IEEE Standard for Interoperable

Local Area Network (LAN) Security (SILS)

Part B -- Secure Data Exchange

Prepared by the IEEE 802.10
Editor of the LAN Security Working Group

This is an unapproved draft which is subject to change and cannot be presumed to reflect the position of Project 802 or The Institute of Electrical and Electronics Engineers.

DO NOT SPECIFY OR CLAIM CONFORMANCE TO THIS DOCUMENT.

Abstract

The Standard for Interoperable Local Area Network (LAN) Security (SILS), Part B - Secure Data Exchange describes a security protocol that can be used to protect IEEE 802 Local Area and Metropolitan Area Networks (LANs, MANs). This Open Systems Interconnection (OSI) Layer 2 security protocol can be used to provide the security services of Confidentiality and Connectionless Integrity. In conjunction with Key Management or System Management, the security services of Data Origin Authentication, and Access Control may also be provided.

Foreword

(This Foreword is not part of the standard, "Standard for Interoperable Local Area Network (LAN) Security (SILS)", produced by IEEE 802.10.)

IEEE 802.10 was formed in May of 1988 to address the security of Local Area and Metropolitan Area Networks (LANs and MANs). It is co-sponsored by IEEE 802 and by the IEEE Technical Committee on Security and Privacy. IEEE 802.10 intends to provide a series of standards to address security for LANs and MANs. The standards are interoperability standards that are compatible with the existing IEEE 802 and OSI architectures. The committee has representation from vendors, government, and users.

Data networks, especially LANs and MANs, have become widespread. LANs and MANs are used by both industry and government for transferring vast amounts of information in the course of daily operations. Because of their ever-increasing use in the private and public sectors, the capabilities of these networks are being expanded to encompass more and more performance requirements. As a result, there is the growing need to standardize network protocols wherever feasible, to ensure that data networks will interoperate effectively.

As standardization practices evolve, several key areas will become critically important. One of these areas is network security. Many LANs and MANs require the capability to exchange data in a secure manner. This is especially important in cases where disclosure of operational information to unauthorized parties would severely undermine an organizations's effectiveness. In addition to disclosure, the integrity of the data is often critical.

Financial and government institutions have traditionally been most aware of the importance of security. However, recent widely publicized cases of computer fraud and related crimes have made security a goal for many other industries as well. As the need for security on LANs and MANs becomes more recognized, the need for a standardized approach to providing such a capability also becomes a priority. Much security standardization has already been started. Where applicable, this standard attempts to incorporate this work.

Committee List

At the time this standard was completed, the LAN Security Working Group had the following membership:

Kimberly Kirkpatrick, Chairman
Jim Randall, Co-Chairman
Russell Housley, Co-Chairman
L. Kirk Barker, Editor
Peter Yee, Recording Secretary

Ken Alonge
Alan Arndt
Kurt Augustine
Bill Birnbaum
Ben Bratcher
Ronald Gibson
Jon Graff
Tom Hunwick
Robert Kolacki
Paul Lambert
Warren Loper
Wen-Pai Lu
Marc Mandel
Ken McCoy
Esther Murphy
Richard Parker
Brian Phillips
James Pyles
Eugene Reilly
Brian Schanning
James Sutherland
Kenta Takumi
Lloyd Taylor
Joseph Williamson

Other individuals who have contributed review and comments are:

Morrie Gasser
B.J. Herbison
Lawrence Kilgalen
John Kimmins
Scott Lawrence
Larry Lunsford
C.E. Reaver
Jim Sanders
Mitch Tannenbaum

Contents

Security (SILS) Part B—Secure Data Exchange

SECTION PAGE

List of Figures

List of Tables

List of Appendices

1. Introduction

1.1 Scope and Purpose. This standard is one of a set of four standards developed by IEEE 802.10 for providing security in IEEE 802 Local Area and Metropolitan Area Networks (IEEE 802 LANs and MANs). The protocol described in this document is not applicable to MANs using IEEE 802.6 Isochronous and Connection-Oriented protocols[11][1]. Nor is it applicable to Integrated Voice and Data Networks using IEEE 802.9[12]. IEEE 802.10a, which describes the model for providing security services, documents the relationship of the four standards. This standard, 802.10b, defines a Secure Data Exchange (SDE) protocol for IEEE 802 LANs and MANs. The other two standards provide for Key Management and System/Security Management in IEEE 802 LANs and MANs. While 802.10b (SDE) is independent of any key management or system management implementation, the security services described in this standard depend on management information provided by management entities.

1.2 Overview. The SDE is an OSI Basic Reference Model [1] Layer 2 entity. This entity provides services that permit the secure exchange of data at Layer 2. As part of the Logical Link Control (LLC) Sublayer, the SDE entity provides a connectionless service immediately above the Medium Access Control (MAC) Sublayer in IEEE 802 LANs and MANs. It provides security across the MAC Sublayer using cryptographic mechanisms and security services provided transparently at the boundary to the LLC entity. Fig 1 shows the relationship of the SDE entity to the IEEE 802 reference model.

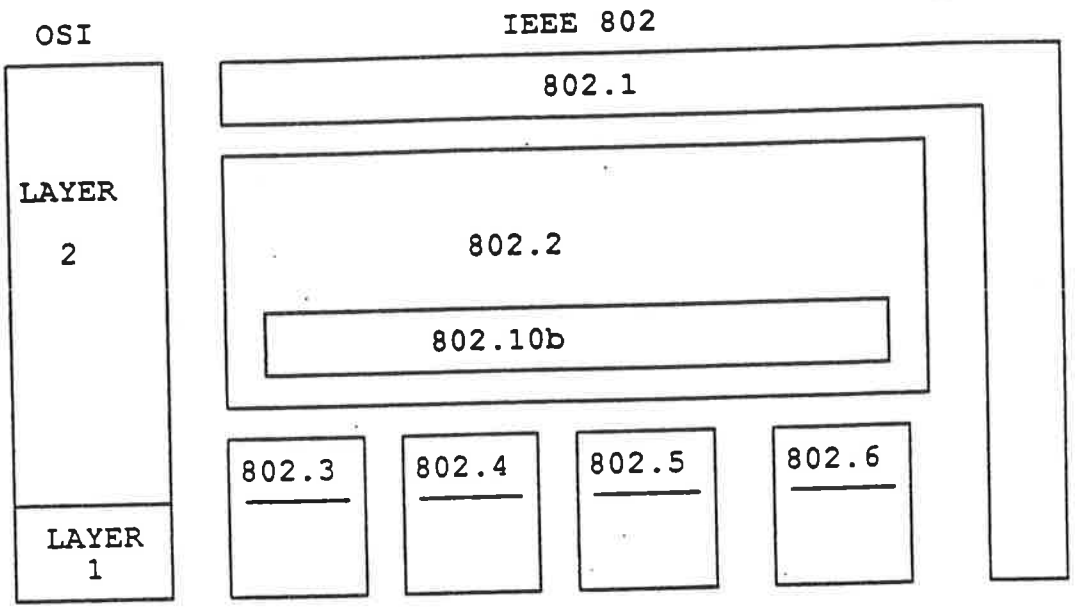


Fig 1
Relationship to IEEE 802 Reference Model

This standard defines the SDE interface services specification to the MAC Sublayer, to the boundary of the LLC entity, and to the SDE Layer Management functions. Section 4 describes the security services provided and the threats these services protect against. Section 5 defines the service specifications and details the interface to the MAC Sublayer and to the LLC entity boundary.

The SDE entity provides security services and an interface at the boundary to the LLC entity. However, it does not specify any of the higher protocols that reside in the User Stack, including those of the LLC sublayer. The SDE interface is equivalent to the unprotected MAC interface and thus requires no change to the existing upper-layer protocols in the User Stack.[2] SDE security services provided to a Key Management Stack or to a System Management Stack require the LLC protocol.

[2] To use the management functionality of IEEE 802.1 and CMIP, the SDE is modeled as part of LLC. If these management protocols are not used, it is possible to model SDE as a Data Link sublayer directly above the MAC sublayer.

Section 6 introduces an SDE-specific Protocol Data Unit (SDE PDU). The SDE PDU has optional elements and fields to satisfy a broad range of potential security applications. A reserved Link Service Access Point (LSAP) in the clear header portion of the SDE PDU distinguishes the SDE PDU from LLC PDUs. Section 6 defines the SDE PDU elements and element fields and describes the transformation of an SDE SDU into an SDE PDU.

A security association is an important concept in this standard. A security association is a cooperative relationship between communicating entities, formed by sharing security management information. This shared information coordinates the transmission and reception processing of the SDE PDU. In practice, there are many defined security associations, but only one applies to the processing of a specific SDE PDU. A Security Association Identifier (SAID) associates a defined security association with a specific SDE PDU. Section 7 defines the contents of the security management information and describes the use of the SAID in finding the applicable security association.

The Layer 2 security services provided by the SDE rely on information from non-Layer 2 key management or system management entities. Management entities communicate the information to the SDE entity through a Security Management Information Base (SMIB). The implementation of the SMIB is a local issue; however, the standard specifies the structure of the information as defined in the Structure of Management Information [6]. Section 7 describes the SMIB, the security management architecture, and the procedures for processing the SDE PDU based upon the security management information contained in the SMIB.

2. Acronyms and Definitions

2.1 Acronyms.

CMIP	Common Management Information Protocol
DSAP	Destination Service Access Point
DA	Destination Address
DEA	Data Encryption Algorithm
ICV	Integrity Check Value
IV	Initialization Vector
LAN	Local Area Network
LM	Layer Manager
LLC	Logical Link Control
LSAP	Link Service Access Point
MAC	Medium Access Control
MAN	Metropolitan Area Network

MDF	Management-Defined Field
MER	Minimum Essential Requirements
MIB	Management Information Base
MSDU	MAC Service Data Unit
OSI	Open System Interconnection
PDU	Protocol Data Unit
SA	Source Address
SAID	Security Association Identifier
SAP	Service Access Point
SDE	Secure Data Exchange
SDU	Service Data Unit
SILS	Standard for Interoperable LAN Security
SMAE	System Management Application Entity
SMIB	Security Management Information Base
SSAP	Source Service Access Point
TCB	Trusted Computing Base

2.2 Definitions. For the purpose of this standard, the following definitions apply:

(Sources for the definitions are indicated by reference numbers. Where references are not indicated, the IEEE 802.10 LAN Security Working Group is the source of the definition.)

access control: The prevention of unauthorized use of a resource, including the prevention of use of a resource in an unauthorized manner. [2]

attribute: A property of a managed object or a property of an association among OSI entities. An attribute has an associated value, which may have a simple or complex structure. [13]

authentication: (See data origin authentication, and peer entity authentication.) Note: In this standard, the term "authentication" is not used in connection with data integrity; the term "data integrity" is used instead.

bootstrap SAID: Four SAID values are reserved for the purpose of establishing initial communication with key management or system management when an SAID has not already been negotiated. These SAID values are called "bootstrap" SAIDs and have a pre-established security association.

ciphertext: Data produced through the use of encipherment, the semantic content of which is not available. Note: Ciphertext may itself be input to encipherment, producing super-enciphered data.

cleartext: Intelligible data, the semantic content of which is available. [2]

compromise: A violation of the security of a system such that an unauthorized disclosure of sensitive information may have occurred. [10]

confidentiality: The property that information is not made available or disclosed to unauthorized individuals, entities, or processes. [2]

connection-oriented confidentiality: The protection of all (N)-service data units from unauthorized disclosure during communications from one (N+1)-entity to one or more (N+1)-entities for which a security association is established for the transfer of data and for the application of confidentiality service between the entities themselves and between each entity and the physical layer.

connection-oriented integrity: A service providing for the integrity of all (N)-service data on a security association and detecting any modification, insertion, deletion or replay of any data within an entire SDU sequence.

connectionless confidentiality: The protection of (N)-service data units from unauthorized disclosure during transmission from one (N+1)-entity to one or more (N+1)-entities, where each entity has an association with the physical layer, and no association is established for the transmission of data or for the application of the confidentiality service between the layer peer-entities themselves.

connectionless integrity: A service providing for the integrity of a single SDU. It may take the form of determining whether or not the received SDU has been modified.

cryptographic checkvalue: Information that is derived by performing a cryptographic transformation (see cryptography) on the data unit. [2]

cryptography: The discipline embodying principles, means, and methods for the transformation of data in order to hide its information content, prevent its undetected modification and/or prevent its unauthorized use. [2].

data deciphering key: A key used for the decipherment of an (N)-layer SDU. (It is not used to decipher other keys.)

data enciphering key: A key used for the encipherment of an (N)-layer SDU. (It is not used to encipher other keys.)

data integrity: The property that data has not been altered or destroyed in an unauthorized

manner. [2]

data origin authentication: The corroboration that the source of data received is as claimed. This service, when provided by the (N)-layer, provides the corroboration to an (N+1)-entity that the source of the data is the claimed peer (N+1)-entity. [2]

decipherment: The reversal of a corresponding reversible encipherment. [2]

encipherment: The cryptographic transformation of data (see cryptography) to produce ciphertext. [2]

Initialization Vector (IV): A binary vector used at the beginning of a cryptographic operation to allow cryptographic chaining. [7]

Integrity Check Value (ICV): A value that is derived by performing an algorithmic transformation on the data unit for which data integrity services are provided. The ICV is sent with the protected data unit and is recalculated and compared by the receiver to detect data modification. (See cryptographic checkvalue.)

key: A sequence of symbols that controls the operations of encipherment and decipherment. [2]

key management: The generation, storage, secure distribution, and application of keys in accordance with a security policy. [2]

Key Management Stack: The protocols residing above SDE that request services via an SDE SAP that is supported by the use of a bootstrap SAID with either of the two values reserved for key management.

layer management: Functions related to the management of the (N)-layer partly performed in the (N)-layer itself according to the (N)-protocol of the layer, and partly performed as a subset of systems management [1].

Layer Manager: A systems management service application for which a particular exchange of systems management information has taken a manager role of the (N)-layer [13].

managed object: The OSI Structure of Management Information [6] term which is an abstract representation of a resource. This managed object has a set of attributes. These attributes are equivalent to data objects.

manipulation detection: A mechanism used to detect whether a data unit has been modified (either accidentally or intentionally). [2]

masquerade: The pretense by an entity to be a different entity. [2]

Management Information Base (MIB): A conceptual data base of information contained in the collection of all the managed object classes and their instances. [3]

misordering data: A form of unauthorized data modification in which the reception sequence of data units is altered from the original transmission sequence in an unauthorized manner. This can be attempted by a combination of techniques involving deleting, delaying, and re-inserting data; or modifying sequence control information; or both.

object: Object in this document refers to a data object which has an identifier (name) and a value.

OSI (N)-service: A capability of the (N)-layer and the layers beneath it, which is provided to the (N)-entities at the boundary between the (N)-layer and the (N+1)-layer. [1]

peer-entity authentication: The corroboration that a peer entity in an association is the one claimed. This service, when provided by the (N)-layer, provides corroboration to the (N+1)-entity that the peer entity is the claimed (N+1)-entity.[2] This is primarily intended for, although not limited to, connection-oriented service and may be either unilateral or mutual. [2, SILS]

reflection: A form of data modification in which PDUs sent by an entity are returned in an unauthorized manner. This can be attempted by a combination of techniques involving deleting, delaying, and re-inserting data; and/or modifying address or sequence control information.

secret key: The traditional cryptographic key known only to the communicating parties and used for both encipherment and decipherment.

security association: A cooperative relationship between entities formed by the sharing of cryptographic keying information and security management objects. This shared information need not be identical, but it shall be compatible.

Security Association Identifier (SAID): A value placed in the clear header of the SDE PDU that is used to identify the security association.

Security Management Information Base (SMIB): A MIB that stores security-relevant objects.

security service: A service, provided by a layer of communicating open systems, which ensures adequate security of the systems or of data transfers. [2] Note that these security services need

not be directly requested at the boundary of the (N)- and (N+1)- layer boundary as is required for an OSI (N)-service.

SDE Layer Manager: The SDE portion of the Layer 2 Manager.

systems management: Functions in the Application Layer related to the management of various OSI resources and their status across all layers of the OSI architecture [1].

System Management Stack: The protocols residing above SDE that request services via an SDE SAP that is supported by the use of a bootstrap SAID with either of the two values reserved for system management.

threat: A potential violation of security. [2]

transparent: A protocol is said to be transparent if all of the following conditions are met:

1. Previously existing protocol implementations are able to recover when receiving packets formed by this new protocol.
2. The implementations of this protocol are able to process packets formed by previously existing protocols without problems.
3. The protocol does not affect the operations of the (N+1) and (N-1)-layer implementations.

trusted functionality: That which is perceived to be correct with respect to some criteria, e.g., as established by a security policy. [2]

unauthorized disclosure: The process of making information available to unauthorized individuals, entities, or processes. [2]

unauthorized data modification: Alteration of data not consistent with the defined security policy.

unauthorized resource use: Use of a resource not consistent with the defined security policy. [2]

User Stack: The protocols residing above SDE that request services from any SDE SAP except those supported by the use of a bootstrap SAID.

3. References

This standard shall be used in conjunction with the following publications:

(1) ISO 7498: 1984, Information Processing Systems--Open Systems Interconnection--Basic Reference Model.

(2) ISO 7498-2: 1988, Information Processing Systems--Open Systems Interconnection--Basic Reference Model--Part 2: Security Architecture.

(3) ISO DIS 7498-4, Information Processing Systems--Open Systems Interconnection--Basic Reference Model--Part 4: Management Framework.

(4) ISO 8802-2: 1989, Logical Link Control Protocol.

(5) ISO DIS 10039, Information Processing Systems--Local Area Networks--MAC Service Definition.

(6) ISO/IEC DIS 10165-2, Information Technology--Open Systems Interconnection -- Structure of Management Information--Part 2: Definition of Management Information.

(7) ANSI X9.17-1985, Financial Institution Key Management (Wholesale).

(8) ANSI X3.92-1981, Data Encryption Algorithm.

(9) IEEE 802.1a: 1990, Overview and Architecture.

(10) NCSG-TG-005-001, National Computer Security Center, "Trusted Network Interpretation of the Trusted Computer System Evaluation Criteria", 31 July 1987.

(11) IEEE P802.6/D14, Distributed Queue Dual Bus (DQDB) Subnetwork of a Metropolitan Area Network (MAN).

(12) IEEE P802.9/D10, Integrated Voice/Data LAN MAC and PHY Specification.

(13) ISO/IEC DIS 10040, Information Processing Systems--Open Systems Interconnection--Systems Management Overview.

4. SDE Security Services

This section contains a description of the security services provided by, or supported by, the SDE entity, and the threats these security services protect against.

The security services are as follows:

- Data Confidentiality -- The SDE entity provides data confidentiality by enciphering the SDE SDU. The SDE entity provides for the use of multiple confidentiality algorithms and depends on an external key management service for establishing a data enciphering key and data deciphering key and for choosing an appropriate cryptographic algorithm.

- Connectionless Integrity -- The SDE entity provides connectionless integrity by calculating an Integrity Check Value (ICV) and placing it in the ICV field of the SDE PDU. The SDE entity depends on an external key management service to establish an integrity algorithm.

- Data Origin Authentication -- Data Origin Authentication is achieved by the use of key management. It is supported by the SDE entity placing a Station ID in the protected header portion of the SDE PDU. The inclusion of the Station ID also prevents undetected reflection of the SDE PDU. Data origin authentication can only be provided in conjunction with the integrity service.

- Access Control -- Access control is provided by key management or system management. The SDE entity's use of security associations supports management's access control decisions. The SDE entity cannot transmit or deliver a PDU unless a security association exists. It is management's responsibility to set up the security associations and the SDE's responsibility to enforce the access control policy. Access control is dependent on both integrity and authentication services. Access control can only be provided in conjunction with integrity and authentication.

The threats that these services protect against are as follows:

- Unauthorized Disclosure
- Masquerading
- Unauthorized Data Modification
- Unauthorized Resource Use

The rationale for addressing these threats is contained in Appendix A. Table 1 shows the dependencies among the security services.

Table 1
Security Service Dependencies

Security Service Dependencies	
Service	Dependency
Confidentiality	No Dependencies
Integrity	No Dependencies
Authentication	Depends on Integrity
Access Control	Depends on Authentication and Integrity

It is not necessary for all stations in the LAN or MAN to employ the SDE protocol. It is possible for entities that do not employ the protocol to communicate with those that do employ the protocol.

The SDE protocol is required to be transparent to existing implementations. Transparency, in the context of this standard, consists of meeting the following requirements. 1) Existing IEEE 802 entities shall be able to recover if they receive an SDE protected packet. 2) SDE entities shall be able to accept non-SDE protected packets without impairment. 3) The addition of security should not modify either the (N+1)-layer or (N-1)-layer implementations. Note that the addition of the SDE protocol may cause certain network management values such as the fragmentation size to change, and still be considered a transparent implementation.

5. SDE Service Specifications

This section defines the services provided by SDE. SDE is modelled as part of the LLC entity and relies on the services provided by the MAC sublayer. There are only two primitives that are used at the SDE boundary: UNITDATA.request and UNITDATA.indication. These primitives are described in detail in ISO DIS 10039 [5].

In subsequent sections of this document, the primitives on the upper boundary of the SDE are prefixed with "SDE" and the primitives on the lower boundary are prefixed with "MA" (see Fig 2). The services provided at the upper SDE boundary include those provided by the MAC sublayer with the addition of those services provided transparently by the SDE.

The primitives used across the SDE service interface are a subset of the MAC primitives defined in ISO DIS 10039 [5]. Additional primitives specified by other MAC interfaces shall be passed unaltered through SDE. Likewise, the minimum set of parameters of these primitives is specified. Other MAC interfaces such as those in IEEE 802.5 are also allowed, and shall be passed through without modification. The MAC primitives that make up the SDE subset are as follows:

UNITDATA.request	Source Address Destination Address MAC Service Data Unit (MSDU)
UNITDATA.indication	Source Address Destination Address MSDU

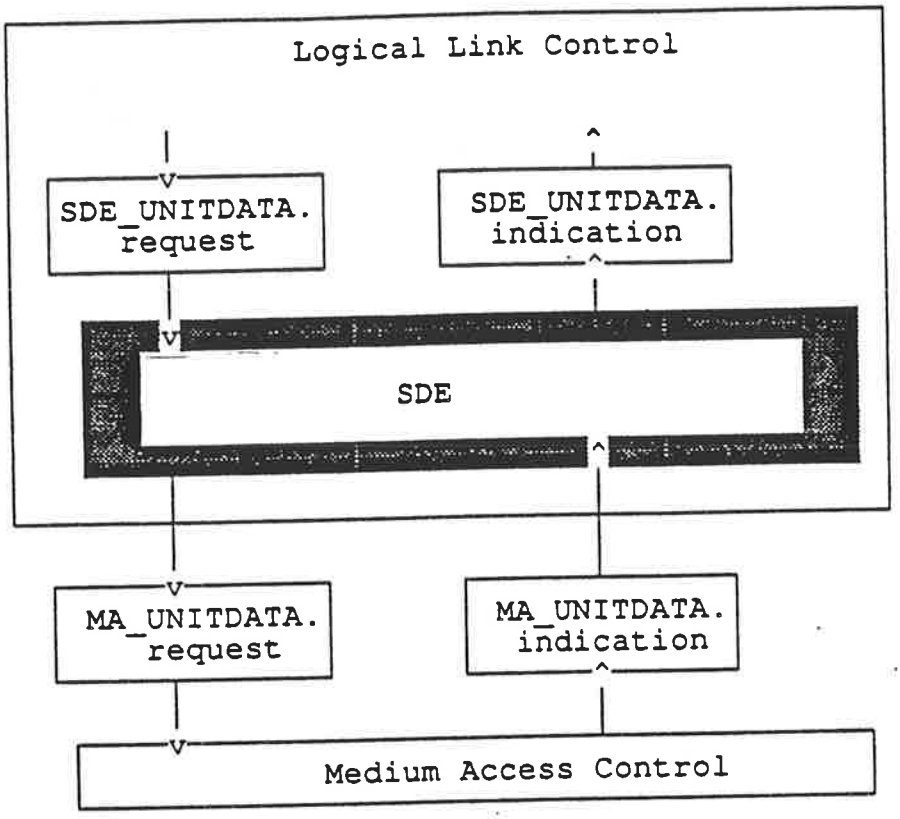


Fig 2
SDE Primitives

5.1 SDE_UNITDATA.request Parameters. The parameters associated with the SDE_UNITDATA.request are defined in ISO DIS 10039 [5].

5.2 SDE_UNITDATA.indication Parameters. The parameters associated with the SDE_UNITDATA.indication are defined in ISO DIS 10039 [5].

5.3 Services Assumed. The service primitives assumed at the lower boundary of SDE are those defined in ISO DIS 10039 [5].

The SDE entity assumes the existence of a Security Management Information Base (SMIB) that is accurately maintained by a method outside the scope of the SDE Entity.

6. SDE Protocol Data Unit (PDU) Structure

This section describes the structure of the SDE PDU. The SDE PDU format is described in 6.1. In 6.2, the relative positions of the various elements of the SDE PDU are defined. This subsection includes descriptions of the fields in terms of size and content. These fields are also defined as either optional or mandatory. Then, 6.3 describes the transformation of an SDE SDU to an SDE PDU.

6.1 SDE PDU Format. SDE uses a single PDU type. The PDU contains an integral number of octets. Fig 3 shows the PDU format, which may contain up to five elements. These elements include the Clear Header, Protected Header, Data (SDE SDU), PAD, and the Integrity Check Value (ICV). All of these elements are optional except Data. The contents of the Protected Header, Data, and PAD may be transformed prior to transmission by the integrity algorithm. The contents of the Protected Header, Data, PAD, and ICV shall always be transformed when the confidentiality algorithm is applied.

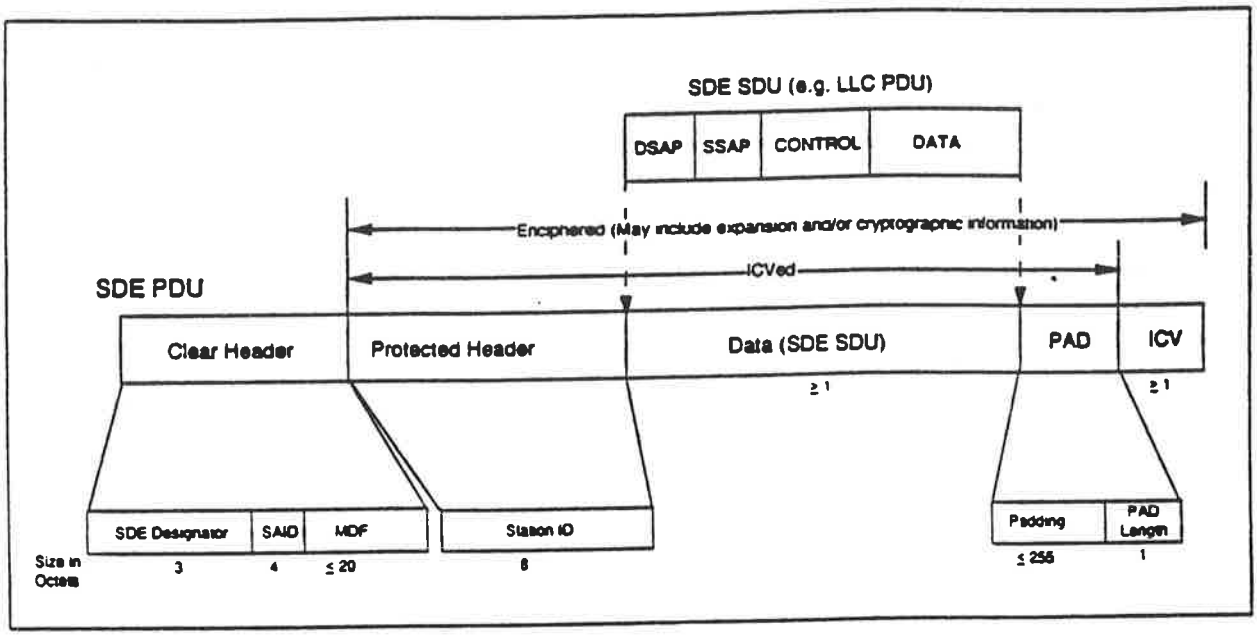


Fig 3
Structure of the SDE PDU

6.2 Elements of the SDE PDU

6.2.1 Clear Header. The Clear Header (see Fig 4) identifies the SDE PDUs and aids in the processing of information contained in these PDUs. The content of the Clear Header is determined during security association setup and is constant for the life of that security association. The use of the Clear Header is optional. When the Clear Header is present, its length will be from seven to twenty-seven octets, inclusive.

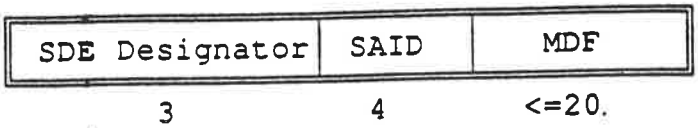


Fig 4
Clear Header

6.2.1.1 SDE Designator. The first three octets of the Clear Header constitute the SDE Designator, which ensures that a non-SDE entity which contains an LLC-entity will not process the SDE PDU. The SDE Designator contains the value of a reserved LSAP in each of the first two octets and the Unnumbered Information control field, as defined in ISO 8802-2 [4] (P-bit equal to zero), in the third octet.[4] In this and subsequent sections, the octets in each field shown are ordered left to right and the leftmost bit is the first bit received from, or sent to, the MAC sublayer. The SDE Designator is mandatory when the Clear Header is present.

6.2.1.2 Security Association Identifier (SAID). The SAID field identifies the security association. It contains the Security Association Identifier associated with the destination SDE entity. If the destination is a group address, the SAID value is common for all the stations in the group and is negotiated by key management or system management. The SAID field is four octets in length and is mandatory when the Clear Header is present.

Fig 5 shows the format of the SAID. The leftmost bit of the SAID is called the G-bit. This is the first bit received from the MAC sublayer. It is used to indicate whether the security association identified by the SAID is common to a group of SDE entities (value set to 1) or an individual SDE entity (value set to 0).

Four SAID values are reserved for the purpose of establishing initial communication with key management or system management when an SAID has not already been negotiated. These SAID values are called "bootstrap" SAIDs, and identify pre-established security associations. If the bootstrap SAID is used for key management, the ID bits contain all zeroes. If the bootstrap SAID is used for system management, the ID bits contain all ones. The use of the bootstrap SAID mechanism is optional. Communication to the System Management and Key Management Stacks may be accomplished via the use of any security association whose

[4] The reserved LSAP contains a value of the form X1XXXXXX that will not conflict with any assigned LSAP values. This will be reserved through ISO and IEEE. It ensures that the reserved LSAP value will not appear in the first or second octet of the MSDU parameter of a MA_UNITDATA.indication unless the MSDU contains an SDE PDU. The value of the Unnumbered Information Control field is "1 1 0 0 0 0 0 0" with the first "1" being the first control field bit received from the MAC sublayer.

SDE_SAP object indicates the appropriate stack. Also note that the function of key management or system management can reside on a User Stack; however, the bootstrap SAIDs cannot be used to support those implementations.

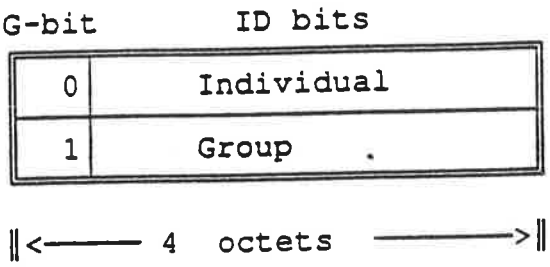


Fig 5
SAID Format

6.2.1.3 Management-Defined Field (MDF). The MDF allows the transfer of information that may facilitate, but is not required for, the processing of the PDU. The MDF is variable in length and is an integral number of octets up to a maximum of twenty. Its value is indicated by an entry in the SMIB. The MDF may contain any value and is not used to determine the appropriate security association. The MDF value is a unidirectional attribute of the security association and is constant for the duration of that security association. The MDF is optional.

An example of the application of the MDF is an SDE implementation that does not retain cryptographic state information. The transfer of cryptographic state information and keying information in the MDF could facilitate reception processing.

6.2.2 Protected Header. The protected header is in the portion of the SDE PDU to which the security services are applied. The Protected Header contains one field, Station ID, which is an optional field.[5] The Station ID uniquely identifies the originating station. It is 8 octets and contains the canonical form of the MAC address as specified in IEEE 802.1a, Section 5.2 [9]. The first octet of the Station ID field shall contain the first octet of the MAC address: the contents of the field after the MAC address is undefined.

[5] See Appendix E on fragmentation for additional uses of the Protected Header.

6.2.3 Data. The Data portion of the SDE PDU contains the SDE SDU, which is the MSDU parameter of the SDE service primitive.

6.2.4 PAD. The PAD consists of the Padding and PAD Length fields. The PAD may be used to provide padding for confidentiality and integrity algorithms.[6] PAD is selected on a per security association basis. If it is allowed, each PDU processed under the association shall contain the PAD Length field.

6.2.4.1 Padding Field. The Padding field is optional but may be required by the specific confidentiality or integrity algorithm selected. The maximum size of the Padding field is 255 octets. The content of the Padding field is a local matter.

The Padding field specifies is an integral number of octets; therefore, the Padding field cannot be used to correct octet alignment problems caused by either the integrity or confidentiality algorithms.

6.2.4.2 PAD Length Field. The value of the PAD Length field contains the number of octets in the Padding field. This value does not include the one octet required by the PAD Length field itself. If no integrity is requested, the PAD Length field is the last octet of the SDE PDU. If integrity is requested, the PAD Length field is the octet before the ICV.

6.2.5 Integrity Check Value (ICV). The ICV field is a security mechanism for detecting data modification. The ICV value, if present, is contained in the last field in the SDE PDU. The length of the ICV is an attribute of the security association. The ICV is calculated over the Protected Header, the Data field, and PAD. It is an optional field.

6.3 Building the SDE PDU. This section describes how the information passed to the MAC is used to construct the SDE PDU. (The MSDU is the SDE SDU.) All of the parameters of the service request except the MSDU are copied unaltered from the SDE_UNITDATA.request to the MA_UNITDATA.request. Likewise, on incoming processing, all parameters except the MSDU

[6] Many confidentiality algorithms take blocks (n bits) of cleartext and transform this cleartext to ciphertext as a unit. This block is known as a cryptographic block. The confidentiality algorithm may require that the input cleartext be a multiple of this block size. If the chosen confidentiality or integrity algorithm has this restriction, then the SDE protocol uses the PAD to make sure that the cleartext is a multiple of the block size. (The PAD follows the Data to allow stream processing for outgoing PDUs.)

are copied unaltered from the MA_UNITDATA.indication to the SDE_UNITDATA.indication. The MSDU is used to generate the SDE PDU as shown in Fig 6. On reception, the process is reversed to reconstruct the MSDU. The encipherment algorithm may require the addition of fields specific to the algorithm. These fields will be added and removed as part of the encipherment or decipherment processing. They will be transmitted as part of the SDE PDU provided in the MSDU of the MAC service primitives. An example of this type of field is the Initialization Vector (IV) required by certain algorithms.

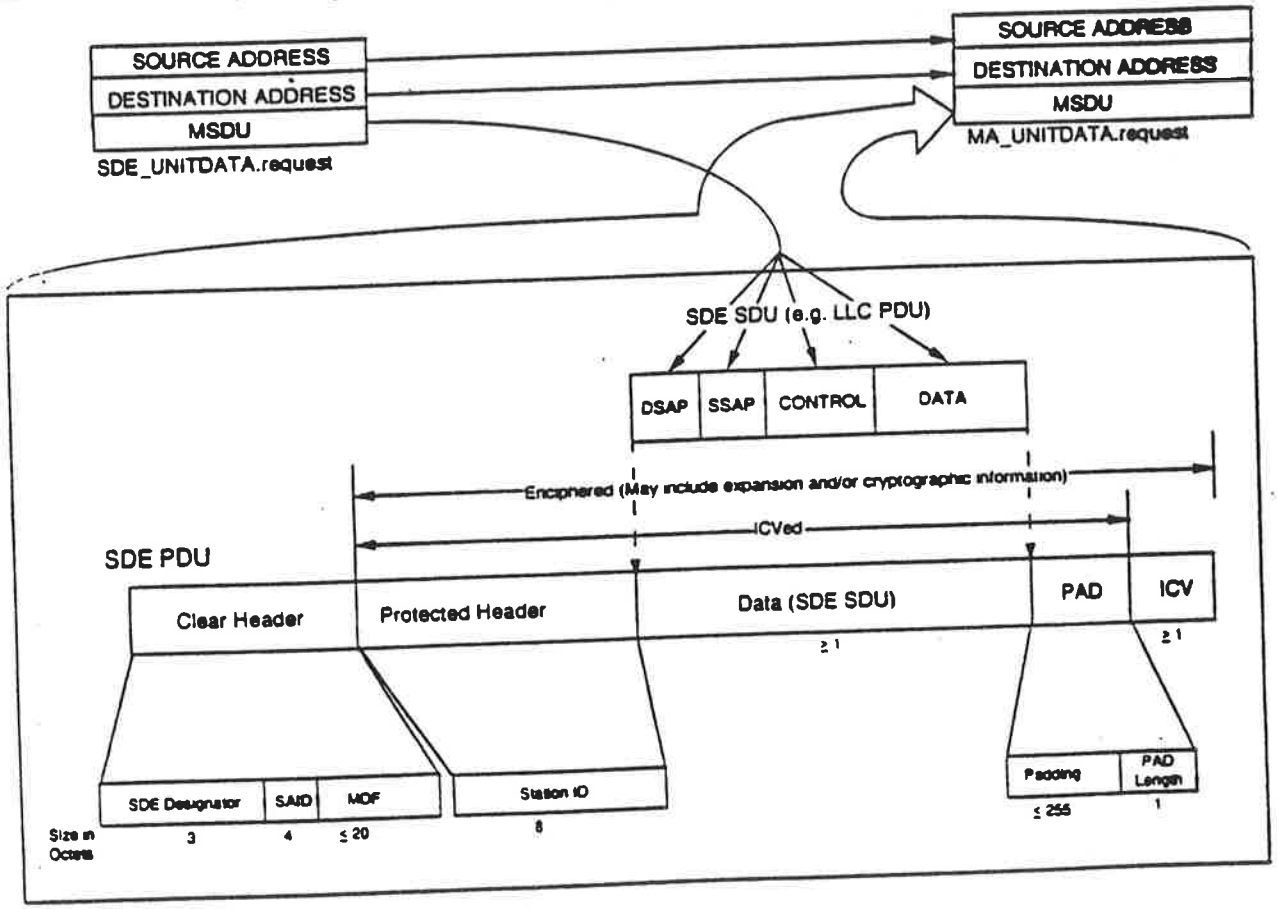


Fig 6
Construction of the SDE PDU

7. SDE Procedure

This section defines all elements of the SDE procedures, including transmission and reception processing and all other elements that direct those procedures. These other elements include management architecture, addressing, the SMIB, and the definitions of the managed objects.

Section 7.1 describes the SDE management architecture. The architectural description includes:

- the relationship between the management application entity and the SDE Layer Manager (LM),
- the role of the SMIB in their relationship,
- how security associations are coordinated through the use and the exchange of SAIDs, and
- the structure of the SDE managed objects.

The type of addressing used by the SDE entity is described in 7.2. The details of the SDE objects which are attributes of SDE managed objects are described in 7.3. Finally, 7.4 and 7.5 describe the transmission and reception procedures, respectively.

7.1 SDE Management Architecture. Each station that employs the SDE protocol has access to a Security Management Information Base (SMIB). The SMIB contains a list of the current security associations. Key Management and/or Security Management is responsible for maintaining this information base.

The SMIB provides the interface between the local System Management Application Entity (SMAE)[1] and the LM of the protocol stack. This is illustrated in Fig 7.

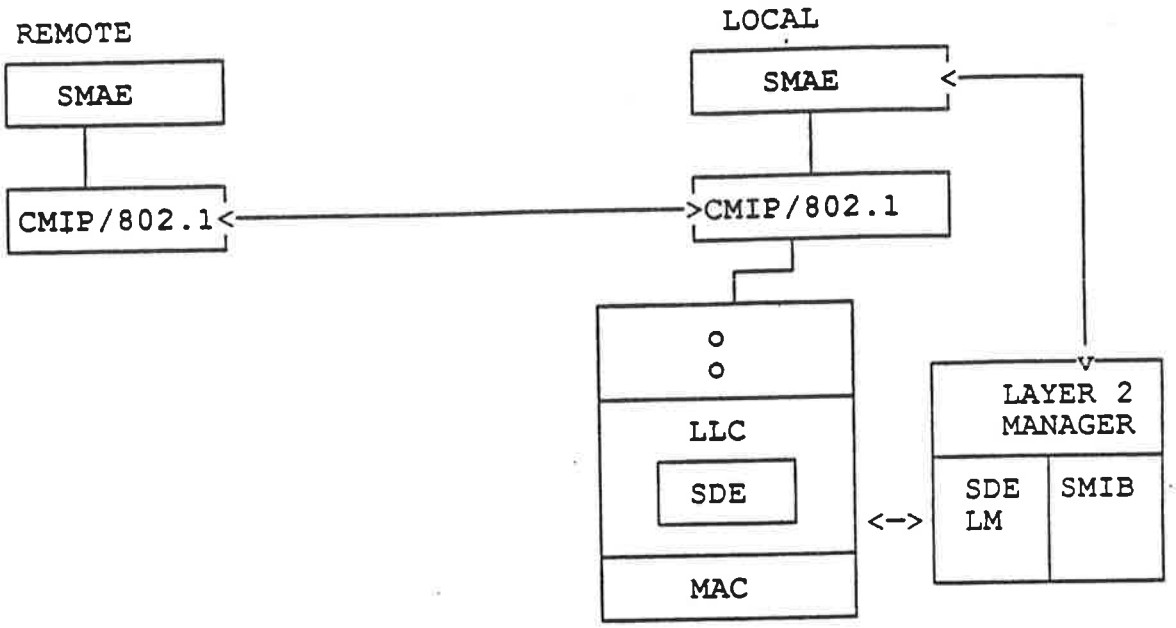


Fig 7
SDE Management Architecture

There are three types of SDE managed objects: station, Service Access Point (SAP), and security association. Station objects, which set certain parameters for the SDE entity, apply to all processing by the SDE entity. The SAP objects apply to a specific SAP. The security association objects apply only to a specific instance of PDU transmission, reception, or both.

Since multiple security associations can exist at any time, the SDE entity shall identify which security association applies to that SDE PDU. For example, this identification may be passed via the optional Security Association Identifier (SAID).

How the value of the SAID is coordinated between SDE entities is independent of the SDE protocol; however, it is useful to examine how a pairwise SAID could be established. During either a key or system management exchange, parties A and B exchange the values of the attributes of the security association managed object. These values specify the security parameters (e.g., the security services employed, keys, etc.) that will be needed for the security association. In this example, the SAID identifies this security association. This process is illustrated in Fig 8.

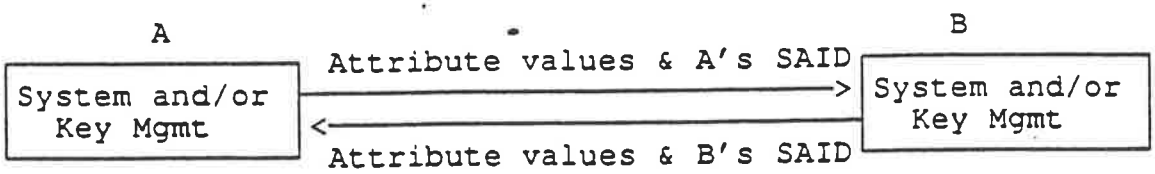


Fig 8
Initial Exchange

System and/or Key Management enters the value for the security association object into the SMIB. Fig 9 illustrates the SMIB which contains a table of security associations and the values of the associated attributes.

Attributes

Security Association #1	Security Association #1 Attribute values
o o o	
Security Association #n	Security Association #n Attribute values

SMIB

Fig 9
Security Associations

The security association shall be selected for each PDU transferred through the SDE entity. outgoing PDUs are PDUs that originate at one of the SDE Stacks (i.e., System Management Stack, Key Management Stack, or one of the User Stacks) and are outward bound to the MAC. Incoming PDUs are PDUs that arrive from the MAC and are to be delivered to one of these stacks. Incoming PDUs may contain the SDE Clear Header which can be used to select the security association; whereas the Clear Header may be created for Outgoing PDUs after the security association has been found. For this reason, the mechanism for selecting the security

associations can be different. Fig 10 shows the different parameters and/or PDU fields which can be used for selecting the appropriate security association from the SMIB.

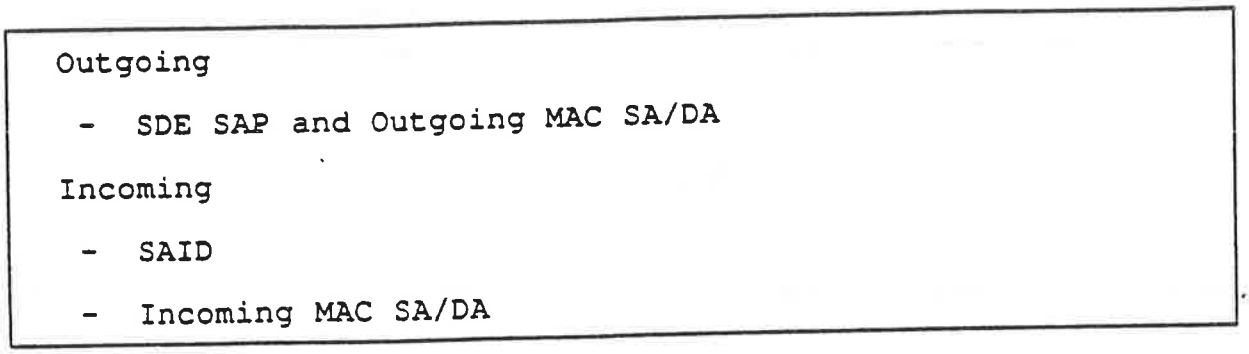


Fig 10
Parameters Used for Selecting Security Association

7.2 Addressing. All addresses referred to in this protocol are either Link Service Access Point (LSAP) addresses or MAC addresses. The LSAP address syntax and semantics are defined in IEEE 802.2 [4], while the specifics of the MAC addresses are defined in IEEE 802.1a [9].

The Station ID contains the MAC address corresponding to the individual address of the station that originated the outgoing PDU. In group transmissions with a shared secret key, the Station ID prevents parties external to the multicast group from tricking the receiving party into believing that the PDU came from a party other than its originator. It does not prevent members within the same group from changing PDUs so that they appear to have originated from a valid member of the same group. The inclusion of a Station ID also provides protection against reflection where that protection is not provided implicitly by the SDE confidentiality or integrity algorithms or from the services of the Key Management protocols.

7.3 SDE Objects. This section describes security managed objects as outlined in 7.1. Subsection 7.3.1 describes security objects that apply to the entire SDE; 7.3.2 describes security objects that apply to transmission to and from an SDE SAP; 7.3.3 describes security objects that are specific to the security associations; and 7.3.4 describes Security Association Identifiers (SAIDs).

7.3.1 Station Objects. The station objects apply to the entire SDE regardless of security association. The formal definition of each of the objects in the SDE standard will be defined by the SDE Layer Management Addendum. The objects described in this and the following two sections are abstractions provided for the purpose of describing the protocol processing. Some implementations may choose only manual management of these objects; in which case, the representation becomes a purely local matter. In this and the following two sections, the object names will be in boldface type.

1. **Station_Clear_Hdr**: Boolean. Station_Clear_Hdr=TRUE indicates that the Clear Header is always used when communicating with other SDE entities.

   Station_Clear_Hdr=FALSE indicates that there is no Clear Header expected on any incoming PDUs, and there is none placed on outgoing PDUs.

   For communication using this mode of the protocol, both stations shall agree to have Station_Clear_Hdr=FALSE. Delivery of SDE PDUs with no clear header (Station_Clear_Hdr=False) will have unpredictable results if the receiving entity is one of the following:

   - o Layer 2 entity not employing SDE
   - o SDE entity with Station_Clear_Hdr=TRUE.

2. Station_MDF: Boolean. Station_MDF shall be set to TRUE if the station sends or desires to receive the Management-Defined Field in the Clear Header. The actual inclusion or exclusion of the MDF is determined by the value of the Assoc_MDF attribute.

7.3.2 SDE SAP Objects. These objects may be defined by Layer Management and in Appendix E.

7.3.3 Security Association Objects. The SDE entity uses security associations available to it via the SMIB to provide the necessary services required for the secure transmission of data. The following are security objects that are attributes of a security association managed object:

1. **Local_SAID**: Octetstring. This contains the value of the SAID expected in incoming PDUs if Station_Clear_Hdr=TRUE.

2. **Remote_SAID**: Octetstring. This contains the value placed in the SAID field of outgoing PDUs if Station_Clear_Hdr=TRUE.

3. **Assoc_MDF**: Boolean. This indicates whether or not the Management-Defined Field is used for the security association. If Station_MDF=FALSE, then this Boolean is always FALSE. The length and value of the MDF field in the PDU are unidirectional characteristics of the security association. Key Management and/or System Management can force this Boolean to FALSE. If the Boolean is TRUE, the value of the following attribute is placed in the MDF of outgoing PDUs:

   a. **Remote_MDF**: Octetstring. This attribute contains the value that will be placed in the MDF field in the Clear Header if the Assoc_MDF=TRUE.

4. Protection Set: These attributes indicate the security services to be provided by SDE.

   a. **Confid**: Boolean. If TRUE, it indicates that data confidentiality is to be provided for the security association.

   b. **Integ**: Boolean. If TRUE, it indicates that connectionless integrity is to be provided for the security association.

5. Security Fields Present: Booleans indicate the presence (TRUE) or absence (FALSE) of security fields. These values shall remain constant over the life of the security association.

   a. **Padding_pres**: Boolean. Flag for the PAD Length field.

   b. **ID_pres**: Boolean. Flag for the Station ID.

6. Confid_Alg_ID: Octetstring. This is a label that specifies a complex object corresponding to a confidentiality algorithm if Confid=TRUE. The definition of the algorithm shall include everything that is necessary for the encipherment or decipherment to occur. This includes, but is not limited to, the length and placement of Initialization Vectors, block size, and mode of operation.

7. **Integ_Alg_ID**: Octetstring. This is a label that specifies a complex object corresponding to an integrity algorithm if Integ=TRUE. The definition of the algorithm shall include everything that is necessary for the ICV to be calculated and verified upon receipt. This includes, but is not limited to, the length and placement of Initialization Vectors, block size, and mode of operation.

8. **SDE_SAP**: Octetstring. This indicates the SDE SAP for the security association. This is used as part of the index into the SMIB for outgoing PDUs. On incoming PDUs, it indicates which protocol stack should receive the PDU.

9. **Remote_SDE**: Boolean. This boolean is TRUE if the remote entity implements SDE protocol and is FALSE otherwise.

10. **Outgoing_Source_MAC_Address:** Octetstring. This corresponds to the individual address of the station that originated the outgoing PDU. It is the value included in the Station ID field of the Protected Header.

11. **Outgoing_Destination_MAC_Address:** Octetstring. This address may be an individual or group address associated with the remote station(s).

12. **Incoming_Destination_MAC_Address:** Octetstring. This may be an individual or group address associated with the local station.

13. **Incoming_Source_MAC_Address:** Octetstring. If the Incoming_Destination_ _MAC_Address is an individual address, this object contains a single individual address. If the Incoming_Destination_MAC_Address is a group address, this object contains a list of individual addresses.

Within the SDE entity, the security association is represented by the security association object. Changing the values of any of the security association attributes (or attributes of the complex objects labelled by the Confid_Alg_ID and the Integ_Alg_ID attributes) causes a new security association to be formed and the prior security association to be invalidated. The SAID is a convenient tag for the identification of these objects.

7.3.4 Security Association IDs (SAIDs). The SAID is primarily used to identify the security association, although it can be used for other purposes. In security associations between two entities, each entity chooses its own SAID and communicates it to the remote entity during a system and/or key management exchange. In security associations for multicast or broadcast groups, it is the responsibility of system management and/or key management to assign and

coordinate the SAID used for that multicast or broadcast group address. Half of the possible values of the SAIDs are reserved as group SAIDs (see Fig 5).

There are bootstrap values (see 6.2.1.2) for the SAID that are sometimes used for communications with the System Management and/or Key Management Stacks. The communications under these bootstrap SAIDs have no security protection (confidentiality, integrity) and do not have a Station ID. In addition, no padding can be applied.

7.4 Transmission Procedures. The transmission procedures are those involved in processing an SDE_UNITDATA.request. The functions are represented as a flow chart shown in Fig 11. (Object values are contained in the SMIB.) Also, Appendix B contains an example of the transmission and reception procedures using specific algorithms.

In response to an SDE_UNITDATA.request from the LLC sublayer, the supplied address parameters and/or the SDE SAP is used to search for a security association in the SMIB.

A. If the search is successful, a security header comprised of a Clear Header and a Protected Header may be created and prepended to the Data field which contains the MSDU of the request. The options of integrity and/or confidentiality may be provided. A PAD may be created and an ICV may be computed and both are appended to the Data Field. The Protected Header, Data, PAD, and the ICV may be enciphered. Finally, a MA_UNITDATA.request is constructed and passed on to the MAC sublayer.

B. If no security association is found, the SDE Layer Manager is notified.

If the expansion causes the PDU to exceed the maximum size the MAC will accept, fragmentation may be required. Fragmentation is not part of this standard; however, if it is implemented, the method of fragmentation specified in Appendix E is the recommended approach.

7.4.1 Obtaining the Security Association. The security association shall be retrieved from the SMIB if a security association exists. The outgoing MAC addresses and the SDE SAP are used to search for the security association in the SMIB. If a security association is found, the values for each object of the security association are returned. If the request is originated by system management and/or key management, the SMIB may contain a bootstrap SAID security association that will allow communication. If no security association is found corresponding to the SDE SAP and addresses specified in the SDE_UNITDATA.request, the SDE entity indicates the error to the SDE Layer Manager.

From this point on, it will be considered that an appropriate security association is already established.

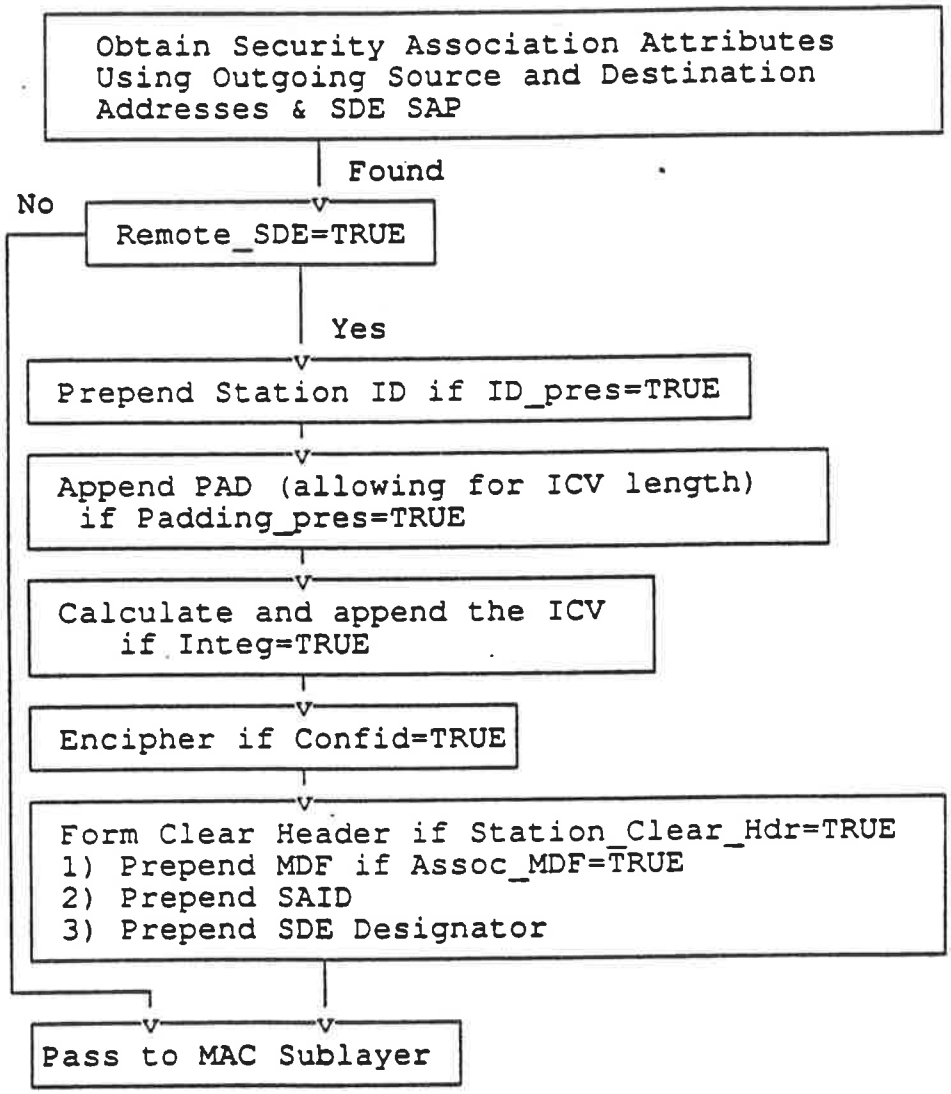


Fig 11
Transmission of an MA_UNITDATA.request

7.4.2 Transmission to Non-SDE Entities. If Remote_SDE = FALSE, bypass further SDE processing and pass the SDE_UNITDATA.request to the MAC sublayer.

7.4.3 Forming the Protected SDE Header. After the security association is retrieved from the SMIB, the Protected Header is formed and prepended to the Data field specified in the SDE_UNITDATA.request.

If ID_pres=TRUE, the Outgoing_Source_MAC_Address is placed in the Station_ID field. The Station_ID is an optional field.

7.4.4 PAD. If padding is required by the security association (Padding_pres=TRUE), the maximum size of PAD is 256 octets (255 Padding octets plus a one octet PAD Length field). PAD may be used to expand the size of the outgoing PDU for the integrity algorithm, for the confidentiality algorithm, or in a local manner.

7.4.5 Calculation of the ICV. If integrity should be applied (Integ=TRUE), the Integrity Check Value (ICV) is computed using the algorithm specified in the SMIB over the Protected Header, Data, and PAD. The ICV is appended to the Data field.

7.4.6 Encipherment of the PDU. If confidentiality is an attribute of the security association (Confid=TRUE), then the Protected Header, Data, PAD, and ICV will be enciphered using the algorithm specified in the SMIB.

7.4.7 Clear Header. The Clear Header is used both to signal the remote SDE entity that the PDU had been processed by the local SDE entity and to supply the necessary information to determine the appropriate security association. If Station_Clear_Hdr=TRUE, the Clear Header is placed in the outgoing PDU.

MDF: If Assoc_MDF=TRUE, then the Remote_MDF is placed in the outgoing PDU. It is an optional field.

Security Association Identifier (SAID): The Remote_SAID shall be placed in the SAID field of the PDU. It is a mandatory field when the Clear Header is present.

SDE Designator: The SDE Designator is placed as the first three octets in the outgoing PDU. It is a mandatory field when the Clear Header is present.

7.4.8 MAC Request. The SDE PDU is passed to the MAC sublayer as the MSDU parameter in the MA_UNITDATA.request. All other parameters are passed through, unaltered, by the SDE entity.

7.5 Reception Procedures. When an MA_UNITDATA.indication is received from the MAC sublayer, processing can vary depending on the local management functions. The security association shall be identified, and the appropriate security mechanisms are applied to the PDU. If appropriate, the PDU is deciphered and the ICV is checked. Finally, the SDE_UNITDATA.indication is forwarded to the designated stack.

If any security-relevant exceptions are encountered during processing by the SDE entity, the PDU in question is discarded and the SDE Layer Manager is notified.

The reception functions are those involved in processing an MA_UNITDATA.indication (see Fig 12).

7.5.1 Requirements for Reception. Before a station can process an incoming SDE PDU, a security association shall exist for communication to be allowed. Note that it is possible to configure the SMIB such that loss of information in the SMIB (e.g., power fail) could prevent automated recovery.

The bootstrap values of the SAID shall have a security association in the SMIB. There are four bootstrap values: Individual Key Management, Group Key Management, Individual System Management, and Group System Management.

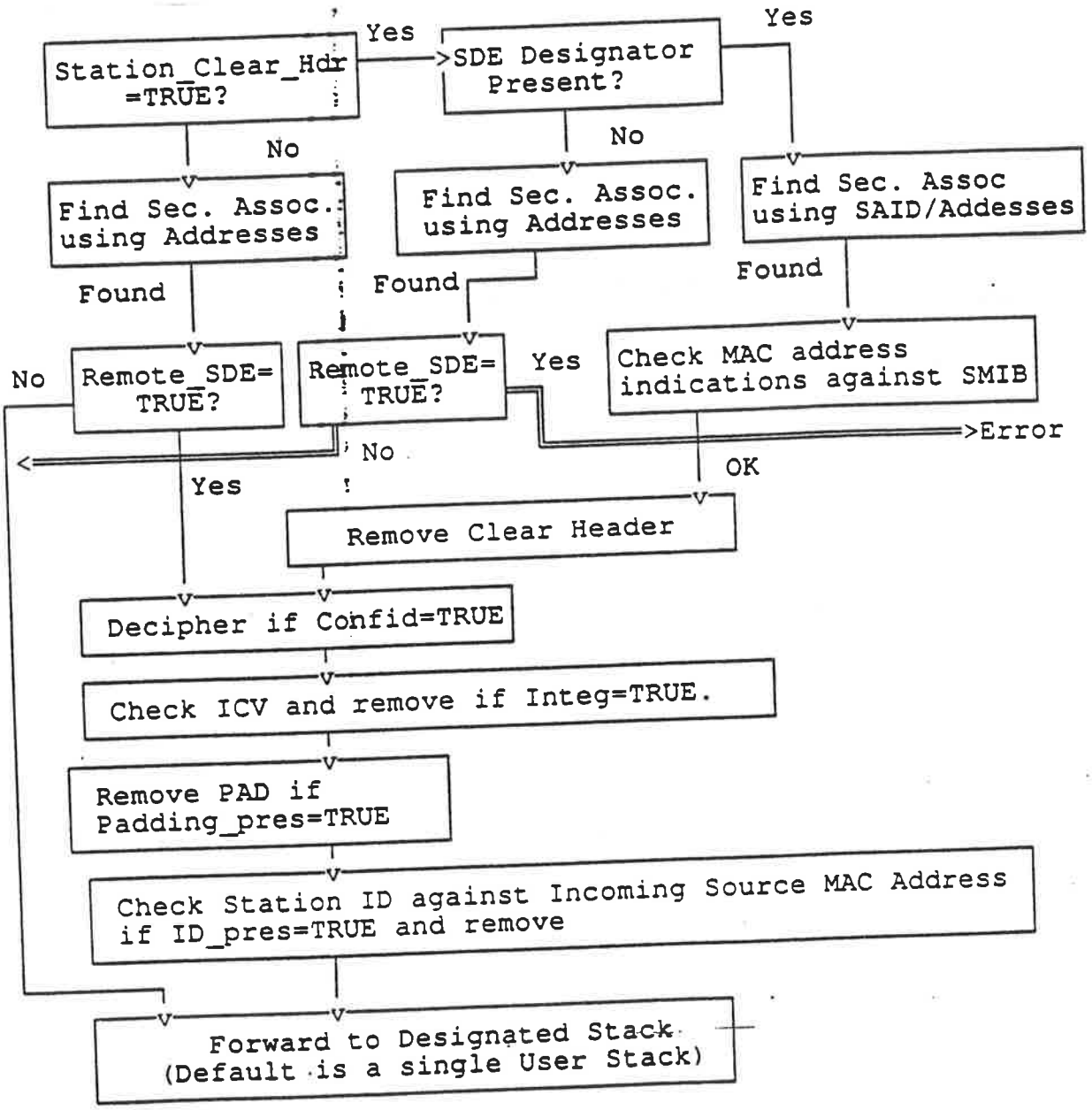


Fig 12
Reception of an MA_UNITDATA.indication

Wild-card entries in the SMIB (i.e., generic entries corresponding to multiple MAC addresses) may be allowed depending on local policy.

7.5.1.1 Station Configured for Clear Header. If Station_Clear_Hdr = TRUE, then the presence of an SDE Designator indicates that an SAID may be used for finding the security association. A security association shall exist for all communications, even with non-SDE entities. Communication with a non-SDE entity will bypass the rest of the security processing and be forwarded to the stack designated by the security association.

The SDE entity checks the Source and Destination Address parameters in the MA_UNITDATA.indication against those denoted by the security association (Incoming_Source_MAC_Address and Incoming_Destination_MAC_Address). The security association in the SMIB may indicate the presence of a MDF. The MDF is used in a locally determined manner. The Clear Header is removed before the PDU is deciphered.

7.5.1.2 Station Configured with No Clear Header. If Station_Clear_Hdr=FALSE, then the security association and the correct protocol stack shall be determined based on source and Destination Addresses in the MA_UNITDATA.indication.

7.5.2 Decipherment of the PDU. If Confid=TRUE, the confidentiality algorithm is selected from the SMIB, and the PDU is deciphered.

7.5.3 ICV Checking. If Integ=TRUE, the PDU is assumed to have an ICV which shall be checked and removed using the chosen algorithm retrieved from the SMIB. If the ICV fails, the SDE Layer Manager is notified.

7.5.4 PAD. The SDE entity strips any PAD that may be present in the PDU.

7.5.5 Station ID. If ID_pres=TRUE, the SDE entity checks that the contents of the Station ID field are the same as the source address in the MA_UNITDATA.indication. The Station ID field is removed.

7.5.6 SDE_UNITDATA.indication. The parameters received in the MA_UNITDATA.indication are passed up to the appropriate protocol stack in the SDE_UNITDATA.indication with the SDE SDU (e.g., the LLC PDU) replacing the received MSDU.

8. Minimum Essential Requirements (MERs)

The MERs are stated in terms of the values of certain management objects in Section 8.1 and 8.2. Additional MERs are contained in Section 8.3 and 8.4. These objects are abstractions used to represent the options for the SDE entity. These MERs do not mean that the objects shall be managed remotely. The effect of setting the object to a particular value shall affect the protocol state as described in the standard. When constrained to the values specified in the following two sections and combined with transmission and reception processing, these objects give the minimally compliant protocol state machine.

8.1 Station Objects.

1) Station_Clear_Hdr: Boolean.

SDE entities shall allow the Station_Clear_Hdr to be TRUE. Entities with Station_Clear_Hdr set to TRUE are not interoperable with stations that have Station_Clear_Hdr set to FALSE.

2) Station_MDF: Boolean.

All entities shall allow the Station_MDF to be FALSE. Entities may support TRUE, but then values for each individual security association are determined by System and/or Key Management. If Station_Clear_Hdr is set to FALSE, then Station_MDF shall also be set to FALSE.

8.2 Security Association Objects.

1) Assoc_MDF: Boolean

This attribute is TRUE if the MDF will be used on the security association. Any party in the negotiation can force the MDF not to be used. The protocol processing shall not depend on the presence of the MDF in any implementation. Each entity shall have the capability of communicating with this attribute set to FALSE. If TRUE is supported, the entity shall have the capability of supporting an integral length (in octets) from 0 to 20.

2) Protection Set: Includes Confid and Integ Booleans

An entity shall be capable of operating with at least one security association having a TRUE

value in at least one of these two Booleans. Entities implementing only Integrity, or only Confidentiality, shall be considered conformant.

3) Padding_pres: Boolean.

PAD is mandatory only if either the integrity or confidentiality algorithm requires padding. Thus, some entities may support this object only being TRUE, and others may support it only set to FALSE. Still others may support both. If the TRUE value is supported, the entity shall be able to accept a maximum length of PAD (256 including PAD Length field)[7]. Negotiation of the Padding_pres by key management and/or system management may allow the value to be set to FALSE where neither cryptographic algorithm requires padding.

4) ID_pres: Boolean.

A device shall be capable of supporting ID_pres=FALSE.

8.3 General Statements.

1) All entities shall implement the protocol processing steps and fields not designated as optional within the standard.

2) All entities shall support the reception of bootstrap, group, and individual SAIDs. In systems with key management appearing on the User Stack, the SDE entity associates the bootstrap SAID with the appropriate stack identified in the SMIB or discards the PDU.

8.4 Security Services. Compliant entities shall support at least the Data Confidentiality Service or the Connectionless Integrity Service.

1) To claim that the entity provides the service of Data Confidentiality, the entity shall allow Confid to be TRUE. The strength of this service is dependent upon the confidentiality algorithm used.

2) To claim that the entity provides the service of Connectionless Integrity, the entity shall allow the Integ to be TRUE. The strength of this service is dependent upon the integrity algorithm used. The entity shall be able to send and receive PDUs with ICVs.

[7] The maximum PAD length must be specified due to its effect on stream processing and buffer sizes. It cannot be restricted to the blocksize of the Integrity or Confidentiality algorithm since this would defeat the objective of algorithm independence and require conformance testing to be tied to a particular cryptographic algorithm.

APPENDIX A--Service Rationale

This appendix contains the rationale for the selection of SDE security services.

LAYER 2 SECURITY SERVICES FOR LOCAL AREA NETWORKS

ABSTRACT

The ISO Security Architecture, ISO 7498-2, was developed using Packet Switched Networks (PSNs) and Wide Area Networks (WANs) as architectural models. Since that time, there have been significant changes in networking practices. Local Area Networks (LANs) have introduced a new range of vulnerabilities that are not present in the Data Link Layer of PSNs and WANs . The point-to-point nature of the Data Link Layer (Layer 2) of PSNs and WANs led to the dismissal of the need for extensive security services at Layer 2. Subnetworks and routing were the focus of the need for inclusion of particular security services at the Network and Transport Layers. However, LANs have introduced subnetworks and routing into the Data Link Layer of many networks. Efforts aimed at providing security services for LANs have found the current Link Layer security service profile in ISO 7498-2 to be deficient. It is necessary to expand this service profile to protect LANs, even in the presence of security services at higher layers in the protocol stack.

INTRODUCTION

In the spring of 1988, preliminary meetings were held to determine interest in security standards for Local Area Networks (LANs). This led to the formation of the IEEE 802.10 LAN Security Working Group, which is sponsored jointly by the IEEE 802 Technical Committee and the IEEE Technical Committee on Security and Privacy. The working group's charter is the development of Standards for Interoperable LAN Security (SILS).

Since its formation, the LAN Security Working Group has concentrated on development of a Secure Data Exchange (SDE) protocol to be inserted between the Media Access Control (MAC) and the Logical Link Control (LLC) sublayers of the link layer in the ISO OSI Basic Reference Model. The working group has recently begun development of a key management protocol and a security management protocol, as well.

In the course of the development of the SDE protocol, the LAN Security Working Group drew up a list of necessary security services. In large part, this list was based on the attributes of emerging LAN security devices. An analysis of the attributes of LANs which make these security services necessary is presented in this appendix. The pertinent attributes are identified and the associated security threats are detailed. Then, the security services necessary to counter those threats are indicated, examples of the benefits of application of those security services are given, and mechanisms for providing the services are discussed.

SECURITY SERVICES UNDER THE ISO SECURITY ARCHITECTURE

ISO 7498-2 identifies five basic security services: access control, authentication, data confidentiality, data integrity, and non-repudiation. These services provide assurance against the security threats of unauthorized resource use, masquerade, unauthorized data disclosure, unauthorized data modification, and repudiation, respectively. This standard also defines the layers within the ISO OSI Basic Reference Model where it is appropriate to apply these services. Appendix B of ISO 7498-2 gives a brief justification for the indicated service placement.

In ISO 7498-2, data confidentiality is the only security service indicated for the Data Link Layer of the ISO OSI Basic Reference Model. Other security services were "not considered useful" at this layer. This appendix details arguments for the inclusion of the services of authentication, access control, and data integrity at the Data Link Layer, as well. It is important to note that the arguments presented in this appendix are based on changes in networking practices since ISO 7498-2 was completed, not on deficiencies intrinsic to ISO 7498-2 as it was originally conceived. LAN standards have only recently begun to appear in the ISO standards arena

(e.g., ISO 8802-2, ISO 88027498-2). Because of changes in of LAN technology, the risks to LANs have become more critical than first considered. High-speed, long distance LANs (e.g., the Fiber Distributed Data Interface, or, FDDI), filtering LAN bridges, and LAN server facilities have increased the range of resources which are vulnerable to abuse. Ring topology networks not only make every Protocol Data Unit (PDU) (e.g., packet, frame) available to every station on the LAN, but require every station on the LAN to receive and then forward every PDU, in order for the LAN to operate properly. These issues have prompted the concerns that lead to this set of arguments. Figure 1 illustrates the differences between the security service profile defined in ISO 7498-2 and the profile proposed for LANs.

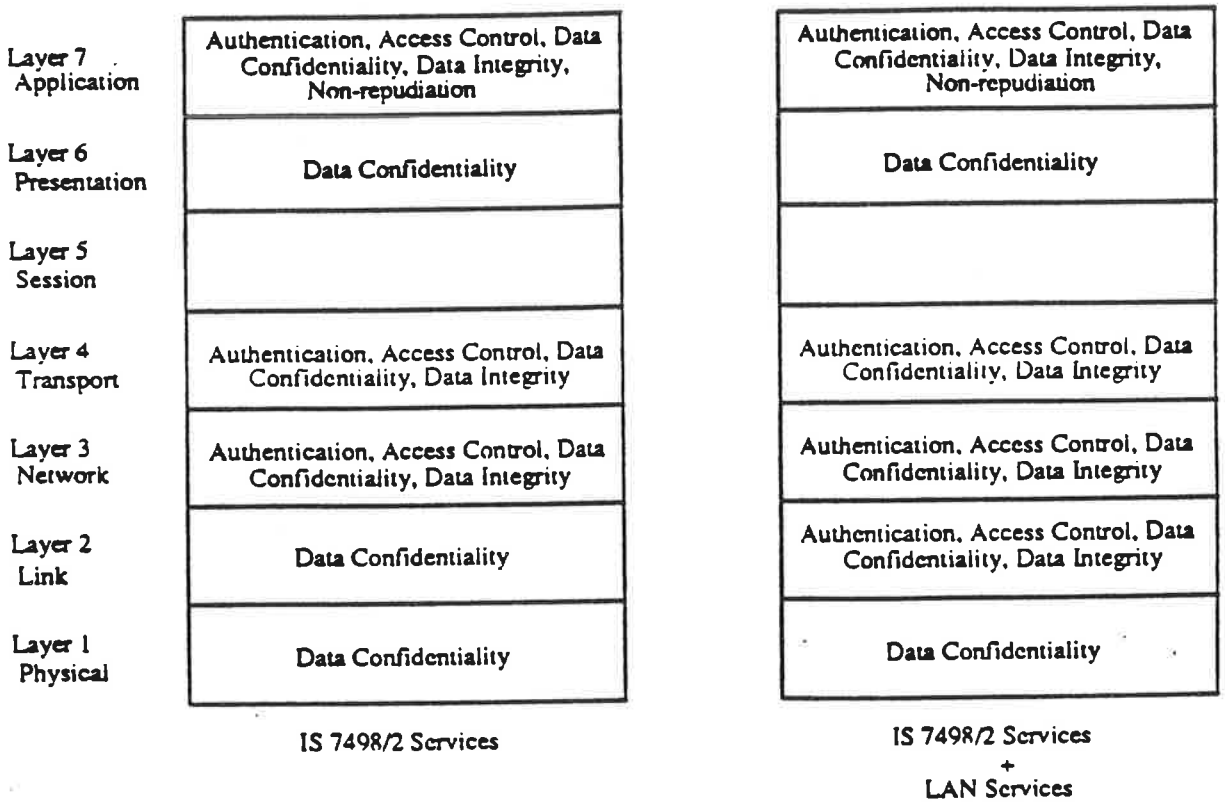


Figure 1

In a specific implementation, a security service can be implemented in any layer at which it is indicated. A service may appear in one layer, more than one layer, or not at all. ISO 7498-2 only indicates where the service can appear, not where the service is required to appear. The security requirements for a particular implementation will determine where the services will be provided. In practice, it is desirable to protect information both at the highest possible point in the protocol stack (i.e., the application layer) and any layers at which subnetworks and routing are implemented.

The ISO Security Architecture was developed using PSNs and WANs as an architectural model. It was assumed that these networks would have a tightly controlled Data Link Layer configuration. In this model, the HDLC Frame was used to represent the Data Link Layer PDU.[1] It was also assumed that the Data Link Layer of LANs had the same attributes as the Data Link Layer of the model. In fact, while LANs are similar to PSNs

[1] While this simplified model may not represent all possible implementations of PSNs and WANs, it does represent the mapping of many PSNs and WANs onto the ISO OSI Basic Reference model. X.25 Packet Level Interface functions are attributed to the Network Layer. The assumption of tightly controlled configurations, in particular, may seem restrictive, but reflects standard practices in the implementation of secure networks.

and WANs at the Data Link Layer, they also exhibit some of the attributes of the Network Layer of PSNs and WANs. For example, the Data Link Layer of LANs exhibits subnetwork and routing functions very similar to those of the Network Layer. These functions are cited as justification for the Network Layer security service profile, which is the same as the security service profile proposed in this appendix for the Link Layer. These similarities and differences are indicated in the following sections as the security-pertinent attributes of LANs are explored.

LAN CHARACTERISTICS THAT NECESSITATE SECURITY SERVICES AT THE DATA LINK LAYER

There are certain characteristics of LANs that necessitate security services at the Data Link Layer: the manner in which data is transmitted, the manner in which data is received, the nature of LANs' address space, and geographic dispersion of LANs. The security threats associated with these characteristics will be identified. Then, the security services required to address these threats will be indicated and how they are applied to LAN data will be shown. Finally, mechanisms for providing these services will be discussed.

DATA TRANSMISSION ON A LAN

The manner in which data is transmitted on LANs is one of the attributes that necessitates additional security services at Layer 2. In a LAN's Data Link Layer, data is transmitted on media that is shared by every attached system. Effectively, every PDU is transmitted to every other station on the LAN and the source of a given transmission is difficult to authenticate.

The nature of data transmission at the Data Link Layer on a LAN presents two security threats. First, any station attached to a LAN can transmit to any other station attached to the LAN. There are no implicit controls at Layer 2 on access to a resource attached to a LAN. Second, since it is difficult to identify the source of a given data transmission, one station can claim to be another station. Any station, or set of stations, can be imitated from a single tap into the LAN. The source of a given PDU is difficult to authenticate. These threats to the security of a LAN are known formally as unauthorized resource use and masquerade.

DATA RECEPTION ON A LAN

The manner in which data is received on LANs, is another attribute that necessitates additional security services at Layer 2. Since data transmission at a LAN's Data Link Layer is over commonly accessible media, every PDU is available to all attached stations. A PDU could traverse any station on its way to its destination. This means that while it may be addressed to a specific entity, every PDU is effectively received by every other station attached to the LAN.

The nature of data reception on a LAN presents two security threats, since any PDU could be intercepted by any attached station. First, a station could receive data for which it is not authorized. Second, and worse yet, a station could change the data in a PDU before it is received at its intended destination. On LANs, data for any station, or set of stations, can be received from a single station on the LAN. This is especially significant in LANs employing a ring topology, where every attached system must receive and retransmit every PDU in order for the LAN to function properly. These threats to the security of a LAN are known formally as unauthorized disclosure and data modification.

LAN ADDRESS SPACE

Assignments within the address space of a LAN are also pertinent to security. Each station interface is permanently assigned a specific address. Since any station interface can be attached to any other station interface through a common medium at Layer 2, LAN addresses must be unique at Layer 2. This means that a station cannot determine, by observation, whether the source address of a PDU is valid or not. There is no hierarchical address assignment in LANs, so any possible link address could be valid on any LAN.

As with data transmission, the nature of address assignment at the Data Link Layer on a LAN presents two security threats. First, any station attached to a LAN can transmit to any other station attached to the

LAN. There are no implicit controls at Layer 2 on access to a station attached to a LAN. Second, since it is difficult to identify the source of a given data transmission, one station can claim to be another station. Any station, or set of stations, can be imitated from a single tap into the LAN. The source of a given PDU is difficult to authenticate. These threats to the security of a LAN are known formally as unauthorized resource use and masquerade.

GEOGRAPHIC DISPERSION OF LANS

LANs span vast geographic areas, rendering them vulnerable to eavesdropping or wiretap. This renders them vulnerable to the threats of unauthorized disclosure and data modification. As indicated previously, there is a significant scope of information and access available on a LAN at Layer 2; any station, or set of stations, can be imitated from a single tap into the LAN.

Wiretapping on a LAN presents two security threats. First, a station can receive data for which it is not authorized. Second, and worse yet, a station can change the data in a PDU before it is received at its intended destination. Again, on LANs, data for any station, or set of stations, can be received from a single tap into the LAN. This is especially significant in LANs employing a ring topology, where every attached system must receive every PDU for the LAN to function properly. These threats to the security of a LAN, are known formally as unauthorized disclosure and data modification.

SECURITY SERVICES

In this section, the type of architecture which requires the indicated security services will be described, the security services themselves will be described in detail, and the formal definition of each service from the ISO Security Architecture will be reviewed. Also, the application of each service to PDUs at the Data Link Layer on a LAN will be examined, making note of the portions of a PDU that are protected by the service.

In figure 2, a LAN has been subdivided into several local segments, or subnetworks, that are interconnected through a backbone network. The subnetworks are effected through the use of bridges, which pass a PDU between a subnetwork and the backbone network only when that PDU is directed from a station on one side of the bridge to a station on the other side of the bridge. Some of the subnetworks have been designated as protected subnetworks, i.e., subnetworks that are safe from attachment of unauthorized stations, as opposed to unprotected networks.

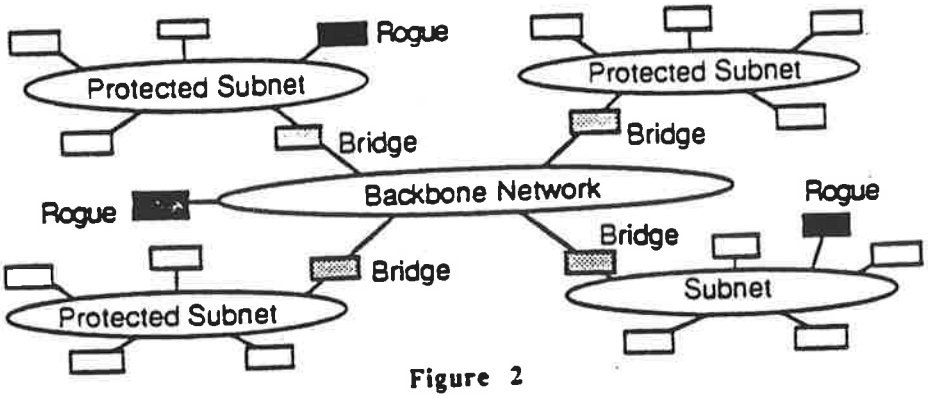


Figure 2

Rogue stations are those that participate in unauthorized activities, whether or not the station is authorized to be attached to the LAN. These rogue stations exploit the risks that have been identified, necessitating the indicated security services. Precautions are necessary to provide protection from these stations wiretapping into the backbone LAN. LAN security services are also necessary to prevent abuse by systems which are authorized to be connected to the LAN, but are being used in an unauthorized fashion. Without the proper security services, even protected subnetworks are susceptible to abuse.

Ultimately, protection of application data can be provided at the application layer. However, in practice, it is desirable to protect information both at the highest possible point in the protocol stack (i.e., the application layer) and any layers at which subnetworks and routing are implemented. This is true for several reasons.

First, security services provided at any layer of a protocol stack, protect only the Service Data Unit (SDU), i.e., the data portion, of that layer's PDU. If data integrity is provided at an upper layer, the header information from that layer and all lower layers is left unprotected. One example of data in a Layer 2 information PDU that is unprotected, even in the presence of higher layer security services, is the security option specified for ISO CLNP, which is included in the U.S. Government Open Systems Interconnection Profile (U.S. GOSIP). Since this data is contained within the Network Layer header, it cannot be protected by security services provided above the Data Link Layer.

Second, PDUs that originate and terminate within Layer 2 are also unprotected in the presence of security services at upper layers. Examples of this type of PDU are the TEST and XID PDUs in ISO 8802-2 LLC, which is also part of the U.S. GOSIP. Network management uses these PDUs, creating a need for protection for this type of PDU as well as information PDUs. ISO 7498-2 considers only information PDUs. It does not address administrative functions and artifacts of protocols. Connectionless data integrity at the Link Layer will provide protection for this type of PDU, as well as information outside the boundary of protection of higher layer security services.

Third, security services provided at the Link Layer provide uniform, common protection for all applications from risks that are intrinsic to LANs and the increased connectivity they provide. Security services provided at another layer can neither take advantage of the attributes of a LAN nor be affected by the deficiencies of a LAN.

Finally, implementations of security at upper layers are developing too slowly to address some users' needs. Emerging LAN security devices can address these needs until upper layer security is available.

CONNECTIONLESS DATA INTEGRITY

ISO 7498-2 defines connectionless data integrity as "the property that the data in a single connectionless PDU has not been altered or destroyed in an unauthorized manner." As the definition indicates, this service inhibits undetected modification of the protected data. This assures the receiving station that the SDU portion of a PDU has not been tampered with since it was transmitted. Given the nature of data transmission and reception at the Link Layer of LANs and the susceptibility of LANs to wiretap, this service is badly needed to protect data on LANs. This service is important not only in its own right, but as a necessary supportive service for authentication services.

Figure 3 illustrates the application of this service to information PDUs. As previously indicated, security services provided at any layer of a protocol stack protect only the SDU portion of that layer's PDU. In implementations where integrity is provided at a higher layer, connectionless data integrity at Layer 2 protects the headers of the layers above the MAC Sublayer up to and including the higher layer at which integrity is provided. The security option specified in the U.S. GOSIP for ISO CLNP is one example of critical data protected in this case. Since this data is contained within the Network Layer Header, it cannot be protected by security services provided above the Link Layer. Modification of the data contained in the security option, combined with the modification of the CLNP header checksum could result in delivery of a PDU to a station not authorized to process that data. In implementations where connectionless data integrity is provided at the Link Layer rather than at a higher layer, application data and all of the headers of the protocol layers above the MAC Sublayer are protected from undetected modification. When implemented at the Data Link Layer, this service also provides protection for logical subnetwork addressing for communities of interest on a common secure backbone LAN.

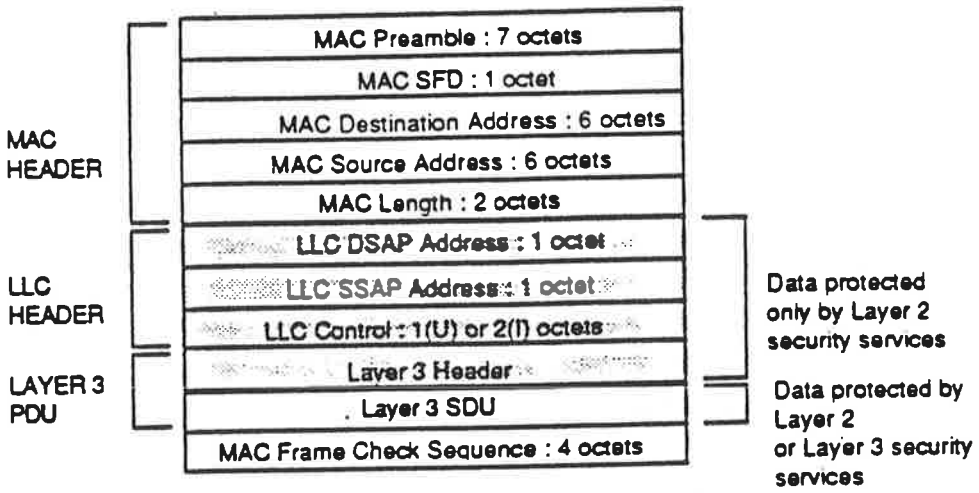


Figure 3

Connectionless data integrity is also necessary at the Data Link Layer to inhibit data modification of the data field of the TEST PDU. Figure 4 illustrates the application of connectionless data integrity to this type of PDU. If the data in a TEST PDU is altered by a third party, either during the request or reply phases, it might result in a bad quality path being marked as good. Distortion of TEST data could also cause a good quality path to be marked as bad, but this is indistinguishable from a failure in the media itself and is, in fact, an indication that there is something wrong with the communications path, anyway. This service also protects the integrity of the LLC header fields, preventing misdelivery of the TEST PDU or modification of the Control field, which identifies the PDU as a TEST PDU. Finally, integrity is also necessary as a supportive service for authentication of this type of PDU, since assurance of authenticity of the source address without assurance of the integrity of the source address is of little value.

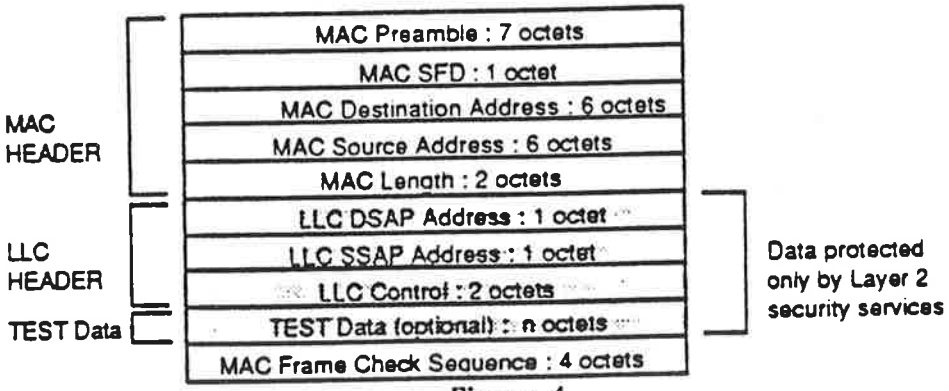


Figure 4

DATA ORIGIN AUTHENTICATION

Data origin authentication inhibits one station from masquerading as another to abuse resources attached to a LAN (i.e., unauthorized resource use). This service assures a receiving station that the SDU portion of a PDU came from the station indicated by the Data Link Layer source address in the PDU header. Data integrity is necessary as a supportive service for data origin authentication, since assurance of authenticity of the source address without assurance of the integrity of the source address is of little value. This service protects resources (e.g., file servers) attached to LANs from one station masquerading as another, whether or not the station is authorized to be connected to the LAN. At Layer 2, this service provides protection for logical subnet addressing for communities of interest on a common secure backbone. Given the nature of data transmission and reception at the Link Layer of LANs and the susceptibility of LANs to wiretap, this service is necessary to protect resources on LANs.

Figure 3 illustrates the application of this service to information PDUs at the Data Link Layer. When authentication is provided at an upper layer, the header data from that upper layer and all lower layers, is left unprotected. Again, an example of data in a Layer 2 information PDU that is unprotected even in the presence of higher layer security services, is the security option specified in the U.S. GOSIP for ISO CLNP. Since this data is contained within the Network Layer Header, it cannot be protected by security services provided above the Link Layer. If an unauthorized station masqueraded as an authorized station and replayed the data contained in the security option from a valid PDU, it could result in delivery of data to a station not authorized to process that data. In implementations where data origin authentication is provided at the Link Layer rather than at a higher layer, application data and all of the headers of the protocol layers above the MAC Sublayer are protected. When implemented at the Link Layer, this service also provides protection for logical subnet addressing for communities of interest on a common secure backbone LAN.

Data origin authentication is also necessary at Layer 2 to inhibit modification of the source address field of the source address field of a TEST PDU. Figure 4 illustrates the application of data origin authentication to this type of PDU. If the source address in a TEST PDU is altered, either during the request or reply phases, it might result in a bad quality path being marked as good. Misrepresentation of the source address in a TEST PDU could also cause a good quality path to be marked as bad, but this is indistinguishable from a failure in the media itself and, in fact, is an indication that there is something wrong with the communications path, anyway. Together with the supportive service of integrity, data origin authentication provides necessary protection for this type of PDU, since assurance of authenticity of the source address without assurance of the integrity of the source address is of little value.

ACCESS CONTROL

Access control inhibits unauthorized use of resources. This service is sometimes thought of as a way to inhibit unauthorized disclosure. But, in fact, data confidentiality is used to protect data from unauthorized disclosure. Access control provides assurance that access to a resource is granted only to authorized stations for authorized purposes. Access control can be applied at either the source of a data transmission or at the destination. However, when access control is applied at a PDU's destination, the data has effectively been transmitted to all stations on a LAN before this service is applied. If nothing else, this leaves stations open to unauthorized depletion of network bandwidth and receiver processing resources. Also, due to the manner in which every PDU is effectively transmitted to every station on a LAN and the susceptibility of LANs to wiretap, access control applied at the destination cannot prevent transmissions to stations not authorized to be connected to a LAN. At the Data Link Layer of a LAN, access control, when applied at the source of a data transmission, can inhibit communications between stations not authorized to communicate with one another, including a station authorized to be connected to the LAN and a station not authorized to be connected to the LAN.

Figure 3 illustrates the application of this service to information PDUs. In implementations where authentication is provided at a higher layer, access control at Layer 2 provides protection from abuse of resources that operate upon data contained in the headers of the higher layer at which the service is provided and all other layers above Layer 2. For example, in a network where access control is provided as a Layer 3 end-to-end service over ISO CLNP, PDUs generated on one LAN could be sent to a remote LAN with particular Quality of Service (QOS) option parameters requested and the Record Route option invoked. This would provide information about the intermediate Network Layer systems to a rogue station on the Remote LAN. By also invoking the Partial Source Routing option and limiting the PDU Lifetime, a single station with partial information on the topology of a set of interconnected subnetworks could develop more complete information from Error Report PDUs, without the participation of a second rogue unit. This information could be used to exploit weaknesses in the network, such as identifying operational characteristics of particular routes (e.g., relative levels of congestion, transit delay, or residual error probability). While access control at Layer 2 cannot limit this type of abuse between stations authorized to communicate with one another, it can inhibit this type of communication between stations not authorized to communicate with one another. In implementations where access control is provided at the Link Layer rather than at a higher layer, this service provides protection from abuse of application data and data in the headers of the protocol layers above Layer 2. For example, this service can limit access to a particular file server to only those stations which required that access. It can also prohibit access to a gateway from unauthorized stations.

At the Link Layer of a LAN, this service can prevent use of the TEST PDU from the LLC Sublayer to create an unauthorized communications association. Figure 4 illustrates the application of access control to this type of PDU. Since the data to be used for a TEST PDU is not defined, the entire data field of this PDU could be filled with any data. By transmitting unnecessary TEST PDUs, cooperating stations could transfer any data. While access control will not limit this type of abuse between stations authorized to communicate, it can inhibit this type of communication between stations not authorized to communicate with one another (e.g., a station authorized to be connected to the LAN and a station not authorized to be connected to the LAN).

DATA CONFIDENTIALITY

Data confidentiality inhibits unauthorized disclosure of the protected data. This assures the sending station that the protected portion of a PDU will be available only to the intended recipient. Given the nature of the Link Layer of LANs and the susceptibility of LANs to wiretap, this service is necessary to protect data on LANs. This service is already indicated as appropriate for Layer 2 in ISO 7498-2.

MECHANISMS FOR PROVISION OF SECURITY SERVICES

Concerns that are raised when one suggests expanding the Layer 2 security service profile are: how can the additional security services be provided and what impact will this have on the complexity and performance of the LAN interface to a station. Data confidentiality is most commonly provided via encryption, also referred to as encipherment. In fact, data confidentiality through encryption is what most people associate with network security. While there are other mechanisms for providing data confidentiality, encryption is one of the simplest and most reliable. Fortunately, the mechanism most commonly used to provide data confidentiality, i.e., encryption, can be used to provide all of the indicated security services. In fact, the additional services can be provided with almost no impact to the performance or the complexity of the LAN interface.

Connectionless data integrity is almost an automatic side effect of data confidentiality via encryption. Most cryptographic algorithms produce a checksum or some other mathematical residue which can only be reproduced with the correct combination of cryptographic algorithm, key material, and data. For systems handling classified data, a cryptographic checksum calculated over the data, using an algorithm and key different from those used for the data confidentiality service, might be required. However, this is unnecessary for unclassified data.

Data origin authentication can easily be provided by including a copy of the source address within the encrypted data field, either as a prefix or a suffix to the Layer 2 SDU[2]. As with connectionless data integrity, in systems handling classified data, a cryptographic checksum calculated over the data, using an algorithm and key different from those used for the data confidentiality service, might be required. Again, however, this is unnecessary for unclassified data.

Access control can be effected implicitly through the management and application of cryptographic association, i.e., keying relationships. If all PDUs are encrypted, only those stations with cryptographic mechanisms and knowledge of the correct keying relationships can exchange information. A station without these facilities will be unable to access any of the protected resources.

With the exception of data origin authentication, all of the additional services can be provided as by-products of encryption when used to provide data confidentiality. And data origin authentication can be included so easily, it is hardly worth noting as an exception. Using the single mechanism of encryption, all of the indicated services can be provided with a minimum of impact to the complexity and performance to the LAN interface of an attached station.

---

[2] Data origin authentication is assured only to the granularity of the cryptographic key. A key that is unique to the source and destination address pair provides assurance of the individual source host identity; a key shared by a group only provides assurance that the source of the PDU is a member of the group.

SUMMARY

Table 1 summarizes the pertinent attributes of LANs that have been identified, the vulnerabilities that those attributes present, the security threat associated with those vulnerabilities, and the security services required to inhibit exploitation of those risks. In each case, the Link Layer of LANs has been shown to have qualities more like the Network Layer of WANs than those of the Link Layer of WANs. Given these arguments, it makes sense to provide the same range of security services for LANs' Link Layer as WANs' Network Layer.

Table 1

LAN Attribute	Vulnerability	Security Threat	Services Indicated
Data Transmission	Any station can transmit to any other station, using any address	Masquerade, unauthorized resource use	Data origin authenucauon, access control
Data Reception	Any station can access any transmission	Data modification, unauthorized disclosure	Connecuonless data integrity, data confidenuality
Address Space	No implicit controls through address management	Masquerade, unauthorized resource use	Data origin authenticauon, access control
Geographic Dispersion	Eavesdropping, wiretapping	Data modification, unauthorized disclosure	Connecuonless data integrity, data confidenuality

REFERENCES

ANSI/IEEE Standard 802.2-1985; Local Area Networks: Logical Link Control; Institute of Electrical and Electronic Engineer, Inc.; December 1984

ANSI/IEEE Standard 802.3-1985; Local Area Networks: Carrier Sense Multiple Access with Collision Detection; Institute of Electrical and Electronic Engineer, Inc.; December 1984

ANSI/IEEE Standard 802.4-1985; Local Area Networks: Token-Passing Bus Access Method; Institute of Electrical and Electronic Engineer, Inc.; February 1985

ANSI/IEEE Standard 802.5-1985; Local Area Networks: Token Ring Access Method; Institute of Electrical and Electronic Engineer, Inc.; April 1985

Berson, Thomas A. and Beth, Thomas, editors; Lecture Notes in Computer Science: Local Area Network Security; Springer-Verlag; April 1989

International Standards Organization 7498-2-1988(E) Information Processing Systems--Open Systems Interconnection--Basic Reference Model--Part 2: Security Architecture

International Standards Organization 8473: 1988 (E) Information Processing Systems--Data Communications--Protocol for providing the connectionless-mode network service

International Standards Organization 8802-2-1987 Information Processing Systems--Data Communications--Logical Link Control

P802.10/D5: Standard for Interoperable LAN Security (Draft); IEEE 802.1- LAN Security Working Group, Kirk Barker, editor; July 1989

Tanenbaum, Andrew S.; Computer Networks; Prentice-Hall, Inc.; 1981

United States Government Open Systems Interconnection Profile (U.S. GOSIP), Version 2: Federal Register; National Institute of Standards and Technology; July 1989

APPENDIX B--Example

This appendix presents an example of the use of the SDE protocol. This example will include two parties (A and B) and will examine the contents of Security Management Information Base (SMIB) and the Protocol Data Unit (PDU) construction. Being an example, it contains some implications for local processing that are not part of the standard. It uses the following objects that are not defined in SDE; however, these objects may subsequently be defined in the Layer Management Addendum.

Station_Max_SDU_Size: The maximum size Service Data Unit (SDU) that the MAC sublayer can support. In this example it is set to 1518 octets for IEEE 802.3.

SAP_Worst_Case Expansion: This is the maximum number of octets that can be added by SDE for SDU's originating at the indicated SDE SAP. The calculation for this object is described later in this document.

MAX_SDE_SDU_Size: This is calculated by subtracting SAP_Worst_Case_Expansion from the Station_MAX_SDU_Size. It is the maximum size SDU that SDE will accept.

1.0 Algorithm Registry

The SDE protocol expects the attributes of any confidentiality algorithm to be registered. This section contains excerpts from the registry.

Algorithm ID: 1
Name: DES CBC mode (ANSI X3.106)
IV length: 64 bits
Key Length: 56 bits + 8 bits parity
Class: Symmetric
Service: Confidentiality
Additional Fields and Placement: none

Algorithm ID: 2
Name: X9.9-1986 (Revised) Binary Option, modified
ICV length: 32 bits
Key Length: 56 bits + 8 bits parity
Class: Symmetric
Service: Integrity
Modifications:
Date of Message Origin: Not used.
Message Identifier: Not used.

2.0 Key Management

In this example, certain parameters are negotiated between the two key management applications to set up parameters for the communication. The effect on the SMIB will be shown in Section 3 of this document. Note that there are some parameters that are set by system management (e.g., Addresses, Remote_SDE). The SAID and the MDF are unique among the negotiated attributes in that each is a uni-directional attribute of the SAID and are simply accepted as opposed to negotiated.

2.1 Party A's Proposed Options

This section contains the proposed options that Party A sends Party B. Party B will select a subset of the provided options. In some cases, Party A specifies an alternate option. Binary fields are represented in hexadecimal.

A's SAID=00000034
Assoc_MDF=TRUE,
MDF= 558977883344
Confid=TRUE, ALT=none
Integ=FALSE, ALT=TRUE
Padding_pres=TRUE, ALT=none
ID_pres=TRUE, ALT=FALSE
Confid_Alg_ID=1, ALT=none
Integ_Alg_ID=None, ALT=2
Station_ID=8ABCDE3456780000

2.2 Party B's Selected Options

Party B chooses the following set from the options provided by Party A:

B's SAID=000000A5
Assoc_MDF=FALSE
Confid=TRUE
Integ=TRUE
Padding_pres=TRUE
ID_pres=FALSE
Confid_Alg_ID=1
Integ_Alg_ID=2

Note that the cryptographic algorithm required padding, so there was no option. The Assoc_MDF is forced to FALSE although that option is stated by A. A shall be able to support FALSE since it is a Minimum Essential Requirement (MER). Also, since ID_pres is selected to be FALSE, no Station ID is supplied.

3.0 Party A's SMIB

This section describes the relevant entries in Party A's SMIB after the key management negotiation.

3.1 Station Parameters

Station_Clear_Hdr=TRUE
Station_MDF=TRUE
Station_Max_MAC_SDU_Size=1518

3.2 SAP Parameters

SAP_Worst_Case_Expansion= 41 = 3 (SDE Designator) + 4 (SAID) + 6 (MDF) + 8 (IV) + 8 (Station ID) + 8 (PAD) + 4 (ICV)

Calculated Max_SDE_SDU_Size=1477

3.3 Security association Parameters

This section contains the relevant parameters in Party A's SMIB after the key management negotiation.

Local_SAID=34

Remote_SAID=A5
Assoc_MDF=FALSE
Confid=TRUE
Integ=TRUE
Padding_pres=TRUE
ID_pres=FALSE
Confid_Alg_ID=1: with key of "763b9d52290886e9"
Integ_Alg_ID=2: with key of "6846c72fab7501a4"
SDE_SAP= reference to User Stack (Set by Key Management)
Remote_SDE=TRUE (Set by Key Management)
Outgoing/Incoming Addresses (Set by System Management)

4.0 Party B's SMIB

This section describes the relevant entries in Party B's SMIB after the key management negotiation.

4.1 Station Parameters

Station_Clear_Hdr=TRUE
Station_User_Def=TRUE
Station_Max_MAC_SDU_Size=1518

4.2 SAP Parameters

SAP_Worst_Case_Expansion= 27 = 3 (SDE Designator) + 4 (SAID) + 8 (IV) + 8 (PAD) + 4 (ICV)

Calculated Max_SDE_SDU_Size=1491

4.3 Security association Parameters

This section contains the relevant parameters in Party B's SMIB after the key management negotiation.

Local_SAID=A5
Remote_SAID=34
Assoc_MDF=FALSE
Confid=TRUE
Integ=TRUE
Padding_pres=TRUE

ID_pres=FALSE
Confid_Alg_ID=1: with key of "763b9d52290886e9"
Integ_Alg_ID=2: with key of "6846c72fab7501a4"
SDE_SAP= reference to User Stack (Set by Key Management)
Remote_SDE=TRUE (Set by Key Management)
Outgoing/Incoming Addresses (Set by System Management)

5.0 Transmission Processing (From Party A)

Assume an SDE_UNITDATA.request with data of length 1005 octets.

5.1 Obtaining the Attributes

This section and the following sections correspond to sections 7.4.2-7.4.8 in the standard. The security association is identified using the SAP and the source and destination outgoing addresses.

5.2 Transmission to Non-SDE

Remote_SDE=TRUE, so this doesn't apply.

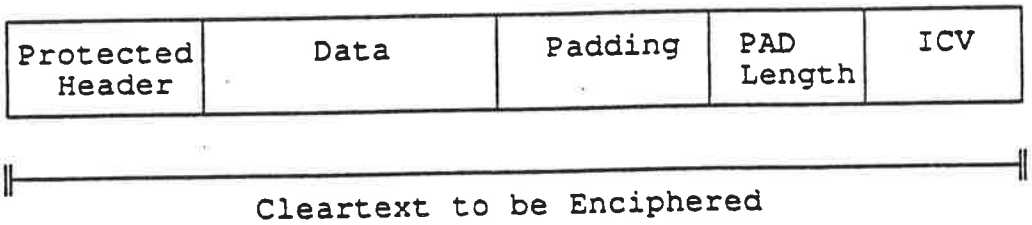
5.3 Oversize SDU

This step is not in the SDE protocol. It is an additional check by the implementation using the objects mentioned in the introduction to this appendix. No fragmentation is needed because 1005 is less than SAP_Max_SDE_SDU_size (1477).

5.4 Forming the Protected Header

ID_pres=FALSE, so this section is not applicable.

5.5 PAD



The figure above illustrates the fields that are enciphered in the SDE protocol. The following is the calculation for the value of the PAD Length.

PAD Length=	8 -		CBC block size
	(	(0	Protected Header
		+ 1005	size of SDE SDU
		+ 1	PAD Length
		+ 4	ICV
		) MOD 8)	CBC block size
	= 8 - (1010 MOD 8) = 8-2 = 6		

The value in the PAD Length field should be 6.

5.6 Calculation of the ICV

A 4 octet ICV is added as specified in X9.9.

5.7 Encipher the PDU

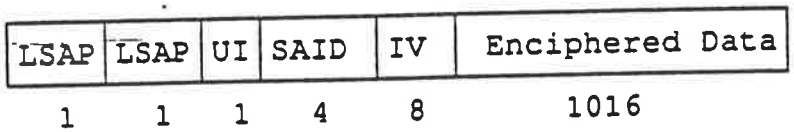
The PDU is enciphered using CBC which adds an 8 octet IV.

5.8 Clear Header

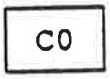
The Clear Header is prepended with the Remote_SAID placed in the SAID field.

5.9 MAC Request

The following appears in the Data field of the MAC request (binary values represented in hexadecimal with leftmost bit most significant):



The UI field contains:



The SAID contains:

000000A5

The IV is 8 octets of random data.

Before encryption (and hopefully after decryption), the enciphered data contains:

LLC PDU	Padding	PAD Length	ICV
1005	6	1	4

The LLC PDU and the Padding can contain any values. The PAD Length field contains "06", and the ICV is calculated based on the contents of the preceding three fields.

6.0 Reception Processing (At Party B)

The following steps correspond to the procedure described in Sections 7.5.1.1 -7.5.6.

6.1 Requirements for reception.

The contents of B's SMIB are contained in Section 4 of this appendix. It is assumed that the values for the bootstrap SAIDs also exist.

Since Station_Clear_Hdr=TRUE, section 7.5.1.1 is applicable. The first three octets of the received PDU correspond to the SDE_Designator, so the next four octets are used as the SAID. The SAID octets indicate "A5". In B's SMIB this indexes into the security association due to the presence of "A5" in the Local_SAID object.

The addresses in the MAC indication are checked against those set by system management in the SMIB. Since they check out as okay in this example, the Clear Header is removed.

6.2 Decipherment of the PDU

Since Confid=TRUE, the PDU is decrypted using the algorithm specified by the Confid_Alg_ID which is CBC. CBC uses the supplied 8 octet IV which is also removed prior to further processing. (The decryption key is part of the complex object pointed to by the Confid_Alg_ID.)

6.3 ICV Checking

Since Integ=TRUE, the ICV is confirmed using the algorithm specified in Integ_Alg_ID (ANSI X9.9). (The key is part of the complex object pointed to by the Integ_Alg_ID.) The ICV field is then removed.

6.4 PAD

The last octet in the PDU after the ICV is checked corresponds to the PAD Length. The number of octets in this field plus 1 (7 total for our example) is removed from the end of the PDU. This leaves the cleartext, integrity-checked Data field (LLC PDU).

6.5 Station ID

ID_pres=FALSE, so this section is not applicable.

6.6 SDE_UNITDATA

The LLC PDU is placed in the data parameter of the indication. All other parameters are transferred unaltered to the LLC.

APPENDIX C--Objectives of SDE

Before the Secure Data Exchange Protocol was defined, the IEEE 802.10 working group drew up a list of objectives that they wanted the protocol specification to meet. These objectives were discussed and refined over the course of several meetings. The objectives have been used to evaluate and develop the SDE proposals that were submitted to the working group. These objectives are present in the standard as the requirements for transparency.

1. Make the data exchange protocol independent of the encryption and integrity check algorithms.

2. Allow SILS protected broadcast and multicast.

3. Choose security mechanisms which allow exportability.

4. Allow co-existence of protected and unprotected traffic.

5. Do not rely on layers above the IEEE 802 architecture to provide SILS security services.

6. Support security service and mechanism (as defined in ISO 7498-2) management by specifying appropriate objects, etc.

7. Maintain the MAC/LLC Interface.

8. Allow encipherment in transparent and non-transparent implementations.

9. Allow the support of multiple MAC addresses behind a MAC bridge entity that implements the SILS Secure Data Exchange.

APPENDIX D--Rationale for Placement

1. Introduction.

IEEE 802 describes a class of Local Area and Metropolitan Area Networks represented by Fig 1. The placement of security within this architecture can logically occur between the Medium Access Control (MAC) and the Logical Link Control (LLC) layer, above the LLC layer, or integrated into either the LLC or MAC sublayer.

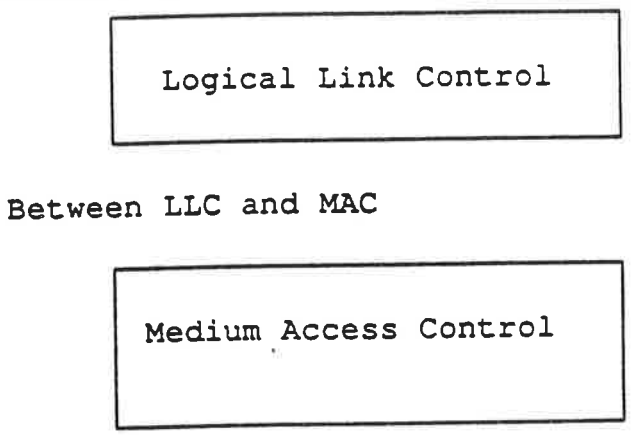


Fig 1
Choices for Placement

This appendix discusses the attributes of each of these placements and recommends that the placement directly above the MAC sublayer as a sublayer or as an LLC entity is the most likely candidate.

2. Integrated into MAC

The MAC sublayer has been developed by several different standards bodies: Carrier Sense Multiple Access/Collision Detect (CSMA/CD; IEEE 802.3), Token Bus (IEEE 802.4), Token

Ring (IEEE 802.5), MAC Bridges (IEEE 802.1, IEEE 802.6), etc. This is further complicated by the fact that these standard bodies often publish multiple standards for different media (e.g. coaxial cable, fiber optic, twisted pair). This implies that integration into the MAC sublayer would probably impact multiple standards, and thus would only apply to a very limited market. Since the security concerns are similar for the different MAC standards, and since a common interface will soon be provided by DIS 10039, the logical choice is to not integrate security services into the MAC sublayer.

IEEE 802.10 did not consider traffic flow analysis a serious threat, but the MAC sublayer is the only place where prevention against traffic flow analysis can be successfully implemented. If traffic flow analysis is a concern for a given implementation, the MAC sublayer would need to be more closely examined.

3. Between LLC and MAC or Lower LLC

The standard places a security entity at the bottom of LLC. With the exception of management, it can be viewed logically between the LLC and MAC sublayers. There are only three primitives that flow between the MAC and LLC layers: MA_UNITDATA request, the MA_UNITDATA indication, and the MA_UNITDATA_STATUS indication. (The contents of these requests currently differ slightly between the various MAC protocols, but there is an effort to determine a common MAC interface.) The simplicity of the interface and the protocol is the biggest advantage of placing the protocol between the MAC and the LLC sublayers.

There are many existing protocols other than LLC that request services directly from MAC. Even though this protocol is referred to as being between LLC and MAC, any protocol that implements the MAC service primitives can reside above the security protocol. This will prove to be an advantage in providing security for existing systems that may not implement LLC.

The security services are as follows:

- Data Confidentiality -- The SDE entity provides data confidentiality by enciphering the SDE SDU. The SDE entity provides for the use of multiple confidentiality algorithms and depends on an external key management service for establishing a data enciphering key and data deciphering key and for choosing an appropriate cryptographic algorithm.

- Connectionless Integrity -- The SDE entity provides connectionless integrity by calculating an Integrity Check Value (ICV) and placing it in the ICV field of the SDE PDU. The SDE entity depends on an external key management service to

establish an integrity algorithm.

- Data Origin Authentication -- Data Origin Authentication is achieved by the use of key management. It is supported by the SDE entity placing a Station ID in the protected header portion of the SDE PDU. The inclusion of the Station ID also prevents undetected reflection of the SDE PDU. Data origin authentication can only be provided in conjunction with the integrity service.

- Access Control -- Access control is provided by key management or system management. The SDE entity's use of security associations supports management's access control decisions. The SDE entity cannot transmit or deliver a PDU unless a security association exists. It is management's responsibility to set up the security associations and the SDE's responsibility to enforce the access control policy. Access control is dependent on both integrity and authentication services. Access control can only be provided in conjunction with integrity and authentication.

4. Integrated into upper LLC

Integration into LLC provides several advantages if it is done correctly. The granularity of security decisions and enforcements can now be at the granularity of the Link Service Access Points (LSAP) instead of the MAC addresses. While this provides added granularity, it shall be realized exactly what this means. Normally LSAP addresses are reserved for applications not processes. For instance, there is a reserved LSAP for ISO Network Layer, another LSAP is reserved for the DoD Internet Protocol. There are also locally administered LSAPs. These LSAPs could be used to separate between security levels, but then, what about the need for different security levels for those applications running above the reserved LSAPs?

LLC provides two types of operation.[1] The first type is a connectionless-mode operation that provides service across a data link with minimum protocol complexity. The second type of operation provides a connection-oriented service across a data link comparable to HDLC. This service includes support of flow control, sequencing, and error recovery. There is no substantial difference in the security services that can be provided over the connectionless-mode operations and those that can be provided by a protocol operating between LLC and MAC. The connection-oriented service, however, can provide additional security services and can allow different mechanisms.

What advantages are connection-oriented security services? With regard to Confidentiality, there

[1] There is a third type (Connectionless Acknowledged), but it is not yet a standard.

is no discernible difference to the service requestor between connection and connectionless confidentiality. However, if encryption is the mechanism used to provide that confidentiality, several advantages are gained if a connection-oriented confidentiality is provided. The first is that the key granularity can be based upon the connection, and not simply between the two peer entities. This provides advantages, since a different key can be used for different connections, providing better security in some cases. Since the key granularity is based on the connection, the protocol can discard keys after receiving disconnect messages for the connection. This is an advantage over connectionless, since connectionless has no concept of connection and uses a key cache of all recently used keys. The judicial use of the disconnects can reduce the size of the key cache in many systems.

The second advantage occurs from the fact that most encryption algorithms chain encryption blocks. A typical block size of an encryption algorithm is 64 bits (such as DEA: ANSI X3.92). If every 64 bits were enciphered separately, then an attacker could look for repetitions of the 64 bits, and thus gain an advantage in breaking the code. To prevent this, there are different modes of operation (such as ANSI X3.106). These modes of operation make each encryption block dependent upon the preceding block(s). While this is nice cryptographically, the order that the blocks are enciphered shall be the same as the order of decryption. If a connectionless service is used, this chaining shall start over for each Protocol Data Unit (PDU) received, since they are unordered. In a connection-oriented service, the chaining can continue across multiple PDUs, thus possibly reducing the overhead of re-initializing the cryptographic algorithm after each PDU.

Connection-oriented integrity is a distinctly different service than connectionless integrity. Connectionless integrity only assures the service-requestor that the chance of unauthorized modification to a single PDU is exceedingly small. Connection-oriented integrity ensures that the data units arrive in sequence, and that all the data units over the connection have arrived.

The effect of providing connection-oriented integrity in LLC is very similar to providing a connection-oriented LLC over a connectionless-integrity layer between LLC and MAC. Since the SDE SDU would be encapsulated in the MA_UNITDATA request, the sequence numbers as well as the data within the SDE SDU would be protected against modification. The only remaining integrity protection is against truncation. Truncation involves an active-wiretap deleting the last of a message in the hopes that a security breach can be caused by the uncompleted transaction. Since the Disconnect is sent enciphered, the interloper cannot generate the Disconnect request. The Disconnect packet does not contain the last received PDU; however, the sender treats all unacknowledged PDUs as if they had been lost. The receiver has no idea that the connection has been truncated, there is no method in LLC to prevent the receiver from thinking that all the valid data has been sent. There would need to be a special Disconnect PDU that contained the last sequence number. Unfortunately, that would involve changing the way that LLC processes, since IS 8802-2 requires that all previously sent

information PDUs "that are unacknowledged when this command [Disconnect] is actioned shall remain unacknowledged."

One additional advantage to a connection-oriented service is a function of the implementation of the LLC protocol. The acknowledgement is provided for PDUs, so the service requestor knows the PDU has been delivered if data origin authentication and integrity are provided. Note, however, as was true with sequencing, the same service is provided by a connection-oriented LLC operating above a protocol providing secure connectionless integrity and data origin authentication.

While the integrated version of LLC appears quite attractive, there are some downsides that convinced us not to choose this option. The most important reason is that all of the existing implementations of LLC would need to change. The connection-oriented security services as described above require changing the way that the PDUs are processed. From a standards point of view, this means that changes to the existing LLC standards will be required. From a vendor's point of view, existing equipment would be made obsolete, and migration to a secure version become difficult.

There are more security services that should be provided by the integrated version than the MAC/LLC proposal. The question is whether these additional security services justify the problems and added complexity. The simplicity of the MAC interface allows a very simple protocol. The integrated LLC protocol shall provide for both modes of operation as well as be extensible to new types of operation that may be defined in the future. It is unclear if all of these security services should be provided at Layer 2. IS 7498 Part 2 takes a much more conservative view of the security services that can be provided at Layer 2. It doesn't allow the provisioning of Access Control and Integrity. While it is believed that this is inappropriate, it should be remembered that the LLC protocol is only at Layer 2, and there may be other higher layers that are more suited for providing these additional services.

5. Above LLC

The protocol operating above LLC shall be cognizant of the different operational modes of LLC (connection and connectionless). It shall tailor its security services to account for these. As such, it will probably not be as simple a protocol as the MAC/LLC protocol. It does have the added benefit of having the granularity of LSAP addresses instead of MAC addresses as did the protocol integrated with LLC.

The reason that it was decided not to seriously consider the placement above LLC is that the only security service added other than finer granularity is the connection-oriented security services as described in the section on the integrated LLC protocol. If the protocol is operating above LLC, it shall duplicate much of the LLC processing if it is to provide these services. For instance,

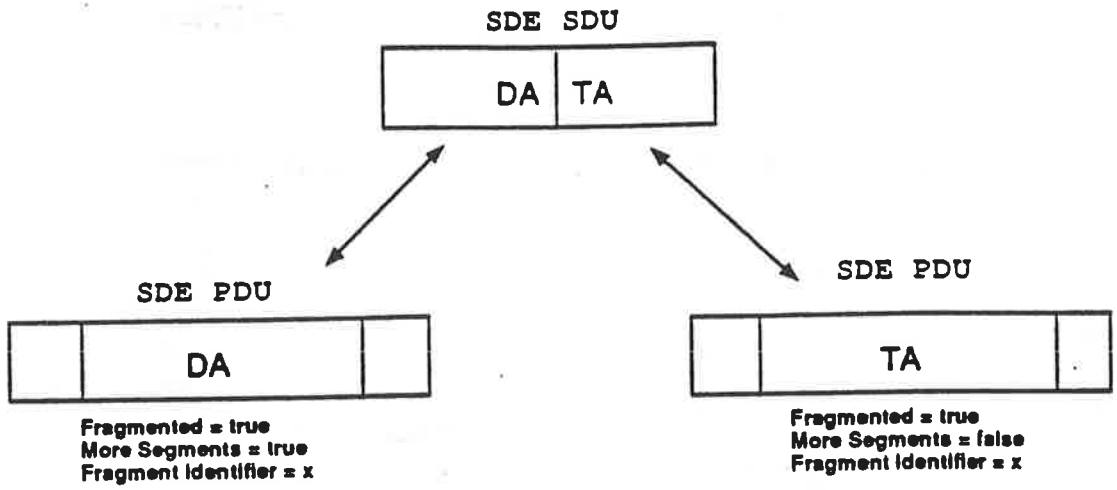
assume a PDU is received that fails the integrity check because it has been modified during transit. If the protocol claims to provide connection-oriented integrity, it cannot deliver the PDU to the next layer. Obviously, its peer shall attempt to resend. Unfortunately, the LLC protocol's error detection did not catch the error and it has already sent an acknowledgement. This necessitates the protocol above LLC to buffer PDUs, and set up a window just like the LLC layer. This involves redundant processing, and eventually becomes almost as complicated as the LLC protocol.

The protocol above LLC could just provide the connectionless services and become much more simple, but then the only motivation for choosing it over the MAC/LLC protocol would be the LSAP granularity.

6. Conclusion

For the reasons stated above, it was felt the best approach was to define a protocol operating between LLC and MAC or lower LLC. Some specific applications will need the additional security services provided by higher layers.

Figure 2-1 Fragment Association



On transmission, the procedure defined by this recommendation fragments the SDE SDU after the SDE entity finds a valid security association for a SDE_UNITDATA.request. Next, the proper fragmentation information is calculated and placed in the protected header of both fragments. Then each fragment is processed independently and finally forwarded to the MAC sublayer. Note that fragmentation occurs before encryption.

On reciept of an SDE PDU from the MAC sublayer, all of the SDE PDU processing is performed before SDE PDU reassembly is attempted. Therefore, when necessary, the SDE PDUs have been decrypted and had their integrity verified. Each security association maintains a list of PDUs awaiting assembly. This set is searched for the other fragment of the SDE SDU. The other fragment is located by finding the SDE PDU with an equal fragment identifier and a different "more segments" value. If a match is found, the SDE SDU is reassembled using the information in the protected header and SDE security processing is continued as usual on the reassembled SDU. If a match is not found, the PDU is placed in the set of PDUs awaiting reassembly.

3 Additional Station Objects

The SDE entity must be able to indicate whether it can support fragmentation. Therefore implentations that support fragmentation must have the Station_Fragmentation_Enabled station object set to true. This object is used by key management when negotiating fragmentation support at security association initialization.

If a station supports fragmentation then the station must support the following objects:

Station_Reassembly_Timer

INTEGER - the number of seconds an SDE entity will store an received SDE PDU that contains a fragment of a SDE SDU.

Station_Reassembly_Expiration_Count

INTEGER - the number of SDE PDUs which have been discarded by the SDE entity when the SDE reassembly timer has expired.

Station_Receive_Fragment

INTEGER - the number of SDE PDUs that contain SDE SDU fragments received by this station.

4 SAP Objects

The procedure defined in this appendix must reference the SAP_Max_SDE_SDU SAP object. This object is defined in the layer management addendum of Standard for Interoperable Local Area Network (LAN) Security (SILS).

5 Additional Association Object

The SDE must be able to determine whether a given association supports fragmentation. Therefore stations which support fragmentation must have the Assoc_Frag_Enab object defined for each security association. Security associations that support fragmentation must have the Assoc_Frag_Enab object set to true.

6 Additional Protected Header Fields

When a security association supports fragmentation, two additional fields must be added to the protected header:

- Flags.
- Fragment Identifier.

Figure 6-1 shows two examples of the protected header formats for an association that supports fragmentation. Both examples assume that association object ID_pres is true. The first example shows the header format for a PDU that contains a fragment, the other example is for PDU that does not contain a fragment.

1
2
3

APPENDIX E -- Fragmentation

1 Introduction

The Secure Data Exchange protocol (SDE) can add additional fields to the data recieved from the SDE service interface (the SDE Service Data Unit) and increase the length of the resulting Meduim Access Control (MAC) Service Data Unit (SDU). This additional length may produce a MAC SDU longer than the maximum allowed MAC SDU length. This is not acceptable, because it would force a MAC sublayer error.

There are two basic methods of insuring that the SDE sublayer does not generate MAC SDUs that are too long:

- Data link users can adjust their maximum Protocol Data Unit (PDU) size to take into account of the additional SDE overhead.
- SDE can fragment and reassemble data link user PDUs transparently to the data link user.

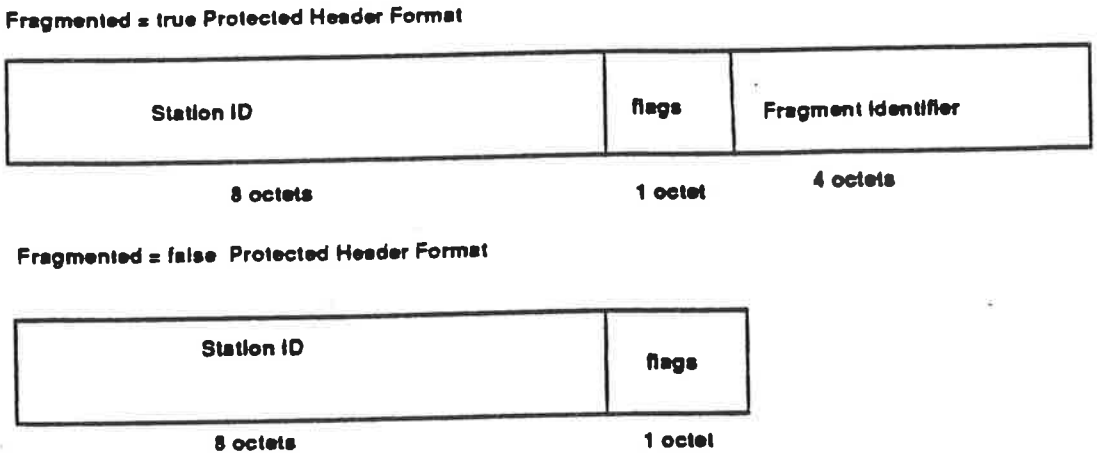
The fragmentation and reassembly of SDE SDUs increases the complexity and reduces the performance of the SDE sublayer. Thus the adjustment of data link users maximum PDU size is the preferred solution. However, it is not always possible to modify the data link users maximum PDU size. This appendix recommends a method for the SDE sublayer to provide SDE fragmentation and reassembly when the data link users maximum PDU size cannot be modified.

2 Overview

The fragmentation and reassembly procedures will be performed only if the security association indicates fragmentation support.

This fragmentation and reassembly procedure splits an SDE SDU into two parts. Each part or fragment will be transmitted as a separate SDE SDU. A PDU that is a fragment of a SDE SDU is identified by a "fragmented" field in the protected header. This field is set true when the PDU contains a fragment of a SDE SDU. Each fragment of a given SDE SDU is assigned the same fragment identifier. The fragment identifier is stored in the SDE SDU's protected header. The fragment identifier must be unique for the duration of the crypto-period. When a SDE SDU is fragmented, the two parts are distinguished by a boolean field in the protected header called "more segments." The first fragment has the "more segments" field set true and the second fragment has the field set false. Figure 2-1 shows the relationships between an SDE SDU and its fragments.

Figure 6-1 Example Protected Header Formats



6.1 Flags Field

The flags field is a mandatory field in the protected header when Assoc_Frag_Enab is true. The format of the field is shown figure 6-2. If ID_pres is true then the flags field follows the Station ID field. If ID_pres is false then flags field is the first field in the protected header. The flags field contains two subfields used for fragmentation and reassembly: fragmented and more segments.

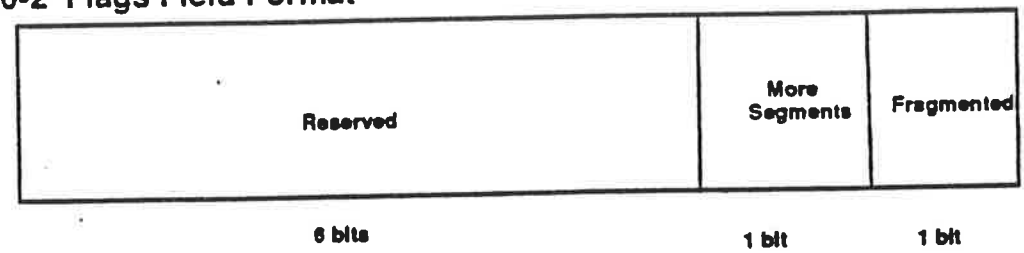
Fragmented

This a boolean field. When the value of this field is true, it indicates that the SDE PDU is a fragment of a SDE SDU and that the fragment identifier field follows the flag field.

More Segments

This is a boolean field which is only meaningful if the fragmented field is true. This field is used to indicate SDE PDU fragment number. If the value of this field is true the SDE PDU contains the first fragment of the SDE SDU. If it is false it contains the second part of a fragmented SDU SDU.

Figure 6-2 Flags Field Format



6.2 Fragment Identifier Field

If the flags field indicates that a SDE PDU contains a fragmented SDE SDU then the fragment identifier field follows the flags field in the protected header. The fragment identifier field is used to associate SDE PDUs with the SDE SDU from which they were derived. The fragment identifier is four octets long.

In order to protect against integrity attacks on fragments, the security association must be re-keyed before the fragment identifier field reuses identifier values. This implies that the SDE entity must be able to inform the key management entity when the fragment identifier space is exhausted.

7 Detailed Functional Specification

7.1 SDE_UNITDATA.request

The following steps are performed after finding the security association (7.4.1[1]) for the SDE_UNITDATA.request and when the security association has the Remote_SDE equal to true and Assoc_Frag_Enab equal to true.

A. If the length of the SDE SDU is greater then maximum SDE SDU length (SAP_Max_SDE_SDU) then perform the following steps:

   1. Generate a fragment identifier. The value of this identifier must be different from all other values used on this association. If a unique fragment identifier value cannot be generated then inform layer management. The handling of this event by layer management is a local manner.

   2. Split the SDE SDU (or the SDE_UNITDATA.request data field) into two pieces. Each piece must small enough so that resulting MAC SDUs are smaller than the maximum MAC SDU length when SDE processing is complete.

   3. Build the fragmentation part of the protected header with fragmented field set true, more segments set true and the value of generated identifier in the fragment identifier field. Prepend these fields to the first fragment.

   4. Build the fragmentation part of the protected header with fragmentation field set true, more segments set false and the value of generated identifier in the fragment identifier field. Prepend these fields to the second fragment.

B. If the length of the SDE SDU is not greater then maximum SDE SDU length (SAP_Max_SDE_SDU) then prepend the flags field to the MSDU specified in the SDU_UNITDATA.request. The flags field has the fragmented field set to false.

C. Forward the outgoing PDU (or both PDUs if the SDU has been fragmented) to the forming protected header step 7.4.3.

[1] All section numbers in this appendix refer to Standard for Interoperable Local Area Network (LAN) Security (SILS) Part B.

7.2 SDE_UNITDATA.indication

The following steps are performed following the Station ID step (7.5.5) in the reception procedures.

If Remote_SDE equals true and Assoc_Frag_Enab equals true and the fragmented field in flags field equals true perform the following:

A. Increment Station_Receive_Fragment station object.

B. Each association has a set of SDE PDUs awaiting reassembly. This set is searched for a SDE PDU that has a fragment identifer that is equal to the fragment identifer of the received SDE PDU.

C. If a matching fragment identifier is found then perform the following:

   1. Compare the more segments subfield of the matching SDE PDUs.
   2. If the subfield values are different perform the following:
      a. Stop the reassembly timer for the stored SDE PDU.
      b. Concatenate the two fragments together
      c. Continue normal processing with the combined SDE PDUs.
   3. If the more segments subfields are the same then stop processing and discard the received SDE PDU.

D. If a matching fragment identifier is not found then perform the following:

   1. Place the SDE PDU in the set of PDUs awaiting reassembly.
   2. Start the reassembly timer for this SDE PDU.
   3. Signal layer management entity of a fragmentation event (Station_Receive_Fragment).

7.3 Build Protected Header

This recommendation requires a slight modification of the SDE build protected header function (7.4.3). This function must always insert the flags field in the protected header if the security association supports fragmentation (Assoc_Frag_Enab = True).

7.4 Fragment Reassembly Timer

The primary function of this timer is to provide a bound for which a SDE PDU will be held for reassembly. When a fragmented PDU is recieved the reassembly timer is started. If the SDE PDU is not reassembled before the timer expires, the SDE PDU is discarded and the layer management entity is notified. The management entity will increment the Station_Reassembly_Expiration_Count object. The timer value is a station object (Station_Reassembly_Timer) and the value of the timer will be a local issue.