INTERNATIONAL STANDARD

ISO/IEC 13888-2

Second edition
2010-12-15

Information technology — Security techniques — Non-repudiation —

Part 2: Mechanisms using symmetric techniques

Technologies de l'information — Techniques de sécurité — Non-répudiation —

Partie 2: Mécanismes utilisant des techniques symétriques

Reference number
ISO/IEC 13888-2:2010(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web 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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. 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.

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.

ISO/IEC 13888-2 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 27, *IT Security techniques*.

This second edition cancels and replaces the first edition (ISO/IEC 13888-2:1998), which has been technically revised.

ISO/IEC 13888 consists of the following parts, under the general title *Information technology — Security techniques — Non-repudiation*:

— *Part 1: General*

— *Part 2: Mechanisms using symmetric techniques*

— *Part 3: Mechanisms using asymmetric techniques*

Information technology — Security techniques — Non-repudiation —

Part 2: Mechanisms using symmetric techniques

1 Scope

The goal of the non-repudiation service is to generate, collect, maintain, make available and validate evidence concerning a claimed event or action in order to resolve disputes about the occurrence or non-occurrence of the event or action. This part of ISO/IEC 13888 provides descriptions of generic structures that can be used for non-repudiation services, and of some specific communication-related mechanisms which can be used to provide non-repudiation of origin (NRO) and non-repudiation of delivery (NRD). Other non-repudiation services can be built using the generic structures described in this part of ISO/IEC 13888 in order to meet the requirements defined by the security policy.

This part of ISO/IEC 13888 relies on the existence of a trusted third party (TTP) to prevent fraudulent repudiation or accusation. Usually, an online TTP is needed.

Non-repudiation can only be provided within the context of a clearly defined security policy for a particular application and its legal environment. Non-repudiation policies are defined in ISO/IEC 10181-4.

2 Normative references

The following referenced documents are indispensable for the application 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 9798-1:1997, *Information technology — Security techniques — Entity authentication — Part 1: General*

ISO/IEC 10118 (all parts), *Information technology — Security techniques — Hash-functions*

ISO/IEC 13888-1, *Information technology — Security techniques — Non-repudiation — Part 1: General*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 13888-1 and the following apply.

3.1
cryptographic check function
cryptographic transformation which takes as input a secret key and an arbitrary string, and which gives a cryptographic check value as output

[ISO/IEC 9798-1]

3.2
data integrity
property that data has not been altered or destroyed in an unauthorized manner

[ISO 7498-2]

3.3
evidence generator
entity that produces non-repudiation evidence

[ISO/IEC 10181-4]

3.4
hash-function
function which maps strings of bits to fixed-length strings of bits, satisfying the following two properties:

— it is computationally infeasible to find for a given output an input which maps to this output;

— it is computationally infeasible to find for a given input a second input which maps to the same output

[ISO/IEC 10118-1]

3.5
key
sequence of symbols that controls the operations of a cryptographic transformation (e.g. encipherment, decipherment, cryptographic check-function computation, signature calculation, or signature verification)

[ISO/IEC 11770-1]

3.6
MAC algorithm
algorithm for computing a function which maps strings of bits and a secret key to fixed-length strings of bits, satisfying the following properties:

— for any key and any input string the function can be computed efficiently;

— for any fixed key, and given no prior knowledge of the key, it is computationally infeasible to compute the function value on any new input string, even given knowledge of the set of input strings and corresponding function values, where the value of the *i*th input string may have been chosen after observing the value of the first *i*-1 function values

[ISO/IEC 9797-1]

3.7
message authentication code
MAC
string of bits which is the output of a MAC algorithm

[ISO/IEC 9797-1]

NOTE A MAC is sometimes called a cryptographic check value (see for example ISO/IEC 7498-2).

3.8
secret key
key used with symmetric cryptographic techniques and usable only by a set of specified entities

[ISO/IEC 11770-1]

3.9
security policy
set of criteria for the provision of security services

[ISO/IEC 7498-2]

3.10
time-stamp
time variant parameter which denotes a point in time with respect to a common time reference

[ISO/IEC 18014]

3.11
time-stamping authority
trusted third party trusted to provide a time-stamping service

[ISO/IEC 18014]

4 Symbols and abbreviated terms

DA	Delivery Authority
GNRT	Generic Non-Repudiation Token
MAC	cryptographic check function of Message Authentication Code
MAC	cryptographic check value of a Message Authentication Code
NRD	Non-Repudiation of Delivery
NRDT	Non-Repudiation of Delivery Token
NRO	Non-Repudiation of Origin
NROT	Non-Repudiation of Origin Token
PoI	non-repudiation policy (or policies) which apply to the evidence
PoI	distinguishing identifier of the non-repudiation policy (or policies) which apply to the evidence
PON	Positive Or Negative, the result of a verification process
SENV	function which generates a Secure ENVelope
SENV	Secure ENVelope
TSA	trusted Time Stamp Authority
TSA	distinguishing identifier of the trusted Time Stamp Authority
TST	Time-Stamping Token generated by the TSA
TTP	Trusted Third Party
TTP	distinguishing identifier of the Trusted Third Party

NOTE In each concatenation an appropriate data encoding must be used since afterwards it is necessary that each concatenated data can be picked up correctly.

5 Notation

5.1 Notation from ISO/IEC 13888-1

For the purposes of this part of ISO/IEC 13888, the following notation from ISO/IEC 13888-1 applies.

A	the distinguishing identifier of entity A
B	the distinguishing identifier of entity B
C	the distinguishing identifier of the evidence generator

D	the distinguishing identifier of the observer, if an independent observer is involved
E	the distinguishing identifiers of other entities involved with the action
f ,f_i	data item (flag) indicating the type of non-repudiation service in effect
Q	optional data that need to be protected
Imp	identity function or a hash-function
Imp(*y*)	imprint of the data string *y*, either (1) the hash-code of data string *y*, or (2) the data string *y*
m	a message for which evidence is generated
$SENV_X(y)$	the secure envelope computed on data *y* using the secret key of entity X
text	a data item forming part of a token that may contain additional information, e.g., key identifier and/or the message identifier
TG	date and time the evidence was generated
T_i	date and time the event or action took place

5.2 Notation unique for the purposes of this part of ISO/IEC 13888

For the purposes of this part of ISO/IEC 13888, the following unique notation applies.

a	a secret key known only to entity A and a TTP
b	a secret key known only to entity B and a TTP
da	secret key of Delivery Authority DA
$MAC_X(y)$	the cryptographic check value computed on the data *y* using the secret key of entity X
ttp	a secret key known only to the TTP to generate non-repudiation tokens
(*y*,*z*)	the ordered pair containing *y* and *z*, in that order
z_1	a data field consisting of data fields relevant for the provision of the NRO token
z_2	a data field consisting of data fields relevant for the provision of the NRD token

6 Requirements

The mechanisms specified in this part of ISO/IEC 13888 have the following requirements.

- Each of the two entities involved is able to communicate separately with the DA, TSA or TTP.
- The two entities wishing to use one of the mechanisms specified in this part of ISO/IEC 13888 must both trust the same third party.
- Prior to the use of these mechanisms, it is assumed that each entity shares a secret key with DA, TSA or TTP. Each of DA, TSA and TTP also holds a single key known only to itself.

  NOTE Key management, key generation and key establishment mechanisms are defined in ISO/IEC 11770.
- A common function Imp is shared by all entities in the non-repudiation service.
- A function MAC chosen for envelope (*SENV*) creation must be held by all participants in the non-repudiation service.
- The TTP generating the non-repudiation tokens shall be able to access the time and date.

The strength of the mechanisms specified in this part of ISO/IEC 13888 is dependent on the security level and strengths of the cryptographic mechanisms and parameters which are used.

7 Secure envelopes

Two entities that share a secret key (known only to them) may send messages to one another using a function for data integrity known as SENV. A *SENV* is formed by protecting the input data items using a secret key. A SENV can also be used by a TTP for generating and verifying evidence, using a shared secret key held only by the TTP.

The function SENV of creating a secure envelope *SENV* is through the use of symmetric integrity techniques. The secret key *x* of entity X is used to compute a cryptographic check value $MAC_X(y)$ which is appended to the data *y*:

$SENV_X(y) = (y, MAC_X(y))$.

8 Generation and verification of non-repudiation tokens

8.1 Creation of tokens by the TTP

In the non-repudiation mechanisms described in this Clause 8, the TTP acts as an evidence generation and verification authority. It is trusted to maintain the integrity of certain records and is directly involved in the resolution of any dispute.

The TTP issues "tokens" to be associated with a message *m*. A token consists of a secure envelope formed by the TTP using its secret key on data specific to a message. Because no other entity knows the secret key *ttp,* the TTP is the only one that can create or verify tokens. ISO/IEC 13888-1 defines the generic non-repudiation token (*GNRT*) as follows:

$GNRT = (text, SENV_X(y))$.

In this case it is:

$GNRT = (text, SENV_{TTP}(y))$.

The TTP should check also the data items present in the evidence request before the tokens are issued.

NOTE Message *m* can be in clear text or cipher text.

8.2 Data items used in the non-repudiation mechanisms

8.2.1 Data items used in secure envelopes

The following data field *z* will form the contents of secure envelopes

$SENV_X(z) = (z, MAC_X(z))$

to be exchanged during the non-repudiation mechanisms described in this part of ISO/IEC 13888:

$z = (Pol, f_i, A, B, C, D, E, TG, T_i, Q, Imp(m))$.

The data field *z* consists of the following data items:

Pol the distinguishing identifier of the non-repudiation policy (or policies) which applies to the evidence, and defines which data items exist in a data field,

f_i the type of non-repudiation service being provided,

A the distinguishing identifier of the originating entity,

B the distinguishing identifier of the entity interacting with the originating entity,

C the distinguishing identifier of the evidence generator,

D the distinguishing identifier of the evidence requester, if the evidence requester is different from the originating entity,

E the distinguishing identifiers of other entities involved with the action (appears depending on service),

TG the date and time the evidence was generated (appears depending on service),

T_i the date and time the event or action took place,

Q optional data that need to be protected,

Imp(*m*) imprint of the message *m*, either (1) the hash-code of the message *m*, or (2) the message *m*.

NOTE Depending on non-repudiation service, index i has values i = 1 or 2.

8.2.2 Data items used in non-repudiation tokens

Non-repudiation tokens consist of a text-field and a secure envelope, as follows:

Non-repudiation token = (*text*, $SENV_{TTP}(z)$)

NOTE Text includes additional data (such as a message identifier or key identifier) that does not need to be cryptographically protected but may be needed to identify the message and the key used in the computation of the check value MAC within SENV calculation. This information depends upon the technique being used.

8.3 Non-repudiation tokens

8.3.1 Provision of evidence

Evidence is provided by non-repudiation tokens, and, if the policy requires it, by additional tokens such as a time stamping token (*TST*), or a token provided by another trusted fourth party (e.g., a Notary) giving additional assurance about an event or action, or about the existence of a message.

If the trusted third party is able to generate a trusted time stamp itself, the addition of a time stamping token (*TST*) as evidence is unnecessary.

NOTE 1 The time included in non-repudiation tokens (*NROT* and *NRDT*) is provided by a trusted authority, i.e. it is regarded secure.

NOTE 2 If the trusted third parties (TTP, DA) are unable to provide a trusted time stamp, then a time stamping token (*TST*) provided by the trusted time stamping authority (TSA) shall be added to the set of non-repudiation information to complete the evidence. The trusted time stamping authority is the instance of trust in providing a trusted time stamp.

8.3.2 Non-repudiation of origin token

A non-repudiation of origin token (*NROT*) is created by the TTP at the request of the originator:

NROT = (*text,* z_1*,* $MAC_{TTP}(z_1)$), with

z_1 = (*Pol*, f_1, *A*, *B*, *C*, *D*, *TG*, T_1, *Q*, *Imp*(*m*)).

The information z_1 necessary for the *NROT* consists of the following data items:

Pol the distinguishing identifier of the non-repudiation policy (or policies) which apply to the evidence,

f_1 a flag indicating non-repudiation of origin,

A the distinguishing identifier of the originator,

B the distinguishing identifier of the intended recipient,

C the distinguishing identifier of the *TTP* generating the evidence,

D the distinguishing identifier of the observer, if an independent observer is involved,

TG the date and time the evidence was generated,

T_1 the date and time the message was originated,

Q optional data that need to be protected,

Imp(m) imprint of the message m, either (1) the hash-code of the message m, or (2) the message m.

8.3.3 Non-repudiation of delivery token

A non-repudiation of delivery token (*NRDT*) is created by the TTP at the request of the recipient:

$NRDT = (text, z_2, MAC_{TTP}(z_2))$, where

$z_2 = (Pol, f_2, A, B, C, D, TG, T_2, Q, Imp(m))$.

The information z_2 necessary for the *NRDT* consists of the following data items:

Pol	the distinguishing identifier of the non-repudiation policy (or policies) which apply to the evidence,
f_2	a flag indicating non-repudiation of delivery,
A	the distinguishing identifier of the originator,
B	the distinguishing identifier of the recipient,
C	the distinguishing identifier of the Trusted Third Party (*TTP*),
D	the distinguishing identifier of the observer, if an independent observer is involved,
TG	the date and time the evidence was generated,
T_2	the date and time the message was delivered,
Q	optional data that need to be protected,
Imp(*m*)	imprint of the message *m*, either (1) the hash-code of the message *m*, or (2) the message *m*.

8.3.4 Time stamping token

The time stamping token (*TST*) provided by the time stamping authority (TSA) can be created using any method from ISO/IEC 18014.

8.4 Verification of tokens by the TTP

8.4.1 Verification process

At some point during the non-repudiation exchange, it may be necessary for the TTP to verify tokens (as defined above) received from an entity. It may also be necessary to re-verify the tokens some time after the exchange is completed, or to provide evidence as to their credibility to some fourth party.

The process of verification includes not only checking that a token was created by the TTP, but also that the token is appropriately associated with the data field of the message for which it was created, as well as the freshness of the time stamp. To check that a token was created for a given message, an entity shall verify the message by comparing the *Imp*(*m*) computed from the message and the *Imp*(*m*) contained in the data field *z,* and then requesting the TTP to verify the token together with its data field.

To verify secure envelopes generated using symmetric integrity techniques, the verification operation involves recalculating the cryptographic check value $MAC_X(y)$ using the appropriate secret key *x* of entity *X* and the data *y* contained in the secure envelope, and comparing this result with that presented.

A TTP shall verify a token using one of the two methods specified in 8.4.2 and 8.4.3.

8.4.2 On-line verification of the token

For this method of verification, the TTP uses a security module containing the secret key *ttp* to verify the token. The security module compares the token with a value that is internally regenerated using the data item z_i and the secret key *ttp*, and returns the outcome of the comparison by specifying whether the token is valid or not. Since the key *ttp* is not known by anyone other than the TTP, the token presented for verification is considered authentic, if the security module returns that the token is valid.

8.4.3 Table of tokens

For this method of verification, a table of all tokens issued by the TTP is stored. For each token created, the TTP records the token along with its associated data field (z_i) and the key identifier of the secret key *ttp.* To verify it, the TTP uses the token as an index into the table to look it up. If the token presented for verification is found in the table, and the data field that is presented with the token (or, is part of the token) corresponds to the data field associated with it in the table, then the token is considered to be authentic.

9 Specific non-repudiation mechanisms

9.1 Mechanisms for non-repudiation

The non-repudiation mechanisms in this Clause 9 allow for generation of evidence for non-repudiation of origin (NRO) and delivery (NRD). In addition the mechanism for generating the time stamp is defined. Entity A wishes to send a message *m* to entity B and thus will be the originator of the non-repudiation transfer. Entity B will be the recipient.

In some mechanisms described in Clause 8, the data field z_i is used. This data field is identical to the z_i data field in the non-repudiation token except it does not contain information on the time at which the evidence is generated. Such time information will be provided by the TTP (or DA) or by the time stamping authority TSA upon request of the TTP (or DA).

NOTE In case that *Imp*(*m*) is the message *m*, it is not necessary to send *m* together with tokens, and the steps for verifying *Imp*(*m*) are also omitted.

9.2 Mechanism for non-repudiation of origin

9.2.1 Transactions and mechanisms

The originator has created a message which he will send to a specified recipient. The recipient can check that this message is from the claimed sender by using the TTP to verify the associated non-repudiation of origin token.

In the first transaction of this mechanism, the originator forms the data and transmits it in a *SENV* to the TTP. The TTP generates the non-repudiation of origin token (*NROT*) and returns this to the originator A. In the second transaction, the *NROT* concatenated to message *m* is sent from the originator A to the recipient B. In the third transaction, the recipient sends the *NROT* enclosed in a secure envelope to the TTP for verification. Non-repudiation of origin is established in the third transaction.

9.2.2 Token Generation

9.2.2.1 Transaction 1 – between originator A and TTP

a. Entity A generates a secure envelope $SENV_A(z'_1)$ using key *a*, where z'_1 is the z_1 specified in 8.3.2 with the data item *TG* being empty. Entity A then requests an *NROT* by sending the secure envelope to the TTP.

b. The TTP verifies that the secure envelope is from entity A and that entity A is the one with the distinguishing identifier *A*. If verification is successful, the TTP then completes z_1 by inserting the data item *TG* and computes the

$NROT = (text , z_1, MAC_{TTP}(z_1))$

using key *ttp* and returns $SENV_{TTP}(NROT)$ to A.

c. Entity *A* verifies that $SENV_{TTP}(NROT)$ is from the TTP and that the z_1 in the *NROT* corresponds to the z'_1 in the original request.

9.2.2.2 Transaction 2 – from originator A to recipient B

Entity A sends to B*:* (*m* , *NROT*)*.*

9.2.2.3 Transaction 3 – between recipient B and TTP

a. Entity B verifies that the policy PoI in z_1 conforms to their security requirements; verifies that the flag f_1 in z_1 indicates a non-repudiation of origin token; verifies that the identities of A, B and C in z_1; verifies the identity of D in z_1 if an independent observer was present; verifies that the time fields *TG* and T_1 are correct; and verifies that the value of *Imp*(*m*) contained in z_1 is correct.

b. Entity B generates $SENV_B(NROT)$ using key *b* and sends it to the TTP in order to request verification of the *NROT* received from A.

c. The TTP verifies that $SENV_B(NROT)$ is from B and also verifies that the *NROT* is authentic. If $SENV_B(NROT)$ is valid, the TTP sends $SENV_{TTP}((PON, NROT))$ to B, where *PON* is positive if the *NROT* is authentic and negative if the *NROT* is not authentic.

d. Entity B verifies that $SENV_{TTP}((PON, NROT))$ is from the TTP. If it is valid and the verification is positive, non-repudiation of origin (i.e., the message came from A) is established.

e. The *NROT* is saved for future non-repudiation of origin.

9.2.3 Token Verification

If the evidence user B wishes to verify, at some future time, the authenticity of a *NROT*, then it will be performed as specified in Transaction 3 of 9.2.2.3.

9.3 Mechanism for non-repudiation of delivery

9.3.1 Transactions and mechanisms

After receipt of the message *m*, entity B sends in the first transaction of this mechanism a request to generate a non-repudiation of delivery token to the TTP enclosed in a secure envelope. The TTP generates the non-repudiation of delivery token (*NRDT*) and returns this to the recipient B. In the second transaction, the *NRDT* is sent by the recipient B to the originator A. In the third transaction, the originator sends the *NRDT* enclosed in a secure envelope to the TTP for verification. Non-repudiation of delivery is established in the third transaction.

9.3.2 Token generation

9.3.2.1 Transaction 1 – between recipient B and TTP

a. Entity *B* generates a secure envelope $SENV_B(z'_2)$ using key *b*, where z'_2 is the z_2 specified in 8.3.3 with the data item *TG* being empty. Entity B then requests an *NRDT* by sending the secure envelope to the TTP.

b. The TTP verifies that the secure envelope $SENV_B(z'_2)$ is from entity B. If it is, the TTP completes z_2 by inserting the data item *TG* and computes the

$NRDT = (text , z_2, MAC_{TTP}(z_2))$

using key *ttp* and returns $SENV_{TTP}(NRDT)$ to B.

c. Entity B verifies that $SENV_{TTP}(NRDT)$ is from the TTP and that the z_2 contained in the token corresponds to z'_2 sent in step a.

9.3.2.2 Transaction 2 – from recipient B to originator A

Entity B sends to A*:* *NRDT.*

9.3.2.3 Transaction 3 – between originator A and *TTP*

a. Entity A verifies that the policy Pol in z_2 conforms to their security requirements; verifies that the flag f_2 in z_2 indicates a non-repudiation of delivery token; verifies that the identities of A, B and C in z_2; verifies the identity of D in z_2 if an independent observer was present; verifies that the time fields *TG* and T_2 are correct; and verifies that the value of *Imp*(*m*) contained in z_2 is correct.

b. Entity A generates $SENV_A(NRDT)$ using key *a* and sends it to the TTP in order to request verification of the *NRDT* received from B.

c. The TTP verifies that $SENV_A(NRDT)$ is from A and also verifies that the *NRDT* is authentic. If $SENV_A(NRDT)$ is valid, the *TTP* sends $SENV_{TTP}((PON, NRDT))$ to A, where *PON* is positive if the *NRDT* is authentic and negative if the *NRDT* is not authentic.

d. Entity A verifies that $SENV_{TTP}((PON, NRDT))$ is from the TTP. If it is valid and the verification is positive, non-repudiation of delivery is established.

e. The *NRDT* is saved for future non-repudiation of delivery.

9.3.3 Token Verification

If the evidence user A wishes to verify, at some future time, the authenticity of a *NRDT*, then it will be performed as specified in Transaction 3 of 9.3.2.3.

9.4 Mechanism for obtaining a time stamping token

When a trusted time reference is required and when the clock provided by the token generating party cannot be trusted, it is necessary to rely on a trusted third party Time-Stamping Authority (TSA).

The communication between an entity (requester) X and the TSA when requesting a time-stamp is described in ISO/IEC 18014.

Annex A
(informative)

Examples of specific non-repudiation mechanisms

A.1 Examples of non-repudiation mechanisms of origin and delivery

The non-repudiation mechanisms in this Annex provide non-repudiation of origin and non-repudiation of delivery between entities A and B. Entity A wishes to send a message to entity B and thus will be the originator of the non-repudiation exchange. Entity B, as the message recipient, will be the recipient. Prior to use of a mechanism, it is assumed that keys *a* and *b* are in place at entity A and entity B, respectively, and that the TTP possesses keys *a* and *b* in addition to its own key *ttp*.

Three different mechanisms (M1, M2, and M3) for non-repudiation using an on-line TTP are provided.

NOTE 1 By letting data of *SENV* also include time stamps, protection against unauthorized delay or replay of messages can be achieved. By letting the *NROT* and *NRDT* include time stamps, future verification of the time stamps at which a message was transferred can be obtained.

NOTE 2 In case that *Imp*(*m*) is the message *m*, it is not necessary to send *m* together with tokens, and the steps for verifying *Imp*(*m*) are also omitted.

A.2 Mechanism M1: Mandatory NRO, optional NRD

A.2.1 Five transactions of mechanism M1

Non-repudiation of origin is established in three transactions between the entities and the TTP. If the optional NRD steps are continued (at the recipient's prerogative), non-repudiation of delivery is established within two more transactions (see Figure A.1).

NOTE 1 While it is up to the recipient to continue the steps required for non-repudiation of delivery, it is important to note that this optional non-repudiation of delivery is fully binding once it has been established.

NOTE 2 This mechanism provides non-repudiation of origin and can optionally be used to provide non-repudiation of delivery. The protocol's use (i.e. whether it is used to provide non-repudiation of origin or both non-repudiation of origin and non-repudiation of delivery) shall be decided between the sender A, the recipient B and the trusted third party TTP before beginning a particular protocol execution.

A.2.2 Transaction 1 – between originator A and TTP

The transactions of mechanism M1 (see Figure A.1) are designated as

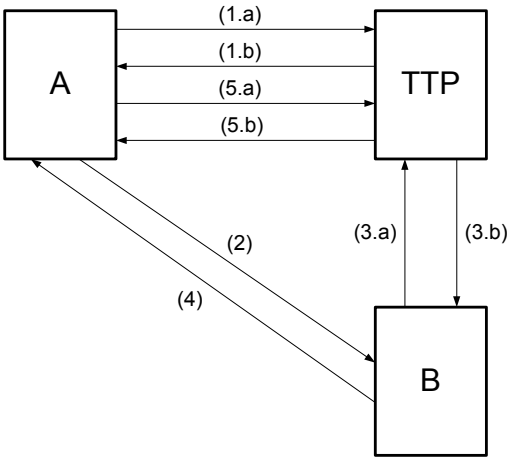
a. Entity A generates a secure envelope $SENV_A(z'_1)$ using key *a*, where z'_1 is the z_1 specified in 8.3.2 with the data item *TG* being empty. Entity A then requests an *NROT* by sending the secure envelope to the TTP.

b. The TTP verifies that the secure envelope is from entity A and that entity A is the one with the distinguishing identifier *A*. If it is, the TTP completes z_1 by inserting the data item *TG* and computes the

$NROT = (text , z_1, MAC_{TTP}(z_1))$

using key *ttp* and returns $SENV_{TTP}(NROT)$ to A.

c. Entity A verifies that $SENV_{TTP}(NROT)$ is from the TTP.



(1.a) $SENV_A$ (z_1')
(1.b) $SENV_{TTP}$ (*NROT*)
(2) (*m, NROT*)
(3.a) ($SENV_B$ (*NROT*)) and $SENV_B$ (z_2')
(3.b) $SENV_{TTP}$ ((*PON, NROT, NRDT*)) or $SENV_{TTP}$ ((*PON, NROT*))
(4) *NRDT*
(5.a) $SENV_A$ (*NRDT*)
(5.b) $SENV_{TTP}$ ((*PON, NRDT*))

Figure A.1 — Mechanism M1

A.2.3 Transaction 2 – from originator A to recipient B

Entity *A* sends to *B:* (*m* , *NROT*).

A.2.4 Transaction 3 – between recipient B and TTP

a. Entity B verifies the value of *Imp*(*m*) contained in z_1, then generates $SENV_B(NROT)$ and $SENV_B(z'_2)$, where z'_2 is z_2 specified in 8.3.3 with the data item *TG* being empty, using key *b* and sends it to the TTP in order to request verification of the *NROT* received from A and generation of the *NRDT*.

b. The TTP checks $SENV_B(NROT)$ and *NROT*. If both are valid, the TTP verifies that $SENV_B(z'_2)$ is from the entity B. If it is, the TTP completes z_2 by inserting the data item *TG* and computes the

$$NRDT = (text , z_2, MAC_{TTP}(z_2))$$

using key *ttp* and sends

$$SENV_{TTP}((PON, NROT, NRDT)),$$

where *PON* is positive, to B. If the secure envelope $SENV_B(NROT)$ is valid, but the *NROT* is not, the TTP sends

$$SENV_{TTP}((PON, NROT)),$$

where *PON* is negative, to B.

c. Entity B verifies that $SENV_{TTP}((PON, NROT, NRDT))$ is from the TTP. If it is valid and *PON* is positive, non-repudiation of origin (i.e., the message came from A) is established.

d. The *NROT* is saved for future non-repudiation of origin.

A.2.5 Transaction 4 – from recipient B to originator A

Entity B sends the *NRDT* to A.

A.2.6 Transaction 5 –between originator A and recipient TTP

a. Entity A verifies the value of *Imp*(*m*) contained in z_2, then generates $SENV_A(NRDT)$ using key *a* and sends it to the TTP in order to request verification of the *NRDT* received from B.

b. The TTP verifies that $SENV_A(NRDT)$ is from A and also verifies that the *NRDT* is authentic. If both are valid, the TTP sends $SENV_{TTP}((PON, NRDT))$, where *PON* is positive, to *A*. If the *NRDT* is not valid, the TTP sends $SENV_{TTP}((PON, NRDT))$, where *PON is* negative, to A.

c. Entity A verifies that $SENV_{TTP}((PON, NRDT))$ is from the TTP. If it is valid and the verification is positive, non-repudiation of delivery is established.

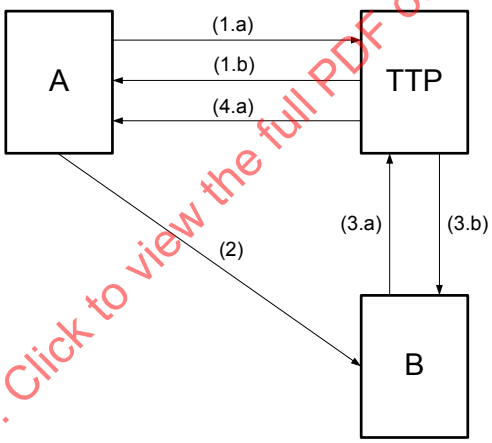
d. Entity A saves the *NRDT* for future non-repudiation of delivery.

A.3 Mechanism M2: Mandatory NRO, mandatory NRD

A.3.1 Four transactions of mechanism M2

Non-repudiation of origin and non-repudiation of delivery are established in four transactions between the two entities and the TTP. In this mechanism, the TTP sends the message receipt directly to A in a *SENV* at the same time that he sends it to B.

The transactions of mechanism M2 (see Figure A.2) are designated as:



(1.a) $SENV_A$ (z_1')
(1.b) $SENV_{TTP}$ (*NROT*)
(2) (*m, NROT*)
(3.a) $SENV_B$ (*NROT*) and $SENV_B$ (z_2')
(3.b) $SENV_{TTP}$ ((*PON, NROT, NRDT*)) or $SENV_{TTP}$ ((*PON, NROT*))
(4.a) $SENV_{TTP}$ (*NRDT*)

Figure A.2 — Mechanism M2

A.3.2 Transaction 1 – between originator A and TTP

a. Entity A generates a secure envelope $SENV_A(z'_1)$ using key *a*, where z'_1 is the z_1 specified in 8.3.2 with the data item *TG* being empty. Entity A then requests an *NROT* by sending the secure envelope to the TTP.

b. The TTP verifies that the secure envelope is from entity A. If it is, the TTP completes z_1 by inserting the data item *TG* and then computes the *NROT* = (*text* , z_1, $MAC_{TTP}(z_1)$) using key *ttp* and returns $SENV_{TTP}(NROT)$ to A using key *a.*

c. Entity A verifies that $SENV_{TTP}(NROT)$ is from the TTP.