INTERNATIONAL TELECOMMUNICATION UNION

# ITU-T X.274

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

(07/94)

## DATA NETWORKS AND OPEN SYSTEM COMMUNICATIONS

## OPEN SYSTEMS INTERCONNECTION – SECURITY PROTOCOLS

---

# INFORMATION TECHNOLOGY – TELECOMMUNICATION AND INFORMATION EXCHANGE BETWEEN SYSTEMS – TRANSPORT LAYER SECURITY PROTOCOL

## ITU-T Recommendation X.274

(Previously "CCITT Recommendation")

---

## Foreword

ITU (International Telecommunication Union) is the United Nations Specialized Agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of the ITU. Some 179 member countries, 84 telecom operating entities, 145 scientific and industrial organizations and 38 international organizations participate in ITU-T which is the body which sets world telecommunications standards (Recommendations).

The approval of Recommendations by the Members of ITU-T is covered by the procedure laid down in WTSC Resolution No. 1 (Helsinki, 1993). In addition, the World Telecommunication Standardization Conference (WTSC), which meets every four years, approves Recommendations submitted to it and establishes the study programme for the following period.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC. The text of ITU-T Recommendation X.274 was approved on 1st of July 1994. The identical text is also published as ISO/IEC International Standard 10736.

---

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

© ITU 1995

All rights reserved. No part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the ITU.

**ITU-T X-SERIES RECOMMENDATIONS**

**DATA NETWORKS AND OPEN SYSTEM COMMUNICATIONS**
(February 1994)

**ORGANIZATION OF X-SERIES RECOMMENDATIONS**

| Subject area | Recommendation series |
|---|---|
| PUBLIC DATA NETWORKS | |
| Services and facilities | X.1-X.19 |
| Interfaces | X.20-X.49 |
| Transmission, signalling and switching | X.50-X.89 |
| Network aspects | X.90-X.149 |
| Maintenance | X.150-X.179 |
| Administrative arrangements | X.180-X.199 |
| OPEN SYSTEMS INTERCONNECTION | |
| Model and notation | X.200-X.209 |
| Service definitions | X.210-X.219 |
| Connection-mode protocol specifications | X.220-X.229 |
| Connectionless-mode protocol specifications | X.230-X.239 |
| PICS proformas | X.240-X.259 |
| Protocol identification | X.260-X.269 |
| Security protocols | X.270-X.279 |
| Layer managed objects | X.280-X.289 |
| Conformance testing | X.290-X.299 |
| INTERWORKING BETWEEN NETWORKS | |
| General | X.300-X.349 |
| Mobile data transmission systems | X.350-X.369 |
| Management | X.370-X.399 |
| MESSAGE HANDLING SYSTEMS | X.400-X.499 |
| DIRECTORY | X.500-X.599 |
| OSI NETWORKING AND SYSTEM ASPECTS | |
| Networking | X.600-X.649 |
| Naming, addressing and registration | X.650-X.679 |
| Abstract Syntax Notation One (ASN.1) | X.680-X.699 |
| OSI MANAGEMENT | X.700-X.799 |
| SECURITY | X.800-X.849 |
| OSI APPLICATIONS | |
| Commitment, concurrency and recovery | X.850-X.859 |
| Transaction processing | X.860-X.879 |
| Remote operations | X.880-X.899 |
| OPEN DISTRIBUTED PROCESSING | X.900-X.999 |

# CONTENTS

## Summary

This Recommendation | International Standard specifies the protocol which can support the integrity, confidentiality, authentication and access control services identified in the OSI security model as relevant to the transport layer. The protocol supports these services through the use of cryptographic mechanisms, security labelling and assigned attributes, such as cryptographic keys.

## 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 and IEC participate in the development of International Standards through technical committees established by the respective 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 JTC1. 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 10736 was prepared by Joint Technical Committee ISO/IEC JTC1, Information technology – Sub-Committee 6 – Telecommunications and information exchange between systems.

Annex A and B form integral parts of this International Standard. Annex C and D are for information only.

## Introduction

The transport protocol specified in ITU-T Rec. X224 | ISO/IEC 8073 provides the connection oriented transport service described in ITU-T Rec. 234 | ISO/IEC 8072. The transport protocol specified in ITU-T Rec. 234 | ISO 8602 provides the connectionless-mode transport service described in ISO 8072/AD1. This Recommendation | International Standard specifies optional additional functions to ITU-T Rec. X.224 | ISO/IEC 8073 and ITU-T Rec. X.234 | ISO 8602 permitting the use of cryptographic techniques to provide data protection for transport connections or for connectionless-mode TPDU transmission.

Annex A and B form integral parts of this International Standard. Annex C and D are for information only.

**INTERNATIONAL STANDARD**

**ITU-T RECOMMENDATION**

# INFORMATION TECHNOLOGY – TELECOMMUNICATION AND INFORMATION EXCHANGE BETWEEN SYSTEMS – TRANSPORT LAYER SECURITY PROTOCOL

## 1 Scope

The procedures specified in this Recommendation | International Standard operate as extensions to those defined in ITU-T Rec. X.224 | ISO/IEC 8073 and ITU-T Rec. X.234 | ISO 8602 and do not preclude unprotected communication between transport entities implementing ITU-T Rec. X.224 | ISO/IEC 8073 or ITU-T Rec. X.234 | ISO 8602.

The protection achieved by the security protocol defined in this Recommendation | International Standard depends on the proper operation of security management including key management. However, this Recommendation | International Standard does not specify the management functions and protocols needed to support this security protocol.

This protocol can support all the integrity, confidentiality, authentication and access control services identified in CCITT Rec. X.800 | ISO 7498-2 as relevant to the transport layer. The protocol supports these services through use of cryptographic mechanisms, security labelling and attributes, such as keys and authenticated identities, pre-established by security management or established through the use of the Security Association – Protocol (SA-P).

Protection can be provided only within the context of a security policy.

This protocol supports peer-entity authentication at the time of connection establishment. In addition, rekeying is supported within the protocol through the use of SA-P or through means outside the protocol.

Security associations can only be established within the context of a security policy. It is a matter for the users to establish their own security policy, which may be constrained by the procedures specified in this Recommendation | International Standard.

The following items could be included in a Security Policy:

a) the method of SA establishment/release, the lifetime of SA;

b) Authentication/Access Control mechanisms;

c) Label mechanism;

d) the procedure of the receiving an invalid TPDU during SA establishment procedure or transmission of protected PDU;

e) the lifetime of Key;

f) the interval of the rekey procedure in order to update key and security control information (SCI) exchange procedure;

g) the time out of SCI exchange and rekey procedure;

h) the number of retries of sci exchange and rekey procedure.

This Recommendation | International Standard defines a protocol which may be used for Security Association establishment. Entities wishing to establish an SA must share common mechanisms for authentication and key distribution. This Recommendation | International Standard specifies one algorithm for authentication and key distribution which is based on public key crypto systems. The implementation of this algorithm is not mandatory; however, when an alternative mechanism is used, it shall satisfy the following conditions:

a) All SA attributes defined in 5.2 are derived.

b) Derived keys are authenticated.

## 2 Normative references

The following Recommendations and International Standards contain provisions which, through reference in this text, constitute provisions of this Recommendation | International Standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this Recommendation | International Standard are encouraged to investigate the possibility of applying the most recent editions of the standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau of the ITU maintains a list of currently valid ITU-T Recommendations.

### 2.1 Identical Recommendations | International Standards

- ITU-T Recommendation X.214 (1993) | ISO 8072:1994, *Information technology – Open Systems Interconnection – Transport service definition*.
- ITU-T Recommendation X.234 (1993) | ISO 8602:1987, *Information technology – Protocol for providing the OSI connectionless-mode transport service*.

### 2.2 Paired Recommendations | International Standards equivalent in technical content

- CCITT Recommendation X.200 (1988), *Reference Model of Open Systems Interconnection for CCITT applications.*

  ISO 7498:1984, *Information processing systems – Open Systems Interconnection – Basic Reference Model*.
- CCITT Recommendation X.800 (1991), *Security architecture for Open Systems Interconnection for CCITT applications*.
- ISO 7498-2:1989, *Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 2: Security Architecture*.
- ITU-T Recommendation X.224 (1993), *Protocol for providing the OSI connection-mode transport service*.
- ISO/IEC 8073:1992, *Information technology – Telecommunications and information exchange between systems – Open Systems Interconnection – Protocol for providing the connection-mode transport service.*
- CCITT Recommendation X.208 (1988), *Specification of Abstract Syntax Notation One (ASN.1)*.

  ISO/IEC 8824:1990, *Information technology – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1)*.
- CCITT Recommendation X.209 (1988), *Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1).*

  ISO 8825:1990, *Information technology – Open Systems Interconnection – Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)*.
- ITU-T Recommendation X.264 (1993), *Transport protocol identification mechanism*.

  ISO/IEC 11570:1992, *Information technology – Telecommunications and information exchange between systems – Open Systems Interconnection – Transport protocol identification mechanism.*

### 2.3 Additional references

- ISO/IEC 7498/AD1:1987, *Information processing systems – Open Systems Interconnection – Basic Reference Model – Addendum 1: Connectionless-mode transmission*.
- ISO 8072/AD1:1986, *Information processing systems – Open Systems Interconnection – Transport service definition – Addendum: Connectionless-mode transmission.*
- ISO/IEC 9834-1:1993, *Information technology – Open Systems Interconnection – Procedures for the operation of OSI Registration Authorities – Part 1: General Procedures.*
- ISO/IEC 9834-3:1990, *Information technology – Open Systems Interconnection – Procedures for the operation of OSI Registration Authorities – Part 3: Registration of object identifier component values for joint ISO/CCITT use.*

# 3 Definitions

This Recommendation | International Standard is based on the concepts developed in the Reference Model for Open Systems Interconnection (CCITT Rec. X.200 | ISO 7498) as well as CCITT Rec. X.800 | ISO 7498-2 on Security Architecture.

## 3.1 Security reference model definitions

This Recommendation | International Standard makes use of the following terms as defined in CCITT Rec. X.800 | ISO 7498-2:

a) access control;

b) asymmetric;

c) ciphertext;

d) cleartext;

e) confidentiality;

f) data integrity;

g) data origin authentication;

h) denial of service;

i) end-to-end encipherment;

j) key;

k) key management;

l) security policy;

m) symmetric.

## 3.2 Additional definitions

For the purposes of this Recommendation | International Standard, the following definitions apply:

**3.2.1 cryptoperiod**: The length of time for which a cryptographic key is permitted to be used. After this time has expired the key must be replaced.

**3.2.2 in-band protocol mechanism**: A protocol mechanism defined in this Recommendation | International Standard.

**3.2.3 out-of-band protocol mechanism**: A protocol mechanism not defined in this Recommendation | International Standard.

**3.2.4 pairwise key**: A pair of related (Public Key) or identical key (Secret Key) values generated for use between two particular parties.

**3.2.5 reflection protection**: A protection mechanism to detect when a protocol data unit has been sent back to the originator.

**3.2.6 security association**: The relationship between communicating entities for which there exists corresponding SA-Attributes.

**3.2.7 security association attributes:** The collection of information required to control the security of communications between an entity and its remote peer(s).

**3.2.8 SE TPDU**: The encapsulated TPDU for security in order to send the TPDU defined in ITU-T Rec. X.224 | ISO/IEC 8073 or ITU-T Rec. X.234 | ISO 8602 after securing it.

# 4 Symbols and abbreviations

This Recommendation | International Standard makes use of the following abbreviations from clause 4 of ITU-T Rec X.224 | ISO/IEC 8073:

| | |
|---|---|
| CR TPDU | Connection request TPDU |
| DC TPDU | Disconnect confirm TPDU |

| | |
|---|---|
| DR TPDU | Disconnect request TPDU |
| DST-REF | Destination reference (field) |
| DT TPDU | Data TPDU |
| ED TPDU | Expedited Data TPDU |
| ED-TPDU-NR | Expedited Data TPDU number (field) |
| ER TPDU | Error TPDU |
| LI | Length indicator (field) |
| NC | Network Connection |
| SN | Sequence Number |
| SRC-REF | Source Reference (field) |
| TC | Transport Connection |
| TPDU | Transport protocol data unit |
| TPDU-NR | DT TPDU number (field) |

Additionally, the following abbreviations are used in this Recommendation | International Standard:

| | |
|---|---|
| CBTSS | Connection Based Transport Security Service |
| Conf_no | Confidentiality is not to be provided |
| Conf_yes | Confidentiality is to be provided |
| DEK | Data Encipherment Key |
| GTSS | General Transport Security Service |
| ICV | Integrity Check Value |
| Integ_no | Integrity is not to be provided |
| Integ_yes | Integrity is to be provided |
| KEK | Key Encipherment Key |
| KEY-ID | Key Identifier |
| Kg_esp | A separate cryptographic key is used for each end system pair |
| Kg_esp_sr | A separate cryptographic key is used for each end system pair and security level set |
| Kg_tc | A separate cryptographic key is used for each Transport connection |
| LABEL | Security Label |
| LLSG | Lower Layer Security Guidelines |
| LME | Layer Management Entity |
| MAC | Message Authentication Code |
| MDC | Manipulation Detection Code |
| NLSP | Network Layer Security Protocol |
| NSAP | Network Service Access Point |
| NSDU | Network Service Data Unit |
| PAD | Padding (field) |
| Ppl_abs | Security Label never used on TPDUs |
| Ppl_pres | Security Label used on every TPDU |
| SA-P | Security Association – Protocol |
| SE TPDU | Security Encapsulation TPDU |
| TLSP | Transport Layer Security Protocol |

# 5 Overview of the Protocol

## 5.1 Introduction

CCITT Rec. X.800 | ISO 7498-2 identifies the following security services as being relevant to the transport layer:

- Peer entity authentication;
- Data origin authentication;
- Access control Service;
- Connection confidentiality;
- Connectionless confidentiality;
- Connection integrity with recovery;
- Connection integrity without recovery;
- Connectionless integrity.

NOTES

1 ITU-T Rec. X.214 | ISO 8072 currently only defines 4 levels of protection quality:

a) no protection features;

b) protection against passive monitoring;

c) protection against modification, replay, addition or deletion;

d) both b) and c),

which are equivalent to the following security services.

CCITT Rec. X.800 | ISO 7498-2 on OSI Security Architecture uses the following terms for these security services:

a) no security services;

b) connection/connectionless confidentiality;

c) connection/connectionless integrity (with or without recovery); and

d) both connection/connectionless confidentiality and integrity.

A Defect Report has been raised on ITU-T Rec. X.214 | ISO 8072 to allow these and possibly other forms of protection.

2 Connectionless integrity does not protect against addition or deletion of connectionless SDUs and only provides limited replay protection.

TLSP used with ITU-T Rec. X.224 | ISO/IEC 8073 can support connection integrity with and without recovery, connection confidentiality, access control service and peer entity authentication with each connection individually protected. However, a key may be shared between several connections.

TLSP used with ITU-T Rec. X.234 | ISO 8602 can support connectionless integrity, connectionless confidentiality, access control service and data origin authentication.

This Recommendation | International Standard specifies protocol extensions for providing confidentiality and integrity data protection, including:

a) procedures incorporating cryptographic techniques in protocol processing;

b) the minimum characteristics of cryptographic algorithms with which these procedures can be used;

c) the structure and encoding of data units necessary to achieve interoperability.

Figures 1 and 2 show the location of TLSP in the seven layer ISO model.

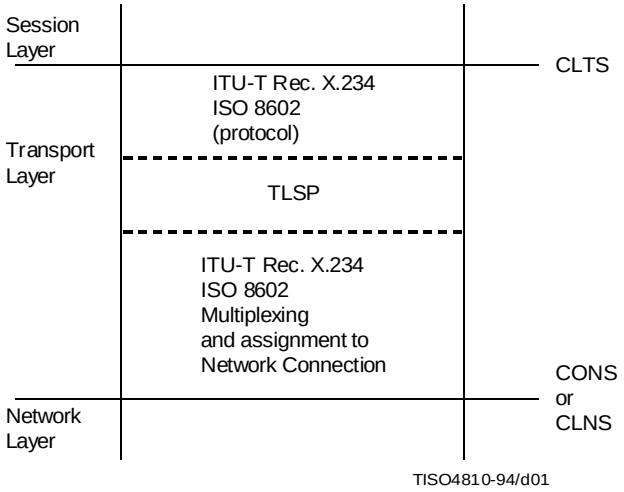



**Figure 1 – TLSP with ITU-T Rec. X.234 | ISO 8602**

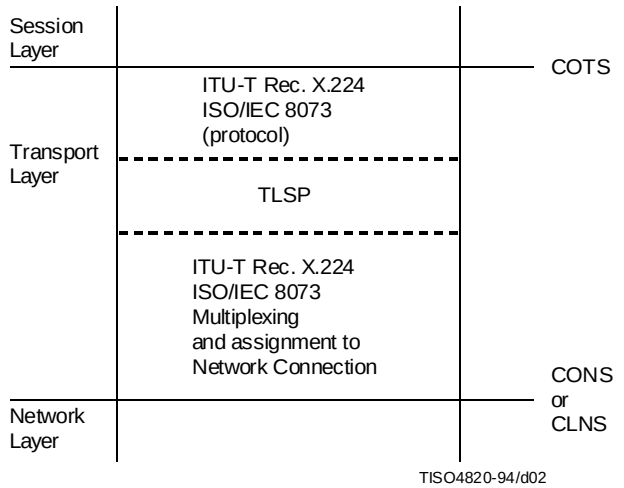



**Figure 2 – TLSP with ITU-T Rec. X.224 | ISO/IEC 8073**

## 5.2 Security Associations and attributes

The specific TLSP processing options used in an instance of communications are determined by the set of security attributes including pairwise protection keys. TLSP assumes that two transport entities share a set of corresponding attributes. The Security Association identifier SA-ID, identifies a set of attributes which may be used to protect an instance of communication.

Each security association is defined by a set of attributes at each end system. The means of establishing all of the attributes which are to be used in an association is currently outside the scope of this Specification. Some could be established by manual exchange of attributes and some of these could be established through use of an Agreed Set of Security Rules (ASSR). An ASSR is a common set of rules which specify the security mechanisms to be used, including

all parameters needed to define the operation of the mechanism for a given protection service(s). Security rules and their identifiers may be registered by third parties. See Annex B for an illustrative example of an ASSR.

Other attributes such as lifetime and timeout of rekey procedure may be defined under the security policy.

TLSP uses these security association attributes to determine processing characteristics of the user data. The following describe the attributes for TLSP and list the mnemonics used to refer to these attributes in this Specification. A set of attributes appropriate for two communicating end systems is dependant on mechanisms used and the security policy.

a) SA Identification

   1) Local_SAID: Octet String The local identifier of the SA.

   2) Peer_SAID: Octet String The remote peer identifier of the SA.

   3) SAID_Len: Integer – Length of the SAID defined by the ASSR
   Integer of range 2 to 126.

   The value of the Local_SAID and Peer_SAID is set up on SA establishment. The value of SAID_Len is defined for a given ASSR.

   When a TLSP entity determines that a particular SA is discontinued, it shall place the SA-ID which it has allocated in a frozen state. While frozen, the SA-ID shall not be re-used. The period in which the SA-ID is frozen shall be greater than the lifetime of the PDU(s) of the underlying network.

b) Indicator of which TLSP entity takes on the role of “initiator” and which entity takes on the role of “responder”. This attribute indicates how the direction indicator should be set to detect reflected TPDUs.

   Initiator: Boolean.

   The value of this attribute is set up on SA establishment.

c) Address of peer TLSP entity(s)

   Peer_Adr: Octet string

   The value of this attribute is set up on SA establishment and indicates either the NSAP address of the transport entity, if the same key is shared between several connections or the connection identifier via the local and remote transport reference numbers, if the key is for only the one connection.

d) Identifier for the agreed set of security rules to be applied for this association

   ASSR_ID: Object Identifier as defined in ASN.1 CCITT Rec. X.208 | ISO 8824

   The value of this attribute is set up on SA establishment or pre-established.

e) Protection QOS selected for the SA

   QOS_Label: Format defined by ASSR

   AC: (Access Control Level) Integer of range defined by the ASSR

   The following QOS parameters are only relevant to TLSP used in conjunction with ITU-T Rec. X.234 | ISO 8602:

   – DOAuth: (Data Origin Authentication level) Integer of range defined by ASSR.

   – CLConf: (Connectionless Confidentiality level) Integer of range defined by ASSR.

   – CLInt: (Connectionless Integrity level) Integer of range defined by ASSR.

   The following QOS parameters are only relevant to TLSP used in conjunction with ITU-T Rec. X.224 | ISO/IEC 8073:

   – Auth: (Peer Entity Authentication level) Integer of range defined by ASSR.

   – CO Conf: (Connection Confidentiality level) Integer of range defined by ASSR.

   – CO Int: (Connection Integrity without recovery) Integer of range defined by ASSR.

   – CO Intr: (Connection Integrity with recovery) Integer of range defined by ASSR.

   – CLConf: (Connectionless Confidentiality level) Integer of range defined by ASSR.

   – CLInt: (Connectionless Integrity level) Integer of range defined by ASSR.

   The value of these attributes are set up on SA establishment or pre-established.

f) Mechanisms selected for the SA

   – Label: Boolean – Explicit labelling TPDUs.

- Conf: Boolean – Confidentiality of a Secure Data Transfer by encipherment.
- ICV: check Boolean – Integrity of a Secure Data Transfer contents using an integrity value.
- SN: Boolean – Connection Integrity Sequence number procedure to be used.
- PE-Authentication: Boolean – Peer Entity Authentication using exchange of encapsulated Connect Request / Connect Response PDUs.
- UNProt: Boolean – Unprotected TPDUs.

g) Label mechanism attributes

The values of these attributes are set up on SA establishment or pre-established. This attribute specifies the set of allowable security labels for the security association.

Label_set: Set of {

Label_Ref: Integer
Label_Defining_Auth: Object Identifier
Label_Content: Format defined by Label_Defining_Auth

}

h) ICV mechanism attributes

- ICV_Alg: Object Identifier
- ICV_Len: Integer
- ICV_Blk: Integer Block size of padding for ICV algorithm

The following attributes are only present if the algorithm is cryptographic:

- ICV_Kg: Integer of value Kg_tc or Kg_esp or Kg_esp_sr

  Key granularity is either:

  - Kg_tc A separate cryptographic key is used for each transport connection
  - Kg_esp A separate cryptographic key is used for each end system pair
  - Kg_esp_sr level A separate cryptographic key is used for each end system pair and security set.

The values of the above attributes are defined by the ASSR given the protection QOS.

- ICV_Gen_key: ICV generation key reference – form defined by ASSR
- ICV_Check_Key: ICV check key reference – form defined by ASSR

i) SN Mechanism attributes

The following attributes are only relevant for TLSP used in conjunction with ITU-T Rec. X.224 | ISO/IEC 8073:

- Data_Local_SN: SN for last normal data sent
- Data_Peer_SN: SN for last normal data received

The initial values of these attributes are set up as part of the normal connection. The SN is the sequence number used by ITU-T Rec. X.224 | ISO/IEC 8073.

j) EXSN Mechanism attributes

The following attributes are only relevant for TLSP used in conjunction with ITU-T Rec. X.224 | ISO/IEC 8073:

- Data_Local_EXSN: EXSN for last expedited data sent
- Data_Peer_EXSN: EXSN for last expedited data received

The initial values of these attributes are set up as part of the expedited procedure. The EXSN is the sequence number used by ITU-T Rec. X.224 | ISO/IEC 8073.

k) Encipherment Mechanism Attributes

- Enc_Alg: Object identifier (e.g. allocated under ISO 9979)
- Enc_Blk: Integer Block size of padding for encipherment algorithm
- Enc_Kg: Integer of value Kg_tc or Kg_esp or Kg_esp_sr

The Key Granularity attributes are defined in h)

The value of this attribute is defined by the ASSR given the protection QOS.

– Enc_Key: Encipherment key reference – form defined by ASSR

– Dec_Key: Decipherment key reference – form defined by ASSR

NOTE – Additional mechanisms and attributes may be identified in future versions of this Recommendation | International Standard and or for private mechanisms.

### 5.2.1 Security services for connection-oriented Transport protocol

When TLSP is used to provide connection oriented security services, the transport entity shall associate a SA-ID with each protected transport connection (Kg_tc), each transport end system pair (Kg_esp) or each transport end system and security level set (Kg_esp_sr). The SA-ID shall be created explicitly for the protected transport connection(s). The security services to be provided on the connection are those defined by the security association. All TPDUs sent or received over a protected transport connection(s) shall be protected according to the services associated with the security association. If Kg_tc, a one to one correspondence exists between a transport connection and a security association.

If connection-oriented integrity is desired, the security services associated with the security association shall include Integrity Check Value (ICV) processing (ICV = True). Any improperly protected TPDUs received shall be discarded. This reception of improperly protected TPDUs is a security relevant event; however further action hereon is outside the scope of this Recommendation | International Standard (e.g. such as filling audit reports).

### 5.2.2 Security Service for connectionless Transport protocol

When TLSP is used to provide security services for the connectionless mode transport service, the transport entity shall associate an SA-ID with either:

– each transport entity pair (Kg_esp);

– each transport entity and security level set pair (Kg_esp_sr).

The sending transport entity shall protect each TPDU according to the attributes associated with the SA-ID and shall place the peer identifier (SA-ID) in the SA-ID parameter of the SE TPDU. Upon receiving an SE TPDU, the key specified by the SA-ID parameter shall be used to decipher the TPDU and or to verify its ICV. Any improperly protected TPDUs received shall be discarded. This reception of improperly protected TPDUs is a security relevant event; however further action hereon is outside the scope of this Recommendation | International Standard (e.g. such as filling audit reports).

## 5.3 Service assumed of the Network Layer

Security services provided by the TLSP protocol are independent of any security services that may be used by the network layer.

## 5.4 Security management requirements

This security protocol requires that the attributes of a security association have been established prior to an instance of protected communication of user data. These attributes may be established through use of security management functions which are outside the scope of this Recommendation | International Standard or through the use of the SA-P.

The degree of protection achieved will depend upon proper management of security including key management. The procedures in this Recommendation | International Standard assume that

a) storage for cryptographic keys is available;

b) both the sending and receiving transport entities have the same cryptographic key available if symmetric keying is used. For asymmetric keying the same cryptographic keys are not available for both the sending and receiving TLSP entities. Either symmetric or asymmetric keying is allowed by this Recommendation | International Standard.

c) cryptographic keys are pairwise, see 3.2.4.

This Recommendation | International Standard does not define how the cryptographic keys are created, updated, or otherwise managed.

## 5.5 Minimum algorithm characteristics

Both the sending and receiving transport entities must use the same cryptographic algorithm or algorithms. The assumptions regarding cryptographic algorithms are as follows:

a) The same algorithm or a different algorithm shall be used for providing both confidentiality and integrity services.

b) Encipherment and decipherment is performed in multiples of octets.

c) Cryptographic synchronization or initialization is realized on an individual TPDU basis.

It is beyond the scope of this Recommendation | International Standard to specify a particular algorithm or to assess the security strengths or weaknesses of particular algorithms.

## 5.6 Security encapsulation function

Encapsulation is used in conjunction with the encipherment and/or integrity check function to provide the connection or connectionless confidentiality and integrity services. The encipherment function is always cryptographically based whereas the and integrity check functions may or may not be cryptographically based. This is dependent on the user's requirements. When used by the sending entity, encapsulation is applied after all protocol processing functions as described in ITU-T Rec. X.224 | ISO/IEC 8073 and ITU-T Rec. X.234 | ISO 8602, except before multiplexing and assignment of network connection. Decapsulation is applied by the receiving entity after demultiplexing and prior to any other protocol processing functions.

### 5.6.1 Data encipherment function

An encipherment mechanism provides data confidentiality. Each SE TPDU contains sufficient information for decipherment independent of information in any other SE TPDU. This includes identification of the security association attributes (SA-ID) to be used for decipherment as well as any cryptographic synchronization or algorithm initialization sequences.

### 5.6.2 Integrity function

This function supports connectionless or connection integrity and data origin authentication The elements of integrity and the mechanisms used to provide them are:

| Protection Against | Mechanism | CBTSS (CO) | GTSS (CL) |
|---|---|---|---|
| Modification | ICV Computed over the protected header and encapsulated PDU | x | x |
| Insertion | ICV and Transport sequence numbers | x | |
| Deletion | ICV and Transport sequence numbers | x | |
| Connection Replay | Separate key per Transport Connection (Kg_tc) or unique connection identifier under each key | x | |
| PDU Replay | Separate key per Transport Connection (Kg_tc)and use of unique sequence numbers under each key or Unique connection identifier and sequence number under each key | x | |
| Reflection | Direction indicator (Flags Field) in each SE TPDU | x | x |
| Masquerade | ICV and, integrity or encipherment key, unique to a transport address | x | |

### 5.6.3 Security label function

Security labelling is an optional function which can be used to associate a security label with each encapsulated TPDU set. The label indicates the sensitivity of the data. The security label supports access control mechanisms.

The structure and interpretation of the Contents of the Label are defined by various Defining Authorities. The Defining Authority is identified by an object identifier, encoded as a content definition as specified in CCITT Rec. X.209 | ISO/IEC 8825.

### 5.6.4 Security padding function

Security padding is an optional function which can be used to extend the length of an encapsulated TPDU set as needed. This supports cryptographic algorithm requirements for both confidentiality and integrity.

### 5.6.5 Peer Entity Authentication function

This function performs peer entity authentication through exchange of encapsulated connection establishment PDUs containing a connection identifier as shown in Figure 3.

The source and destination references must be:

– integrity protected; and

– unique within the lifeline of the integrity key.

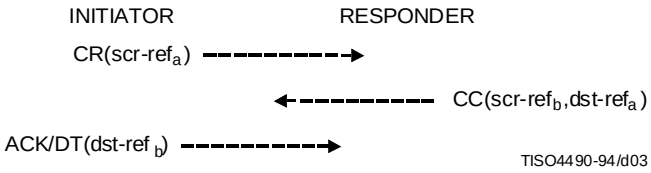



**Figure 3 – Illustration of exchanges to support peer entity authentication**

### 5.6.6 SA Function using in band SA-P

This protocol can be initialized by using procedures defined in ITU-T Rec. X.224 | ISO/IEC 8073 to support the transfer of SA-P PDUs, but that initialization must be done before the establishment of transport connection, or via local management channels. If the procedures defined in ITU-T Rec. X.224 | ISO/IEC 8073 are used a local reference number shall be used to uniquely identify that this is for use within the transport layer for the establishment, maintenance, and release of the SA.

Note – If systems implementing ITU-T Rec. X.234 | ISO 8602 believe that the level of reliability associated with establishing a security association is not available, they may decide to not use the in band SA-P method of establishing a Security Association.

## 6 Elements of procedure

The elements of procedure are as specified in the Connection-oriented Transport Protocol specification (see ITU-T Rec. X.224 | ISO/IEC 8073) and Protocol for Providing the Connectionless-mode Transport Service (see ITU-T Rec. X.234 | ISO 8602), with the following additions.

The protocol mechanisms described below are those used for data encapsulation. A SE TPDU contains:

a) a clear text header;

b) a protected content, length, and flag; if confidentiality is not used, this header is also clear text;

c) a single TPDU or set of TPDUs concatenated according to the rules in ITU-T Rec. X.224 | ISO/IEC 8073;

d) an ICV parameter field, if integrity protection is used.

e) appropriate padding fields for integrity and confidentiality.

f) a security label, if label mechanism is selected.

A TPDU shall be protected based on the attributes of the security association and encapsulated in a SE TPDU. On receipt of a SE TPDU, the transport entity shall verify that all the protection specified by the security association key attributes is present. An improperly protected TPDU (not protected according to SA attributes) shall be discarded.

NOTE – This reception of improperly protected TPDUs is a security relevant event; however further action hereon is outside the scope of this Recommendation | International Standard (e.g. such as filling audit reports).

If the security encapsulation function is invoked for a TPDU for which a suitable SA doesn't exist, the TLSP may invoke either an SA Establishment Protocol as specified in this Recommendation | International Standard or take any other appropriate action.

## 6.1 Concatenation and separation

The procedure for concatenation and separation is as specified in 6.4 of the Connection-oriented Transport Protocol specification (see ITU-T Rec. X.224 | ISO/IEC 8073), with the following changes:

a) Concatenation shall only take place prior to encapsulation. Any TPDU defined in ITU-T Rec. X.224 | ISO/IEC 8073 may be transferred after being encapsulated within an SE TPDU. Only TPDUs which are to be protected under the same security association key may be concatenated.

b) A SE TPDU shall never itself be encapsulated within another SE TPDU.

NOTE – This procedure is not used with the connectionless transport protocol (ITU-T Rec. X.234 | ISO 8602).

## 6.2 Confidentiality

### 6.2.1 Purpose

Confidentiality may be used by the transport protocol connection and connectionless mode for end-to-end protection of TPDU and security control information in transit between communicating transport entities.

### 6.2.2 TPDUs and parameters used

The procedure makes use of the following TPDU and parameters:

– SE-TPDU;

– SA-ID;

– Crypto-synch;

– Encipherment Pad.

### 6.2.3 Procedure

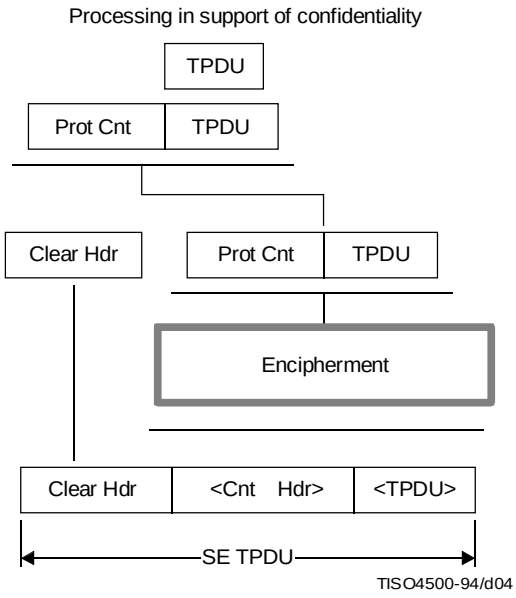
If confidentiality is specified for a security association (Conf = true), then all TPDUs shall be protected by being encapsulated within an SE TPDU. All octets following the SA-ID (protected header and TPDU) shall be enciphered. See Figure 4. If a crypto-synch field is required by the encipherment algorithm, it shall be pre-pended to the protected contents and after the clear header.

Before encipherment an encipherment pad shall be placed, if necessary, at the end of the SE-TPDU so that the length of the protected contents (including the protected content length field), plus the length of the ICV and ICV pad field (if integrity has been requested), plus the length of the encipherment pad is an integral multiple of the encipherment block size (SA Attribute Enc_Blk). On receipt the crypto- synch field if present will be used for synchronization.

The cryptographic algorithm is specified by an attribute of the security association which is identified by the security association identifier (SA-ID).

Upon receipt of a SE TPDU the transport entity uses the key identified by the SA-ID in the SE TPDU to identify the security service and to decipher the SE TPDU. On receipt the content of the encipherment pad field shall be ignored. If the key is not available, the SE TPDU is discarded.

NOTE – This reception of an SE-TPDU with an invalid SA-ID is a security relevant event; however further action hereon is outside the scope of this Recommendation | International Standard (e.g. such as filling audit reports).




NOTE – Quantities in brackets are enciphered quantities.

**Figure 4 – TLSP Encapsulation Methods**

**(TLSP's method for encapsulation and encipherment in support of Confidentiality as indicated in 6.2)**

## 6.3 Integrity processing

The following procedures are used to provide connectionless and connection-oriented integrity services.

### 6.3.1 Integrity Check Value (ICV) processing

#### 6.3.1.1 Purpose

ICV processing may be used by TLSP for both Transport Protocol connection mode (ITU-T Rec. X.224 | ISO/IEC 8073) and connectionless mode (ITU-T Rec. X.234 | ISO 8602) to detect unauthorized modification of user data and security control information while in transit between communicating transport entities.

#### 6.3.1.2 TPDUs and parameters used

The procedure makes use of the following TPDU and parameters:

- SE TPDU;
- SA-ID;
- Integrity PAD;
- ICV.

#### 6.3.1.3 Procedure

There are two types of ICV processing: message authentication code (MAC) and manipulation detection code (MDC). The difference between the use of MAC or MDC is directly related to what is specified -integrity or integrity and confidentiality. If only integrity is selected then a cryptobased MAC shall be used. If integrity and confidentiality is selected the ICV may either be a non-cryptobased manipulation detection code (MDC) such as XOR or checksum or it may be cryptobased such as MAC. It does not need to be cryptobased because the whole protected contents will be encrypted since confidentiality was also selected. If only confidentiality is selected then there is no ICV field.

If data integrity is specified (Integ = True) for a cryptographic association, then an ICV shall protect every SE TPDU. The message authentication code (MAC) is carried in the ICV parameter and occurs as the last field in the SE TPDU. The ICV is computed over the protected contents and encapsulated TPDU. If confidentiality is specified (Conf = True) in addition to integrity, the manipulation detection code (MDC) or the cryptobased MAC is computed prior to encipherment. If necessary an integrity pad shall be placed in the protected contents so that the length of the protected content (including the protected content field) is an integral multiple of the ICV block size (SA Attribute ICV_Blk). On receipt the content of the integrity pad field shall be ignored. See Figure 5.

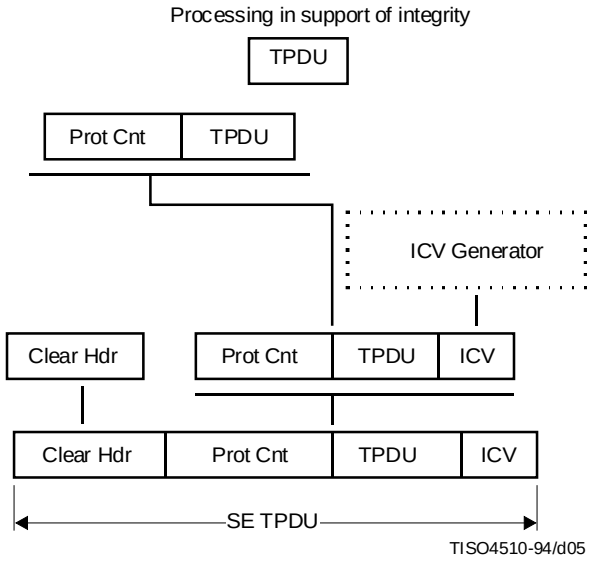



**Figure 5 – TLSP Encapsulation Methods**

**(TLSP's method for encapsulation and ICV generation in support of integrity as indicated in 6.3)**

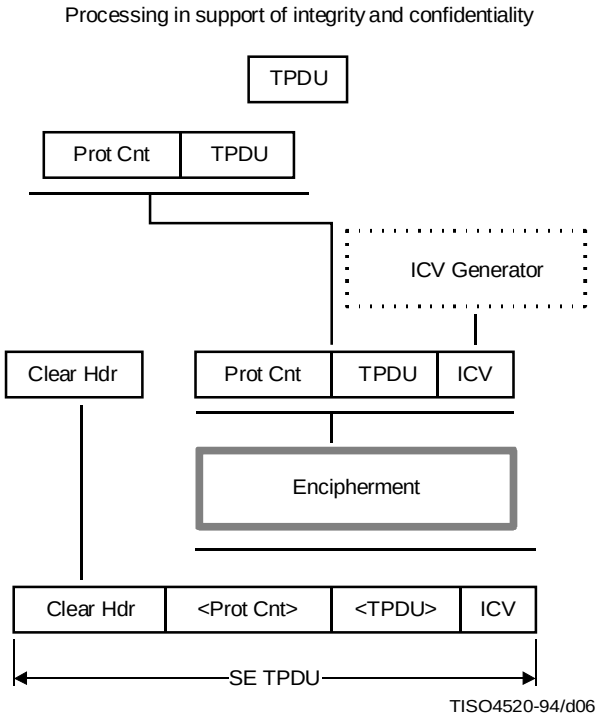
The integrity check function and ICV field length are attributes of the security association.

Upon receiving a SE TPDU on a security association with integrity protection, the ICV field shall be verified by computing a test Integrity Check Value over the protected contents and encapsulated TPDU set. If the Security Association identified by the SA-ID is not available or the test Integrity Check Value is not equal to the ICV field, then the entire SE TPDU shall be discarded.

NOTE – The failure of the ICV check is a security relevant event; however further action hereon is outside the scope of this Recommendation | International Standard (e.g. such as filling audit reports).

If decipherment is also required, the testing of the Integrity Check Value shall be performed subsequent to decipherment.

Figure 6 depicts both integrity and confidentiality.




NOTE – Quantities in brackets are enciphered quantities.

**Figure 6 – TLSP Encapsulation Method**

**(TLSP's method for encapsulation and ICV generation in support of "Integrity and Confidentiality" as indicated in 6.2 and 6.3)**

### 6.3.2 Direction indicator processing

#### 6.3.2.1 Purpose

The purpose of the direction indicator is to provide reflection protection.

#### 6.3.2.2 TPDUs and parameters used

The procedure makes use of the following TPDU and parameters:

– SE TPDU;

– FLAGS.

#### 6.3.2.3 Procedure

Each SE TPDU shall contain the direction indicator bit (FLAGS field) indicating the sender of the TPDU. The parties involved in the security association have already agreed on who is the responder and who is the initiator. When a SE TPDU is sent by the initiator of the security association, the direction indicator bit shall be set to 1. When a SE TPDU is sent by the responder of the security association, the direction indicator bit shall be set to 0. Upon receipt of a SE TPDU the transport entity shall validate the direction indicator bit. If a SE TPDU is received with an incorrect direction indicator the TPDU shall be discarded.

NOTE – The receipt of a SE TPDU with the incorrect direction indicator is a security relevant event; however further action hereon is outside the scope of this Recommendation | International Standard (e.g. such as filling audit reports).

### 6.3.3 Connection integrity sequence number processing

Replay, insertion, and deletion detection requires that each TPDU in a security association have a unique sequence number. When connection-oriented integrity is specified for a connection (Kg_tc and Integ_yes), this is provided using a key per connection in conjunction with the unique sequence number procedure (see 6.3.3.1). This procedure is not used with ITU-T Rec. X.234 | ISO 8602.

#### 6.3.3.1 Unique sequence numbers

The unique sequence numbers are the same sequence numbers as those stated in ITU-T Rec. X.224 | ISO/IEC 8073 (see 6.10 and 6.11).

#### 6.3.3.2 Purpose

Unique sequence numbers is an optional procedure to uniquely identify each DT and ED TPDU (Normal and Expedited Transport Data) within a connection. This procedure is only applicable to ITU-T Rec. X.224 | ISO/IEC 8073 (Classes 2, 3, and 4)

#### 6.3.3.3 Procedure

If the connection-oriented integrity service is specified for a transport connection (Kg_tc and Integ = True), each TPDU shall have unique sequence numbers in a Security Association. Neither transport entity shall transmit a new DT or ED TPDU bearing a sequence number (either TPDU NR or ED TPDU NR) which was previously used with that key. Retransmissions as part of normal error control and recovery may repeat the sequence number under the original key or use a new key. When either the DT or ED sequence number space is exhausted on a particular connection, a different cryptographic key than any previously used to protect data using that connection identifier (DST-REF) may be used for transmitting any further data TPDUs. The key replacement procedure (see 6.7) shall be invoked. If no such key exists, the connection may be released. Upon receipt of a DT or ED TPDU which duplicates a previously received sequence number on the current cryptographic key the transport entity shall discard the TPDU.

NOTE – The reception of a DT or ED TPDU with a duplicate sequence number is a security relevant event; however further action hereon is outside the scope of this Recommendation | International Standard (e.g. such as filling audit reports).

The unique sequence number is the Transport sequence number used in classes 2, 3, and 4. It is recommended that extended sequence numbers be used to avoid rekeying.

## 6.4 Peer address check processing

### 6.4.1 Purpose

This procedure is to counter masquerade attacks and support data origin authentication.

### 6.4.2 Procedure

Upon receipt of a TPDU, the peer address associated with the cryptographic key shall be compared to the source address of the TPDU. If the addresses do not match, the SE TPDU shall be discarded.

NOTE – The reception of a SE TPDU with an invalid address is a security relevant event; however further action hereon is outside the scope of this international standard (e.g. such as filling audit reports).

For each type of key granularity there is a corresponding degree of peer address information that requires verification. When per end-system (Kg-esp) keying is used, the NSAP address of the peer transport entity is checked with the negotiated peer address. When per end-system and security level (Kg-esp-sr) keying is used, the security label of the SE TPDU is checked with the negotiated security level set, in addition to checking the NSAP address of the peer transport entity. Since the security label is not strictly speaking address information and may be used optionally with single level security associations, security label checking is done independently as discussed elsewhere (see 6.5, Security Labels for Security Associations).

When per connection (Kg-tc) keying is used, the procedure becomes a bit more complex since the transport connection identifiers (SRC-REF, DST-REF) conveyed within individual TPDUs must be verified, in addition to the NSAP address of the peer transport entity. The SRC-REF is checked against the remote transport reference number portion of the peer address security attribute and the DST-REF against the local reference for the connection. Note that a transport entity initiating a connection may not know the local reference used by its peer, and may be unable to verify the SRC-REF of an incoming CC TPDU. This situation occurs when the peer dynamically determines the local reference upon processing a CR TPDU, and no value is available for the key manager to convey at the time the security attributes are established. Providing that the DST-REF field of the CC TPDU is the same as the local reference for the connection, the TPDU may be accepted and the value of SRC-REF field retained.

## 6.5 Security labels for Security Associations

### 6.5.1 Purpose

Security labels are used to provide support for access control and to provide support for data separation based on sensitivity.

### 6.5.2 TPDUs and parameters used

The procedure makes use of the following TPDU and parameters:

- SE TPDU;
- SA-ID;
- LABEL.

### 6.5.3 Procedure

When a security association specifies use of an explicit security label on every TPDU, the label shall be sent in the LABEL field of the protected header of each SE TPDU. Upon receipt of a SE TPDU containing the LABEL parameter, the transport entity shall verify that the LABEL parameter falls within the set of acceptable security levels for the security association. If a SE TPDU is received with an improper LABEL, the TPDU shall be discarded.

NOTE – The reception of a SE TPDU that fails the label check is a security relevant event; however further action hereon is outside the scope of this Recommendation | International Standard (e.g. such as filling audit reports).

## 6.6 Connection release

If the connection-oriented service (Kg_tc) is in use, the key associated with a connection shall be deselected as part of the connection release procedure.

## 6.7 Key replacement

The key replacement procedure is used if the cryptoperiod of a key expires. When the connection-oriented service is in use (Kg_tc) it may also be used when the sequence number spaces have been exhausted (see 6.3.3.1).

Key replacement associates a new cryptographic key with ongoing transport connection(s). The new Security Association shall have attributes which are identical to the old Security Association except for the new key. If no such key exists, the Security Management entity shall be notified and the original cryptographic key shall not be used for transmission. After the key replacement procedure has been executed, the old cryptographic key shall be discarded. It is a security related event if no suitable new key exists, however further action hereon such as filing an audit report is considered to be a local matter.

NOTE – The new key should be available within the transport activity timer (for class 4) or the TWR time (class 3) otherwise the connection may be terminated by the transport protocol.

Following a key replacement, unacknowledged DT and ED TPDUs requiring retransmission shall be sent under the new key.

## 6.8 Unprotected TPDUs

The security policy may allow secure Transport connections and non-secure connections between communicating entities. The means by which this is achieved is a local matter.

On transmission if the SA-Attribute UNProt is true the TPDU is passed through unprotected without the addition of PCI processing under TLSP.

On reception if the SA-Attribute UNProt is true the received TPDU is passed through without any processing under TLSP procedures.

## 6.9 Protocol identification

If this protocol is used over a network connection, it shall be explicitly identified by the explicit identification procedures defined in ISO/IEC 11570. The identifying UN TPDU itself may be protected by the protocol specified in this Recommendation | International Standard. If the UN TPDU is unprotected and if it specifies this Recommendation | International Standard in conjunction with either ITU-T Rec. X.224 | ISO/IEC 8073 or ITU-T Rec. X.234 | ISO 8602, then the TPDUs will be protected after successful completion of network connection establishment in accordance with the SA attributes. If the UN TPDU is protected and if it specifies either ITU-T Rec. X.224 | ISO/IEC 8073 only or ITU-T Rec. X.234 | ISO 8602 only then the single specified protocol is used after successful completion of network connection establishment.

NOTES

1 Whether or not unprotected communication is supported depends on the SA attributes.

2 If unprotected TCs are supported, whether or not they are multiplexed with protected TCs over the same NC depends on SA attributes and the security policy of the entity.

The explicit identification procedure defined in ISO/IEC 11570 is not used if ITU-T Rec. X.224 | ISO/IEC 8073 (Class 4) is being run over the OSI Connectionless Network Service as defined in ISO 8348.

## 6.10 Security Association-Protocol

A Security Association-Protocol (SA-P) is carried out by exchange of a SA PDUs and is designed to enable the establishment and customization of a SA.

The precise fields used within the SA PDU used for exchange of security information depend on the specific mechanism to be used to provide the SA. Whichever mechanism is used for SA-P, it shall provide the following:

a) derivation of all SA attributes required for the selected form of protection;

b) authentication of derived keys;

c) establishment of initial information for the purpose of authentication, and integrity if required;

d) rekey;

e) release of the security association.

A symmetric or asymmetric algorithm can be used for this function. It is recommended that an asymmetric algorithm be used. Appendix B contains an example of such a mechanism.

During that part of SA establishment that requires exchange of information in an unprotected form SA-PDUs shall be used. Exchanges of protected information as required for SA establishment can be carried in either SA-PDUs or SE-TPDUs.

Immediately following receipt of the final SA PDU in the SA-P protocol, if a TPDU is awaiting security encapsulation it is processed and transmitted.

NOTE – It is required that the last SA PDU with security control information sets the flag from responder to initiator in the SA-P. If necessary a SA PDU with Local SA-ID and Peer SA-ID as the only content may be sent.

If the expected sequence of PDUs does not occur within a specified timeout a SA PDU used for security control information (SCI) exchange may be repeated any number of times. Receipt of an SA PDUs with SCI previously received will result in the SA PDUs previously sent in response being resent. SA PDUs with SCI that are out of the expected sequence shall be ignored.

A TLSP entity may abort an SA establishment procedure and ignore subsequent SA PDUs with SCI if any check fails.

# 7 Use of elements of procedure

Table 1 gives an overview of which elements of procedure are included in each class of ITU-T Rec. X.224 | ISO/IEC 8073 and in ITU-T Rec. X.234 | ISO 8602.

**Table 1 – TLSP elements of procedure**

| Protocol mechanism | Reference (subclause) | ISO/IEC 8073, Class ITU-T X.224 | | | | | ISO 8602 ITU-T X.234 |
|---|---|---|---|---|---|---|---|
| Cryptographic Confidentiality | 6.2 | m | m | m | m | m | m |
| ICV Processing | 6.3.1 | m | m | m | m | m | m |
| Direction Indicator Processing | 6.3.2 | * | * | * | * | * | * |
| Unique Sequence Nos. | 6.3.3.1 | NA | NA | o | o | o | NA |
| Peer Address Check Processing | 6.4 | * | * | * | * | * | * |
| Security Labels for Cryptographic Assoc. | 6.5 | o | o | o | o | o | o |
| Connection Release | 6.6 | o | o | o | o | o | NA |
| Key Replacement | 6.7 | o | o | o | o | o | o |

* Procedure always included in class.

NA Not applicable.

o Negotiable procedure whose implementation in equipment is optional.

m Negotiable procedure whose implementation in equipment is mandatory.

NOTE – All negotiation is either outside the scope of this Recommendation | International Standard or via the SA-P procedure in 6.10 which allows this negotiation to be done at any time before the connection as part of TLSP.

# 8 Structure and encoding of TPDUs

## 8.1 Structure of TPDU

The structure of the TPDU, or concatenated TPDUs, before encapsulation (i.e. placed in “protected data” field of a TPDU, see 8.2) is as defined in 13.2 of ITU-T Rec. X.224 | ISO/IEC 8073.

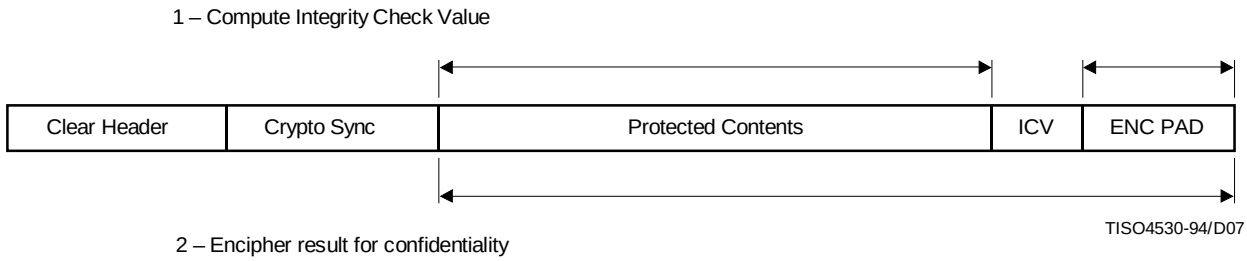
## 8.2 Security encapsulation TPDU

All the transport protocol data units (SE TPDUs) shall contain an integral number of octets. The octets in a SE TPDU are numbered starting from 1 and increasing in the order they are put into an NSDU. The bits in an octet are numbered from 1 to 8, where bit 1 is the low-order bit.

When consecutive octets within the SE TPDU are used to represent a binary number, the lower octet number has the most significant value.

For each fixed length field of the SE TPDU, the number of octets for the field is listed below the field in the following figures.

The structure of the TPDU shall be as follows:




a) This field is dependant on whether the encipherment algorithm selected requires an independent encipherment pad.

**Figure 7 – Structure of the TPDU**

### 8.2.1 Clear header

See Figure 8.

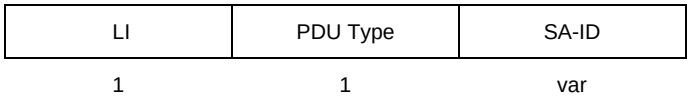



**Figure 8 – Format of the clear header**

#### 8.2.1.1 PDU clear header length

The PDU Clear Header Length indicator field (LI) contains the length PDU Type and SA-ID in octets, excluding the length indicator field itself.

#### 8.2.1.2 PDU type

This field contains the PDU TYPE code. It is used to define the structure of the remaining header. The value of the PDU TYPE code is: 0100 1000.

#### 8.2.1.3 SA-ID

The Security Association identifier field (SA-ID) contains the remote identifier of the cryptographic key used to protect the TPDU.

### 8.2.2 Crypto sync

This is an optional field which may contain synchronization data for specific encipherment algorithm identifier contained in the Security Association attributes.

NOTE – The size of the field would be known by the participating entities and part of the Security Association attributes.

### 8.2.3 Protected contents

Figure 9 shows the protected contents format for the Secure PDU.

| Content Length | Flag/type | Label | Protected Data | INT PAD |
|---|---|---|---|---|
| 1-3 | 1 | (tlv) | (tlv) | (tlv)[a)] |

a) May be a single octet pad.

**Figure 9 – Protected contents**

#### 8.2.3.1 Structure of protected contents field

The protected content fields are type, length, and value (tlv) encoded.

The content Field Type has the following allocation.

| *Value* | *Content Field Type* |
|---|---|
| 00-7F | Reserved for Private Use |
| 80-BF | Reserved |
| C0 | Protected Data |
| C1-C5 | Reserved |
| C6 | Label |
| C7-CF | Reserved |
| D0 | Reserved |
| D1 | Single Octet Pad |
| D2 | Reserved |
| D3 | Integrity Pad |
| D4 | Encipherment Pad |
| D5-FF | Reserved for future use |

If a two octet pad field is required for either Integrity or Encipherment PAD, the length field shall have the value 0 with the appropriate content field type.

The content field length contains the length of the content field value in octets. The content field length can be one, two or three octets long.

a) If one octet long then bit 8 is 0 and the remaining 7 bits define a value length up to 127 octets.

b) If two octets long then the first octet is encoded as 1000 0001 and the remaining octets defines the fields length up to 255 octets.

c) If three octets long then the first octet is encoded as 1000 0010 and the remaining two octets define the field length up to 65, 535 octets.

Other values of the first octet are reserved for future use.

#### 8.2.3.2 Content length

The length field contains the length of the Protected Contents in octets, excluding the content length field (i.e. flag, label, protected data, and icv pad). It has a maximum value of 65 535 ($2^{16}-1$).

#### 8.2.3.3 Flags

See Figure 10.

| Value |
|---|
| 1 |

**Figure 10 – Flags field**

The currently defined bits in this field are:

– *bit 1 direction indicator*

0 = responder to initiator;

1 = initiator to responder;

– *bit 4 Outward/Response*

0 = outward;

1 = response.

bits 2 to 3 and 5 to 8 Unused flag bits are set to zero on transmission.

#### 8.2.3.4 Label

See Figure 11.

| C6 Hex | Label Length | Def Auth Length | Defining Authority | Value |
|---|---|---|---|---|
| 1 | 1-3 | 1-3 | var | var |

**Figure 11 – Format of the label field**

The format of the Value field is defined by the Defining Authority.

NOTE – It is expected that these labels will be registered under procedures defined by ISO and CCITT. A Defining Authority will be registered as an Object Identifier value in accordance with ISO 8824, encoded in accordance with ISO 8825, and using procedures defined in ISO/IEC 9834.

#### 8.2.3.5 Protected data

The data field contains a TPDU or concatenated set of TPDUs as per ITU-T Rec. X.224 | ISO/IEC 8073 or ITU-T Rec. X.234 | ISO 8602 (see Figure 12).

| C0 Hex | Length | Protected data |
|---|---|---|
| 1 | var | var |

**Figure 12 – Format of the protected data field**

#### 8.2.3.6 Integrity PAD

The Value field contains arbitrary data required for integrity mechanisms.

The length of padding is defined by

a) The padding required by the integrity mechanism.

The integrity mechanism being used has known characteristics which will include block length defined for the use in the security association (if mechanism used in block mode). The starting point of the integrity process to the end of the integrity pad must be an integral multiple of the block length.

b) The padding required by the block encipherment mechanism to bring the end of the ICV up to the end of the block size, if a separate encipherment pad is not required.

The choice of padding value is a local matter. If a single octet pad is required a Single Octet Pad (Type = D1 – without length or value) is used instead of the Integrity Pad.

### 8.2.4 ICV

The ICV field contains the Integrity Check Value. The length of this field is implied by the ICV algorithm identifier contained in the Security Association attributes.

### 8.2.5 Encipherment PAD

The block size for encipherment is a known characteristic of the encipherment algorithm. The starting point of the confidentiality process to the end of the ICV must be an integral multiple of the block length. The presence of the encipherment pad following the ICV is dependent on whether the encipherment algorithm selected requires an independent encipherment pad.

The choice of padding value is a local matter. If a single octet pad is required a Single Octet Pad (Type = D1 – without length or value) is used instead of the Encipherment Pad.

## 8.3 Security Association PDU

The format of the SA PDU is shown in Figure 13.

| LI | PDU Type | SA-ID | SA-P-Type | SA PDU Contents |
|---|---|---|---|---|
| 1 | 1 | var | lv | var |

**Figure 13 – SA PDU Structure**

### 8.3.1 LI

This field contains the length of the PDU type field plus the SA-ID. If the SA-ID needs to signal that it does not know its peer's SA-ID (for example on establishing a new SA), the length field shall be set such that the SA-ID field is not present (that is, value 1).

### 8.3.2 PDU Type

This field contains the PDU type value of 0100 1001 to indicate a Security Association PDU.

### 8.3.3 SA-ID

The SA-ID field contains the Security Association Identifier of the recipient (that is, the SA attribute Peer_SAID). This field is not required when the SA-P is being used to establish a new SA (that is, the recipient has not yet assigned an SA-ID).

### 8.3.4 SA-P Type

This field contains an object identifier indicating the set mechanisms used to provide the SA Protocol as below.

The object identifier assigned for Exponential Key Exchange (as defined in Annex D is joint-ccitt-iso (2) tlsp (21) sa-p-kte (1) eke (1).

Use of other algorithms with the SA-P may be indicated by further object identifiers allocated in accordance with ISO 9834-1 (Registration Procedures).

### 8.3.5 SA PDU Contents

The internal structure of this field is dependent on the mechanism providing the SA Protocol as specified in 8.3.4 above. Annex B defines one such SA-Protocol which uses token key exchange and digital signature mechanisms.

# 9 Conformance

## 9.1 General

A Protocol Implementation Conformance Statement (PICS) shall be completed with respect to any claim for conformance of an implementation to this Recommendation | International Standard. The PICS shall be produced in accordance with the relevant PICS proforma.

## 9.2 Common static conformance requirements

a) A conformant implementation shall support at least TLSP with either ITU-T Rec. X.224 | ISO/IEC 8073 or ITU-T Rec. X.234 | ISO 8602.

b) A conformant implementation shall support implementation in an end system.

c) Each system claiming conformance to TLSP shall be capable of encapsulation and extraction of Userdata within a Secure Data Transfer PDU.

d) Each system claiming to provide confidentiality security services shall support at least the encipherment mechanism.

e) Each system claiming to provide integrity security services shall support at least the ICV mechanism.

### 9.3 TLSP with ITU-T Rec. X.234 | ISO 8602 static conformance requirements

Each system claiming conformance to the TLSP protocol shall provide at least one of the following security services:

a) Connectionless confidentiality;

b) Connectionless integrity.

### 9.4 TLSP with ITU-T Rec. X.224 | ISO/IEC 8073 static conformance requirements

Each system claiming conformance to TLSP shall provide at least one of the following security services:

a) Connection confidentiality

b) Connection integrity without recovery

c) Peer entity authentication.

### 9.5 Common dynamic conformance requirements

Each system claiming to be conformant to this Recommendation | International Standard shall have the following behaviour:

a) Detection of all mandatory and optional fields within a Secure Data Transfer PDU in a sequence.

b) Unrecognized fields within a Secure Data Transfer PDU shall be treated as an error as described in clause 6.

### 9.6 TLSP with ITU-T Rec. X.234 | ISO 8602 dynamic conformance requirements

Each system claiming to be conformant to the TLSP protocol shall have the following behaviour:

– When data origin authentication is provided then either the encipherment mechanism or a cryptographic ICV mechanism shall be invoked.

### 9.7 TLSP with ITU-T Rec. X.224 | ISO/IEC 8073 dynamic conformance requirements

Each system claiming to be conformant to the TLSP protocol shall have the following behaviour:

– When peer entity or data origin authentication is provided then either the encipherment mechanism or a cryptographic ICV mechanism shall be invoked.

## 10 Protocol implementation conformance statement (PICS)

The supplier of a protocol implementation which is claimed to conform to this Recommendation | International Standard shall complete a copy of the PICS proforma provided in Annex A, including the information necessary to identify fully both the supplier and the implementation.

# Annex A

# PICS proforma[a)]

(This annex forms an integral part of this Recommendation | International Standard)

## A.1 Introduction

### A.1.1 Background

The supplier of a protocol implementation which is claimed to conform to Recommendation | International Standard 10736 shall complete the Transport Layer Security Protocol (TLSP), Protocol Implementation Conformance Statement (PICS) proforma. A completed PICS proforma becomes the PICS for the implementation in question. The PICS is a statement identifying the capabilities and options of the protocol that have been implemented. The PICS can have a number of uses, including:

- use by the protocol implementer, as a check list to reduce the risk of failure to conform to the standard through oversight;
- use by the supplier and receiver of the implementation, as a detailed indication of its capabilities, stated relative to the common basis of understanding provided by the standard PICS proforma;
- use by the user of the implementation, as a basis for checking the possibility of interworking with another implementation;
- use by a protocol tester, as the basis for selecting appropriate tests against which to assess the claim for conformance of the implementation.

### A.1.2 Approach

The first part of the PICS proforma, the Implementation Identification and Protocol Summary, is to be completed as indicated with the information necessary to identify fully both the supplier and the implementation. The main part of the PICS proforma is a fixed-format questionnaire divided into subclauses, each containing a group of individual items. Answers to the questionnaire items are to be provided in the rightmost column, either by simply marking an answer to indicate a restricted choice (usually “Yes” or “No”), or by entering a value or a set or range of values. Note that there are some items where two or more choices from a set of possible answers can apply. Therefore, all relevant choices are to be marked.

Each item is identified by an reference index in the first column; the second column contains the item to be addressed; the third column contains the reference(s) to the location of the item in the main body of the standard. For optional items, additional columns indicate the status of the item (i.e. whether support is mandatory, optional, or conditional), and provide space or a choice or items for the implementation support response.

The following status column notations described in ISO/IEC JTC1/ SC6 N6233, Catalogue of PICS Proforma Notations, are used for this PICS proforma:

| *Symbol* | *Meaning* |
|---|---|
| m | Mandatory |
| o | Optional |
| – | Not applicable (N/A) |
| o.<n> | Optional, but support of at least one of the group of options labelled by the same numeral <n> is required |
| <cid>: | Conditional requirement, according to the condition or item index identified by <cid> |
| <item>:: | Simple predicate condition, dependent on the support marked for <item> |

---

a) Copyright release for PICS proforma

Users of this Recommendation | International Standard may freely reproduce the PICS proforma in this annex so that it can be used for the intended purpose and may further publish the completed PICS.

## A.2 Implementation identification

See Table A.1.

**Table A.1 – TLSP Implementation Identification**

| Item | Information |
|---|---|
| Supplier | |
| Contact point for queries about this PICS | |
| Implementation Name(s) andVersion(s) | |
| Other information necessary for full identification [e.g. Names and Version(s) for machines and operating systems, System Name(s)] | |

NOTES

1 Only the first three items are required for each implementation. Other information may be completed as appropriate in meeting the requirements for full identification.

2 The terms “Name” and “Version” should be interpreted appropriately to correspond with a supplier’s terminology (e.g. using Type, Series, Model).

## A.3 General statement of conformance

Table A.2 codifies the general statement of conformance for the implementation.

**Table A.2 – General Conformance Statement**

| Index Item | | Support | |
|---|---|---|---|
| SP | Does the implementation claim conformance with ISO/IEC 10736? | Y | N |
| SPMAN | Are all mandatory features of ISO/IEC 10736 implemented? | Y | N |

## A.4 Protocol implementation

Table A.3 identifies common abbreviations used in this PICs, these same abbreviations are found in ITU-T Rec. X.224 | ISO/IEC 8073, but are identified here for help in conforming to this PICS.

**Table A.3 – CO and CL Transport Implemented**

| Index Transport Class Network Service | |
|---|---|
| C0 | Class 0 over cons |
| C1 | Class 1 over cons |
| C2 | Class 2 over cons |
| C3 | Class 3 over cons |
| C4 | Class 4 over cons |
| C4L | Class 4 over clns |
| CLTP | Connectionless transport protocol |

## A.5 Security services supported

Tables A.4 to A.7 identify for each Class of Transport (COTP::), the security services available through the TLSP and their level of support within the implementation. The security services listed are taken from CCITT Rec. X.800 | ISO 7498-2.

**Table A.4 – Service Element Proforma for C0**

| Index Service Element | Status | Support | |
|---|---|---|---|
| TOSE0 Confidentiality | o.1 | Y | N |
| TOSE1 Connection Confidentiality | TOSE0:m | Y | N |
| TOSE2 Connectionless Confidentiality | – | | |
| TOSE3 Integrity | o.1 | Y | N |
| TOSE4 Connection Integrity w Recovery | – | | |
| TOSE5 Connection Integrity wo Recovery | | Y | N |
| TOSE6 Connectionless Integrity | TOSE3:m | | |
| TOSE7 Peer Entity Authentication | o | Y | N |
| TOSE8 Access Control | o | Y | N |
| TOSE9 IN BAND SA-P | o | Y | N |

**Table A.5 – Service Element Proforma for C1, C2, C3**

| Index Service Element | Status | Support | |
|---|---|---|---|
| T3SE0 Confidentiality | o.1 | Y | N |
| T3SE1 Connection Confidentiality | T3SE0:m | Y | N |
| T3SE2 Connectionless Confidentiality | – | | |
| T3SE3 Integrity | o.1 | Y | N |
| T3SE4 Connection Integrity w Recovery | – | | |
| T3SE5 Connection Integrity wo Recovery | T3SE3:o.2 | Y | N |
| T3SE6 Connectionless Integrity | T3SE3:o.2 | Y | N |
| T3SE7 Peer Entity Authentication | o | Y | N |
| T3SE8 Access Control | o | Y | N |

**Table A.6 – Service Element Proforma for C4**

| Index Service Element | Status | Support | |
|---|---|---|---|
| T4SE0 Confidentiality | o.1 | Y | N |
| T4SE1 Connection Confidentiality | T4SE0:m | Y | N |
| T4SE2 Connectionless Confidentiality | – | | |
| T4SE3 Integrity | o.1 | Y | N |
| T4SE4 Connection Integrity w Recovery | T4SE3:o.2 | Y | N |
| T4SE5 Connection Integrity wo Recovery | – | | |
| T4SE6 Connectionless Integrity | T4SE3:o.2 | Y | N |
| T4SE7 Peer Entity Authentication | o | Y | N |
| T4SE8 Access Control | o | Y | N |

**Table A.7 – Service Element Proforma for C4L**

| Index Service Element | Status | Support | |
|---|---|---|---|
| TLSE0 Confidentiality | o.1 | Y | N |
| TLSE2 Connectionless Confidentiality | TLSE0:m | Y | N |
| TLSE1 Connection Confidentiality | – | | |
| TLSE3 Integrity | o.1 | Y | N |
| TLSE4 Connection Integrity w Recovery | TLSE3:o.2 | | |
| TLSE5 Connection Integrity wo Recovery | – | | |
| TLSE6 Connectionless Integrity | TLSE3:o.2 | Y | N |
| TLSE7 Peer Entity Authentication | o | Y | N |
| TLSE8 Access Control | o | Y | N |

Table A.8 identifies for connectionless Transport (CLTP::), the security services available through the TLSP and their level of support within the implementation.

**Table A.8 – Service Element Proforma for CLTP**

| Index Service Element | Status | Support | |
|---|---|---|---|
| TCSE0 Confidentiality | o.1 | Y | N |
| TCSE1 Connection Confidentiality | – | | |
| TCSE2 Connectionless Confidentiality | TCSE0:m | Y | N |
| TCSE3 Integrity | o.1 | Y | N |
| TCSE4 Connection Integrity w Recovery | – | | |
| TCSE5 Connection Integrity wo Recovery | – | | |
| TCSE6 Connectionless Integrity | TCSE3:m | Y | N |
| TCSE7 Data Origination Authentication | o | Y | N |
| TCSE8 Access Control | o | Y | N |

## A.6 Supported functions

Tables A.9 to A.16 identify the mandatory and optional functions implemented for each class of Transport (COTP::) supported.

**Table A.9 – Mandatory Functions for C0**

| Index | Function | Reference (subclause) | Status | Support | |
|---|---|---|---|---|---|
| T0SF1 | Verification of peer address | 5.6.2, 6.4 | m | Y | |
| T0SF2 | Reflection detection | 5.6.2, 6.3.2 | m | Y | |
| T0SF3 | Security encapsulation | 5.6 | m | Y | |
| T0SF4 | Reporting of security events | Notes | m | Y | |

**Table A.10 – Optional Functions for C0**

| Index | Function | Reference (subclause) | Status | Support | |
|---|---|---|---|---|---|
| T0SF5 | Data encipherment | 6.2 | o.1 | Y | N |
| T0SF6 | Integrity protection | 6.3 | o.1 | Y | N |
| T0SF7 | Integrity padding | 6.3.1.3 | o | Y | N |
| T0SF8 | Explicit security labeling | 6.5 | o | Y | N |
| T0SF9 | Encipherment padding | 6.2.2 | o | Y | N |

**Table A.11 – Mandatory Functions for C1**

| Index | Function | Reference (subclause) | Status | Support | |
|---|---|---|---|---|---|
| T1SF1 | Verification of peer address | 5.6.2, 6.4 | m | Y | |
| T1SF2 | Reflection detection | 5.6.2, 6.3.2 | m | Y | |
| T1SF3 | Separation after decapsulation | 6.1 | m | Y | |
| T1SF4 | Security encapsulation | 5.6 | m | Y | |
| T1SF5 | Reporting of security events | Notes | m | Y | |

**Table A.12 – Optional Functions for C1**

| Index | Function | Reference (subclause) | Status | Support | |
|---|---|---|---|---|---|
| T1SF6 | Data encipherment | 6.2 | o.1 | Y | N |
| T1SF7 | Integrity protection | 6.3 | o.1 | Y | N |
| T1SF8 | Pre-encapsulation concatenation | 6.1 | o | Y | N |
| T1SF9 | Integrity padding | 6.3.1.3 | o | Y | N |
| T1SF10 | Explicit security labeling | 6.5 | o | Y | N |
| T1SF11 | Encipherment padding | 6.2.2 | o | Y | N |

**Table A.13 – Mandatory Fuctions for C2, C3**

| Index | Function | Reference (subclause) | Status | Support | |
|---|---|---|---|---|---|
| T3SF1 | Verification of peer address | 5.6.2, 6.4 | m | Y | |
| T3SF2 | Reflection detection | 5.6.2, 6.3.2 | m | Y | |
| T3SF3 | Separation after decapsulation | 6.1 | m | Y | |
| T3SF4 | Secure multiplexing | Implicit | m | Y | |
| T3SF5 | Security encapsulation | 5.6 | m | Y | |
| T3SF6 | Reporting of security events | Notes | m | Y | |

**Table A.14 – Optional Functions for C2, C3**

| Index | Function | Reference (subclause) | Status | Support | |
|---|---|---|---|---|---|
| T3SF7 | Data encipherment | 6.2 | o.1 | Y | N |
| T3SF8 | Integrity protection | 6.3 | o.1 | Y | N |
| T3SF9 | Integrity sequence number space | 6.3.3 | o | Y | N |
| T3SF10 | Pre-encapsulation concatenation | 6.1 | o | Y | N |
| T3SF11 | Integrity padding | 6.3.1.3 | o | Y | N |
| T3SF12 | Explicit security labeling | 6.5 | o | Y | N |
| T3SF13 | Encipherment padding | 6.2.2 | o | Y | N |

**Table A.15 – Mandatory Functions for C4, C4L**

| Index | Function | Reference (subclause) | Status | Support | |
|---|---|---|---|---|---|
| T4SF1 | Verification of peer address | 5.6.2, 6.4 | m | Y | |
| T4SF2 | Reflection detection | 5.6.2, 6.3.2 | m | Y | |
| T4SF3 | Separation after decapsulation | 6.1 | m | Y | |
| T4SF4 | Secure multiplexing | Implicit | m | Y | |
| T4SF5 | Security encapsulation | 5.6 | m | Y | |
| T4SF6 | Reporting of security events | Notes | m | Y | |

**Table A.16 – Optional Functions for C4, C4L**

| Index | Function | Reference (subclause) | Status | Support | |
|---|---|---|---|---|---|
| T4SF7 | Data encipherment | 6.2 | o.1 | Y | N |
| T4SF8 | Integrity protection | 6.3 | o.1 | Y | N |
| T4SF9 | Integrity sequence number | 6.3.3 | o | Y | N |
| T4SF10 | Pre-encapsulation concatenation | 6.1 | o | Y | N |
| T4SF11 | Integrity padding | 6.3.1.3 | o | Y | N |
| T4SF12 | Explicit security labeling | 6.5 | o | Y | N |
| T4SF13 | Encipherment padding | 6.2.2 | o | Y | N |

Tables A.17 and A.18 identify the mandatory and optional functions implemented for connectionless Transport (CLTP::).

**Table A.17 – Mandatory Functions for CLTP**

| Index | Function | Reference (subclause) | Status | Support | |
|---|---|---|---|---|---|
| TLF1 | Verification of peer address | 5.6.2, 6.4 | m | Y | |
| TLF2 | Reflection detection | 5.6.2, 6.3.2 | m | Y | |
| TLF3 | Security encapsulation | 5.6 | m | Y | |
| TLF4 | Reporting of security events | 5.2.1, 6 | m | Y | |

**Table A.18 – Optional Functions for CLTP**

| Index | Function | Reference (subclause) | Status | Support | |
|---|---|---|---|---|---|
| TLF5 | Data encipherment | 6.2 | o.1 | Y | N |
| TLF6 | Integrity protection | 6.3 | o.1 | Y | N |
| TLF7 | Integrity padding | 6.3.1.3 | o | Y | N |
| TLF8 | Explicit security labeling | 6.5 | o | Y | N |
| TLF9 | Encipherment padding | 6.2.2 | o | Y | N |

## A.7 Supported Protocol Data Units (PDUs)

### A.7.1 Supported Transport PDUs (TPDUs)

As indicated in Table A.19 the SE TPDU is supported for both transmission and receipt, for both the connection oriented (COTP::) and connectionless Transport Protocol (CLTP::).

**Table A.19 – TPDUs Supported**

| Index | TPDU | Item | Status | Support | |
|---|---|---|---|---|---|
| STS1 | SE | Transmission COTP or CLTP | m | Y | |
| STS2 | SE | Receipt COTP or CLTP | m | Y | |

### A.7.2 Supported parameters of issued TPDUs

Tables A.20 and A.21 indicate which parameters are mandatory or optional when a SE TPDU is issued by Transport (COTP:: or CLTP::).

**Table A.20 – Mandatory Parameters for COTP, CLTP**

| Index | Parameter | Reference (subclause) | Status | Support | |
|---|---|---|---|---|---|
| SPI1 | Key Identifier must be present. | 6.2, 6.3 | m | Y | |
| SPI2 | Bit one of Protected Header Flag must be set as direction indicator. | 8.2.3.3 | m | Y | |

**Table A.21 – Optional Parameters for COTP, CLTP**

| Index | Parameter | Reference (subclause) | Status | Support | |
|---|---|---|---|---|---|
| SPI3 | Label | 8.2.3.4 | o | Y | N |
| SPI4 | Integrity Pad | 8.2.3.6 | o | Y | N |
| SPI5 | ICV | 8.2.4 | o | Y | N |
| SPI6 | Encipherment Pad | 8.2.5 | o | Y | N |

### A.7.3 Supported parameters of received TPDUs

Implementations shall be capable of receiving and processing all possible parameters of the SE TPDU as indicated in Table A.22.

**Table A.22 – Mandatory parameters for COTP, CLTP**

| Index | Parameter | Reference (subclause) | Status | Support | |
|---|---|---|---|---|---|
| SPR1 | Key Identifier must be present. | 6.2, 6.3 | m | Y | |
| SPR2 | Bit one of Protected Header Flag | 8.2.3.3 | m | Y | |
| SPR3 | Label | 8.2.3.4 | m | Y | |
| SPR4 | Integrity Pad | 8.2.3.6 | m | Y | |
| SPR5 | ICV | 8.2.4 | m | Y | |
| SPR6 | Encipherment Pad | 8.2.5 | m | Y | |

Allowed values of issued TPDU parameters are given in Table A.23.

**Table A.23 – Values for Parameters of issued TPDUs for COTP, CLTP**

| Index | Parameter | Values | |
|---|---|---|---|
| | | Allowed | Supported |
| AVI1 | SA-ID | 2-126 octets | |
| AVI2 | Prot Header Flags | 0 or 1 | |
| | Label | | |
| AVI3 | Defining Authority | 1-n octets | |
| AVI4 | Value | 1-m octets | |
| | ICV Padding | | |
| AVI5 | Length | 1-254 | |
| AVI6 | Value | 1-254 octets | |
| AVI7 | | ICV 1-indef octets | |
| | ENC PADDING | | |
| AVI8 | Length | 1-254 | |
| AVI9 | Value | 1-254 octets | |

### A.7.4 Allowed values of issued TPDU parameters

See Table A.24.

**Table A.24 – Values for parameters of received TPDUs for COTP, CLTP**

| Index | Parameter | Values | |
|---|---|---|---|
| | | Allowed | Supported |
| AVR1 | SA-ID | 2-126 octets | |
| AVR2 | Prot Header Flags | 0 or 1 | |
| | Label | | |
| AVR3 | Defining Authority | 1-n octets | |
| AVR4 | Value | 1-m octets | |
| | ICV Padding | | |
| AVR5 | Length | 1-254 | |
| AVR6 | Value | 1-254 octets | |
| AVR7 | ICV | 1-indef octets | |
| | ENC PADDING | | |
| AVR8 | Length | 1-254 | |
| AVR9 | Value | 1-254 octets | |

## A.8 Service, function, and protocol relationships

### A.8.1 Relationship between services and functions

Table A.25 gives a mapping between OSI security services provided by TLSP and the associated functions needed in an implementation. The consistency between supported functions and security services shall be maintained accordingly.

**Table A.25 – Mapping of security services to supported functions**

| Security Service | Functions |
|---|---|
| Confidentiality | Data encipherment padding |
| Connection Integrity | Integrity sequence number space<br>Integrity protection<br>Reflection detection padding |
| Connectionless Integrity | Integrity protection<br>Reflection detection padding |
| Peer Entity or | Verification of peer address |
| Data Orig. Authentication | Security encapsulation<br>Use of either:<br>integrity protection or data encipherment |
| Access Control | Explicit security labeling<br>Secure meltiplexing<br>Security encapsulation |

### A.8.2 Relationship between services and protocol

Table A.26 gives a mapping between OSI security services provided by TLSP and the SE TPDU protocol control information (PCI) and parameter fields employed by the underlying security mechanisms. The consistency between supported security parameters and SE TPDU parameter fields shall be maintained accordingly.

**Table A.26 – Mapping of security services to SE TPDU parameters**

| Security Service | TPDU Parameters/PCI |
|---|---|
| Confidentiality | Encrypted data<br>Confidentiality padding |
| Connectionless Integrity | Integrity check value<br>Direction indicator<br>Integrity padding |
| Connection Integrity | Integrity check value<br>Direction indicator<br>Integrity padding<br>DT/ED send sequence number<br>(final sequence number) |
| Data Orig. Authentication | Peer address |
| Peer Entity Authentication | Key identifier<br>Key identifier employed in:<br>integrity check value or encrypted data |
| Access Control | Security labels<br>Key identifier<br>Key identifier employed in:<br>integrity check value or encrypted data |

## A.9 Supported algorithms

Table A.27 identifies the set of confidentiality and integrity algorithms supported by this implementation.

**Table A.27 – Supported algorithms**

| Index | Item | Reference (subclause) | Algorithm Identifier [a)] |
|---|---|---|---|
| ALG1 | Data Encryption | 6.2.3 | |
| ALG2 | Cryptographichic ICV | 6.3.1.3 | |
| ALG3 | Non-Cryptographic ICV | 6.3.1.3 | |
| a) Algorithms supported (if appropriate) under the registration scheme defined in ISO/IEC 9979 or ISO/IEC 9834. | | | |

## A.10 Error handling

### A.10.1 Security errors

Table A.28 contains the mandatory security error actions to be taken upon receipt of an SE TPDU corresponding to the event description.

**Table A.28 – Mandatory security error actions for COTP, CLTP**

| Index | Event | Reference (subclause) |
|---|---|---|
| SEA1 | An improperly protected TPDU received shall be discarded. | 6.0 |
| SEA2 | A TPDU with an invalid value in the SA-ID identified shall be discarded. | 6.2.3 |
| SEA3 | A TPDU with an invalid ICV shall be discarded. | 6.3.1.3 |
| SEA4 | A TPDU with an invalid direction indicator shall be discarded. | 6.3.2.3 |
| SEA5 | A TPDU with an improper label shall be discarded. | 6.5.3 |
| SEA6 | A TPDU with an improper Integrity Pad shall be discarded. | 6.3.1.3 |
| SEA7 | A TPDU with a duplicate sequence number shall be discarded. | 6.3.3.3 |
| SEA8 | A TPDU with an invalid peer address shall be discarded. | 6.4 |
| SEA9 | A TPDU with an improper Encipherment Pad shall be discarded. | 6.2.2 |

NOTES

1 In item SEA1, an improperly protected TPDU includes both those SE TPDUs where non-negotiated options are used, and those where negotiated options are not used.

2 Item SEA7 apply only to the connection oriented Transport Protocol (COTP::) when integrity sequence number space and truncation protection have been negotiated for C2-C4, C4L.

### A.10.2 Protocol errors

Table A.29 identifies the protocol error actions to be taken upon receipt of an SE TPDU corresponding to the event description.

**Table A.29 – Protocol error actions for COTP, CLTP**

| Index | Event | Reference (subclause) | Action | |
|---|---|---|---|---|
| | | | Allowed | Supported |
| PEA1 | An undefined parameter encountered in the protected contents. | 8.2.3 | | |
| PEA2 | Out of sequence parameters discovered in the protected contents. | 8.2.3 | | |

## A.11 Security Association

### A.11.1 SA Generic Fields

See Table A.30.

**Table A.30**

| Item | Questions/Features | Reference (subclause) | Status | Support on transmission | Support on receipt |
|---|---|---|---|---|---|
| SaLI | Length Indicator field tansmitted in each SA PDU? | 8.3.1 | SA:M | Yes<br>N/A | Yes<br>N/A |
| SaPDUType | PDU Type field with value 01001001 in each SA PDU | 8.3.2 | SA:M | Yes<br>N/A | Yes<br>N/A |
| SaSAID | SA-ID field | 8.3.3 | SA:M | Yes<br>N/A | Yes<br>N/A |
| SA-PType | SA-P TYPE field | 8.3.4 | SA:M | Yes<br>N/A | Yes<br>N/A |
| SA-RK | Is the SA REKEY Supported? | B.5.3 | SA:O | Yes<br>No<br>N/A | Yes<br>No<br>N/A |
| SSLYR* | Is the example SA protocol using Key Token Exchange supported? | Annex B | SA:O | Yes<br>No<br>N/A | Yes<br>No<br>N/A |

### A.11.2 Content Fields Specific to Key Exchange SA-P

See Table A.31.

**Table A.31**

| Item | Questions/Features | Reference (subclause) | Status | Support on Transmission | Support on receipt |
|---|---|---|---|---|---|
| SAExchId | ExchangeID | B.6.1 | SAKTE:M | Yes<br>N/A | Yes<br>N/A |
| ContLen | Is the Length Indicator field transmitted in each SA PDU? | B.6.2 | SAKTE:M | Yes<br>N/A | Yes<br>N/A |
| MySAID | My SAID Content field | B.6.3.1 | SAKTE:M | Yes<br>N/A | Yes<br>N/A |
| OldYrSAID | Old Your SAID Content field | B.6.3.2 | SAKTE:M | Yes<br>N/A | Yes<br>N/A |
| KeyTokens | Key Token 1 and Key Token 2 Content Fields | B.6.3.3 | SAKTE:M | Yes<br>N/A | Yes<br>N/A |
| AuthFields | Authentication digital signature and Authentication certificate Content fields | B.6.3.4 | SAKTE:M | Yes<br>N/A | Yes<br>N/A |
| ServSel | Service Selection Content field | B.6.3.5 | SAKTE:O | Yes<br>No<br>N/A | Yes<br>No<br>N/A |
| SARejReas | SA Rejection Reason Content field | B.6.3.6 | SAKTE:O | Yes<br>No<br>N/A | Yes<br>No<br>N/A |
| SAAbReas | SA Abort/Release Reason Content field | B.6.3.7 | SAKTE:M | Yes<br>No<br>N/A | Yes<br>No<br>N/A |
| LabDef | Label Definition Content field | B.6.3.8 | SAKTE:O | Yes<br>No<br>N/A | Yes<br>No<br>N/A |
| KeySel | Key Selection Content field | B.6.3.9 | SAKTE:O | Yes<br>No<br>N/A | Yes<br>No<br>N/A |
| KeyUse | Usage Flags sub-field | B.6.3.9. | KeySel:M | Yes<br>No<br>N/A | Yes<br>No<br>N/A |
| KeySelInfo | Key Selection Information sub-field | B.6.3.9. | KeySel:M | Yes<br>No<br>N/A | Yes<br>No<br>N/A |
| KeyRefx | Key Reference sub-field | B.6.3.9. | KeySel:O | Yes<br>No<br>N/A | Yes<br>No<br>N/A |
| SaFlags | SA Flags Content field | B.6.3.10 | SAKTE:O | Yes<br>No<br>N/A | Yes<br>No<br>N/A |
| ASSR | ASSR Content field | B.6.3.11 | ServSel:M | Yes<br>No<br>N/A | Yes<br>No<br>N/A |

# Annex B

## Security Association Protocol Using Key Token Exchange and Digital Signatures

(This annex forms an integral part of this Recommendation | International Standard)

### B.1 Overview

This annex defines a protocol for use of an asymmetric mechanism to perform the SA establishment, maintenance, and abort/release. It allows the communicating TLSP entities to:

a) authenticate the two entities to each other;

b) initialize SA attributes including keys; and

c) establish initial information for use in providing integrity.

This annex describes a SA-Protocol which logically performs the following distinct functions:

a) Key token exchange (KTE), is used to establish a shared secret. This mechanism supports an exchange of key tokens. The form of these tokens is mechanism specific. An example of mechanism specific key tokens, supporting exponential key exchange, also known as Diffie-Hellman exchange, is outlined in Annex C.

b) Certificates, digital signatures and elements from the KTE are used to achieve authentication.

c) Protocol exchanges are used to negotiate SA attributes.

d) Protocol exchanges to signal that the SA is being released.

Prior to establishing an SA using this SA-Protocol each TLSP entity must have pre-established the following information:

a) The mechanisms it supports as expressed by:

   1) a list of ASSRs supported; and

   2) the set of security services supported for each of the ASSRs identified above.

b) An asymmetric key pair for each asymmetric algorithm supported which can be used by the TLSP entity to sign data for authentication purposes.

c) A certificate from a trusted authority for each asymmetric algorithm supported which identifies the TLSP entity, and its public asymmetric key, for authentication purposes.

d) The public keys, and the implied asymmetric algorithms, of any trusted certification authorities which would issue certificates to TLSP entities which this TLSP entity will be communicating with.

This SA-Protocol dynamically establishes the following security information which it needs to secure its own communication:

a) negotiation of the encipherment algorithm to protect SA-Protocol communication;

b) negotiation of the asymmetric algorithm and digital signature scheme used to provide SA-Protocol authentication;

c) generation of keying information needed by the encipherment algorithm to protect SA-Protocol communication.

This SA-Protocol establishes the following shared information between two TLSP Entities:

a) local and remote SA-IDs;

b) security services to be used between the associated entities for instances of communication;

c) the mechanisms and their parameters as implied through the security services selected;

d) initial shared keys for integrity, encipherment mechanisms and authentication of an instance of communication;

e) the set of security labels that may be used on this association for access control.

An SA can be established using the same selected security services, mechanisms and their parameters and set of security labels from a previously established SA. In this case only the SA-ID and keys are changed, all other attributes shall remain the same.

Whenever a new SA is established new key values shall be established.

In the case of the connectionless mode TLSP, after an SA has been released, the SA-ID shall not be re-used. The period in which the SA-ID is frozen shall be greater than the maximum lifetime of a PDU in the underlying network.

The SA Attribute adr_served is established by means outside this protocol.

The SA Attribute Initiator is set to true for the initiator of the SA Protocol exchange and false for the responder.

The protocol exchanges for SA establishment are illustrated in Figure D.1.

## B.2 Key Token Exchange (KTE)

The TLSP entities start their SA-Protocol with a key token exchange (KTE) to generate a shared secret (i.e. bit string) between the entities. The TLSP entities then use a subset of this secret bit string in conjunction with a private key algorithm to encipher the remainder of the communications between them, thus providing confidentiality to the remainder of the SA-Protocol exchanges.

The KTE involves the exchange of two values, Key Token 1 and Key Token 2 calculated from mechanism specific parameters along with locally generated numbers using mechanism specific algorithms such as those outlined in Annex D. The exchanged values are then used by the communicating entities to generate the shared secret bit string.

A subset of this bit string is used in conjunction with a private key algorithm to encipher the remainder of the SA Protocol exchange supporting SA-Protocol Authentication and SA Attribute negotiation. In addition, subset of this bit string is also referenced to be used as key and ISN attributes of the security association being established. This is referenced either:

1) by exchanging position information in the SA Attribute Negotiation; or

2) via *a priori* knowledge.

## B.3 SA-Protocol Authentication

In order for an TLSP entity to authenticate another during SA establishment, it requires an authentication certificate and public key pair.

The TLSP entities exchange certificates and digital signatures (such as defined in ISO 9594-8) to verify each other's identity. A certificate contains, as a minimum, some identifying information for an TLSPE plus the entity's public key (see Figure D.1).

The certificate is certified by a trusted authority and provided to the TLSP using a procedure outside the scope of the TLSP protocol. The certificate carries the authentication signature of the trusted authority. An TLSP entity partaking in this SA-Protocol must have the public key of the trusted authority which issued the certificate. The method used to attain the trusted authority's public key is outside the scope of this standard. For an TLSP entity to demonstrate that it owns a particular certificate, it must prove that it knows the secret key corresponding to the public key in the certificate.

Proof of timeliness and prevention of replay attacks is addressed by the signed data consisting of the specific numbers jointly determined and specific to this protocol operation. This is done as follows for two communicating entities A (the initiator of the SA) and B (the responder):

a) The SA contents is created, including the TLV encoded fields carrying:

   A'Certificate,

   the SA Attribute negotiation (see B.4) or Abort/Release reasons (see B.6),

   Key Token 3 calculated using an algorithm such as described in Annex D,

   but excluding the exchange ID and the content length, are then signed (e.g. using the authentication signature defined in ISO 9594-8). The SA Contents, including the signature, encoded as a TLV, and the content length is then enciphered. The encipherment key is the first n bits of the bit string produced by the KTE exchange, where n is the number of bits required by the algorithm used.

b) The SA contents is then created, signed, and enciphered using equivalent information relating to B and Key Token 4 instead of Key Token 3.

Each entity verifies the authentication signature of the peer entity by first decrypting the received exchange, then verifying the signature and checking the Key Token to protect against replay attack. Verification requires use of the peer entity's public key, and the agreed process for signature verification.

## B.4 SA Attribute Negotiation

### B.4.1 Service Negotiation

Based on its local security policy, the initiating TLSP Entity issues a set of one or more acceptable security service selections. Each element in the set contains the following:

a) the ASSR_ID which defines the semantics of the security services selected (listed below) for this element in the set; and

b) a Service selection values (semantics defined by the ASSR_ID) for each of: Confidentiality, Authentication, Access Control, Integrity, and Traffic Flow Confidentiality.

Based on its local security policy, the recipient TLSP Entity will return the following PCI to the originator:

a) If one of the proposed set of services is acceptable, the recipient will return single selected service element.

b) If none of the proposed set of services is acceptable, the recipient will reject the SA by returning a status indicating the reason for rejecting the SA.

NOTE – This negotiation allows both TLSP entities to select security services which are consistent with its local security policy.

### B.4.2 Label Set Negotiation

Based on its local security policy, the initiating TLSP Entity issues a set of security labels and references which it is willing to have transferred under the protection of this SA. Each element in the set contains the full semantics of the label.

Based on its local security policy, the recipient TLSP Entity will determine which of the proposed set of labels it is willing to have transferred under protection of this SA. The recipient TLSP Entity will return the following PCI to the originator:

a) if one or more labels in the proposed set is acceptable, the recipient will return a subset of the proposed set of references. Null sets are not allowed.

b) If no labels in the proposed set are acceptable, the recipient will reject the SA by returning a Status indicating the reason for rejecting the SA.

NOTE – This negotiation allows either TLSP entity to select a label set which is consistent with its local security policy. The above is only applicable if the label attribute has been selected.

### B.4.3 Key and ISN Selection

Based on its local security policy, the initiating TLSP Entity selects those portions of the bit string resulting from the KTE for use as keys and/or ISNs during communications (that is, TLSP communications and not SA-Protocol communications) to the recipient TLSP entity. The key/ISN is identified by communicating the starting bit position within the KTE resultant bit string. The key/ISN length is determined from the parameters associated with the selected service. A set of pointers is sent to the recipient TLSP entity for the following:

a) Normal Data Encipherment Key;

b) Expedited Data Encipherment Key;

c) Normal Data Integrity Check Generation Key;

d) Expedited Data Integrity Check Generation Key;

e) My ISN for Normal Data;

f) My ISN for Expedited Data; and

g) Authentication Generation Key.

Similarly, the recipient TLSP Entity will determine via its local security policy which portions of the KTE resultant bit string it will use for its keys/ISNs. The recipient TLSP Entity will return the following PCI to the originator:

a) If the recipient chooses to use the same bit positions as proposed by the initiating TLSP entity, no explicit PCI is returned.

b) If the recipient is rejecting the SA due to other negotiation failures, no explicit PCI is returned.

c) If the recipient selects different bit positions for its keys/ISNs, it will return a set of pointers.

NOTE – The same key value may be used for multiple purposes by providing the same pointer for more than one key/ISN.

### B.4.4 Miscellaneous SA Attribute Negotiation

Based on its local security policy, the initiating TLSP Entity determines the value of the following SA attributes for the SA being established such as retain these SA attributes on disconnect (ITU-T Rec. X.224 | ISO/IEC 8073 selected).

The initiating TLSP entity sends the recipient TLSP entity this set of proposed SA attributes in a Miscellaneous flags field.

Based on its local security policy, the recipient TLSP Entity will return the following PCI to the originator:

a) If the recipient accepts all of the proposed SA attributes then no explicit PCI is returned. If the recipient does not reject the SA, it implies that the SA attributes are acceptable to the recipient TLSP entity.

b) If any one of the attributes is not acceptable, the recipient rejects the SA by returning a status indicating which attributes caused the rejection.

### B.4.5 Re-keying Overview

If an SA is being established to rekey an old SA then only Key and ISN Selection are carried out. Instead of service, label set and miscellaneous SA Attribute negotiation the reference to the old SA from which these attributes are to be inherited is place in Old_Your_SA-ID.

### B.4.6 SA Abort/Release Overview

A security association may be released by the following methods:

a) through the exchange of security association protocol data units;

b) using external mechanisms outside the scope of the lower layer protocol;

c) implicitly by closing a connection;

d) implicitly when a key within the SA expires.

These methods for the release of Security Association fall into two categories, out-of-band method and in-band method. In the case of in-band method it is possible to release the SA by either the disconnect request of Transport Connection or the issuance of SA release (SA PDU with a Content Field Type SA abort/release Reason. See B.6.3 for further details.

## B.5 Mapping of SA-Protocol Functions to Protocol Exchanges

This SA-Protocol performs the three functions described above during two distinct protocol exchanges:

a) The first exchange consists of KTE and certificate exchange and has no encipherment applied.

b) The second exchange consists of a security negotiation protected to provide authentication as defined in B.3.

c) A separate exchange initiated when the SA is no longer required consisting of a reason code protected to provide authentication as defined in B.3.

### B.5.1 KTE (First) Exchange

#### B.5.1.1 Request to Initiate the SA-Protocol

The TLSP Entity or local security management initiates the SA-Protocol.

The initiating TLSP entity performs the following functions and sends the following information to the recipient:

a) An available SA-ID is selected and sent as the originator’s My_SA-ID.

b) KTE is started and the following is sent: Key Token 1.

c) A list of proposed confidentiality mechanisms which could be used to protect the second SA-Protocol exchange. This list is expressed as a set of one or more elements which include: ASSR_ID, and confidentiality security service selected. This list need not be sent if mechanisms have been agreed in advance.

d) A list of proposed integrity mechanisms, one of which would be used to digitally sign the second SA-Protocol exchange. This list is expressed as a set of one or more elements which include: ASSR_ID and integrity security services selected. This list need not be sent if mechanisms have been agreed in advance.

NOTE – The confidentiality security services selected should only identify a symmetric encipherment algorithms and its mode of operation. The integrity security services selected should only identify an asymmetric algorithm and its associated digital signature scheme. Items c) and d) may be known *a priori*.

In the CO case, if no PDU is returned for the first exchange after a timeout, the SA is not established and no further attempts are made.

In the CL case, if no PDU is returned for the first exchange after a timeout, the initiating TLSP entity retransmits its first exchange PDU. Retransmissions are limited to a finite number which is locally defined.

#### B.5.1.2 Receipt of the First Exchange PDU by Recipient

Upon receipt of the first exchange PDU, the recipient TLSP entity performs the following functions and sends the following information to the initiator:

a) The received My_SAID is placed in the Your_SAID field of the generic header as described in 8.3.

b) An available SAID is selected and sent as the originator's My_SAID.

c) Based on its local security policy, the recipient TLSP Entity will return the following PCI to the originator:

   1) If the recipient accepts one of the proposed confidentiality mechanism, then it returns the selected mechanism. If the initiator proposed a single mechanism, no explicit PCI is returned.

   2) If all of the confidentiality mechanisms are not acceptable, the recipient rejects the SA by returning a status indicating the cause of rejection.

d) Based on its local security policy, the recipient TLSP Entity will return the following PCI to the originator:

   1) If the recipient accepts one of the proposed confidentiality mechanism, then it returns the selected mechanism. If the initiator proposed a single mechanism, no explicit PCI is returned.

   2) If all of the confidentiality mechanisms are not acceptable, the recipient rejects the SA by returning a status indicating the cause of rejection.

e) Provided both a confidentiality and integrity mechanism have been selected, the KTE calculation is started and Key Token 2 is sent.

In the CO case, if a PDU from the second exchange is not returned after a timeout, the SA is not established and no further attempts are made.

In the CL case, if a PDU from the second exchange is not returned after a timeout, the initiating TLSP entity retransmits its first exchange PDU. Retransmissions are limited to a finite number which is locally defined.

In the CL case, if the PDU from the first exchange is received again, the return PDU is resent.

### B.5.2 Authentication and Security Negotiation (Second) Exchange

#### B.5.2.1 Receipt of First Exchange PDU by Initiator

On receipt of the first exchange SA PDU, the initiating TLSP entity performs the following functions and sends the following information to the recipient:

a) The received My_SAID is placed in the Your_SAID field of the generic header as described in 8.3.

b) The initiator certificate associated with the selected integrity mechanism is placed in Content Field Certificate.

c) The initiator generates Key Token 3.

d) A list of proposed security services which could be used to protect the TLSP communication are placed in Content Field Service Selection.

e) A set of proposed labels which could be protected using this SA during TLSP communication are placed in Label_Def.

f) A set of key/ISN pointers are placed in Key Selection.

g) The miscellaneous SA attributes required for this SA are placed in SA Flags.

h) If SA establishment is to rekey an old SA then Old Your SA-ID is set to the SA-ID for the old SA being rekeyed. If this process is carried out d), e) and g) above shall not be carried out.

i) Protect SA contents as described in B.3.

In the CO case, if a PDU from the second exchange is not returned after a timeout, the SA is not established and no further attempts are made.

In the CL case, if a PDU from the second exchange is not returned after a timeout, the initiating TLSP entity retransmits its second exchange PDU. Retransmissions are limited to a finite number which is locally defined.

In the CL case, if the PDU from the first exchange is received again, the second exchange PDU is resent.

#### B.5.2.2 Receipt of the Second Exchange PDU by Recipient

Upon receipt of the second exchange PDU, the recipient TLSP entity performs the following functions and sends the following information to the initiator:

a) The received My_SAID is placed in the Your_SAID field of the generic header as described in 8.3.

b) The following items are checked. If any item fails its check, the SA is rejected and a Status field is returned indicating the cause of rejection:

   1) The received digital signature is checked to be valid.

   2) The received Key Token 3 is checked to be valid.

   3) The set of proposed security services is checked to determine if any are acceptable. Only one of the proposed security services can be selected.

   4) The set of proposed labels is checked to determine if any are acceptable.

   5) The miscellaneous SA attributes are checked to determine if all are acceptable.

c) If Old Your SA-ID is present in the received PDU then the appropriate SA are copied from the referenced SA-ID. In this case use of the fields described in c, d below cannot be sent.

Provided all the checks pass the following items are sent:

a) The initiator certificate associated with the selected integrity mechanism is sent.

b) The selected security services to be used to protect the TLSP communication are sent. If the set of proposed services contained one element, no PCI is returned.

c) The recipient generates Key Token 4.

d) The selected subset of proposed labels which could be protected using this SA during TLSP communication is sent.

e) A set of key/ISN pointers are sent. If the initiator's pointers are acceptable to the recipient, no PCI is sent.

f) Protect SA contents as described in B.3.

In the CL case, if the PDU from the second exchange is received again, the recipient resends its second exchange PDU.

### B.5.3 Rekey Procedure

The TLSP entities may update keys at any time during a Security Association. This is achieved through an exchange of SCI. This exchange is transparent to the TLSP user and no TLSP service primitives are defined to invoke it.

A possible mode of operation is at regular intervals during a connection (e.g. every hour or every 10 000 SE-TPDUs) to exchange SCI. Rekeying will cause a new SA-ID to be selected; however, attributes from the current SA can be inherited.

The rekey information can contain either:

a) a new key enciphered with a mutual KEK;

b) a new key enciphered with the public key of the recipient;

c) a reference to a previously distributed key;

d) rekey information used by a previously agreed key distribution method.

The rekey procedures are based on the exchange of two SA PDUs with rekey information (called outward and response) and normal SE-TPDUs as follows:

A SA PDU containing outward rekey information is prepared with:

a) the outward/response flag in the flag octet set to 0;

b) if label mechanism selected then the Label-Ref is set to the reference for the appropriate label;

c) the key information is set as required by the key distribution mechanism;

d) Protected Initial ISN is set the sequence number of which SE TPDU should be encrypted by using updated key;

e) SA Flag Rekey Field three is set to 1.

On receipt of a SA PDU containing outward rekey information:

a) if Label mechanism is selected, then the label-ref field is checked to be a value valid for this SA;

b) the key information is processed as required by the key distribution mechanism;

c) check that the Initial ISN is appropriate or not;

d) Check Flag Rekey field three is set to 1.

Then a SA PDU containing response rekey information is prepared with:

a) the outward/response flag of the flag octet set to 1;

b) if label mechanism selected then the Label-Ref is set to the reference for the appropriate label;

c) the key information is set as required by the key distribution mechanism;

d) Protected Initial ISN is set the sequence number of which SE TPDU should be encrypted by using updated key;

e) SA Flag Rekey field three is set to 1.

On receipt of a SA PDU containing Response Rekey Information:

a) if Label mechanism is selected, then the Label-Ref field is checked to be a value valid for this SA;

b) the key information is processed as required by the key distribution mechanism;

c) check that the Initial ISN is appropriate or not;

d) Check Flag Rekey field three is set to 1.

Following successful checking of the response if the TLSP entity has no TPDU awaiting encapsulation, a SA PDU containing no data is sent to complete the Rekey Procedure.

When a TLSP entity having sent a SE-TPDU containing outward rekey information receives a SE-TPDU encapsulated by using the former key, the TLSP entity should not discard the SE-TPDU by the former key unless the security policy indicates that this should be done.

If the rekey procedure fails, then the association is reestablished either by use of the SA-P or by any other appropriate means.

### B.5.4 SA Release / Abort Exchange

#### B.5.4.1 Request to Initiate SA Release / Abort

The TLSP entity or local security management initiates the SA Release / Abort. The initiator of an SA Abort / Release need not be the initiator of the SA establishment.

a) If the local entity is the SA establishment initiator, then Key Token 3 is generated else Key Token 4 is generated. In either case the generated token is placed in the SA contents.

b) The appropriate reason code is placed in SA Content field Abort/Release Reason.

c) Protect SA contents as described in B.3.

In the CO case, if a confirm PDU from the abort/release request is not returned after a timeout, the SA is not established and no further attempts are made.

In the CL case, if a PDU from the abort/release exchange is not returned after a timeout, the initiating TLSP entity retransmits its SA release/abort request PDU. Retransmissions are limited to a finite number which is locally defined.

#### B.5.4.2 Receipt of SA Abort/Release Requests

Upon receipt of the SA Abort/Release Confirm PDU, the recipient TLSP entity performs the following functions and sends the following information to the initiator:

a) If the local entity is the SA establishment initiator, then Key Token 3 is generated else Key Token 4 is generated. In either case the generated token is placed in the SA contents.

b) The appropriate reason code is placed in the SA Content field – Abort/Release Reason.

c) Protect SA contents as described in B.3.

In the CL case, if the PDU from the abort/release request is received again, the recipient re-sends its second exchange PDU up to a given limited number of times.

## B.6 SA PDU – SA Contents

For this specific SA-Protocol, the format of the SA Contents field of the SA PDU defined in 8.4 is shown in Figure B.2.

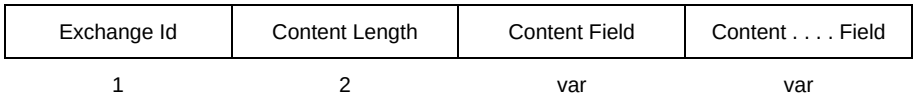



**Figure B.2 – SA Contents**

### B.6.1 Exchange ID

This field contains a value of 00000000 if the PDU is associated with the first KTE Exchange and a value of 00000001 if the PDU is associated with the second Authentication/Negotiation Exchange. This field contains a value of 10000000 if the PDU is associated with a SA ABort/Release request 10000001 if the PDU is associated with a SA Abort/Release confirm.

### B.6.2 Content Length

The length in octets of all Content fields but excluding the Content Length field.

### B.6.3 Content Fields

The content field type encoding is defined in 8.2. The SA content fields (that is, 00-BF) used by the procedures in this appendix are given below:

| *Value* | *Content Field Type* |
|---|---|
| A0 | My SA-ID |
| A1 | Old Your SA-ID |
| A2 | Key Token 1 |
| A3 | Key Token 2 |
| A4 | Authentication digital signature |
| A5 | Authentication certificate |
| A6 | Service Selection |
| A7 | SA Rejection Reason |
| A8 | SA Abort/Release Reason |
| A9 | SA Flags |
| AA | Key Selection |
| AB | ASSR |
| AC | Initiator |
| AD | Integrity Algorithm |
| AE | Confidentiality algorithm |
| AF | ICV length |
| B1 | Encipherment Key |
| B2 | Decipherment Key |
| B3 | Authentication Mechanism |
| B4 | Access Control Mechanism |
| B5 | Key Token 3 |
| B6 | Key Token 4 |
| B7-BF | Reserved for future use |

NOTE – Further codes are reserved for private use in 8.2 in the main body of this Recommendation | International Standard.

The Service Selection, SA Rejection Reason, Label-Def, SA Flags, and Key Selection Fields are optional within this specific SA-Protocol content definition.

#### B.6.3.1 My SA-ID

This mandatory field is used in the first exchange only. This parameter is the local identifier for a Security Association.

#### B.6.3.2 Old Your SA-ID

This field is used on the second exchange if attributes, other than keys, are to be inherited from the old SA.

#### B.6.3.3 Key Token 1, Key Token 2, Key Token 3, and Key Token 4

These mandatory fields are used to support the KTE as described earlier in this annex.

#### B.6.3.4 Authentication Digital Signature, Certificate

These mandatory fields are used to support the authentication as described earlier in this annex.

#### B.6.3.5 Service Selection

This optional field is used in both the first and second exchanges:

a) If used during the first exchange, it is used to identify proposed confidentiality and/or integrity mechanisms to be used during the second SA-Protocol exchange. In this case, only the first two octets are present.

b) If used during the second exchange, it is used to propose all mechanisms to be used during the TLSP communications protected by the SA being established.

This field may be included one or more times within either the first or second exchange PDU to form a proposed set of security services for negotiation.

This parameter contains a sequence of octets indicating the levels of security services selected. The semantics of the levels is defined as part of the security policy. The octets for each of the security services appear in the order indicated below. The sequence of octets can be truncated if the truncated octets all relate to the services that have the QOS value 0. A single octet of value 255 indicates that selected security services have been pre-established.

| *Octet* | *Meaning* |
|---|---|
| 1 | Connectionless Confidentiality / Connection Confidentiality |
| 2 | Connectionless Integrity / Connection Integrity with or without Recovery |
| 3 | Data Origin Authentication / Peer Entity Authentication |
| 4 | Access Control |
| 5 | End System Protection |
| 6 | Per connection Protection |

#### B.6.3.6 SA Rejection Reason

This optional field may be present in either the first or second exchange PDU. It is present to indicate a rejection of the SA during its establishment. It contains the reason for rejection as follows:

| *Value* | *Meaning* |
|---|---|
| 1 | Confidentiality Mechanism not supported |
| 2 | Integrity Mechanism not supported |
| 3 | Access Control Mechanism not supported |
| 4 | Authentication Mechanism not supported |
| 5 | End System not supported |
| 6 | Per Connection not supported |
| 7 | Confidentiality Mechanism rejected |
| 8 | Integrity Mechanism rejected |
| 9 | Access Control Mechanism rejected |
| 10 | Authentication Mechanism rejected |
| 11 | Authentication signature invalid |
| 12 | Certificate invalid |
| 13 | Proposed Label set rejected |
| 14 | Retain_on_Disconnect rejected |
| 15 | Param_Prot rejected |
| 16 | End System rejected |
| 17 | Per Connection rejected |

#### B.6.3.7 SA Abort/Release Reason

This mandatory field is present in the SA Abort/Release request and indication. It is used to indicate the reason of a SA Abort release.

It is set to 0 to abort and 1 for normal release. Value 2 to 127 are reserved for future use. Other values can be used for privately defined reason codes.

#### B.6.3.8 Label

This optional field is for use in the second exchange PDU only and is used as defined in 8.2.3.4. The originator will propose a set of security labels. The recipient may select either the full set or a subset of what the originator sent. If the original set is not acceptable the recipient may propose a different set of labels.

#### B.6.3.9 Key Selection

This optional field is for use in the second exchange PDU only. It can occur any number of times within the SA Contents.

This field is sub-divided into three sub-fields:

a) Usage Flags;

b) Key Selection Information;

c) Key Reference.

##### B.6.3.9.1 Usage Flags

This field contains up to seven values indicating the position within the KTE resultant bit string where certain keys are to take their value. The length of the key is determined from the associated security service selected which identifies the associated algorithm. Multiple keys may use the same bit position (i.e. the same key). The allowable combinations will be dependant on the local security policy.

| *Octet* | *Related Key/ISN position in EKE bit string* |
|---|---|
| 1-2 | Normal Data Encipherment Key |
| 3-4 | Expedited Data Encipherment Key |
| 5-6 | Normal Data Integrity Check Generation Key |
| 7-8 | Expedited Data Integrity Check Generation Key |
| 9-10 | My ISN for Normal Data |
| 11-12 | My ISN for Expedited Data |
| 13-14 | Authentication Generation Key |

If the recipient wishes to use the same keys as the originator, this field shall not be present in the recipient's second exchange PDU.

##### B.6.3.9.2 Key Selection Information

This field indicates the position within the KTE resultant bit string where selected keys are to take their value. The length of the key is determined from the associated security service selected which identifies the associated algorithm. Multiple keys may use the same bit position (i.e. the same key). The allowable combinations will be dependent on the local security policy.

##### B.6.3.9.3 Key Reference

This optional sub-field can be used to enable later reference to the key. This may be used for example for auditing purposes or for selection of a new key for a connection using the SA PDU. The value of this reference shall be unique for the Security Association.

#### B.6.3.10 SA Flags

The following bit positions are used to signal the identified SA attributes. Value 0 means false; value 1 means true.

| *Bit* | *SA Attribute* |
|---|---|
| 1 | Retain-on-Disconnect |
| 2 | Param_Protect |
| 3 | Rekey |
| 4 | Outward/Response |
| 5-8 | reserved for future use |

Bits 5-8 are set to 0 on transmission and ignored on receipt.

#### B.6.3.11 ASSR

This field must be present if the Service Selection field is present. It is the object identifier (as defined ISO/IEC 9834) which identifies the set of security rules which define the mechanisms to be applied given the protection QOS selected.

# Annex C

## An example of an agreed set of security rules (ASSR)

(This annex does not form an integral part of this Recommendation | International Standard)

An Agreed Set of Security Rules (ASSR) establish the security mechanisms to be used including all parameters needed to define the operation of the mechanism for a given protection QOS.

ASSR-ID { joint-iso-ccitt (2) identified organization (3) oiw (14) secsig (3) oiwsecsigassrobjectidentifier (5) rule (1) } (Object Identifier)

SA-ID Length 4 Octets

Protection QOS Definition Module

| | |
|---|---|
| PE Auth: | low |
| AC: | none |
| Confid: | high |
| Integ: | high |
| Security Label: | none |

Protection of all service parameters
For Protection QOS: Integ = high Confid = high

Mechanism Module – Security Labels for Access Control
For Protection QOS: AC = high or Conf = high

Label_Def_Auth XYZ

Explicit indication: Yes

Mechanism Module – Integrity Check Value
For Protection QOS: Integ .none or Auth = High
or Mechanism for Security Labels

| | |
|---|---|
| ICV_Alg_ID | XYZ |
| ICV_Block_size | 8 octets |
| Rekey after | 15 000 PDUs |
| Key Distribution mech | Asymmetric |

Mechanism Module – Integrity Sequence Number
For Protection QOS: Integ = high Auth = high

ISN_Len 4 octets

Mechanism Module – Encipherment
For Protection QOS: Conf > low

| | |
|---|---|
| Enc_Alg_ID | XYZ |
| Mode | Chained |
| Enc_Block_Size | 8 octets |
| Rekey after | 10 000 PDUs |
| Key Distribution mech | Asymmetric |

Mechanism Module – Connection Authentication
For Protection QOS: AC > low or PE Auth > Low

Enc_Alg_ID XYZ

Mechanism Module – Asymmetric Key Distribution
For mechanism encipherment or Integrity check value

PKC_Alg_ID RSA

Mechanism Module – Symmetric Key Distribution
For mechanism encipherment or Integrity check value

PKC_Alg_ID DES (X9.17)

# Annex D

## Overview of EKE Algorithm

(This annex does not form an integral part of this Recommendation | International Standard)

There are two parameters required for EKE: one is a large prime number p (such that p-1 has a large prime factor) and the other is a number “a” which is on the range 1, a, p-1.

Let A and B be the two communicating parties (see Figure D.1). EKE begins with A selecting a large random number X, and B selecting a large random number Y. A then calculates (a**X mod p) and sends a, p, and (a**x mod p) to B, and B calculates (a**XY mod p) and sends it to A. Both A and B calculate (a**XY mod p). An eavesdropper only sees (a**X mod p) and (a**Y mod p). An eavesdropper cannot determine X or Y and, therefore, cannot calculate (a**XY mod p).

A and B may subsequently use subsets of the bits in (a**XY mod p) as keys.

The values described in SA protocol defined in Annex B are:

- The shared EKE bit string is (a**XY mod p).
- Key Token 1 is a, p,(a**X mod p) where the ‘a’, ‘p’, and (a**X mod p) are encoded as an concatenated bit strings.
- Key Token 2 is (a**Y mod p).
- Key Token 3 is information derived from the shared KTE bit string (a**XY mod p) to counter replay attacks.
- Key Token 4 is information derived from the shared KTE bit string (a**XY mod p) to counter replay attacks.

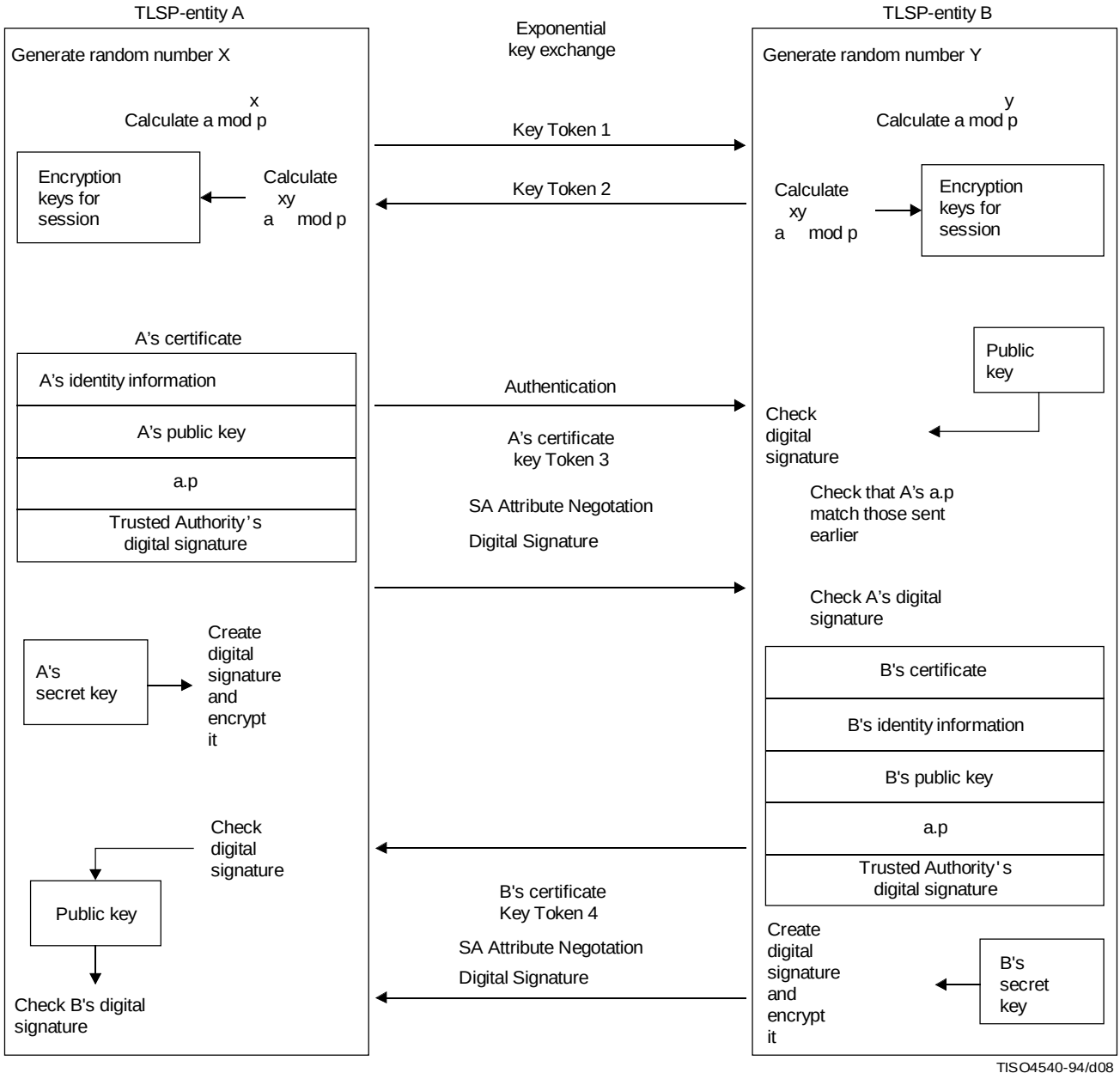



**Figure D.1 – Illustration of On-Line Key Derivation and Digital Signature using EKE**