INTERNATIONAL STANDARD

ISO 13492

First edition
1998-05-15

Banking — Key management related data element (retail)

Banque — Élément de données lié à la gestion des clés (services aux particuliers)

Reference number
ISO 13492:1998(E)

Contents

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

Printed in Switzerland

Foreword

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

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

International Standard ISO 13492 was prepared by Technical Committee ISO/TC 68, *Banking, securities and other financial services*, Subcommittee SC 6, *Retail financial services*.

Annexes A and B of this International Standard are for information only.

Introduction

This International Standard describes the structure and contents of a key management related data element that may be conveyed in electronically transmitted messages within the retail banking environment to support the secure management of cryptographic keys, where the retail banking environment involves the communications between a card-accepting device and an acquirer, and between an acquirer and a card issuer. Key management of keys used in an Integrated Circuit Card (ICC) and the related data elements are not covered in this International Standard.

This International Standard provides compatibility with the existing ISO standard on bank card originated messages (see ISO 8583).

Banking — Key management related data element (retail)

1 Scope

This International Standard describes a key management related data element that may be transmitted either in transaction messages to convey information about cryptographic keys used to secure the current transaction or in cryptographic service messages to convey information about cryptographic keys to be used to secure future transactions.

This International Standard addresses the requirements for the use of the key management related data element within ISO 8583, using the following two ISO 8583 data elements: Security Related Control Information (bit 53) or Key Management Data (bit 96). However, the transportation of key management related data is not limited to ISO 8583.

This International Standard is applicable to either symmetric or asymmetric cipher systems.

Key management procedures for the secure management of the cryptographic keys within the retail banking environment are described in ISO 11568. Security related data, such as PIN data and MACs, are described in ISO 9564 and ISO 9807, respectively.

2 Normative references

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

ISO/IEC 7812-1:1993, *Identification cards — Identification of issuers — Part 1: Numbering system.*

ISO/IEC 7812-2:1993, *Identification cards — Identification of issuers — Part 2: Application and registration procedures.*

ISO 8583:1993, *Financial transaction card originated messages — Interchange message specifications.*

ISO 8908:1993, *Banking and related financial services — Vocabulary and data elements*.

ISO 9564-1:1991, *Personal Identification Number management and security — Part 1: PIN protection principles and techniques.*

ISO 9807:1991, *Banking and related financial services — Requirements for message authentication (retail).*

ISO 11568-1:1994, *Banking — Key management (retail) — Part 1: Introduction to key management.*

ISO 11568-2:1994, *Banking — Key management (retail) — Part 2: Key management techniques for symmetric ciphers.*

ISO 11568-3:1994, *Banking — Key management (retail) — Part 3: Key life cycle for symmetric ciphers*.

ANSI X3.92:1987, *Data Encryption Algorithm.*

3 Definitions

For the purposes of this International Standard, the definitions given in ISO 8908 and the following definitions apply.

3.1
asymmetric cipher
a cipher in which the encipherment and decipherment keys are different and it is computationally infeasible to deduce the decipherment key from the encipherment key

3.2
cipher
a pair of operations that effect transformations between plaintext and ciphertext under the control of a parameter called a key

NOTE The encipherment operation transforms data (plaintext) into an unintelligible form (ciphertext). The decipherment operation restores the original text.

3.3
cryptographic algorithm
a set of rules specifying the procedures required to perform encipherment and decipherment of data

NOTE The algorithm is designed so that it is not possible to determine the control parameters (e.g. keys) except by exhaustive search.

3.4
cryptographic key; key
the control parameter of a cryptographic algorithm that cannot be deduced from the input and output data except by exhaustive search

3.5
cryptographic service message
a message for transporting keys or related information used to control a keying relationship

3.6
primary key
that key for a transaction from which other keys for the transaction are produced (e.g. by means of variants or transformations)

3.7
symmetric cipher
a cryptographic method using the same secret cryptographic key both for encipherment and decipherment

3.8
transaction message
a message used to convey information related to a financial transaction

4 Requirements for key management related data element

A key management related data element that conveys information about the associated transaction's key(s) is normally divided into sub-fields. This data element may be transmitted in a transaction where the nature of the sub-fields are implicitly known to the communicating parties. In environments where such transactions are exchanged, the parties may use the key management related data element as a private-use field and define its sub-fields in any mutually agreeable way. In other environments, transactions are exchanged where the nature of the sub-fields

are not implicitly known and therefore must be structured using a standardized representation to support interoperability. However, other environments, both types of transactions may be exchanged.

To distinguish between those transactions where the key management related data element must have a standardized representation and those transactions where it is used for private use, the first byte of the key management related data element shall be structured as a "control byte", where the control byte is defined as follows:

00-9F: The first sub-field of the key management related data element is a variable-length "key set identifier," as defined in 4.1 and 4.2.

A0-FF: The key management related data element is a private-use field, where the nature of the sub-fields are implicitly known to both communicating parties.

The use of key set identifiers provides a standardized way to convey any type of key management related information associated with a key management system. This approach eliminates the need to recognize specific key management techniques and to specify specific sub-fields to meet the needs of each such technique.

When the key management related data element begins with a key set identifier, the remainder of the data element contains whatever type of information is required to determine the key(s) needed to cryptographically process the transaction. Thus, there is no specified structure to the sub-fields contained in the remainder of the data element. Any information that may vary on a per transaction basis is conveyed following the key set identifier. This information normally includes the identity of a particular key(s) within a key set.

Key management related information that does not change from one transaction to the next need not be conveyed with every transaction. Rather, it may be implicitly known, or it may be installed concurrent with, and stored in association with, the corresponding key. Examples of information that may be implicitly known include the following:

— Key management technique used for the transaction's keys (e.g. static key, unique key per transaction).

— Format of enciphered or authenticated data (e.g. PIN block format).

— Encipherment algorithm used.

— Number of different keys used with the transaction and the purpose of each such key.

In some key management schemes, it may not be necessary to transmit a key management related data element in transaction messages. The need to transmit such a data element is discussed in annex A.

4.1 Concept of key set identifiers

A key set identifier is a number that uniquely identifies a key set, where a key set is a group of related keys that are all different but have certain characteristics in common, most notably:

— All are managed using the same key management method.

— The same high level key is used to encipher (for database storage) or derive all keys of the set.

— The remainder of the key management related data element (beyond the key set identifier) is identically structured for all keys of the set and is interpreted using the same logic.

Associated with any given key set is logic (e.g. computer software) at the acquiring host that may interpret the key management related data element to determine what key(s) is to be used with that transaction and how each such key is to be used.

Multiple key sets, with different key set identifiers, may use the exact same logic, differing only, for example, in the key encipherment key or the derivation key used to decipher or derive the key for the associated transaction.

The first byte of the key set identifier is the control byte (00-9F). Key set identifiers are assigned as described in 4.2. Key set identifiers are of variable length and do not have a specified maximum length. The length of the key set identifier is implicit. Therefore, the key management related data element shall not contain a "length" sub-field preceding the key set identifier that indicates the length of the key set identifier. Similarly, it is unnecessary for the key set identifier to be followed by a specified delimiter. (Note that if the key management related data element is transmitted in a variable length field, the key management related data element itself may be preceded by a length sub-field indicating the length of the entire data element, as is required in ISO 8583 for data elements Security Related Control Information and Key Management Data.)

Since key set identifiers are of variable length and the length is implicit, the acquiring host should store in the table of the key set identifiers that it recognizes the length of each key set identifier. When a host receives a transaction from, e.g. a POS terminal, the host should attempt to match the key set identifier in each such table entry with as many leftmost key management related data element digits as are specified for the particular table entry. Such a match indicates that this table entry holds the key set identifier that applies to the just-received key management related data element.

4.2 Assignment of key set identifiers

To prevent institutions from assigning duplicate key set identifiers, key set identifiers shall be assigned using either the six-digit Issuer Identification Numbers (IINs) as defined in ISO 7812 or the six-digit Institution Identification Codes (IICs) as defined in ISO 8583. The ISO Registration Authority assigns IINs to institutions that issue cards and IICs to institutions that do not issue cards. Since IINs and IICs are unique to the institution to which they are assigned and these two sets of numbers do not overlap, this ensures that, if two cryptographic environments are combined, key set identifiers that were unique in each separate environment will be unique in the combined environment.

An organization that wishes to obtain a key set identifier but has not been assigned an IIN or IIC may also obtain such an identifier from an institution that has been assigned an IIN or IIC. Such an institution shall ensure that it never assigns duplicate key set identifiers.

An institution may use an IIN or IIC directly as a key set identifier provided it will never need more key set identifiers than the quantity of IINs or IICs it has been assigned. If the institution requires additional key set identifiers, it shall concatenate one or more hexadecimal digits to the right of an IIN or IIC and in this way obtain multiple key set identifiers from a single IIN or IIC.

The institution allocating key set identifiers should choose how many digits (if any) to concatenate with its IIN or IIC to obtain its key set identifiers, prior to allocating any key set identifiers based on the IIN or IIC. For example, if an institution chooses to use seven-digit key set identifiers by concatenating a single digit with an IIN or IIC, it may not, after having used all 16 such seven-digit numbers, subsequently add an eighth digit to obtain additional key set identifiers. Such an eight-digit key set identifier would match, in the first seven digits, a key set identifier already assigned. For example, if the key set identifier seven-digit "1362047" already exists, the key set identifier eight-digit "13620475" is not allowed, and vice versa, because the one key set identifier is totally included within the other key set identifier. For an example of the usage of key set identifiers, see annex B.

5 Implementation in ISO 8583

When the key management related data element described in clause 4 is used with ISO 8583 to convey key management information for the current transaction message, the contents of the key management related data element shall be transmitted using ISO 8583 Security Related Control Information, which is a variable-length binary data element up to 48 bytes.

NOTE 1 Examples of ISO 8583 messages in which Security Related Control Information may be transmitted are an authorization or financial request that contains the Personal Identification Number (PIN) Data (bit 52) or a file action or network management message that contains the Message Authentication Code Field (bit 64 or 128).

When the key management related data element is used with ISO 8583 in cryptographic service messages to convey keying information for future use, the contents of the key management related data element shall be transmitted using ISO 8583 Key Management Data, which is a variable length binary data element up to 999 bytes.

NOTE 2 Examples of ISO 8583 messages in which Key Management Data may be transmitted are a network management request or request response either for confirming the synchronization of current keys or for exchanging future keys.

Annex A
(informative)

Uses for transmitted key management related data

A.1 Purpose for conveying data

There are two main reasons for conveying key management related information in transaction messages:

— To identify the key(s) used to protect the current transaction message (this normally requires identifying only a single primary key).

— To convey information about keys to be used to protect future transaction messages.

The second purpose is related primarily to the master-key session-key technique, the conveyed information being either a new session key ("working" or "transaction" key) enciphered under a master key, or a new master key enciphered under the previous master key.

It should be noted that, in retail banking systems conforming to ISO standards, the master-key session-key technique is of significant value if and only if:

— The master key is substantially less susceptible to exhaustive determination than is the session key; or

— The master key itself is replaced (e.g. by transmitting a new master key enciphered under the old master key) after the new session key has been successfully received and deciphered.

Since such key replacements are normally relatively infrequent (e.g. not more often than every few hours), efficiency in conveying information about future keys is not especially important. For example, if (as in ISO 8583) a primary bit map is used to convey transaction data, a data element to convey the key replacement information could reasonably be placed in a secondary bit map.

A.2 Description of data

When two parties exchange cryptographically protected transaction data, both parties should unambiguously understand the answers to the following questions for each message associated with such a transaction.

a) Question 1: What key is used for each cryptographic process associated with this transaction message?

b) Question 2: What cryptographic algorithm (e.g. DEA) is used for each cryptographic process?

c) Question 3: How does each cryptographic process use the indicated algorithm (e.g. "cipher-block-chaining")?

d) Question 4: What transaction data elements are cryptographically protected by each cryptographic process?

e) Question 5: How are the data elements formatted, in plaintext, for input into the cryptographic process?

f) Question 6: How is each cryptographic result formatted for inclusion in the transaction message?

g) Question 7: Where in the transaction message is each cryptographic result placed?

h) Question 8: How is the key management related information included in the transaction message to be interpreted?

Question 1 normally requires the identification of only one key — the primary key of the transaction. In most systems, the various keys used in a transaction are obtained by taking variants (or transformations) of a single primary key. To meet the requirements of this International Standard, the primary key used between one pair of communicating parties is normally different from that used between any other pair. It is also common for such a key to be frequently replaced, even to the point of using a unique key for each transaction.

Unlike question 1, the answers to questions 2-8 are normally unchanged for the life of the equipment or at least for an extended time. Since each cryptographic device is given an initial key, normally by a manual process, the answers to questions 2-8 above may generally be specified at the time of equipment installation or initial key distribution. It would be unusual for any such item to change and not be followed by a repeat of the initial key distribution process.

Quite commonly, two communicating parties hold a shared key that each party stores in enciphered form (the "stored key" method). Thus, the answers to questions 2-8, if not specified when the equipment is installed, may be exchanged during the initial key distribution process and may be stored with the enciphered key. Therefore, it is normally unnecessary to convey the specification of such items in transaction messages.

Not only are the answers to questions 2-8 normally fixed for long periods of time, but they are also commonly the same for many parties with which an electronic data processing (EDP) facility communicates. However, as indicated above, the primary cryptographic key (question 1) is normally different for each pair of communicating parties and commonly changes with time.

A.3 Explicit key identification

The first purpose for conveying key management related information in a transaction message — to identify the key(s) used to protect this message — requires the presence of a key management related data element in every such transaction message. Thus, efficiency is important, and such a data element should be carried in a primary bit map. Furthermore, as discussed subsequently, it is desirable to have some standardization in the structure of this data element when it serves as a key identifier.

The inclusion of key identifying information in every transaction message of a given type is called explicit key identification. The alternative is called "implicit" key identification. With implicit key identification, the key(s) used for the transaction message in question are determined from other transaction related information. For example, the key to be used might be determined by the communications line on which the message is received, or it might be determined from a "terminal identifier" included in the message for other purposes as well. Virtually all host-to-host transaction messages use implicit key identification, and many terminal-to-host transaction messages also use it.

A.3.1 Explicit key identification characteristics

The characteristics of explicit key identification make it especially useful in the POS environment, where a very large number of POS terminals interface with a single host. In such an environment the key may not be associated with the terminal *per se*, but perhaps with a PIN pad that is a physically separate unit that is installed or replaced independently of the terminal.

The use of explicit key identification may substantially eliminate the key management logistics that are otherwise required. A POS terminal may be installed, and a PIN pad connected to it, without any regard for key management considerations. Every transaction message from such a terminal includes a key identifier that identifies the key(s) used with that message. Therefore, there is no need to establish a keying relationship prior to the first transaction from such a terminal. Should the terminal's PIN pad fail and be replaced, there is no need to inform the acquiring host of the replacement prior to the first transaction using the new PIN pad.

An additional benefit of explicit key identification is that it may prevent the loss of cryptographic synchronization. When such a loss occurs, it is because the originating party uses a different key than the receiving party expects. The explicit identification of the key in the transaction message may eliminate any such misunderstanding.

Another benefit of explicit key identification is that it permits the use of "derived key" techniques as an alternative to more conventional stored key techniques. With stored key techniques, the acquiring host stores an (enciphered)

key for every terminal with which it communicates (using the POS environment as an example). With derived key techniques, such storage is unnecessary. The host holds a relatively small number of derivation keys, each common to many POS terminals. When the host receives a transaction from such a terminal, it is able to derive the key(s) used at that terminal by cryptographically processing the key identifier included in the transaction using the appropriate derivation key. As a simple example, the key identifier might be enciphered using the derivation key, the resulting cipher being the terminal's key. Note that all such terminal related keys are unique (assuming all have unique key identifiers), and that knowledge of one terminal's key provides no feasibly usable information about any other terminal's key.

With this technique there is no need for the host to maintain the database of enciphered terminal related keys that is needed for stored key techniques because any such key may be derived, using the key identifier included in the transaction and the common derivation key, when needed. The elimination of such a database may simplify key management logistics and operations.

It should be noted that key identifiers are non-secret and provide no information that an adversary could feasibly use to determine any associated key.

A.3.2 Key sets

When using explicit key identification with POS terminals, it is convenient to use the concept of a key set. Many (e.g. thousands of) terminals may use keys that are in a single key set. Although all the keys in a key set are different (except by chance), all such keys have the following in common:

— If the primary keys of the set are stored in enciphered form in the host's database, the same key encipherment key is used to decipher all of them. If the primary keys of the set are derived, the same derivation key is used by the host to cryptographically compute all of them.

— The primary keys of the set are managed identically.

— Any additional keys are obtained from the associated primary key in the exact same manner (e.g. by using the same variants).

— The key management related data elements that identify the set's keys are identical in structure and are interpreted using identical logic.

— The answers to questions 2-8 listed in A.2 are implemented identically for all devices holding keys in the set, or else the key management related data elements specify the implementation of any item for which the implementation may vary from one of these devices to another.

For each key set it is necessary for the host to store the high level key used to decipher or derive each primary key, as well as information concerning how this key is used to implement the required cryptographic functions. (This information may take the form of a computer program, a list of parameters, or a combination of the two.) When a stored key technique is used, each key set also requires a table of enciphered keys. When a derived key technique is used, no per-terminal information need be stored for use in determining a terminal related key. (However, some per-terminal information should be stored for auditing and control purposes, but the accidental loss of this information does not impact the host's ability to determine a terminal's key.) Thus, the concept of key sets may be used to minimize and systematize the key management information that a host needs to store, and therefore simplify the host's implementation of key management procedures.

A.3.3 Key identifiers

When key sets are used with explicit key identification, it is useful to include in the identification of a key the identification of the set to which that key belongs. This is called a key set identifier, as described in 4.2 and 4.3, and, when used, forms the most significant (leftmost) portion of the key identifier.

When a key set identifier is used, a key identifier may consist of two or three fields, as follows:

— key set identifier;

— device identifier;

— key replacement counter (optional).

The device identifier indicates a specific device whose key(s) is within the indicated key set. For stored key systems, the device identifier may be used to locate, in the host's key table for the indicated key set, the location that holds the key(s) related to that particular device. For derived key systems, the device identifier is used to cryptographically compute the key initially loaded into that device, using the derivation key of that particular key set.

In systems that implement automatic key replacement, a key replacement counter may be included in the key identifier. This counter indicates how many key replacements have occurred since the initial key was loaded. When a key replacement occurs on every transaction, this field becomes a transaction counter. The use of a key replacement counter (or transaction counter) is necessary when automatic key replacements occur in derived key techniques and useful (to detect and recover from loss of cryptographic synchronization) when they occur in stored key techniques.

It is not necessary to have a standardized manner for representing device identifiers or key replacement counters. The representation of these fields is fixed for any key set (as indicated in the preceding key set definition). Therefore, associated with each key set may be computer software and/or parameters that define the security related sub-fields.

It should be noted that key set identifiers assigned in accordance with this International Standard provide a standardized methodology in which explicit key identification may be used with non-standardized security and key management techniques.

When explicit key identification is used in a POS environment, it is generally used only in terminal-originated transaction messages. Once the acquiring host has received a terminal-originated message, the host determines (from the message's key identifier) the primary key currently associated with this terminal and uses this key or another related key for the cryptographic protection of data to be transmitted back to the terminal. Since the terminal already knows this key, there is no need to send a key identifier back to the terminal.