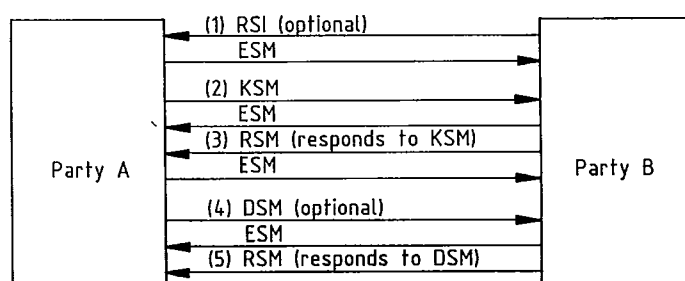
13.6.2 Point-to-Point environment

Figures 5 and 6 show the flow of Cryptographic Service Messages for the Point-to-Point environment.



Note—Either Party A or Party B may initiate the disconnect (DSM) process. Initiation by Party A is shown.

Figure 5 — Point-to-Point environment
(normal message flow in sequence)



Note—Either Party A or Party B may initiate the disconnect (DSM) process. Initiation by Party A is shown.

Figure 6 — Point-to-Point environment
(message flow with error messages)

ISO 8732: 1988 (E)

Table 3 defines the fields and their order for each message in this environment. Message flow shall be as follows:

- a) If one logical party of a communicating pair (Party B):
- 1) wishes to communicate with another logical party (Party A); and
 - 2) shares a (*)KK with Party A, and
 - 3) does not have key generation capability or access to keys;
- then

Party B sends an RSI to Party A requesting that Party A send key(s) and, optionally, an IV to Party B. If Party A receives an RSI from Party B with an error in it, an ESM shall be returned to Party B.

- b) If one logical party of a communicating pair (Party A):
- 1) wishes to send key(s) to another logical party (Party B); and
 - 2) shares a (*)KK with Party B; and
 - 3) has key generation or acquisition capability; then

Party A generates or acquires keys and optionally an IV and sends a KSM to Party B containing the key(s) (and IV).

If a (*)KK is sent, it shall be enciphered under a (*)KK shared with Party B and the accompanying KD(s) sent in the Cryptographic Service Message shall be enciphered under the (*)KK sent in that message.

If a (*)KK is not sent, the KD(s) sent in the Cryptographic Service Message shall be enciphered under a (*)KK shared with Party B.

The IV (if enciphered) shall be enciphered under the KD (the second KD if two KDs are sent) in that Cryptographic Service Message.

The Cryptographic Service Message shall be authenticated using the KD(s) sent in that message.

Party B shall respond with an RSM if the KSM is received correctly, or with an ESM if there is an error in the received KSM. If Party A receives an RSM which contains an error(s), Party A shall return an ESM to Party B.

Party A may resend a KSM to Party B an arbitrary number of times, but Party A shall not send a new KSM (ie, utilising new keys or a new count for the (*)KK specified in a message) until the old KSM is acknowledged by an RSM or an ESM.

- c) If either logical party of a communicating pair wishes to terminate a keying relationship or wishes to discontinue the use of a specific key(s), that party may send a DSM to the second party.

The second party shall respond with an RSM if the DSM is received correctly and all information contained in the DSM was applicable. Otherwise, the second party shall respond with an ESM.

When a DSM is sent, the key named by the IDA field (or the only data key shared between the originating and recipient parties if no IDA field is present) shall be retained to authenticate the subsequent RSM. That key shall then be discontinued. When a (*)KK is discontinued, all keys enciphered under that (*)KK shall also be discontinued without being named in the DSM.

When an RSM is received in error, no ESM shall be sent and manual recovery procedures are required.