INTERNATIONAL STANDARD

ISO/IEC 11770-2

Third edition
2018-10

IT Security techniques — Key management —

Part 2: Mechanisms using symmetric techniques

Techniques de sécurité IT — Gestion de clés —

Partie 2: Mécanismes utilisant des techniques symétriques

Reference number
ISO/IEC 11770-2:2018(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Fax: +41 22 749 09 47
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 27, *Security techniques*.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

This third edition cancels and replaces the second edition (ISO/IEC 11770-2:2008), which has been technically revised. It also incorporates ISO/IEC 11770-2:2008/Cor 1:2009.

The main changes compared to the previous edition are as follows:

— the list of requirements in Clause 5 has been updated;

— an optional message and mechanism identifier to the encrypted strings sent within each of the mechanisms has been added;

— the set of inputs for calculation of the key in Mechanism 5 has been expanded;

— minor changes have been made to the fourth message in Mechanism 8 and the second message in Mechanism 10.

A list of all parts in the ISO/IEC 11770 series can be found on the ISO website.

Introduction

The purpose of key management is to provide procedures for handling cryptographic keying material to be used in symmetric or asymmetric cryptographic algorithms according to the security policy in force.

Key establishment mechanisms using symmetric cryptographic techniques can be derived from the entity authentication mechanisms of ISO/IEC 9798-2 and ISO/IEC 9798-4 by specifying the use of text fields available in those mechanisms. Other key establishment mechanisms exist for specific environments. Besides key establishment, the goals of such a mechanism can include unilateral or mutual authentication of the communicating entities. Further goals can be the verification of the integrity of the established key, or key confirmation.

IT Security techniques — Key management —

Part 2:
Mechanisms using symmetric techniques

1 Scope

This document defines key establishment mechanisms using symmetric cryptographic techniques.

This document addresses three environments for the establishment of keys: Point-to-Point, Key Distribution Centre (KDC), and Key Translation Centre (KTC). It describes the required content of messages which carry keying material or are necessary to set up the conditions under which the keying material can be established.

This document does not indicate other information which can be contained in the messages or specify other messages such as error messages. The explicit format of messages is not within the scope of this document.

This document does not specify the means to be used to establish initial secret keys; that is, all the mechanisms specified in this document require an entity to share a secret key with at least one other entity (e.g. a TTP). For general guidance on the key lifecycle, see ISO/IEC 11770-1. This document does not explicitly address the issue of inter-domain key management. This document also does not define the implementation of key management mechanisms; products complying with this document are not necessarily compatible.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 11770-1, *Information technology — Security techniques — Key management — Part 1: Framework*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 11770-1 and the following apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at http://www.electropedia.org/

3.1
distinguishing identifier
information which unambiguously distinguishes an entity

3.2
entity authentication
corroboration that an entity is the one claimed

[SOURCE: ISO/IEC 9798-1:2010]

3.3
explicit key authentication from *A* to *B*
assurance for entity *B* that entity *A* (and possibly additional identified trusted parties) is the only other entity that is in possession of the correct key

Note 1 to entry: Implicit key authentication from *A* to *B* is assurance for entity *B* that entity *A* (and possibly additional identified trusted parties) is the only other entity that can possibly be in possession of the correct key.

Note 2 to entry: Implicit key authentication from *A* to *B* and key confirmation from *A* to *B* together imply explicit key authentication from *A* to *B*.

Note 3 to entry: Where a mechanism is claimed to provide explicit key authentication, this assumes that the mechanism has completed successfully, i.e. both parties send and receive all messages correctly and succeed in processing them correctly. In any protocol, it is not possible from within the protocol itself for the sender of the final message to know whether this message has been received correctly by the intended recipient. If assurance is required that this has occurred, then this can typically be obtained from the context of use of the mechanism, e.g. if a message is received from the recipient making use of an established key.

[SOURCE: ISO/IEC 11770-3:2015]

3.4
key confirmation
key confirmation from *A* to *B*
assurance for entity *B* that entity *A* is in possession of the correct key

Note 1 to entry: Where a mechanism is claimed to provide key confirmation, this assumes that the mechanism has completed successfully, i.e. both parties send and receive all messages correctly and succeed in processing them correctly. In any protocol, it is not possible from within the protocol itself for the sender of the final message to know whether this message has been received correctly by the intended recipient. If assurance is required that this has occurred, then this can typically be obtained from the context of use of the mechanism, e.g. if a message is received from the recipient making use of an established key.

[SOURCE: ISO/IEC 11770-3:2015]

3.5
key control
ability to choose the key, or the parameters used in the key computation

3.6
key derivation function
KDF
function which takes as input a number of parameters, at least one of which is secret, and which gives as output keys appropriate for the intended algorithm(s) and applications

Note 1 to entry: In ISO/IEC 11770-2:2008, such a function was referred to as a key generating function.

[SOURCE: ISO/IEC 11770-6:2016, 3.3 — modified, Note 1 to entry has been changed and Note 2 to entry has been removed.]

3.7
point-to-point key establishment
direct establishment of keys between entities, without involving a third party

3.8
random number
time variant parameter (3.11) whose value is unpredictable

3.9
redundancy
information that is known and can be checked

3.10
sequence number
time variant parameter (3.11) whose value is taken from a specified sequence which is non-repeating within a certain time period

3.11
time variant parameter
data item used to verify that a message is not a replay, such as a *random number* (3.8), *sequence number* (3.10), or a time stamp

3.12
unknown key share attack
attack in which all the entities involved in the mechanism complete the mechanism successfully to establish a shared secret key K, but do not agree on the identity of the entity with which they share the key.

4 Symbols and abbreviated terms

A, B	entities between which a key is established
$e_K(Z)$	result of encrypting data Z with a symmetric encryption algorithm using the secret key K
f	key derivation function
F	keying material
F_X	keying material originated by entity X
I_X	the distinguishing identifier of entity X
K	secret key that is established between entities A and B as a result of the use of one of the mechanisms specified in this document; K can be part of F or computed from F using a key derivation function NOTE K is not to be confused with K_{AB}, the long-term secret key shared by A and B.
KDC	Key Distribution Centre
KTC	Key Translation Centre
K_{XY}	secret key associated with entities X and Y
MAC	Message Authentication Code
$MAC_K(Z)$	result of applying a MAC function to data Z using the secret key K
P	Key Distribution Centre or Key Translation Centre
R	random number
R_X	random number issued by entity X
SID^i_m	constant uniquely identifying the mechanism (m) and the instance of encryption (i) within the mechanism
T/N	time stamp or sequence number

$Text_1$, $Text_2$, ..., fields that can contain optional data for use in applications outside the scope of this document (they can be empty). Their relationship and contents depend upon the specific application. One such possible application is message authentication (see Annex C for an example). Likewise, optional plaintext text fields may be included as a prefix, or appended, to any of the messages. They have no security implications and are not explicitly included in the mechanisms specified in this document.

TVP Time Variant Parameter

TVP_X Time Variant Parameter issued by entity X

T_X/N_X time stamp or sequence number issued by entity X

$X||Y$ The result of the concatenation of data items X and Y in the order specified. In cases where the result of concatenating two or more data items is encrypted as part of one of the mechanisms specified in this document, this result shall be composed so that it can be uniquely resolved into its constituent data strings, i.e. so that there is no possibility of ambiguity in interpretation. This latter property can be achieved in a variety of different ways, depending on the application. For example, it can be achieved by (a) fixing the length of each of the strings throughout the domain of use of the mechanism, or (b) encoding the sequence of concatenated strings using a method that guarantees unique decoding, e.g. using the distinguished encoding rules defined in ISO/IEC 8825-1[1].

[...] Data items that are optional in the mechanisms are shown in square brackets, [thus].

5 Requirements

The key establishment mechanisms specified in this document make use of symmetric cryptographic techniques, more specifically, symmetric encryption algorithms, MACs, and/or key derivation functions. The cryptographic algorithms and their key life-times shall be chosen such that it is computationally infeasible for a key to be deduced during its lifetime. If the following additional requirements are not met, the key establishment process can be compromised.

a) For those mechanisms making use of a symmetric encryption algorithm, either assumption 1) or assumption 2) is required.

   1) The encryption algorithm, its mode of operation, and the redundancy in the plaintext shall provide the recipient with the means to detect forged or manipulated data.

   2) The integrity of the encrypted data shall be ensured by a MAC.

   In order to achieve 1) or 2), it is recommended that choices for encryption and integrity algorithms be in accordance with the following.

      i) Assumption 1) above is guaranteed if an authenticated encryption technique is used; use of one of the techniques standardized in ISO/IEC 19772[10] is recommended.

      ii) The choice for a symmetric encryption algorithm should be chosen from amongst those standardized in ISO/IEC 18033-3, ISO/IEC 18033-4, ISO/IEC 29192-2 and ISO/IEC 29192-3.

      iii) If a block cipher encryption algorithm is used, then the mode of operation employed should be one of those standardized in ISO/IEC 10116[4].

      iv) If a MAC is used, then the techniques should be chosen from amongst those standardized in ISO/IEC 9797 (all parts)[2] and ISO/IEC 29192-6.

   NOTE 1 When a KDC or KTC is involved, assumptions 1) and 2) are not always equivalent in terms of the ability to unambiguously detect on which link an active attack is being performed. See Annex C for examples.

b) In each exchange specified in the mechanisms of 6, 7 and 8, the recipient of a message shall know the claimed identity of the originator. If this is not the case, i.e. if the context of use of the mechanism does not establish the claimed identity, then this can, for example, be achieved by the inclusion of identifiers in additional plaintext text fields of one or more of the messages.

The specifications of many of the mechanisms in this document require the correctness of an identifier included in a message to be checked. This should be done by comparing the received identifier with the expected identifier (as specified in the mechanism concerned). If the identifier in question is that of the originator of the message, then the recipient should know the value of the expected identifier because of requirement b).

c) Keying material can be established using either secure or insecure communication channels. When using only symmetric cryptographic techniques, at least the long-term key(s) K_{AB} (and K_{AP}, K_{BP}, where relevant) shall be exchanged between two entities using a secure channel in order to allow secure communications.

d) The key establishment mechanisms in this document require the use of time variant parameters, such as time stamps, sequence numbers, or random numbers. In this context, the use of the term random number also includes unpredictable pseudo-random numbers. The properties of these parameters, in particular that they are non-repeating, are important for the security of these mechanisms. For additional information on time variant parameters, see ISO/IEC 9798-1:2010, Annex B[3]. For means of generating random numbers, see ISO/IEC 18031 [8].

e) The secret key(s) used in implementations of any of the mechanisms specified in this document shall be distinct from keys used for any other purposes.

f) The data strings encrypted at various points in a key establishment mechanism shall not be composed so that they can be interchanged.

NOTE 2 This can be enforced by including the following elements in each encrypted data string, e.g. within a text field.

— The object identifier as specified in Annex A, in particular identifying the ISO/IEC standard number (11770), the part number (2) and the authentication mechanism.

— A constant that uniquely identifies the encrypted string within the mechanism. This constant may be omitted in the mechanisms that involve only one encrypted string. A specific proposal for including this information in the form of a special identifier is provided in ISO/IEC 9798-2[3], and is included as an optional element in the mechanism specifications in this document.

If any additional element is included in an encrypted string, then its presence shall be checked by the recipient after decryption, and the mechanism shall fail if any such check fails.

g) In the mechanisms involving a KDC or a KTC, the holder of a key K_{AP} (or K_{BP}) shall always use it in the same way, i.e. acting either as the KDC or KTC P or the entity A (or B). That is, no entity shall act as the KDC or KTC in one instance of a mechanism and act as A or B in another instance of the mechanism, and use the same key in both cases. That is, if a single entity uses a mechanism acting as P in one instance and as A (or B) in the other, then the keys it uses in the two instances shall be distinct.

The mechanisms specified in this document are believed to offer protection against unknown key share attacks. However, additional protection against such attacks can be provided if a key K established using one of the mechanisms specified in this document is subject to further processing before use by applying one of the key derivation functions (KDFs) specified in ISO/IEC 11770-6[7]. The inputs to the KDF, which should be agreed in advance by the relevant parties, should include the key K and the identifiers of the parties who will share the secret key. The output of the KDF can then be used as an operational key.

Annex A defines object identifiers that shall be used to identify the mechanisms specified in this document.

6 Point-to-point key establishment

6.1 General

Underlying every key management scheme is a point-to-point key establishment procedure, the use of which requires that the entities already share a key so that further keys can be established directly between the two entities. In Clause 6, six point-to-point key establishment mechanisms are specified.

For the implementation of the mechanisms specified in Clause 6, it is required that:

a) a long-term secret key K_{AB} is shared by entities A and B. This key is assumed to be bilateral, i.e. it is capable of being used for encrypting and decrypting data strings (or as input to a key derivation function in the case of Mechanism 1) by both A and B;

b) at least one of A and B is able to generate, acquire or contribute to a secret key, K, as described in the individual mechanism;

c) the security requirements for the mechanism include the confidentiality of the established secret key, K, and the detection of modification or replay of keys and messages.

The mechanisms specified in Clause 6 should not be used to update the long-term secret key, K_{AB}.

6.2 Key establishment mechanism 1

In key establishment mechanism 1, the key, K, is derived from a time variant parameter TVP, i.e. a random number, R, a time stamp, T, or a sequence number, N, using a key derivation function. Key establishment mechanism 1 does not provide authentication of the key, K, established by the mechanism. The mechanism requires that A is able to generate a TVP.

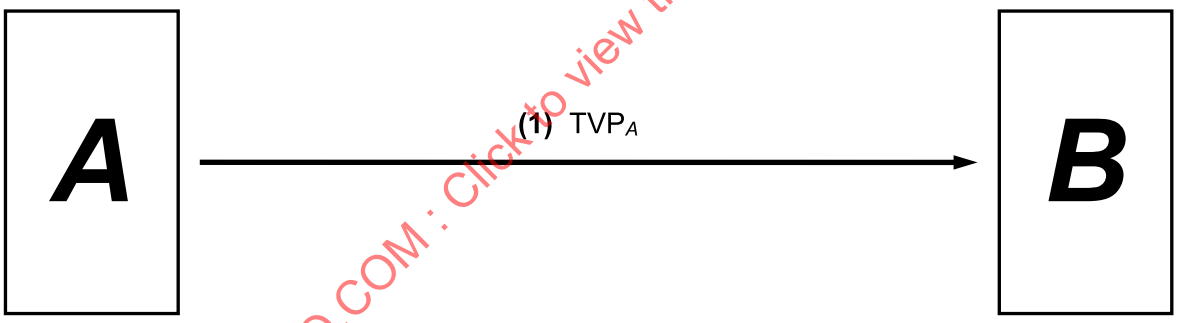

Figure 1 — Mechanism 1

The mechanism involves the following steps (see Figure 1).

1) A generates a time variant parameter, TVP_A, which can be a random number, R_A, a time stamp, T_A, or a sequence number, N_A, and transfers it to B. Both A and B then derive the key, K, by using a key derivation function, f, which takes as inputs the shared secret key, K_{AB}, and the time variant parameter, TVP_A:

$$K = f(K_{AB}, \text{TVP}_A)$$

The function, f, should be one of the one-step key derivation functions specified in ISO/IEC 11770-6:2016, Clause 6; in the notation of ISO/IEC 11770-6:2016, $s = K_{AB}$, should be used as the secret input and $t = \text{TVP}_A$ as the salt.

NOTE 1 A poor choice for the key derivation function, f, can result in an insecure key establishment process. Implementors of this mechanism are very strongly advised to use a KDF specified in ISO/IEC 11770-6:2016, Clause 6. Use of an ISO/IEC 11770-6 KDF for f has not been made mandatory in order to allow implementations complying with a previous edition of this document to also be compliant with this document.

NOTE 2 In order to also provide entity authentication, key establishment mechanism 1 can be combined with an authentication mechanism as specified in ISO/IEC 9798-2 or ISO/IEC 9798-4. See Annex C for an example.

6.3 Key establishment mechanism 2

In key establishment mechanism 2 the key, *K*, is supplied by entity *A*. The mechanism does not provide authentication of the key, *K*, established by the mechanism, nor does it provide entity authentication.

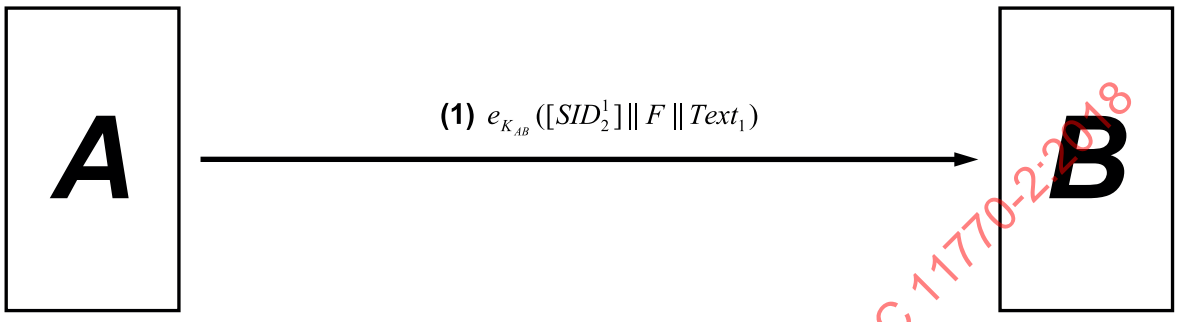

Figure 2 — Mechanism 2

The mechanism involves the following steps (see Figure 2).

1) *A* sends *B* the keying material, *F* (made up of a key, *K*, together with optional data), encrypted using the key, K_{AB}. On receipt of the message, *B* deciphers the encrypted part, and thus obtains the key, *K*.

6.4 Key establishment mechanism 3

Key establishment mechanism 3 is derived from the one-pass unilateral authentication mechanism specified in ISO/IEC 9798-2. In this mechanism, the key, *K*, is supplied by entity *A*.

Key establishment mechanism 3 provides unilateral authentication, i.e. the mechanism enables entity *B* to authenticate entity *A*. It also provides key confirmation and explicit key authentication from *A* to *B*.

Uniqueness/timeliness is controlled by time stamps or sequence numbers. The mechanism requires that both *A* and *B* are able to maintain mechanisms for generating or verifying the validity of time stamps, T_A, or sequence numbers, N_A, respectively.

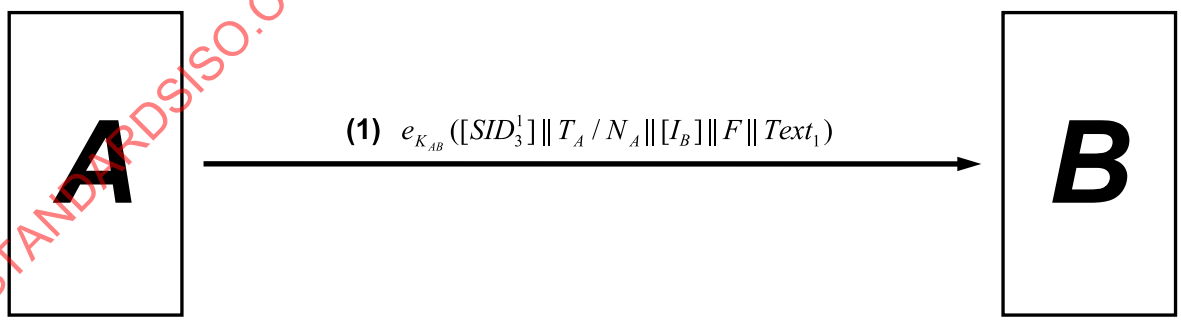

Figure 3 — Mechanism 3

The mechanism involves the following steps (see Figure 3).

1) *A* sends *B* a time stamp, T_A or sequence number, N_A, the distinguishing identifier, I_B, and the keying material, *F* (made up of a key, *K*, together with optional data). The inclusion of the distinguishing identifier, I_B, is optional. The data fields are encrypted using the key, K_{AB}. On receipt of the message, *B* deciphers the encrypted part, checks the correctness of its distinguishing identifier, if present, checks the time stamp or sequence number, and obtains the key, *K*.

NOTE Distinguishing identifier I_B is included in step 1) to prevent a substitution attack, i.e. the re-use of this message by an adversary masquerading as *B* to *A* (see Annex B). In environments where such attacks cannot occur, the identifier can be omitted.

6.5 Key establishment mechanism 4

Key establishment mechanism 4 is derived from the two-pass unilateral authentication mechanism specified in ISO/IEC 9798-2. In this mechanism, the key, *K*, is supplied by entity *A*.

Key establishment mechanism 4 provides unilateral authentication, i.e. the mechanism enables entity *B* to authenticate entity *A*. It also provides key confirmation and explicit key authentication from *A* to *B*.

Uniqueness/timeliness is controlled by a random number, R_B. The mechanism requires that *B* is able to generate random numbers.

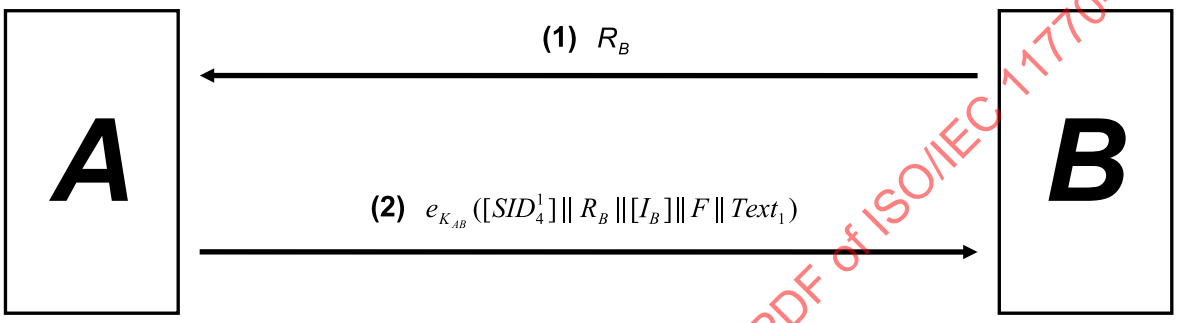

Figure 4 — Mechanism 4

The mechanism involves the following steps (see Figure 4).

1) *B* sends *A* a random number, R_B.

2) *A* sends *B* the received number, R_B, the distinguishing identifier, I_B, and the keying material, *F* (made up of a key, *K*, together with optional data). The inclusion of the distinguishing identifier, I_B, is optional. The data fields are encrypted using the key, K_{AB}. On receipt of message 2, *B* deciphers the encrypted part, checks the correctness of its distinguishing identifier, if present, checks that the random number, R_B, sent to *A* in step 1, was used in constructing message 2, and obtains the key, *K*.

NOTE Distinguishing identifier I_B is included in step 2 to prevent a substitution attack, i.e. the re-use of this message by an adversary masquerading as *B* to *A* (see Annex B). In environments where such attacks cannot occur, the identifier can be omitted.

6.6 Key establishment mechanism 5

Key establishment mechanism 5 is derived from the two-pass mutual authentication mechanism specified in ISO/IEC 9798-2. This mechanism enables both *A* and *B* to contribute part of the established key, *K*.

Key establishment mechanism 5 provides mutual authentication between *A* and *B*. It also provides key confirmation and explicit key authentication from *B* to *A*.

Uniqueness/timeliness is controlled by time stamps or sequence numbers. The mechanism requires that both *A* and *B* are able to maintain mechanisms for generating and verifying the validity of time stamps, *T*, or sequence numbers, *N*.

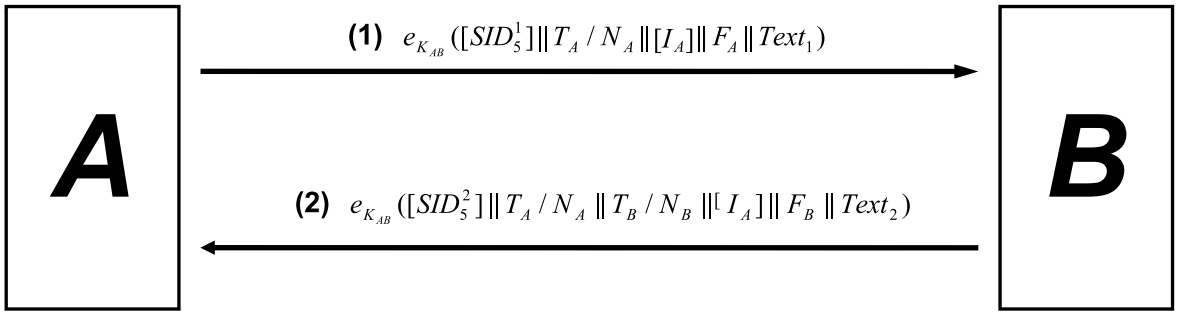

Figure 5 — Mechanism 5

The mechanism involves the following steps (see Figure 5).

1) *A* sends *B* a time stamp, T_A, or sequence number, N_A, the distinguishing identifier, I_B, and the keying material, F_A. The inclusion of the distinguishing identifier, I_B, is optional. The data fields are encrypted using the key, K_{AB}. On receipt of message 1, *B* deciphers the encrypted part, checks the correctness of its distinguishing identifier, if present, and checks the time stamp or sequence number.

2) *B* sends *A* the time stamp, T_A, or sequence number, N_A, it received from *A*, its own time stamp, T_B, or sequence number, N_B, the distinguishing identifier, I_A, and the keying material, F_B. The inclusion of the distinguishing identifier, I_A, is optional. The data fields are encrypted using the key, K_{AB}.

   i) On receipt of message 2, *A* deciphers the encrypted part, checks the correctness of its distinguishing identifier, if present, checks that the sequence number, T_A, or sequence number, N_A, equal the value it sent in message 1, and checks the time stamp, T_B, or sequence number, N_B, provided by *B*.

   ii) Both *A* and *B* derive the key, *K*, using a key derivation function, *f*, taking as input the secret keying material fields, F_A and F_B, and (at least) the time stamps or sequence numbers, T_A or N_A and T_B or N_B:

$$K = f(F_A, F_B, T_A/N_A, T_B/N_B, [I_A], [I_B], [\text{Text1}], [\text{Text2}], [SID_5^1]).$$

In steps 1 and 2, either of the two keying material fields, F_A and F_B, may be empty, but not both.

The function, *f*, should be one of the two-step key derivation functions specified in ISO/IEC 11770-6:2016, Clause 7; in the notation of ISO/IEC 11770-6:2016, $s = F_A \| F_B$ should be used as the secret input to the key extraction function and $t = T_A/N_A \| T_B/N_B [\| I_A] [\| I_B] [\| Text_1] [\| Text_2] [\| SID_5^1]$ as the salt input to the key expansion function.

NOTE 1 A poor choice for the key derivation function, *f*, can result in an insecure key establishment process. Implementors of this mechanism are very strongly advised to use a KDF specified in ISO/IEC 11770-6:2016, Clause 7. Use of an ISO/IEC 11770-6 KDF for *f* has not been made mandatory in order to allow implementations complying with a previous edition of this document to also be compliant with this document.

NOTE 2 If either of the keying material fields, F_A and F_B, is empty, the key is computed using the key derivation function, *f*, as described above, but with the relevant one of the two inputs equal to either the empty string or a fixed string (depending on the nature of the function *f*).

NOTE 3 Distinguishing identifier, I_B, is included in step 1 to prevent a substitution attack, i.e. the re-use of this message by an adversary masquerading as *B* to *A* (see Annex B). For similar reasons, distinguishing identifier, I_A, is present in step 2. In environments where such attacks cannot occur, one or both of the identifiers can be omitted.

6.7 Key establishment mechanism 6

Key establishment mechanism 6 is derived from the three-pass mutual authentication mechanism specified in ISO/IEC 9798-2. This mechanism enables both *A* and *B* to contribute part of the established key *K*.

Key establishment mechanism 6 provides mutual authentication between *A* and *B*. It also provides key confirmation and explicit key authentication from *B* to *A*.

Uniqueness/timeliness is controlled by random numbers. The mechanism requires that both *A* and *B* are able to generate random numbers.

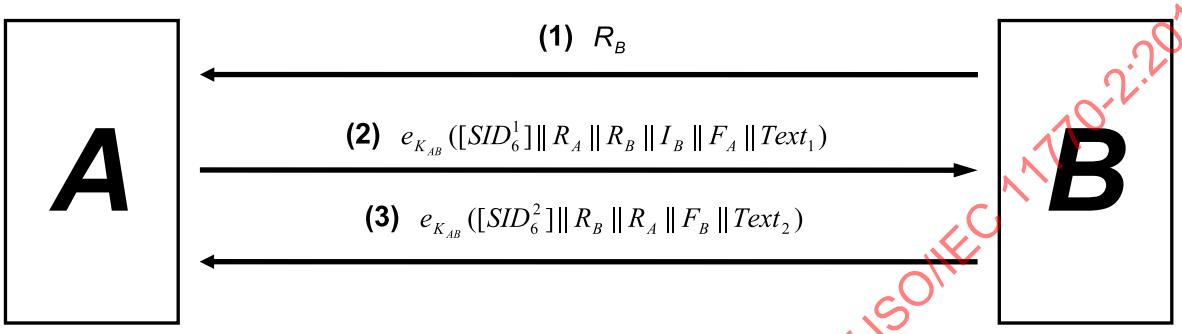

Figure 6 — Mechanism 6

The mechanism involves the following steps (see Figure 6).

1) *B* sends *A* a random number, R_B, in message 1.

2) *A* sends *B* a random number, R_A, the received number, R_B, the distinguishing identifier, I_B, and the keying material, F_A, in message 2. The inclusion of the distinguishing identifier, I_B, is optional. The data fields are encrypted using the key, K_{AB}. On receipt of message 2, *B* deciphers the encrypted part, checks the correctness of its distinguishing identifier, if present, and checks that the random number, R_B, sent to *A* in step 1, was used in constructing message 2.

3) *B* sends *A* the random numbers, R_B and R_A, and the keying material, F_B, in message 3. The data fields are encrypted using the key, K_{AB}.

   i) On receipt of message 3, *A* deciphers the encrypted part and checks that the random numbers, R_B and R_A, sent in messages 1 and 2, respectively, were used in constructing message 3.

   ii) Both *A* and *B* derive the key, *K*, using a key derivation function, *f*, taking as input the secret keying material fields, F_A and F_B, and (at least) the random numbers, R_A and R_B:

$$K = f(F_A, F_B, R_A, R_B, [I_B], [Text_1], [Text_2], [SID_6^1]).$$

In steps 2 and 3, either of the two keying material fields, F_A and F_B, may be empty, but not both.

The function *f* should be one of the two-step key derivation functions specified in ISO/IEC 11770-6:2016, Clause 7; in the notation of ISO/IEC 11770-6:2016, $s = F_A \| F_B$ should be used as the secret input to the key extraction function and $t = R_A \| R_B [\| I_B] [\| Text_1] [\| Text_2] [\| SID_6^1]$ as the salt input to the key expansion function.

NOTE 1 A poor choice for the key derivation function, *f*, can result in an insecure key establishment process. Implementors of this mechanism are very strongly advised to use a KDF specified in ISO/IEC 11770-6:2016, Clause 7. Use of an ISO/IEC 11770-6 KDF for *f* has not been made mandatory in order to allow implementations complying with a previous edition of this document to also be compliant with this document.

NOTE 2 Distinguishing identifier, I_B, is included in step 2 to prevent reflection attacks (see Annex B). In environments where such attacks cannot occur, the identifier can be omitted.

NOTE 3 A variant of key establishment mechanism 6 can be constructed from two parallel instances of key establishment mechanism 4, one started by entity A and the other by entity B. In this case, the key would be computed as in step 3 above, i.e. as $K = f(F_A, F_B, R_A, R_B, [I_A], [I_B], [Text_1], [Text_2])$.

7 Mechanisms using a Key Distribution Centre

7.1 General

The purpose of a Key Distribution Centre (KDC) is to first generate or acquire and then distribute a key to a pair of entities that both share a key with the KDC.

In Clause 7, four key establishment mechanisms using a KDC are specified.

— In the first three mechanisms, one of the two entities requests a key, K, from the KDC for later distribution to the other entity. The KDC generates or acquires the key, K, and sends a message to the requesting entity protected by a key shared with this entity. This message contains a second message protected by a key shared between the KDC and the second entity, which can then be forwarded by the requesting entity to the ultimate recipient.

— In the fourth mechanism the KDC generates or acquires the key, K, and sends it directly to both communicating entities. The two messages are protected using the key that the KDC shares with the corresponding entities.

For all of these mechanisms, only the KDC is required to have the ability to generate or otherwise acquire keys. Following the distribution of a key by the KDC, the two entities can use this key to support the use of a point-to-point key establishment mechanism.

For the implementation of the mechanisms specified in Clause 7 it is required that:

a) entities A and B share secret keys, K_{AP} and K_{BP}, respectively, with a trusted third party, P, that acts as a KDC; the KDC shall be able to generate or otherwise acquire a key, K;

b) the KDC shall have a means of communication with the entity requesting a key;

c) the security requirements for the mechanism include the confidentiality of K, and the detection of modification, substitution or replay of keys and messages.

7.2 Key establishment mechanism 7

Key establishment mechanism 7 does not provide authentication of the key, K, established by the mechanism.

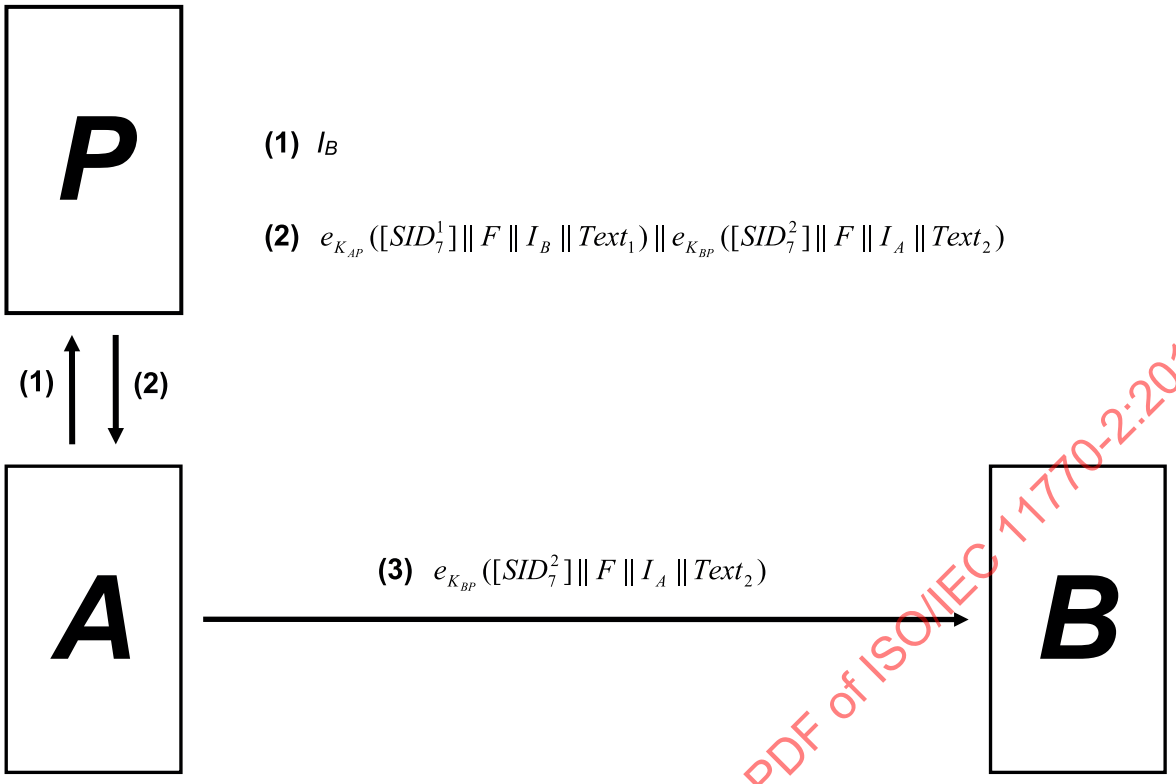

Figure 7 — Mechanism 7

The mechanism involves the following steps (see Figure 7).

1) *A* requests keying material from the KDC by sending message 1 to the KDC containing the distinguishing identifier, I_B, of the second entity *B*.

2) The KDC sends message 2 to *A* containing the keying material, *F* (made up of a key, *K*, together with optional data). This message consists of two main parts:

   — $e_{K_{AP}}([SID_7^1] \| F \| I_B \| Text_1)$;

   — $e_{K_{BP}}([SID_7^2] \| F \| I_A \| Text_2)$.

   On receipt of message 2, *A* deciphers the first part, checks the correctness of the distinguishing identifier, I_B, and obtains the key, *K*.

3) *A* forwards the second part of message 2 to *B* in message 3. On receipt of message 3, *B* deciphers the encrypted part, checks the correctness of the distinguishing identifier, I_A, and obtains the key, *K*.

7.3 Key establishment mechanism 8

Key establishment mechanism 8 is derived from the four-pass mutual authentication mechanism specified in ISO/IEC 9798-2.

Key establishment mechanism 8 optionally provides mutual authentication between *A* and *B*. It also provides key confirmation and explicit key authentication from *A* to *B*, and, optionally, from *B* to *A*.

Uniqueness/timeliness is controlled by time stamps or sequence numbers. The mechanism requires that *A*, *B* and the KDC are able to maintain mechanisms for generating and verifying the validity of time stamps, *T*, or sequence numbers, *N*.

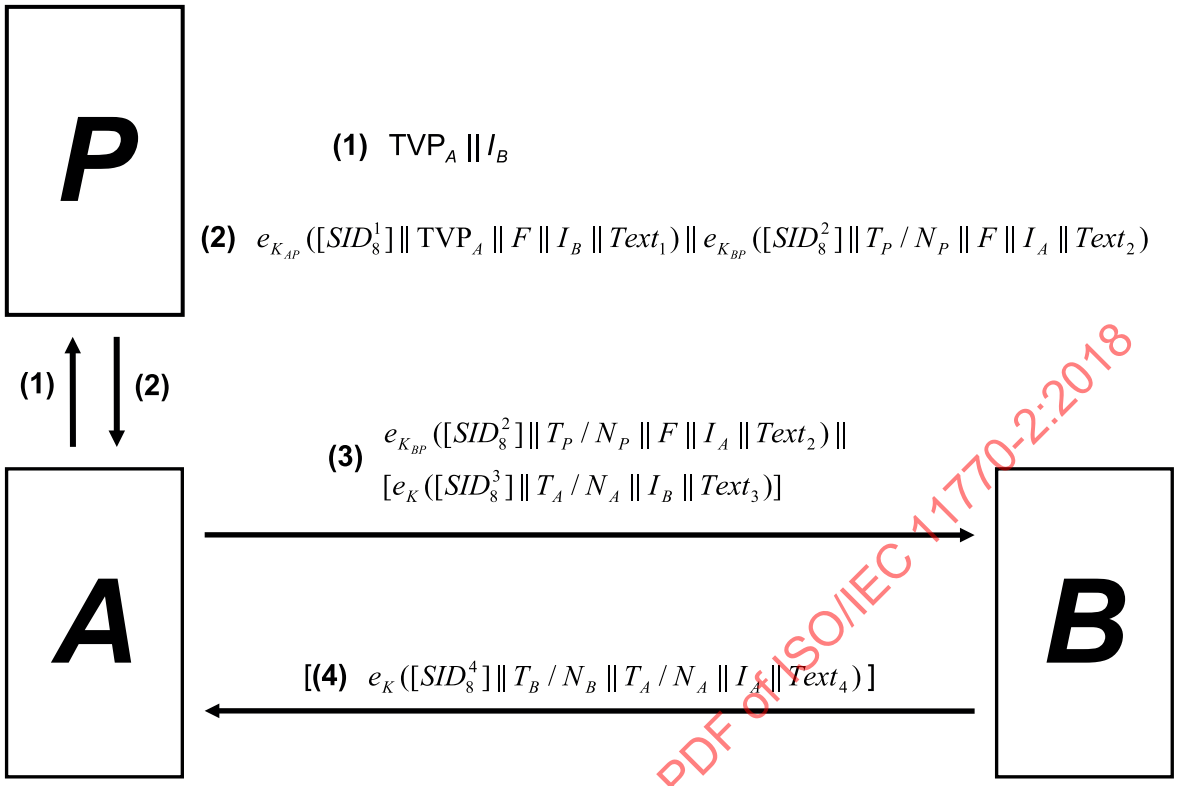

Figure 8 — Mechanism 8

The mechanism involves the following steps (see Figure 8).

1) *A* requests keying material from the KDC by sending message 1 to the KDC containing a time variant parameter, TVP_A (which can be a random number, a time stamp, or a sequence number), and the distinguishing identifier, I_B, of the second entity *B*.

2) The KDC sends message 2 to *A* containing the keying material, *F*, (made up of a key, *K*, together with optional data). This message consists of two main parts:

   — $e_{K_{AP}}([SID_8^1] || TVP_A || F || I_B || Text_1)$;

   — $e_{K_{BP}}([SID_8^2] || T_P / N_P || F || I_A || Text_2)$.

   On receipt of message 2, *A* deciphers the first part, checks that the time variant parameter, TVP_A, sent to the KDC in step 1, was used in constructing the first part of message 2, checks the correctness of the distinguishing identifier I_B, and obtains the key *K*.

3) *A* forwards the second part of message 2 to *B* in message 3. Message 3 optionally also contains a second part, a data field:

   $e_K([SID_8^3] || T_A / N_A || I_B || Text_3)$

   which enables *B* to check the integrity of the key, *K*, retrieved from, *F*.

   NOTE 1 The timestamp, T_A, or sequence number, N_A, included in message 3 is unrelated to the time variant parameter, TVP_A, included in message 1.

   i) On receipt of message 3, *B* deciphers the first part, checks the correctness of the time stamp or sequence number, checks the correctness of the distinguishing identifier, I_A, and obtains the key, *K*.

ii) *B* deciphers the second part of message 3, if present, and checks the correctness of the time stamp or sequence number and of the distinguishing identifier, I_B.

The fourth step given below is optional; it can be omitted if either no entity authentication or only unilateral authentication is required.

4) *B* returns $e_K([SID_8^4]||T_B / N_B || T_A / N_A || I_A || Text_4)$ to *A* in message 4, thereby acknowledging that it shares the key, *K*. On receipt of message 4, *A* deciphers it and checks the correctness of the time stamps or sequence numbers and of the distinguishing identifier, I_A.

NOTE 2 The encryption algorithm, *e*, used in the optional key confirmation process (i.e., part 2 of message 3 and message 4) can differ from the encryption algorithm (also denoted by *e*) used for key distribution.

NOTE 3 Mutual authentication, key confirmation and explicit key authentication from *B* to *A*, and conformance with the four-pass entity authentication mechanism specified in ISO/IEC 9798-2 are only achieved if the optional message (parts) in steps 3 and 4 are included.

NOTE 4 If required, authentication of the requesting entity by the KDC can be provided by the inclusion of a MAC, computed over TVP_A using a secret key shared by *A* and the KDC, in a plaintext text field of message 1. This can only work correctly if TVP_A is of a form (e.g. a time stamp) whose correctness can be verified by the KDC.

7.4 Key establishment mechanism 9

Key establishment mechanism 9 is derived from the five-pass mutual authentication mechanism specified in ISO/IEC 9798-2.

Key establishment mechanism 9 optionally provides mutual authentication between *A* and *B*. It also provides key confirmation and explicit key authentication from *A* to *B*, and, optionally, from *B* to *A*.

Uniqueness/timeliness is controlled by random numbers. The mechanism requires that *A*, *B* and the KDC are able to generate random numbers.

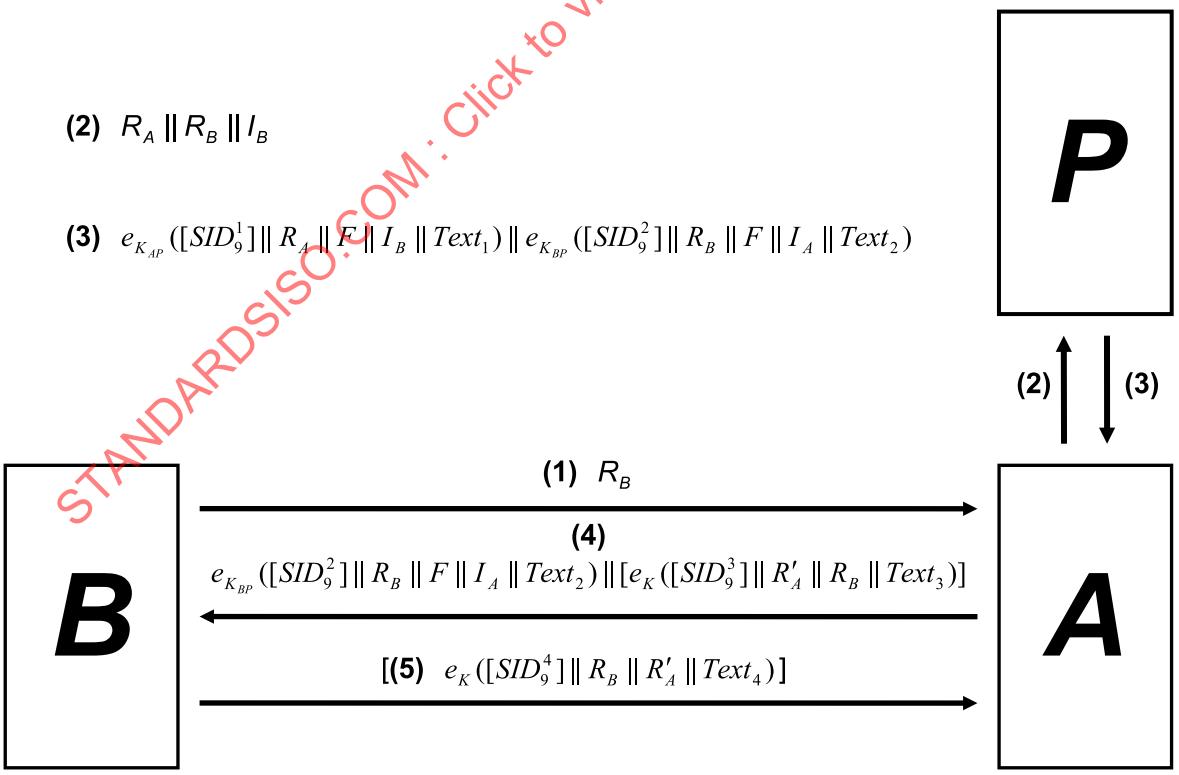

Figure 9 — Mechanism 9

The mechanism involves the following steps (see Figure 9).

1) B initiates the mechanism by sending a random number, R_B, to A in message 1.

2) A requests keying material from the KDC by sending message 2 to the KDC containing a random number, R_A, the random number, R_B, sent in message 1, and the distinguishing identifier, I_B, of B.

3) The KDC sends message 3 to A containing the keying material, F (made up of a key, K, together with optional data). This message consists of two main parts:

   — $e_{K_{AP}}([SID_9^1]||R_A||F||I_B||Text_1)$;

   — $e_{K_{BP}}([SID_9^2]||R_B||F||I_A||Text_2)$.

   On receipt of message 3, A deciphers the first part, checks that the random number, R_A, sent to the KDC in step 2, was used in constructing the first part of message 3, checks the correctness of the distinguishing identifier, I_B, and retrieves the key, K.

4) A forwards the second part of message 3 to B in message 4. Message 4 optionally also contains a second part, a data field:

   $e_K([SID_9^3]||R'_A||R_B||Text_3)$

   which incorporates random numbers, R_B and R'_A, and enables B to check the integrity of the key, K, retrieved from F.

   i) On receipt of message 4, B deciphers the first part, checks that the random number, R_B, sent to A in step 1 was used in constructing the first part of message 4, checks the correctness of the distinguishing identifier, I_A, and obtains the key, K.

   ii) B deciphers the second part of message 4, if present, and checks that the random number, R_B, sent to A in step 1, was used in constructing the second part of message 4.

   The fifth step given below is optional; it can be omitted if either no entity authentication or only unilateral authentication is required. This fifth step can only be used if the second part of message 4 is also sent.

5) B returns $e_K([SID_9^4]||R_B||R'_A||Text_4)$ to A in message 5, thereby acknowledging that it shares the key, K. On receipt of message 5, A deciphers it and checks that the random number, R'_A, sent to B in step 4, was used in constructing message 5.

NOTE 1 The encryption algorithm e used in the optional key confirmation process (i.e., part 2 of message 4 and message 5) can differ from the encryption algorithm (also denoted by e) used for key distribution.

NOTE 2 Mutual authentication, key confirmation and explicit key authentication from B to A, and conformance with the five-pass entity authentication mechanism specified in ISO/IEC 9798-2 are only achieved if the optional message (parts) in steps 4 and 5 are included.

7.5 Key establishment mechanism 10

Key establishment mechanism 10 provides mutual authentication between A and the KDC and unilateral authentication of the KDC to B.

Uniqueness/timeliness is controlled by time stamps or sequence numbers. The mechanism requires that A, B, and the KDC are able to maintain mechanisms for generating and/or verifying the validity of time stamps, T, or sequence numbers, N.

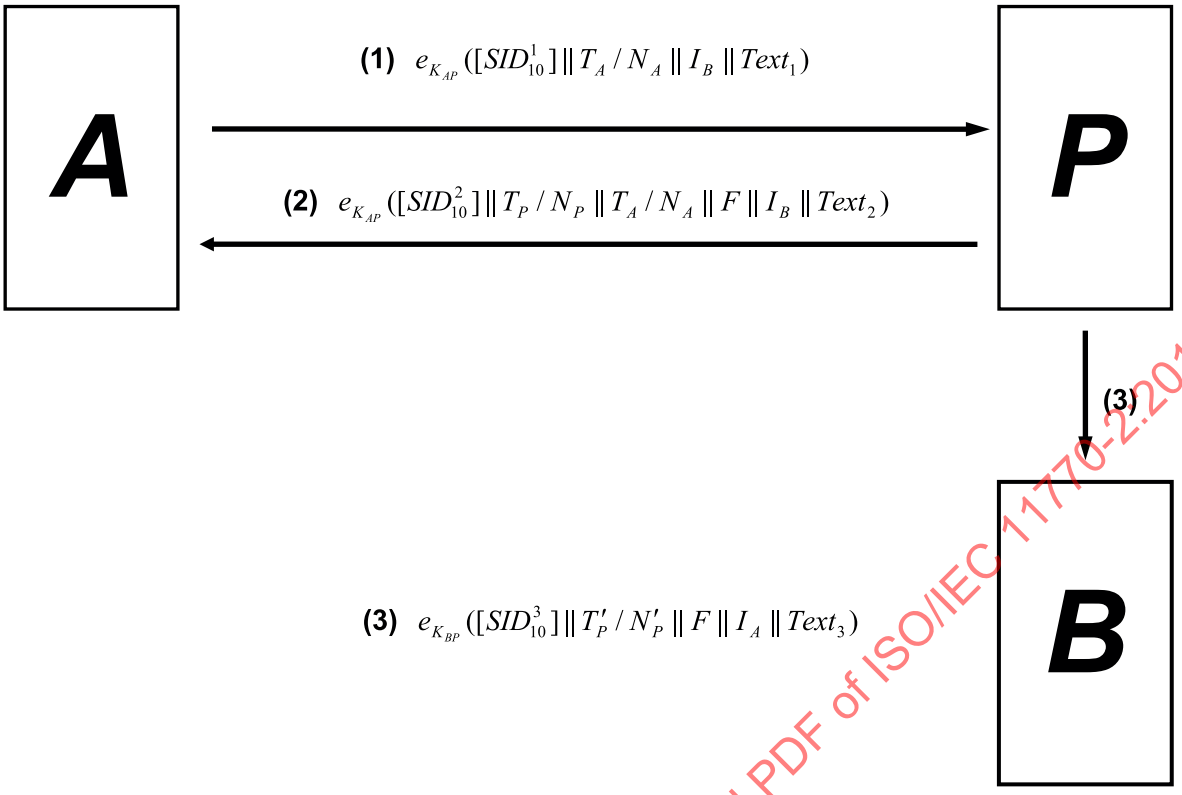

Figure 10 — Mechanism 10

The mechanism involves the following steps (see Figure 10).

1) *A* requests keying material from the KDC by sending message 1 to the KDC containing a time stamp T_A or sequence number, N_A, and the distinguishing identifier, I_B, of *B*. The data fields are encrypted using the key, K_{AP}. On receipt of message 1, the KDC deciphers it and checks the correctness of the time stamp or sequence number.

2) The KDC sends message 2 to *A* containing a time stamp, T_P, or sequence number, N_P, the distinguishing identifier, I_B, and the keying material, *F* (made up of a key, *K*, together with optional data). The data fields are encrypted using the key, K_{AP}. On receipt of message 2, *A* deciphers it, checks the correctness of the distinguishing identifier, I_B, checks the correctness of the time stamps or sequence numbers and obtains the key, *K*.

3) The KDC sends message 3 to *B* containing a time stamp, T'_P, or sequence number, N'_P, the distinguishing identifier, I_A, and the keying material, *F*. The data fields are encrypted using the key, K_{BP}. On receipt of message 3, *B* deciphers it, checks the correctness of the time stamp or sequence number, and obtains the key, *K*. The distinguishing identifier, I_A, indicates to *B* that the key was requested by *A*.

NOTE 1 The order in which steps 2 and 3 are executed is at the discretion of the implementer.

NOTE 2 Entity authentication between *A* and *B* is not achieved by this mechanism. If entity authentication between *A* and *B* is required, it can be achieved using the key, *K*, established by the mechanism with one of the mechanisms specified in ISO/IEC 9798-2 or ISO/IEC 9798-4.

NOTE 3 Entity authentication of the requesting entity by the KDC is provided.

8 Mechanisms using a Key Translation Centre

8.1 General

The purpose of a Key Translation Centre (KTC) is to enable a key to be transferred between a pair of entities that both share a key with the KTC.

In Clause 8, three key establishment mechanisms using a KTC are specified. In each of these mechanisms, one of the two entities (the originator) sends a key, *K*, to the KTC, encrypted using a key shared between the originator and the KTC. The KTC deciphers the key, *K*, and re-enciphers it with a key shared with the second entity (i.e. the ultimate recipient) – this process produces what is known as the translated key. The KTC then either:

a) sends the translated key back to the originator who then forwards it to the ultimate recipient; or

b) forwards the translated key to the ultimate recipient directly.

For the implementation of the mechanisms specified in Clause 8 it is required that:

a) entities *A* and *B* share secret keys K_{AP} and K_{BP}, respectively, with a trusted third party, *P*, that acts as a KTC;

b) the KTC shall have a means of communication with the entity requesting key translation (the originator);

c) the originator shall be able to generate or otherwise acquire a key, *K*;

d) the security requirements for the mechanism include the confidentiality of *K*, and the detection of modification, substitution or replay of keys and messages.

8.2 Key establishment mechanism 11

In key establishment mechanism 11, the key, *K*, is provided by entity *A*.

Key establishment mechanism 11 provides key confirmation and explicit key authentication from *A* to *B*.

The mechanism requires that *A*, *B* and the KTC are able to maintain mechanisms for generating and verifying the validity of time stamps, *T*, or sequence numbers, *N*.

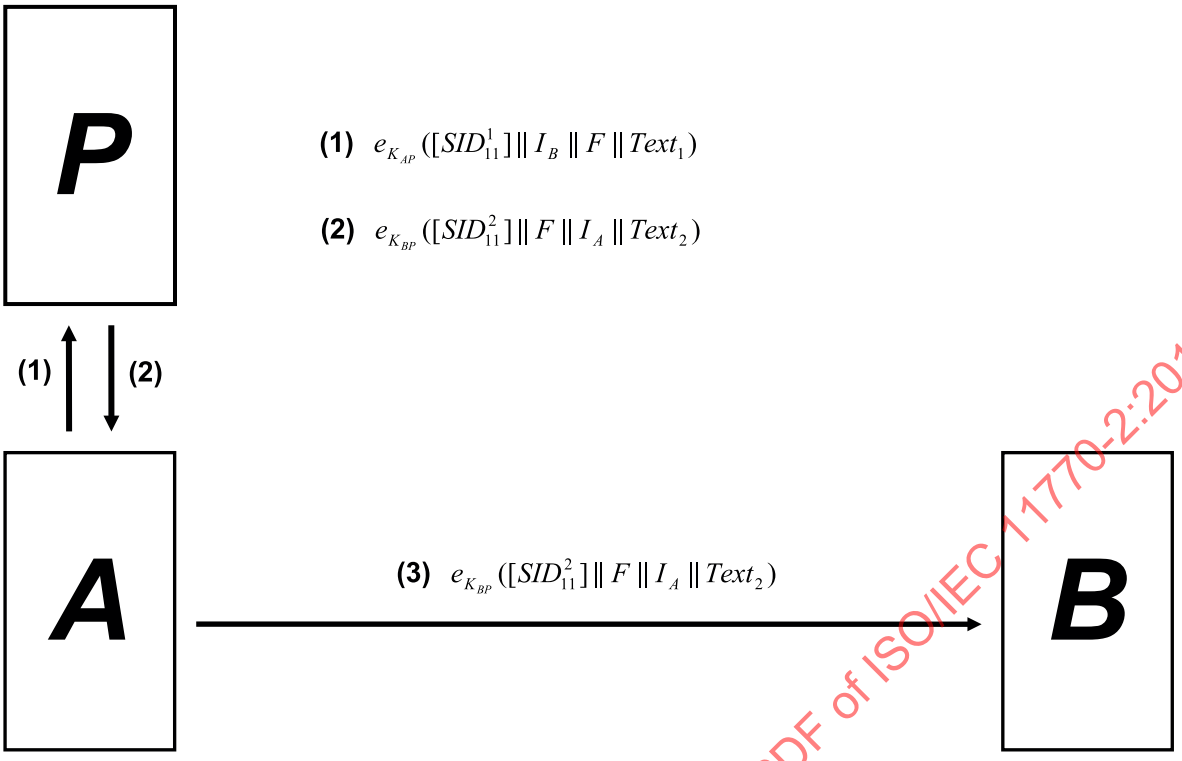

Figure 11 — Mechanism 11

The mechanism involves the following steps (see Figure 11).

1) *A* requests key translation by sending message 1 to the KTC, consisting of $e_{K_{AP}}([SID_{11}^{1}] || I_B || F || Text_1)$, encrypted using the key, K_{AP}, containing the distinguishing identifier, I_B, of the second entity *B*, and the keying material, *F* (made up of a key, *K*, together with optional data). On receipt of message 1, the KTC deciphers it to obtain, *F*, adds the distinguishing identifier, I_A, and re-enciphers both using the key, K_{BP}, to obtain $e_{K_{BP}}([SID_{11}^{2}] || F || I_A || Text_2)$.

2) The KTC returns the re-enciphered keying material to *A* in message 2.

3) *A* forwards $e_{K_{BP}}([SID_{11}^{2}] || F || I_A || Text_2)$ to *B* in message 3. On receipt of message 3, *B* deciphers the encrypted part, checks the correctness of the distinguishing identifier, I_A, and, thus, obtains the key, *K*.

8.3 Key establishment mechanism 12

Key establishment mechanism 12 is derived from, but is not fully compatible with, the four-pass authentication mechanism of ISO/IEC 9798-2:2018, 6.1. In this mechanism, the key, *K*, is supplied by entity *A*. Uniqueness/timeliness is controlled by time stamps or sequence numbers.

Key establishment mechanism 12 optionally provides mutual authentication between *A* and *B*. It also provides key confirmation and explicit key authentication from *A* to *B*, and, optionally, from *B* to *A*.

The mechanism requires that *A*, *B* and the KTC are able to maintain mechanisms for generating and verifying the validity of time stamps, *T*, or sequence numbers, *N*.

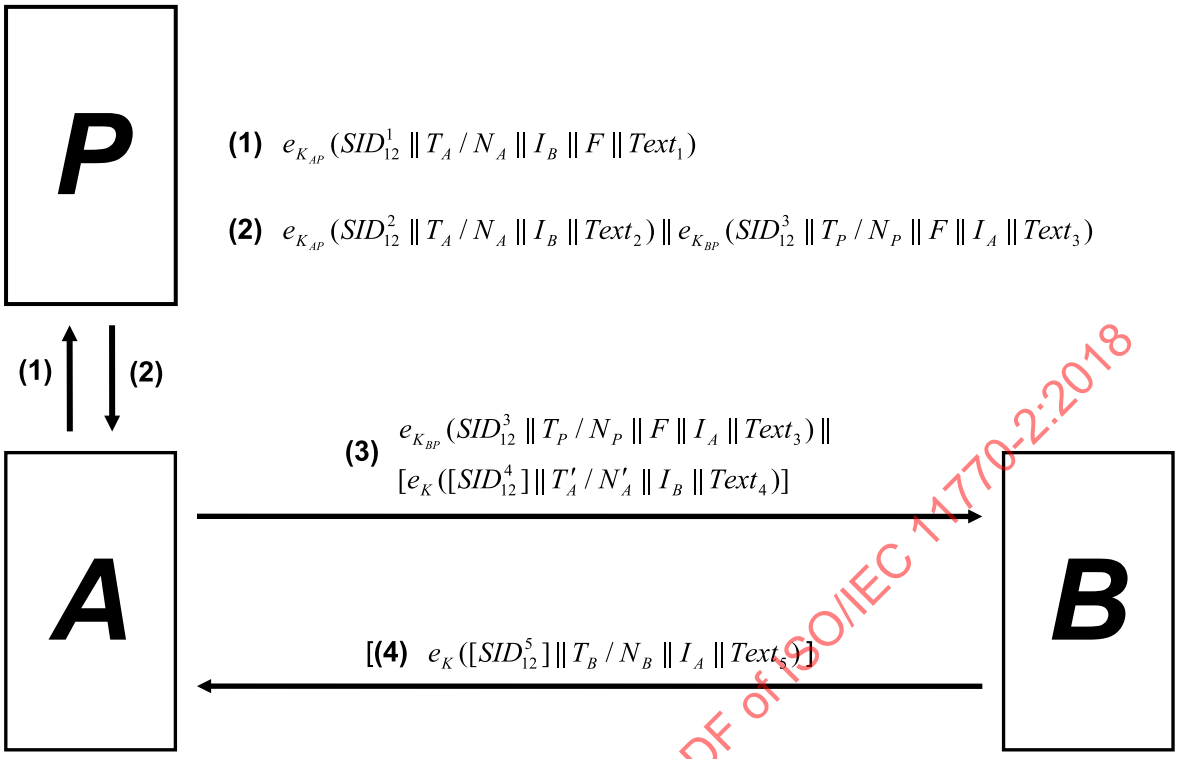

Figure 12 — Mechanism 12

The mechanism involves the following steps (see Figure 12).

1) *A* requests key translation by sending message 1 to the KTC, containing the identifier, SID_{12}^{1}, a time stamp, T_A, or sequence number, N_A, the distinguishing identifier, I_B, of *B*, and the keying material, *F* (made up of a key *K*, together with optional data). The data fields are encrypted using the key, K_{AP}.

   On receipt of message 1, the KTC deciphers it, checks the presence of the identifier, SID_{12}^{1}, checks the time stamp, T_A, or sequence number, N_A, and recovers the encrypted keying material, *F*.

2) The KTC sends message 2 to *A*, that consists of two main parts:

   — $e_{K_{AP}}(SID_{12}^{2} || T_A / N_A || I_B || Text_2)$;

   — $e_{K_{BP}}(SID_{12}^{3} || T_P / N_P || F || I_A || Text_3)$.

   On receipt of message 2, *A* deciphers the first part, and checks the presence of the identifier, SID_{12}^{2}, and the distinguishing identifier, I_B, and that the time stamp, T_A, or the sequence number, N_A, sent to the KTC in step 1, was used in constructing the first part of message 2.

3) *A* forwards the second part of message 2 to *B* in message 3. Message 3 optionally also contains a second part, a data field:

   $e_K([SID_{12}^{4}] || T'_A / N'_A || I_B || Text_4)$

   which incorporates an (optional) identifier, SID_{12}^{4}, a time stamp, T'_A, or sequence number, N'_A, and enables *B* to check the integrity of the key, *K*, retrieved from *F*.

   i) On receipt of message 3, *B* deciphers the first part, checks the presence of the identifier, SID_{12}^{3}, and the distinguishing identifier, I_A, and obtains the key, *K*.

ii) B deciphers the second part of message 3, if present, and checks the time stamp, T'_A, or sequence number, N'_A, and the presence of the distinguishing identifier, I_B.

The fourth step given below is optional; it can be omitted if either no entity authentication or only unilateral authentication is required. This fourth step can only be used if the second part of message 3 is also sent.

4) B returns $e_K([SID_{12}^5]||T_B / N_B || I_A || Text_5)$ to A in message 4, thereby acknowledging that it shares the key K. On receipt of message 4, A deciphers it and checks the time stamp T_B or sequence number N_B and the presence of the distinguishing identifier I_A.

NOTE 1 The encryption algorithm e used in the optional key confirmation process (i.e., part 2 of message 3 and message 4) may differ from the encryption algorithm (also denoted by e) used for key distribution.

NOTE 2 In order to achieve mutual authentication, key confirmation, and explicit key authentication from B to A, the optional message (parts) in steps 3 and 4 need to be included.

NOTE 3 Unlike other mechanisms in this document, the inclusion of identifiers: SID_{12}^1, SID_{12}^2 and SID_{12}^3 is made mandatory to maximise compatibility with the previous edition of this document.

8.4 Key establishment mechanism 13

Key establishment mechanism 13 is derived from, but is not fully compatible with, the five-pass mutual authentication mechanism specified in ISO/IEC 9798-2:2018, 6.2.

Key establishment mechanism 13 optionally provides mutual authentication between A and B. It also provides key confirmation and explicit key authentication from A to B, and, optionally, from B to A.

Uniqueness/timeliness is controlled by random numbers. The mechanism requires that A, B and the KTC are able to generate random numbers.

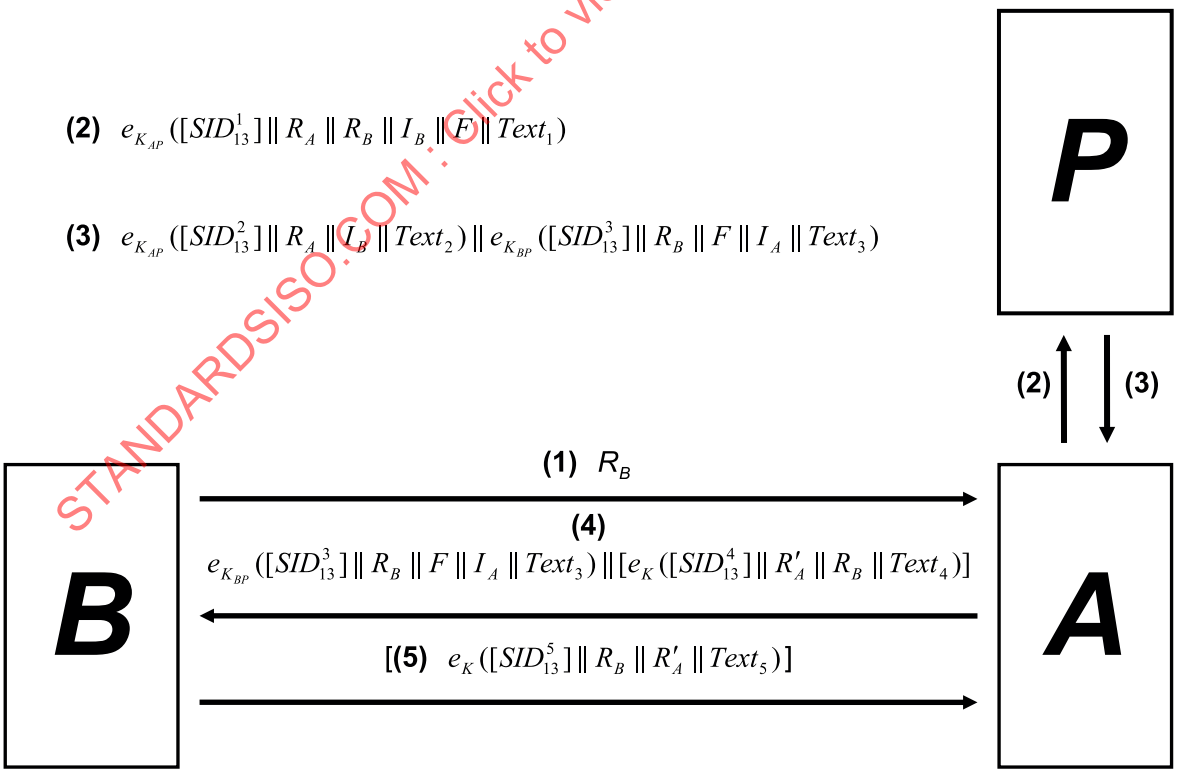

Figure 13 — Mechanism 13

The mechanism involves the following steps (see Figure 13).

1) B initiates the mechanism by sending a random number R_B to A in message 1.

2) A requests key translation by sending message 2 to the KTC containing a random number, R_A, the random number, R_B sent in message 1, the distinguishing identifier, I_B, of B, and the keying material, F (made up of a key K, together with optional data). The data fields are encrypted using the key, K_{AP}.

   On receipt of message 2, the KTC deciphers the encrypted keying material, F, and re-enciphers it together with additional (optional) data fields.

3) The KTC sends message 3 to A, that consists of two main parts:

   — $e_{K_{AP}}([SID_{13}^{2}]||R_A||I_B||Text_2)$;

   — $e_{K_{BP}}([SID_{13}^{3}]||R_B||F||I_A||Text_3)$.

   On receipt of message 3, A deciphers the first part, and checks the distinguishing identifier, I_B, and that the random number, R_A, sent to the KTC in step 2, was used in constructing the first part of message 3.

4) A forwards the second part of message 3 to B in message 4. Message 4 optionally also contains a second part, a data field:

   $e_K([SID_{13}^{4}]||R'_A||R_B||Text_4)$

   which incorporates random numbers, R_B and R'_A, and enables B to check the integrity of the key, K, retrieved from F.

   i) On receipt of message 4, B deciphers the first part, checks the distinguishing identifier, I_A, and that the random number, R_B, sent to A in step 1 was used in constructing the first part of message 4, checks the correctness of the distinguishing identifier, I_A, and obtains the key, K.

   ii) B deciphers the second part of message 4, if present, and checks that the random number, R_B, sent to A in step 1, was used in constructing the second part of message 4.

   The fifth step given below is optional; it can be omitted if either no entity authentication or only unilateral authentication is required. This fifth step can only be used if the second part of message 4 is also sent.

5) B returns $e_K([SID_{13}^{5}]||R_B||R'_A||Text_5)$ to A in message 5, thereby acknowledging that it shares the key, K. On receipt of message 5, A deciphers it and checks that the random numbers, R'_A and R_B, sent to B in step 4, were used in constructing message 5.

NOTE 1 The encryption algorithm, e, used in the optional key confirmation process (i.e., part 2 of message 4 and message 5) can differ from the encryption algorithm (also denoted by e) used for key distribution.

NOTE 2 Mutual authentication, key confirmation, and explicit key authentication from B to A are only achieved if the optional message (parts) in steps 4 and 5 are included.

Annex A
(normative)

Object identifiers

A.1 Formal definition

```
KeyManagementSymmetricTechniques {
   iso(1) standard(0) key-management(11770) part(2) asn1-module(0)
      key-management-mechanisms(0) }
DEFINITIONS EXPLICIT TAGS ::= BEGIN
-- IMPORTS None; --

OID ::= OBJECT IDENTIFIER

KeyEstablishmentMechanism ALGORITHM ::= {
   ke-mechanism1      |
   ke-mechanism2      |
   ke-mechanism3      |
   ke-mechanism4      |
   ke-mechanism5      |
   ke-mechanism6      |
   ke-mechanism7      |
   ke-mechanism8      |
   ke-mechanism9      |
   ke-mechanism10      |
   ke-mechanism11      |
   ke-mechanism12      |
   ke-mechanism13
}

-- Synonyms --

is11770-2 OID ::= { iso(1) standard(0) key-management(11770) part2(2) }
mechanism OID ::= { is11770-2 mechanisms(1) }

-- Point-to-point key establishment --

ke-mechanism1 OID ::= { mechanism 1 }
ke-mechanism2 OID ::= { mechanism 2 }
ke-mechanism3 OID ::= { mechanism 3 }
ke-mechanism4 OID ::= { mechanism 4 }
ke-mechanism5 OID ::= { mechanism 5 }
ke-mechanism6 OID ::= { mechanism 6 }

-- Mechanisms using a key distribution centre --

ke-mechanism7 OID ::= { mechanism 7 }
ke-mechanism8 OID ::= { mechanism 8 }
ke-mechanism9 OID ::= { mechanism 9 }
ke-mechanism10 OID ::= { mechanism 10 }

-- Mechanisms using a key translation centre --

ke-mechanism11 OID ::= { mechanism 11 }
ke-mechanism12 OID ::= { mechanism 12 }
ke-mechanism13 OID ::= { mechanism 13 }

END  -- KeyManagementSymmetricTechniques --
```