INTERNATIONAL STANDARD

ISO/IEC 9798-2

First edition
1994-12-15

Information technology — Security techniques — Entity authentication —

Part 2:
Mechanisms using symmetric encipherment algorithms

Technologies de l'information — Techniques de sécurité — Authentification d'entité —

Partie 2: Mécanismes utilisant des algorithmes de chiffrement symétriques

Reference number
ISO/IEC 9798-2:1994(E)

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

International Standard ISO/IEC 9798-2 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 27, *IT Security techniques*.

ISO/IEC 9798 consists of the following parts, under the general title *Information technology — Security techniques — Entity authentication mechanisms*:

— *Part 1: General model*

— *Part 3: Entity authentication using a public key algorithm*

ISO/IEC 9798 also consists of the following parts, under the general title *Information technology — Security techniques — Entity authentication*:

— *Part 2: Mechanisms using symmetric encipherment algorithms*

— *Part 4: Mechanisms using a cryptographic check function*

— *Part 5: Mechanisms using zero knowledge techniques*

NOTE — The introductory element of the titles of parts 1 and 3 will be aligned with the introductory element of the titles of parts 2, 4 and 5 at the next revision of parts 1 and 3 of ISO/IEC 9798.

Further parts may follow.

Annexes A, B and C of this part of ISO/IEC 9798 are for information only.

ISO/IEC Copyright Office • Case postale 56 • CH-1211 Genève 20 • Switzerland

Printed in Switzerland

Information technology — Security techniques — Entity authentication — Part 2: Mechanisms using symmetric encipherment algorithms

1 Scope

This part of ISO/IEC 9798 specifies entity authentication mechanisms using symmetric encipherment algorithms. Four of them deal with authentication mechanisms between two entities where no trusted third party is involved; two of these four are concerned with the authentication of a single entity (unilateral authentication), while the other two are mechanisms for mutual authentication of two entities. The remaining mechanisms require a trusted third party for the establishment of a common secret key, and realize mutual or unilateral entity authentication.

The mechanisms specified in this part of ISO/IEC 9798 use time variant parameters such as time stamps, sequence numbers, or random numbers, to prevent valid authentication information from being accepted at a later time.

If no trusted third party is involved and a time stamp or sequence number is used, one pass is needed for unilateral authentication, while two passes are needed to achieve mutual authentication. If no trusted third party is involved and a challenge and response method employing random numbers is used, two passes are needed for unilateral authentication, while three passes are required to achieve mutual authentication. If a trusted third party is involved, any additional communication between an entity and the trusted third party requires two extra passes in the communication exchange.

2 Normative reference

The following standard contains provisions which, through reference in this text, constitute provisions of this part of ISO/IEC 9798. At the time of publication, the edition indicated was valid. All standards are subject to revision, and parties to agreements based on this part of ISO/IEC 9798 are encouraged to investigate the possibility of applying the most recent edition of the standard indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO/IEC 9798-1: 1991, *Information technology — Security techniques — Entity authentication mechanisms — Part 1: General model.*

3 Definitions and notation

For the purposes of this part of ISO/IEC 9798 the definitions and notation described in ISO/IEC 9798-1 apply.

4 Requirements

In the authentication mechanisms specified in this part of ISO/IEC 9798 an entity to be authenticated corroborates its identity by demonstrating its knowledge of a secret authentication key. This is achieved by the entity using its secret key to encipher specific data. The enciphered data can be deciphered by anyone sharing the entity's secret authentication key.

The authentication mechanisms have the following requirements. If any one of these is not met then the authentication process may be compromised or it cannot be implemented.

a) A claimant authenticating itself to a verifier shares a common secret authentication key with that verifier, in which case the mechanisms of clause 5 apply, or each entity shares a secret authentication key with a common trusted third party, in which case the mechanisms of clause 6 apply. Such keys shall be known to the involved parties prior to the commencement of any particular run of an authentication mechanism. The method by which this is achieved is beyond the scope of this part of ISO/IEC 9798.

b) If a trusted third party is involved it is trusted by both the claimant and the verifier.

c) The secret authentication key shared by a claimant and a verifier, or by an entity and a trusted third party, is known only to those two parties and, possibly, to other parties they both trust.

NOTE 1 — The encipherment algorithm and the key life-time should be chosen so that it is computationally infeasible for a key to be deduced during its life-time. In addition, the key life time should be chosen to prevent known plaintext or chosen plaintext attacks.

d) Either assumption d1) or assumption d2) is met.

d1) The encipherment algorithm and the mode of operation used in the authentication mechanisms shall provide the recipient with the means to detect forged or manipulated data. This requires that sufficient redundancy is present in the data, and that any modification in the plaintext results in an unpredictable modification of a large number of ciphertext bits.

A possible way to provide sufficient redundancy is to append a hash-code to the data before encipherment.

NOTE 2 — Hash-functions are standardized in ISO/IEC 10118.

If a block cipher algorithm is used for encipherment and the block size is smaller than the length of the data to be enciphered, then the replacement of any block shall be detectable.

d2) The integrity of the enciphered data shall be ensured by an independent data integrity mechanism.

NOTE 3 — A data integrity mechanism is standardized in ISO/IEC 9797.

5 Mechanisms not involving a trusted third party

In these authentication mechanisms the entities A and B shall share a common secret authentication key K_{AB} prior to the commencement of any particular run of the authentication mechanisms.

The mechanisms require the use of time variant parameters such as time stamps, sequence numbers or random numbers. The properties of these parameters, in particular that it is most unlikely for them to repeat within the life-time of an authentication key, are important for the security of these mechanisms. For additional information see annex B.

All text fields specified in the following mechanisms are available for use in applications outside the scope of this part of ISO/IEC 9798 (they may be empty). Their relationship and contents depend upon the specific application. See annex A for information on the use of text fields.

5.1 Unilateral authentication

Unilateral authentication means that only one of the two entities is authenticated by use of the mechanism.

5.1.1 One pass authentication

In this authentication mechanism the claimant A initiates the process and is authenticated by the verifier B. Uniqueness / timeliness is controlled by generating and checking a time stamp or a sequence number (see annex B).

The authentication mechanism is illustrated in figure 1.

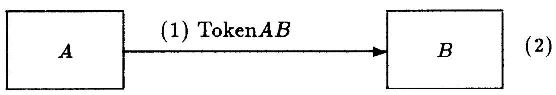

Figure 1

The form of the token (TokenAB), sent by the claimant A to the verifier B is:

$$\text{Token}AB = \text{Text2}\|eK_{AB}\left(\genfrac{}{}{0pt}{}{T_A}{N_A}\|B\|\text{Text1}\right),$$

where the claimant A uses either a sequence number N_A or a time stamp T_A as the time variant parameter. The choice depends on the technical capabilities of the claimant and the verifier as well as on the environment.

The inclusion of the distinguishing identifier B in TokenAB is optional.

NOTE — Distinguishing identifier B is included in TokenAB to prevent the re-use of TokenAB on entity A by an adversary masquerading as entity B. Its inclusion is made optional so that, in environments where such attacks cannot occur, it may be omitted.
The distinguishing identifier B may also be omitted if entities A and B share a secret key K'_{AB} used only for the authentication of A by B. The token then becomes:

$$\text{Token}AB = \text{Text2}\|eK'_{AB}\left(\genfrac{}{}{0pt}{}{T_A}{N_A}\|\text{Text1}\right).$$

(1) A sends TokenAB to B.

(2) On receipt of the message containing TokenAB, B verifies TokenAB by deciphering the enciphered part and checking the correctness of the distinguishing identifier B, if present, as well as the time stamp or the sequence number.

5.1.2 Two pass authentication

In this authentication mechanism the claimant A is authenticated by the verifier B who initiates the process. Uniqueness / timeliness is controlled by generating and checking a random number R_B (see annex B).

The authentication mechanism is illustrated in figure 2.

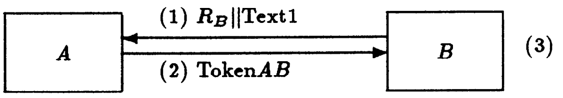

Figure 2

The form of the token (TokenAB), sent by the claimant A to the verifier B is:

$$\text{Token}AB = \text{Text3}||eK_{AB}(R_B||B||\text{Text2}).$$

The inclusion of the distinguishing identifier B in TokenAB is optional.

> NOTE — Distinguishing identifier B is included in TokenAB to prevent a so-called reflection attack. Such an attack is characterized by the fact that an intruder "reflects" the challenge R_B to B pretending to be A. The inclusion of the distinguishing identifier B is made optional so that, in environments where such attacks cannot occur, it may be omitted.
>
> The distinguishing identifier B may also be omitted if entities A and B share a secret key K'_{AB} used only for the authentication of A by B. The token then becomes:
>
> $$\text{Token}AB = \text{Text3}||eK'_{AB}(R_B||\text{Text2}).$$

(1) B sends a random number R_B and, optionally, a text field Text1 to A.

(2) A sends TokenAB to B.

(3) On receipt of the message containing TokenAB, B verifies TokenAB by deciphering the enciphered part and checking the correctness of the distinguishing identifier B, if present, and that the random number R_B, sent to A in step (1), agrees with the random number contained in TokenAB.

5.2 Mutual authentication

Mutual authentication means that the two communicating entities are authenticated to each other by use of the mechanism.

The two mechanisms described in 5.1.1 and 5.1.2 are adapted in 5.2.1 and 5.2.2, respectively, to achieve mutual authentication. In both cases this requires one more pass resulting in two more steps.

> NOTE — A third mechanism for mutual authentication can be constructed from two instances of the mechanism specified in 5.1.2, one started by entity A and the other by entity B.

5.2.1 Two pass authentication

In this authentication mechanism uniqueness / timeliness is controlled by generating and checking time stamps or sequence numbers (see annex B).

The authentication mechanism is illustrated in figure 3.

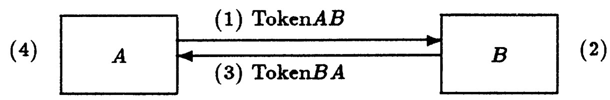

Figure 3

The form of the token (TokenAB), sent by A to B, is identical to that specified in 5.1.1.

$$\text{Token}AB = \text{Text2}||eK_{AB}\left(\begin{smallmatrix}T_A\\N_A\end{smallmatrix}||B||\text{Text1}\right).$$

The form of the token (TokenBA), sent by B to A, is:

$$\text{Token}BA = \text{Text4}||eK_{AB}\left(\begin{smallmatrix}T_B\\N_B\end{smallmatrix}||A||\text{Text3}\right).$$

The inclusion of the distinguishing identifier B in TokenAB and the inclusion of the distinguishing identifier A in TokenBA are (independently) optional.

> NOTE 1 — Distinguishing identifier B is included in TokenAB to prevent the re-use of TokenAB on entity A by an adversary masquerading as entity B. For similar reasons the distinguishing identifier A is present in TokenBA. Their inclusion is made optional so that, in environments where such attacks cannot occur, one or both may be omitted.
>
> The distinguishing identifiers A and B may also be omitted if entities A and B share two secret keys K'_{AB} and K'_{BA}, used respectively for the authentication of A by B and B by A. The tokens then become:
>
> $$\text{Token}AB = \text{Text2}||eK'_{AB}\left(\begin{smallmatrix}T_A\\N_A\end{smallmatrix}||\text{Text1}\right),$$
>
> $$\text{Token}BA = \text{Text4}||eK'_{BA}\left(\begin{smallmatrix}T_B\\N_B\end{smallmatrix}||\text{Text3}\right).$$

The choice of using either time stamps or sequence numbers in this mechanism depends on the capabilities of the claimant and the verifier as well as on the environment.

Steps (1) and (2) are identical to those specified in 5.1.1, one pass authentication.

(3) B sends TokenBA to A.

(4) The message in step (3) is handled in a manner analogous to step (2) of 5.1.1.

NOTE 2 — The two messages of this mechanism are not bound together in any way, other than implicitly by timeliness; the mechanism involves independent use of mechanism 5.1.1 twice. If it is desired to bind these messages further, appropriate use could be made of text fields (see annex A).

5.2.2 Three pass authentication

In this authentication mechanism uniqueness / timeliness is controlled by generating and checking random numbers (see annex B).

The authentication mechanism is illustrated in figure 4.

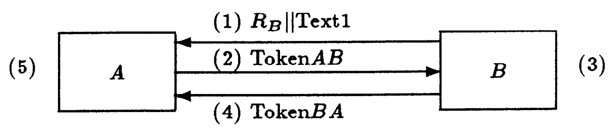

Figure 4

The tokens are of the following form:

TokenAB = Text3||eK_{AB} (R_A||R_B||B||Text2),
TokenBA = Text5||eK_{AB} (R_B||R_A||Text4).

NOTE 1 — The inclusion of R_B in TokenBA prevents the derivation of TokenBA from TokenAB.

The inclusion of the distinguishing identifier B in TokenAB is optional.

NOTE 2 — Distinguishing identifier B is included in TokenAB to prevent a so-called reflection attack. Such an attack is characterized by the fact that an intruder "reflects" the challenge R_B to B pretending to be A. The inclusion of the distinguishing identifier B is made optional so that, in environments where such attacks cannot occur, it may be omitted.
The distinguishing identifier B may also be omitted if entities A and B share two secret keys K'_{AB} and K'_{BA}, used respectively for the authentication of A by B and B by A. The tokens then become:

TokenAB = Text3||eK'_{AB} (R_A||R_B||Text2),
TokenBA = Text5||eK'_{BA} (R_B||R_A||Text4).

(1) B sends a random number R_B and, optionally, a text field Text1 to A.

(2) A sends TokenAB to B.

(3) On receipt of the message containing TokenAB, B verifies TokenAB by deciphering the enciphered part and checking the correctness of the distinguishing identifier B, if present, and that the random number R_B, sent to A in step (1), agrees with the random number contained in TokenAB.

(4) B sends TokenBA to A.

(5) On receipt of the message containing TokenBA, A verifies TokenBA by deciphering the enciphered part and checking that the random number R_B, received from B in step (1) agrees with the random number contained in TokenBA and that the random number R_A, sent to B in step (2), agrees with the random number contained in TokenBA.

6 Mechanisms involving a trusted third party

These authentication mechanisms do not make use of a secret key shared by the two entities prior to the authentication process. They do, however, make use of a trusted third party (with distinguishing identifier TP) with which the entities A and B each share a secret key, K_{AT} and K_{BT} respectively. In each mechanism one of the entities requests a key K_{AB} from the trusted third party. This is followed by an adaptation of the mechanisms described in 5.2.1 and 5.2.2, respectively.

As described below certain passes may be omitted from each mechanism if unilateral authentication is required.

NOTE — The mechanisms do not provide any guarantee to the trusted third party regarding the identities of entities A and B. In addition, if the authentication fails, there is no information on which exchange has been modified or created by an intruder.

The mechanisms require the use of time variant parameters such as time stamps, sequence numbers or random numbers. The properties of these parameters, in particular that it is most unlikely for them to repeat within the life-time of an authentication key are important for the security of these mechanisms. For additional information see annex B.

All text fields specified in the following mechanisms are available for use in applications outside the scope of this part of ISO/IEC 9798 (they may be empty). Their relationship and contents depend upon the specific application. See annex A for information on the use of text fields.

6.1 Four pass authentication

In this mutual authentication mechanism uniqueness / timeliness is controlled by using time variant parameters (see annex B).

The authentication mechanism is illustrated in figure 5.

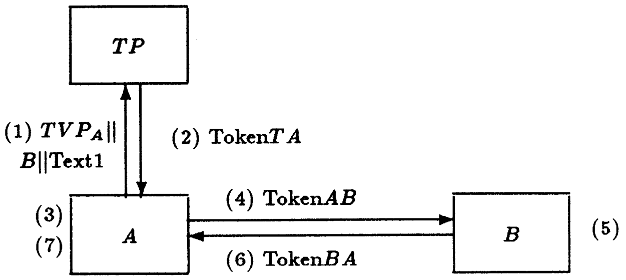

Figure 5

The form of the token (TokenTA), sent by TP to A, is:

$$\text{Token}TA = \text{Text4}||eK_{AT}(TVP_A||K_{AB}||B||\text{Text3})|| eK_{BT}\left(\tfrac{T_{TP}}{N_{TP}}||K_{AB}||A||\text{Text2}\right).$$

The form of the token (TokenAB), sent by A to B, is:

$$\text{Token}AB = \text{Text6}||eK_{BT}\left(\tfrac{T_{TP}}{N_{TP}}||K_{AB}||A||\text{Text2}\right)|| eK_{AB}\left(\tfrac{T_A}{N_A}||B||\text{Text5}\right).$$

The form of the token (TokenBA), sent by B to A, is:

$$\text{Token}BA = \text{Text8}||eK_{AB}\left(\tfrac{T_B}{N_B}||A||\text{Text7}\right).$$

The choice of using either time stamps or sequence numbers in this mechanism depends on the capabilities of the entities involved as well as on the environment.

The use of the time variant parameter TVP_A in steps (1) through (3) of figure 5, as specified below, is somewhat different from its normal use. It allows A to associate the response message (2) with the message request (1). The important property of the time variant parameter here is its non-repeatability, to limit the possible reuse of a previously used Token TA.

(1) A sends a time variant parameter TVP_A, the distinguishing identifier B and, optionally, a text field Text1 to the trusted third party TP.

(2) The trusted third party TP sends TokenTA to A.

(3) On receipt of the message containing TokenTA, A verifies TokenTA by deciphering the data enciphered under K_{AT} and checking the correctness of the distinguishing identifier B and that the time variant parameter, sent to TP in step (1), agrees with the time variant parameter contained in TokenTA. In addition, A retrieves the secret authentication key K_{AB}. A then extracts $eK_{BT}\left(\tfrac{T_{TP}}{N_{TP}}||K_{AB}||A||\text{Text2}\right)$ from TokenTA and uses it to construct TokenAB.

(4) A sends TokenAB to B.

(5) On receipt of the message containing TokenAB, B verifies TokenAB by deciphering the enciphered parts and checking the correctness of the distinguishing identifiers A and B as well as the time stamp(s) or the sequence number(s). In addition, B retrieves the secret authentication key K_{AB}.

(6) B sends TokenBA to A.

(7) On receipt of the message containing TokenBA, A verifies TokenBA by deciphering the enciphered part and checking the correctness of the distinguishing identifier A as well as the time stamp or the sequence number.

Steps (6) and (7) may be omitted if only unilateral authentication of A to B is required.

6.2 Five pass authentication

In this mutual authentication mechanism uniqueness / timeliness is controlled by using random numbers (see annex B).

The authentication mechanism is illustrated in figure 6.

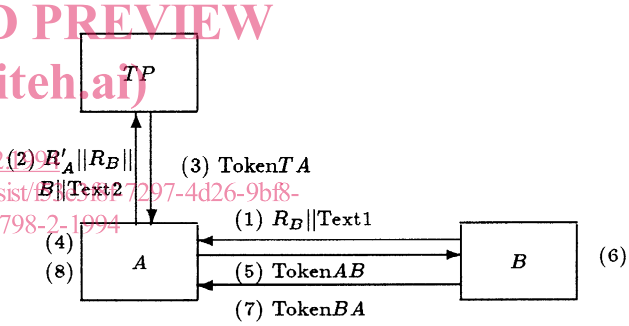

Figure 6

The form of the token (TokenTA), sent by TP to A, is:

$$\text{Token}TA = \text{Text5}||eK_{AT}(R'_A||K_{AB}||B||\text{Text4})|| eK_{BT}(R_B||K_{AB}||A||\text{Text3}).$$

The form of the token (TokenAB), sent by A to B, is:

$$\text{Token}AB = \text{Text7}||eK_{BT}(R_B||K_{AB}||A||\text{Text3})|| eK_{AB}(R_A||R_B||\text{Text6}).$$

The form of the token (TokenBA), sent by B to A, is:

$$\text{Token}BA = \text{Text9}||eK_{AB}(R_B||R_A||\text{Text8}).$$

(1) B sends a random number R_B and, optionally, a text field Text1 to A.

(2) A sends the random numbers R_B and R'_A, the distinguishing identifier B and, optionally, a text field Text2 to the trusted third party TP.

(3) The trusted third party TP sends TokenTA to A.

(4) On receipt of the message containing TokenTA, A verifies TokenTA by deciphering the data enciphered under K_{AT} and checking the correctness of the distinguishing identifier B and that the random number R'_A, sent to TP in step (2), agrees with the random number contained in TokenTA. In addition, A retrieves the secret authentication key K_{AB}. A then extracts $eK_{BT}(R_B||K_{AB}||A||\text{Text3})$ from TokenTB and uses it to construct TokenAB.

(5) A sends TokenAB to B.

(6) On receipt of the message containing TokenAB, B verifies TokenAB by deciphering the enciphered parts and checking the correctness of the distinguishing identifier A and that the random number R_B, sent to A in step (1), agrees with both copies contained in TokenAB. In addition, B retrieves the secret authentication key K_{AB}.

(7) B sends TokenBA to A.

(8) On receipt of the message containing TokenBA, A verifies TokenBA by deciphering the enciphered part and checking that the random number R_B, received from B in step (1), agrees with the random number contained in TokenBA and that the random number R_A, sent to B in step (5), agrees with the random number contained in TokenBA.

Steps (7) and (8) may be omitted if only unilateral authentication of A to B is required.

Annex A

(informative)

Use of text fields

The tokens specified in clauses 5 and 6 of this part of ISO/IEC 9798 contain text fields. The actual use of and the relationships between the various text fields in a given pass depend on the application. Some examples are given below.

If the tokens do not contain (sufficient) redundancy, the enciphered text fields may be used to provide additional redundancy.

Any information requiring confidentiality or data origin authentication should be placed in the enciphered part of the token.

Text fields may contain additional time variant parameters. For instance, a time stamp may be included in the text field(s) of Token*AB* in mechanism 5.1.1 if this is used with sequence numbers. This would allow the detection of forced delays by requiring the recipient of a message to verify that any time stamp contained in the message is within a prespecified time window (see also annex B).

If more than one valid key exists, then the cleartext text field may include the key identifier. If more than one trusted third party exists, then text fields could be used to include the distinguishing identifier of the trusted third party in question.

Text fields could also be used for the distribution of keys (see ISO/IEC 11770-2).

Should any of the mechanisms specified in this part of ISO/IEC 9798 be embedded in an application which allows either entity to initiate the authentication by using an additional message prior to the start of the mechanism, certain intruder attacks may become possible. Text fields may be used to state which entity requests the authentication in order to counteract such attacks, which are characterized by the fact that an intruder may reuse a token obtained illicitly.