IEEE P802.1AE/D?

Draft Standard for Local and Metropolitan Area Networks: Media Access Control (MAC) Security

Sponsor

LAN MAN Standards Committee
of the
IEEE Computer Society

This individual contribution to the Project under consideration by the Link Security Task Group of IEEE 802.1 is intended to faciliate future progress. ***It has no official standing whatsoever and has not been reviewed by the task group.***

Abstract: This standard specifies how all or part of a bridged local or metropolitan area network can be secured transparently to communicating peers. MAC security (MACsec) entities in end stations and in MAC Bridges use secure associations and media access independent protocols to provide connectionless user data confidentiality, frame data integrity, and data origin authenticity between authorized ports.
Keywords: For keywords refer to the title page proper, following the editors' foreword.

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards.

Use of an IEEE Standard is wholly voluntary. The IEEE disclaims liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon this, or any other IEEE Standard document.

The IEEE does not warrant or represent the accuracy or content of the material contained herein, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained herein is free from patent infringement. IEEE Standards documents are supplied "**AS IS**."

The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE Standard is subjected to review at least every five years for revision or reaffirmation. When a document is more than five years old and has not been reaffirmed, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE Standard.

In publishing and making this document available, the IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is the IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing this, and any other IEEE Standards document, should rely upon the advice of a competent professional in determining the exercise of reasonable care in any given circumstances.

Interpretations: Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration.

Comments for revision of IEEE Standards are welcome from any interested party, regardless of membership affiliation with IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Comments on standards and requests for interpretations should be addressed to:

Secretary, IEEE-SA Standards Board
445 Hoes Lane
P.O. Box 1331
Piscataway, NJ 08855-1331
USA

Note: Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. The IEEE shall not be responsible for identifying patents for which a license may be required by an IEEE standard or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

IEEE is the sole entity that may authorize the use of certification marks, trademarks, or other designations to indicate compliance with the materials set forth herein.

Authorization to photocopy portions of any individual standard for internal or personal use is granted by the Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; (978) 750-8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

Editors' Foreword

<<Notes>>

<<Throughout this document, all notes such as this one, presented between angle braces, are temporary notes inserted by the Editors for a variety of purposes; these notes and the Editors' Foreword will all be removed prior to publication and are not part of the normative text.>>

<<Comments and participation in 802.1 standards development

Comments on this draft are encouraged. **PLEASE NOTE: All issues related to IEEE standards presentation style, formatting, spelling, etc. are routinely handled between the 802.1 Editor and the IEEE Staff Editors prior to publication, after balloting and the process of achieving agreement on the technical content of the standard is complete.** Readers are urged to devote their valuable time and energy only to comments that materially affect either the technical content of the document or the clarity of that technical content. Comments should not simply state what is wrong, but also what might be done to fix the problem.

Full participation in the development of this draft requires individual attendance at IEEE 802 meetings. Information on 802.1 activities, working papers, and email distribution lists etc. can be found on the 802.1 website:

http://ieee802.org/1/

Use of the email distribution list is not presently restricted to 802.1 members, and the working group has had a policy of considering ballot comments from all who are interested and willing to contribute to the development of the draft. Individuals not attending meetings have helped to identify sources of misunderstanding and ambiguity in past projects. Non-members are advised that the email lists exist primarily to allow the members of the working group to develop standards, and are not a general forum.

Comments on this document may be sent to the 802.1 email exploder, to the editors, or to the Chairs of the 802.1 Working Group and Link Security Task Group. :

Dolors Sala
Chair, 802.1 Link Security Task Group

Email:dolors@ieee.org

Tony Jeffree
Chair, 802.1 Working Group
11A Poplar Grove
Sale
Cheshire
M33 3AX
UK
+44 161 973 4278 (Tel)
+44 161 973 6534 (Fax)
Email: tony@jeffree.co.uk

PLEASE NOTE: Comments whose distribution is restricted in any way cannot be considered, and may not be acknowledged.
>>

<<The draft text and accompanying information

This document currently comprises:

- A temporary cover page, preceding the Editors' Forewords. This cover page will be removed following working group approval of this draft, i.e. prior to sponsor ballot.
- IEEE boilerplate text.
- The editors' forewords, including this text. These include an unofficial and informal appraisal of history and status, introductory notes to each draft that summarize the progress and focus of each successive draft, and requests for comments and contributions on major issues.
- A title page for the proposed standard including an Abstract and Keywords. This title page will be retained following approval.
- IEEE boilerplate text (identical to the above).
- An introduction to the family of 802 standards.
- The introduction to this standard, as revised by this proposed draft. This follows the preceding item and is actually important.
- A record of participants (not included in early drafts but added prior to publication).
- The proposed revision proper.
- An Annex Z comprising the editors' discussion of issues. This annex will be deleted from the document prior to sponsor ballot.

During the early stages of draft development, 802.1 editors have a responsibility to attempt to craft technically coherent drafts from the resolutions of ballot comments and the other discussions that take place in the working group meetings. Preparation of drafts often exposes inconsistencies in editors instructions or exposes the need to make choices between approaches that were not fully apparent in the meeting. Choices and requests by the editors' for contributions on specific issues will be found in the editors' introductory notes to the current draft, at appropriate points in the draft, and in Annex Z. Significant discussion of more difficult topics will be found in the last of these.

The ballot comments received on each draft, and the editors' proposed and final disposition of comments, are part of the audit trail of the development of the standard and are available, along with all the revisions of the draft on the 802.1 website (for address see above).

>>

<<History and Scope

A PAR for this project was drafted at the June 2003 802.1 interim meeting, and has been submitted to the 802 SEC for circulation to and to solicit comment from other P802 Working Groups. It is anticipated that a final proposed PAR will be forwarded for SEC consideration by vote of the 802.1 Working Group at its closing plenary during the July 2003 meeting of P802.

>>

<<Introductory notes to the current draft

This document, P802.1AE/D?, is not a working group or task group draft, but an individual contribution intended to facilitate future progress.
>>

<<Notes to prior drafts (excerpts of continuing relevance).

P802.1D/D0:

P802.1/D0 was the first of the full revision series. The prior development of P802.1y is archived on the IEEE 802.1 website, and the notes in Annex Z summarize prior technical progress.

In July 2002 we discussed what we might do if there were objections to removing the STP material in clause 8, on the grounds that it should still be accessible for historical reasons. Having worked on the draft it seems clear that not removing the material would either be an obstacle to correctly expressing the technical content of newer material or necessitate a thorough revision of the way the old material fits into the document. It seems unlikely that we would expend the effort or have the enthusiasm to do the latter well. Moreover cutting out dead wood references to withdrawn standards creates its own problem of historical reference, while leaving it in carries forward a maintenance load. The only sensible approach to this would seem to be to retain an archive of withdrawn and superseded copies of our standards, accessible to those who have a need to retrieve historical information. In turn this helps us cut out further dead wood, saving the reader from plowing through irrelevancies, and helping maintenance.
>>

<<Editors' final checklist (items noted in development, to be applied to final text.

The published standards are inconsistent and a bit of a mess when it comes to PDF bookmarks, this makes using them rather than final working group text difficult. P802.1p/D9 was very good. In particular it provides bookmarks for all figures at the end of a clause (see clause 7 for an example), need to copy that example.
>>

IEEE P802.1AE/D?

Draft Standard for Local and Metropolitan Area Networks: Media Access Control (MAC) Security

Sponsor

LAN MAN Standards Committee
of the
IEEE Computer Society

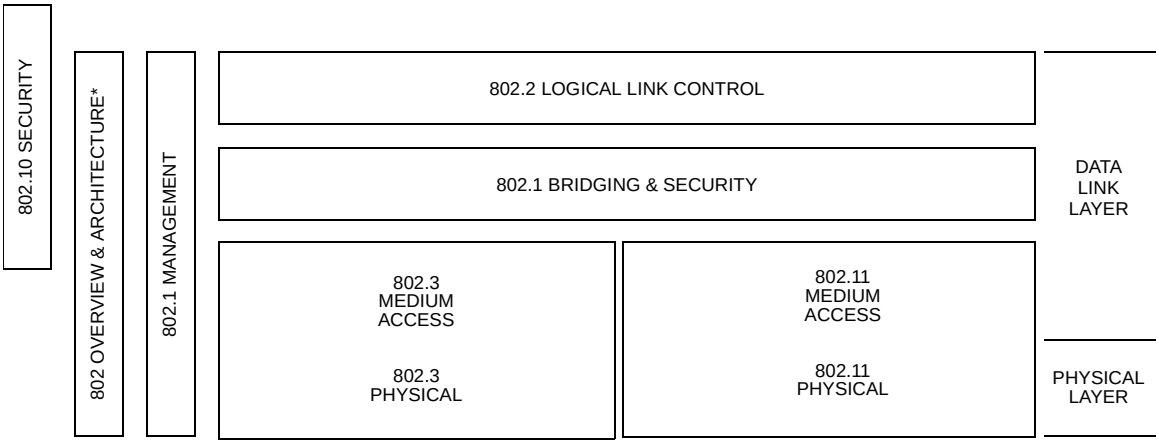
Contribution to the Link Security Task Group of IEEE 802.1

Abstract: This standard specifies how all or part of a bridged local or metropolitan area network can be secured transparently to communicating peers. Secure associations, between protocol entities in end stations and in MAC Bridges, are used by media access independent protocols to support connectionless user data confidentiality, frame data integrity, and data origin authenticity between authorized ports.
Keywords: local area networks, LANs, metropolitan area networks, MANs, security, MAC security confidentiality, integrity, data origin authenticity, port based network access control, MAC Service, MSAP, service access point, transparent bridging, MAC Bridges, port based network access control, authorized port, secure association.

Introduction to IEEE 802 Local And Metropolitan Area Network Standards

[This introduction is not part of IEEE Std 802.1AE, 200X Edition, IEEE Standard for Local and Metropolitan Area Networks : Media Access Control (MAC) Security.]

This standard is part of a family of standards for local and metropolitan area networks. The relationship between the standard and other members of the family is shown below. (The numbers in the figure refer to IEEE standard numbers.)



The IEEE 802 family of standards deals with the Physical and Data Link layers as defined by the International Organization for Standardization (ISO) Open Systems Interconnection (OSI) Basic Reference Model (ISO/IEC 7498-1 : 1994). The access standards define medium access technologies and associated physical media, each appropriate for particular applications or system objectives.

The standards defining the technologies noted above are as follows:

- IEEE Std 802 *Overview and Architecture.* This standard provides an overview to the family of IEEE 802 Standards.
- IEEE Std 802.1B *LAN/MAN Management.* Defines an OSI management-compatible architecture, and services and protocol elements for use in a LAN/MAN environment for performing remote management.
- IEEE Std 802.1D *Media Access Control (MAC) Bridges.* Specifies an architecture and protocol for the interconnection of IEEE 802 LANs below the MAC service boundary.
- IEEE Std 802.1AE *Media Access Control (MAC) Security.* Specifies an architecture and protocol for IEEE 802 LAN Security.
- IEEE Std 802.1F Common Definitions and Procedures for IEEE 802 Management Information
- IEEE Std 802.2 Logical link control
- IEEE Std 802.3 CSMA/CD access method and physical layer specifications
- IEEE Std 802.11 Wireless LAN Medium Access Control (MAC) and physical layer specifications

IEEE Std 802.AE, 200X Edition

.....

Relationship between IEEE Std 802.1AE and other IEEE Std 802.1 standards

IEEE Std 802.1X specifies ...

IEEE Std 802.1A? specifies key management and the establishment of secure associations used by IEEE Std 802.1AE.

Contents

IEEE P802.1AE/D?

Draft Standard for Local and Metropolitan Area Networks: Media Access Control (MAC) Security

1. Overview

1.1 Introduction

IEEE 802 Local Area Networks (LANs) are often used ...

<<in applications where security matters and there is a security threat ... an informal and easy to read description in in one paragraph of no more than 5 lines>>

MAC Security, as defined by this Standard, allows ...

<< an informal and easy to relate introduction to the effects and benefits of MAC Security in one paragraph of not more than 5 lines>>

MAC Security provides for

<<between 3 and a maximum of 8 specific bullet points to backup the preceding paragraph>>

a) specific benefit 1
b) specific benefit 2
c) specific benefit 3.

1.2 Scope

This standard specifies provision of connectionless user data confidentiality, frame data integrity, and data origin authenticity by media access independent protocols and entities that operate transparently to MAC Clients. To this end it

a) Specifies the requirements for MAC Security in terms of provision of the MAC Service and the preservation of the semantics and parameters of service requests and indications.
b) Describes the threats, both intentional and accidental, to correct provision of the service.
c) Specifies the security services supported by MAC Security to prevent, mitigate, or restrict the impact or scope of attacks that present these threats.
d) Examines the potential impact of both the threats and the use of MAC Security on the Quality of Service, specifying constraints on the design and operation of entities and protocols that provide MAC Security.
e) Models support of the secure MAC Service in terms of the operation of media access control method independent MAC Security Entities (SecYs) within the MAC Sublayer in End Stations and Bridges.
f) Identifies the functions to be performed by each SecY, and provides an architectural model of its internal operation in terms of Processes and Entities that provide those functions.
g) Establishes the requirements for security wrapper to provide security services for each MAC Protocol Data Unit (MPDU) delivered or accepted by a SecY to or from client entities and communicated to or from peer SecYs.

h) Establishes the requirements for a protocol between peer SecYs to identify the potential end points of the secure associations (SAs) used by security wrappers <<improve wording>>.
i) Specifies the interface/exchanges between a secY and its associated and collocated Port Access Entity (PAE) that provides and updates cryptographic keying and secure association identification (SAID) information for the SecY.
j) Establishes the requirements for management of MAC Security, identifying the managed objects and defining the management operations for MAC Security Entities.
k) Specifies performance requirements and recommends default values and applicable ranges for the operational parameters of a SecY.
l) Specifies the requirements to be satisfied by equipment claiming conformance to this standard.

<<what else?>>

This standard does not

m) specify key management or key distribution protocols, but makes use of ...

<<find words for this>>

n) blah ...

<<what else?>>

2. References

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

ANSI X3.159-1989, American National Standards for Information Systems—Programming Language—C.[1]

IEEE Std 802-2001, IEEE Standards for Local and Metropolitan Area Networks: Overview and Architecture.[2]

IEEE Std 802.1D-2003, IEEE Standards for Local and Metropolitan Area Networks: Media Access Control (MAC) Bridges.

IEEE Std 802.1Q-2003, IEEE Standards for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks.

IEEE Std 802.1X-2001, IEEE Standards for Local and Metropolitan Area Networks—Port Based Network Access Control.

IEEE Std 802.2, 1998 Edition [ISO/IEC 8802-2: 1998], Information technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements—Part 2: Logical link control.[3]

IEEE Std 802.3, 2002 Edition, IEEE Standards for Local and Metropolitan Area Networks, Supplement to Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications: Aggregation of Multiple Link Segments.

IEEE Std 802.11, 1999 Edition [ISO/IEC 8802-11: 1999], Information technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications.

IETF RFC XXXX[4]

ISO 6937-2: 1983, Information processing—Coded character sets for text communication—Part 2: Latin alphabetic and non-alphabetic graphic characters.[5]

ISO/IEC 7498-1: 1994, Information processing systems—Open Systems Interconnection—Basic Reference Model—Part 1: The Basic Model.

---

[1]ANSI publications are available from the Sales Department, American National Standards Institute, 11 West 42nd Street, 13th Floor, New York, NY 10036, USA.

[2]IEEE publications are available from the Institute of Electrical and Electronics Engineers, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331, USA. IEEE publications can be ordered on-line from the IEEE Standards Website: http://www.standards.ieee.org

[3]ISO [IEEE] and ISO/IEC [IEEE] documents are available from ISO Central Secretariat, 1 rue de Varembé, Case Postale 56, CH-1211, Genève 20, Switzerland/Suisse; and from the Institute of Electrical and Electronics Engineers, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331, USA. ISO [IEEE] and ISO/IEC [IEEE] documents can be ordered on-line from the IEEE Standards Website: http://www.standards.ieee.org.

[4]Internet RFCs are available from the Internet Engineering Task Force website at http://www.ietf.org/rfc.html.

[5]ISO and ISO/IEC documents are available from the ISO Central Secretariat, 1 rue de Varembé, Case Postale 56, CH-1211, Genève 20, Switzerland/Suisse; and from the Sales Department, American National Standards Institute, 11 West 42nd Street, 13th Floor, New York, NY 10036, USA.

ISO/IEC TR 11802-2: 1997, Information technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Technical reports and guidelines—Part 2: Standard Group MAC addresses.

ISO/IEC 14882: 1998, Information Technology—Programming languages—C++.

ISO/IEC 15802-1: 1995, Information technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Common specifications—Part 1: Medium Access Control (MAC) service definition.

3. Definitions

For the purposes of this standard, the following terms and definitions apply.

3.1 Bridged Local Area Network

A concatenation of individual IEEE 802 LANs interconnected by MAC Bridges.

NOTE—Unless explicitly specified the use of the word ‘network’ in this Standard refers to a Bridged Local Area Network. The term Bridged Local Area Network is not otherwise abbreviated. The term Local Area Network and the abbreviation LAN are used exclusively to refer to an individual LAN specified by a MAC technology without the inclusion of Bridges. This precise use of terminology within this specification allows a Bridged Local Area Network to be distinguished from an individual LAN that has been bridged to other LANs in the network. In more general usage such precise terminology is not required, as it is a explicit goal of this standard that MAC Security is transparent to the users of the MAC Service.

3.2 Client

N-Service User (a client of an N-entity is the N-Service User of the service provided by that entity).

3.3 IEEE 802 Local Area Network (LAN)

IEEE 802 LANs (also referred to in the text simply as LANs) are LAN technologies that provide a MAC Service equivalent to the MAC Service defined in ISO/IEC 15802-1. IEEE 802 LANs include IEEE Std 802.3 (CSMA/CD), and IEEE Std 8802.11 (Wireless).

3.4 Port Access Entity

<<see 802.1X>>

4. Abbreviations

The following abbreviations are used in this standard.

CRC	Cyclic Redundancy Check
FCS	Frame Check Sequence
ICV	Integrity Check Value
kB/s	Kilobit per second (1 kB/s is equivalent to 1000 bits per second)
MAC	Media Access Control
MB/s	Megabit per second (1 MB/s is equivalent to 1,000,000 bits per second)
RSTP	Rapid Spanning Tree Algorithm and Protocol
RST BPDU	Rapid Spanning Tree Bridge Protocol Data Unit
SAID	Secure Association IDentifier
TB/s	Terabit per second (1 TB/s is equivalent to 1,000,000 MB/s)
SAID	Secure Association IDentifier

5. Conformance

5.1 Required Capabilities

An implementation of a MAC Security Entity (MACsecY) for which conformance to this standard is claimed shall

a) Implement the MAC Security Protocol (MACsec), as specified in Clause 9.
b) Encode transmitted SMPDUs and validate received SMPDUs as specified in Clause 9.
c) Specify the following parameters of the implementation
    1) Filtering Database Size, the maximum number of entries.
    2) Permanent Database Size, the maximum number of entries.
d) Specify the following performance characteristics of the implementation
    1) .....
    2) .....

5.2 Optional Capabilities

An implementation of a MAC Security Entity (MACsecY) for which conformance to this standard is claimed may

a) .....
b) .....

NOTE—The term capability is used to describe a set of related detailed provisions of this Standard. Each capability can comprise both mandatory provisions, required if implementation of the capability is to be claimed, and optional provisions. Each detailed provision is specified in on or more of the other clauses of this standard. The PICS, described below, provides a useful checklist of these provisions.

5.3 Protocol Implementation Conformance Statement

The supplier of an implementation that is claimed to conform to this standard shall complete a copy of the PICS proforma provided in Annex A (normative) and shall provide the information necessary to identify both the supplier and the implementation.

6. Secure support of the MAC Service

MACsec provides security for communications between authenticated and authorised peer MAC Security Entities attached to and providing service at service access points in

a) End stations attached to Local Area Networks; and
b) MAC Bridges that connect Local Area Networks into a Bridged Local Area Network.

This clause discusses the following aspects of secure service provision

a) Provision of the MAC Service to MAC Service users in End Stations
b) Provision of the MAC Internal Sublayer Service to MAC Relay Entities in MAC Bridges
c) Provision of the MAC Enhanced Internal Sublayer Service in VLAN-aware Bridges
d) Preservation of the MAC Service
e) Security threats
f) Maintenance of Quality of Service.

<<The strategy is: What are the defining characteristics of the MAC Service? What threats are these services vulnerable to? What security services or features can be provided to prevent, mitigate, these attacks?>>

6.1 Provision of the MAC Service

The service provided to End Stations is the (unconfirmed) connectionless-mode MAC Service defined in ISO/IEC 15802-1. The MAC Service is defined as an abstraction of the features common to a number of specific MAC Services; it describes the transfer of user data between source and destination end stations, via MA-UNITDATA request primitives and corresponding MA-UNITDATA indication primitives issued at MAC service access points. Each MA-UNITDATA request and indication primitive has four parameters: Destination Address, Source Address, MAC Service data unit (MSDU), and Priority.

The service provided to the MAC Relay Entity of a MAC Bridge is the MAC Internal Sublayer Service (ISS) specified in Clause 6.4 of IEEE Std 802.1D. The ISS is derived from the MAC Service by augmenting the specification of the MA_UNITDATA.request and MA_UNITDATA.indication primitives with two parameters necessary to the performance of the relay function. These are frame_type and frame_check_sequence. The definition of the ISS does not add any new service primitives to those defined by the LAN MAC Service Definition.

The service provided to the MAC Relay Entity of a VLAN-aware MAC Bridge is the MAC Enhanced Internal Sublayer Service (EISS) specified in Clause 6.4 of IEEE Std 802.1Q. The ISS is derived from the EISS by augmenting the specification of the M_UNITDATA.request and M_UNITDATA.indication primitives with parameters necessary to the operation of the tagging and untagging functions of the VLAN Bridge. A canonical_format_indicator, vlan_classification, option rif_information, and an include_tag are added to the request primitive. A canonical_format_indicator, vlan_identifier, and optional rif_information are added to the indication primitive. The definition of the EISS does not add any new service primitives to those defined by the ISS.

End stations and MAC Bridges may restrict the provision of the service, i.e., may restrict the service access points at which request primitives can be issued and corresponding indication primitives can occur, to those that are bound to authenticated and authorized devices. Unauthorized devices can be denied access to a LAN or to a Bridged Local Area Network, other than to allow protocol exchanges required by an authentication process.

<<‘bound to a device’ means accessible only by entities within that device.>>

<<The above specifications are not complete for the purposes of security - some fudamental properties of the service (such as symmetrical and transitive communication)need to be added as described below.>>

<<Should mention MAC status parameters (802.1D Clauses 6.4.2, 6.4.3) somewhere, are they a part of the service that needs to be explicitly provided, preserved, and supported from the security point of view? Perhaps MAC Operational needs to be set to reflect temporary failures of the secure support to preserve the necessary connectivity.>>

<<Multipoint service needs to be defined, probably in P802.1ac, and discussed.>>

<<Shared media, yellow coax (or wireless), messages can be read everywhere. The MAC service for wired is defined in 802.1D, constrained to that definition. A bridge does not forward a frame other than to the destination MAC address. Therefore the shared media ability to read the frame anywhere is countervened. The only service a higher layer can depend on is that defined in .1D, which is delivery to the specified MAC destination address. In an EPON the ability of a station to read a message not addressed to it, is a security threat. The countermeasure is to make the message unintelligible except to the destination addressee - i.e., encryption.>>

<<Which variants of multi-hop are we doing? Should be treated in cl07, Principles of Network Operation.>>

NOTE—Authentication and authorization is outside the scope of this standard, which just ensures secure communication between mutually authenticated and authorized service access points.

6.2 Preservation of the MAC service

Preservation of the MAC Service comprises both

a) delivering the ‘correct’ parameters on a service indication (see Section 6.3), and
b) causing service indications to occur at the ‘correct’ places (Section 6.4).

The MAC Service consists of important and expected features. Attacks, or security breaches, prevent or distort the delivery of these features. Attacks may be intentional, perpetuated by an attacker, or they may come about accidentally through unintentional means, such as misconfiguration. Well-known security threats constitute attacks against provision of the expected MAC Service.

<<Specify important, interesting features of the MAC Service. What well-known attacks are threats to these services? How do threats jeopardize this service? What security countermeasures could prevent or tamper with each feature? >>

In providing safeguards or countermeasures, the MAC Service must be preserved. Security features must not alter or hamper the expected MAC Service, either in their design or implementation.

<<How do the security mechanisms need to be designed in order to avoid breaking these services?>>

6.3 Deliver ‘correct’ parameters

Preserving the MAC Service requires delivery of the ‘correct’ parameters on a service indication, i.e., the same values of the parameters as supplied by the corresponding service request, the parameters cannot undergo an unauthorized change. There may be specified exceptions in which specific parameters can be modified (or delivered with a certain probability of accidental modification) by the service provider. Each MA-UNITDATA request and indication primitive has four parameters: Destination Address, Source Address, MAC Service Data Unit (MSDU), and Priority. Each of these is further discussed as a subclause.

<<Needs blow by blow, parameter by parameter discussion, some here and some under the QoS Maintenance heading. >>

6.3.1 Destination Address

An endpoint is uniquely identified. User data is delivered ONLY to specified endpoints. If it arrives elsewhere, there is a security breach.

A frame can arrive at an unintended destination through wiretapping, a man-in-the-middle attack, address spoofing the destination address.

Cryptographic techniques can protect against such threats.

6.3.2 Source Address

Each endpoint is uniquely identified. User data comes ONLY FROM the place where the request is issued

An attacker can take on the identity of an endpoint by a man-in-the-middle attack, masquerading, spoofing the source address. Wiretapping.

Cryptographic techniques can protect against such threats.

6.3.3 MAC Service Data Unit (MSDU)

The MSDU should not undergo an unauthorized change.

<<There has been some discussion that it is not the job of MACSec to provide confidentiality for user data. This service can be provided through IPSec, for example, at a higher layer.>>

An attacker can change an MSDU via a man-in-the-middle attack, wiretapping.

Unauthorized changes to MSDU can be prevented with encryption and authentication.

6.3.4 User Priority

The User Priority parameter should not undergo an unauthorized change. A network device may need to change the priority parameter in order to accomplish drop precedence.

<<Security Associations may cause difficulty with user priority. What about multi-hop SAs? Not purely point to point, because there's one bridge between end stations. End-to-end can mean many different things : end2end and nothing else; and everything else. What's between the ends? A provider bridge between the ends? >>

6.4 'Correct' occurrence of service indications

Service indications need to occur at the 'correct' places. 'Correct' means, first, as constrained by the basic service specification - service access points compose a symmetric transitive group. Secondly, 'correct' means as required by security - only authenticated and authorized members are permitted in the group. Correct places can (by design or accident) be a subset of the places to which that unsecured can deliver.

<<There is an approved PAR for (improving) the specification of the MAC Service (P802.1ac).>>

6.4.1 MAC Service is transitive

If station A can contact Station B, and if Station B can contact Station C, then Station A can contact Station C. This feature provides connectivity. Many network services depend on this feature, e.g., OSPF.

Security Association may break transitivity. Authentication may break transitivity.

6.4.2 MAC Service is symmetric.

If station A can contact station B, then station B can contact station A.

<<The MAC Service provided by a Bridged Local Area Network is similar to that provided by a single LAN (6.5). In consequence

A Bridge is not directly addressed by communicating end stations, except as an end station for management purposes: frames transmitted between end stations carry the MAC Address of the peer-end station in their Destination Address field, not a MAC Address of the Bridge.

All MAC Addresses need to be unique within the network.

MAC Addresses of end stations are not restricted by the topology and configuration of the network.>>

6.5 Quality of service maintenance

.Attacks and security breaches also affect the quality of the MAC Service provided. Providing security must make sure to preserve the quality of service.

Quality of Service comprises

a) Service availability
b) Frame loss
c) Frame misordering
d) Frame duplication
e) Frame transit delay
f) Frame lifetime
g) Undetected frame error rate
h) Maximum service data unit size supported
i) Frame priority
j) Throughput

6.5.1 Service availability

Service availability is measured as a fraction of some total time during which the MAC Service is provided. The operation of MACsec has the potential to lower the service availability.

Service availability can be lowered by DoS attacks caused by masqerading, unauthorized data modification.

6.5.2 Frame loss

The MAC Service does not guarantee the delivery of Service Data Units. Frames transmitted by a source station arrive, uncorrupted, at the destination station with high probability. The operation of MACsec introduces minimal additional frame loss.

A frame transmitted by a source station can fail to reach its destination station as a result of DoS caused by unauthorized resource use.

6.5.3 Frame misordering

The MAC Service (9.2 of ISO/IEC 15802-1) permits a negligible rate of reordering of frames with a given user priority for a given combination of destination address and source address. MA_UNITDATA.indication service primitives corresponding to MA_UNITDATA.request primitives, with the same requested priority and for the same combination of destination and source addresses, are received in the same order as the request primitives were processed.

NOTE 1—MACsec does not mis-order or duplicate frames.

6.5.4 Frame duplication

The MAC Service (9.2 of ISO/IEC 15802-1) permits a negligible rate of duplication of frames. MACsec does not duplicate user data frames.

6.5.5 Frame transit delay

The MAC Service introduces a variable frame transit delay that is dependent on media types and media access control methods. Frame transit delay is the elapsed time between an MA_UNITDATA.request primitive and the corresponding MA_UNITDATA.indication primitive. Elapsed time values are calculated only on Service Data Units that are successfully transferred.

Since the MAC Service is provided at an abstract interface within an end station, it is not possible to specify the total frame transit delay precisely. It is, however, possible to measure the media access and frame transmission and reception, and the transit delay introduced by an intermediate system, in this case a Bridge.

The minimum additional transit delay introduced by MACsec is ...

6.5.6 Frame lifetime

The MAC Service mandates an upper bound to the transit delay experienced for a particular instance of communication. This maximum frame lifetime is necessary to ensure the correct operation of higher layer protocols. The additional transit delay introduced by MACsec is ...

6.5.7 Undetected frame error rate

The MAC Service introduces a very low undetected frame error rate in transmitted frames. Undetected errors are protected against by the use of an FCS that is appended to the frame by the MAC Sublayer of the source station prior to transmission, and checked by the destination station on reception.

It is necessary for a SecY to recalculate the FCS when .....

NOTE—Application of the techniques described in IEEE Std 802.1D Annex F (informative) allow an implementation to achieve an arbitrarily small increase in undetected frame error rate, even in cases where the data that is within the coverage of the FCS is changed.

6.5.8 Maximum Service Data Unit Size

The Maximum Service Data Unit Size that can be supported by an IEEE 802 LAN varies with the MAC method and its associated parameters (speed, electrical characteristics, etc.). It may be constrained by the owner of the LAN. .....

7. Principles of Secure Network Operation

<<This clause provides a network wide view of the operation of security and its impact on the operation of the network as a whole. The impact is almost trivial if MACsec is restricted to point to point (only two possible peers) 'single hop' (one LAN) operation. It becomes more complex if Security Associations subdivide, separate, or distinguish the connectivity that would otherwise be available. For example, such disturbances in connectivity can come about if (a) some of the stations attached to a shared media LAN participate in a secure association while others are excluded (b) the stations are grouped into a number of secure associations (c) group-wise associations are distinguished, such as those used for network configuration protocols. It becomes much more complex if the connectivity assumed by some associations rests upon the connectivity provided by others, as is likely in multi-hop scenarios. In these cases it is exceedingly necessary to keep clear if a combination of multiple associations and multi-hop are present.>>

This clause establishes the principles and a model of Secure Network operation. It specifies the context necessary to understand:

a) the operation of each MAC Security Entity (SecY) (Clause 8);
b) the establishment of secure associations between peer SecYs;
c) the operation of the MAC Security Protocol (Clause 9) between those SecYs;
d) the relationship between SecYs and the other entities that compose Systems attached to LANs (Clause 8);
e) how to support, preserve, and maintain the quality and security (Clause 6) of the Secure MAC Service, including
f) the operation of network configuration protocols

NOTE 1—Unless explicitly stated the use of the term ‘secure network’ in this Standard refers to a Bridged Local Area Network in part of which MAC Security is active.

7.1 Secure Network Overview

The principal elements of Secure Network operation comprise:

a) blah.
b) more blah.

and may also include:

c) optional blah.

The position of the bridging function within the MAC Sublayer is shown in Figure 7-1.

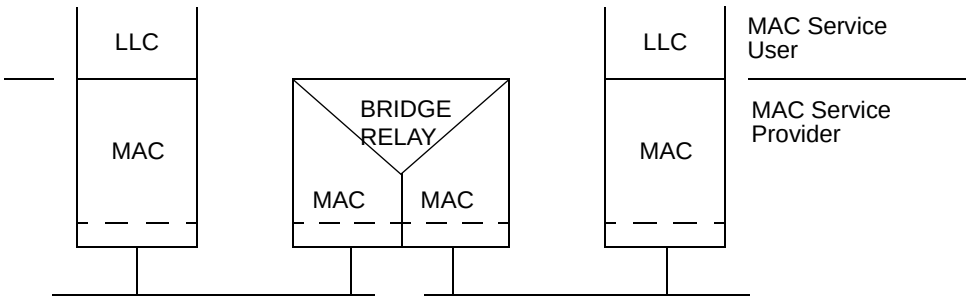


Figure 7-1—<<replace with diagrams that show SecYs in the appropriate places>>

8. Principles of MAC Security Entity operation

<<Material borrowed from 802.1D is scattered through this clause as a prompt to the editor and reviewers to supply analogous material for MAC Security, if appropriate.>>

This clause

a) Introduces the principal elements of MAC Security Entity (SecY) operation and the functions that support these elements.
b) Describes an architectural model for a MAC Security Entity that governs the provision of these functions.
c) Provides a model of SecY operation in terms of Processes and Entities that support the functions.
d) Details the addressing requirements and specifies the addressing of SecYs.

8.1 Elements of SecY operation

The principal elements of SecY operation are:

a) Element 1
b) Element 2

8.1.1 Element 1

A SecY .....encodes/wraps user data frames between the underlying service and users connected to its Authorized and Uncontrolled Ports. The functions that support ... encoding and wrapping of frames ... and maintain Quality of Service are:

a) Frame reception.
b) Discard on received frame in error (6.3.2).
c) Frame discard if the frame_type is not user_data_frame (6.4).
d) Frame discard on transmittable service data unit size exceeded (6.3.8).
e) Selection of traffic class, following the application of filtering information.
f) Mapping of service data units and recalculation of Frame Check Sequence, if required (6.3.7, 8.7.6).
g) Frame transmission.

8.1.2 Element 2

A ...

h) .

8.1.3 MACsec management

The functions that support Bridge Management control and monitor the provision of the above functions. They are specified in Clause ??.

8.2 SecY architecture

A SecY is modeled as comprising:

a) first bit;
b) second bit;
c) interfaces to Higher layer entities, including at least a Key Management Entity.

The SecY handles the Media Access Method Independent Functions of ... It uses the Internal Sublayer Service (6.4, 6.5) provided by the separate MAC Entities of [each Port].

8.3 Model of operation

The model of operation is simply a basis for describing the functionality of a SecY. It is in no way intended to constrain real implementations; these may adopt any internal model of operation compatible with the externally visible behavior that this standard specifies. Conformance of equipment to this standard is purely in respect of observable protocol.

Subclauses ?? and 8.6 specify the MAC SecY's use of the Extended Internal Sublayer Service. Port State information (8.4) governs each Port's participation in the Bridged Local Area Network.

8.4 Frame reception

The individual MAC Entity of each Bridge Port examines all frames transmitted on the attached LAN.

All error-free received frames give rise to M_UNITDATA indications which are handled as follows.

Frames with an M_UNITDATA.indication frame_type of user_data_frame (6.4), shall be processed by the SecY. Frames with other values of frame_type shall be discarded.

<<more to be said>>

8.5 Frame transmission

.....

A frame queued for transmission on a Port shall be removed from that queue if the associated Port leaves the Forwarding State. Removal of a frame from a queue for any particular Port does not of itself imply that it is to be removed from a queue for transmission on any other Port.

8.5.1 Transmission selection

The following shall be supported by as the default algorithm for selecting frames for transmission:

a) For each Port, frames are selected for transmission on the basis of the traffic classes that the Port supports. For a given supported value of traffic class, frames are selected from the corresponding queue for transmission only if all queues corresponding to numerically higher values of traffic class supported by the Port are empty at the time of selection;
b) For a given queue, the order in which frames are selected for transmission shall maintain the ordering requirement specified in 8.7.3.

Additional algorithms that meet the requirements of 8.7.3 may be selected by management.

8.5.2 FCS recalculation

<<Since fields are to be inserted/and removed from the frame something needs to be said here, probably similar to the text in 802.1D, which is included here for the editors convenience.>>

When a frame is forwarded between two MAC Entities of the same IEEE 802 LAN type, and the data that is within the FCS coverage is not modified, the FCS received in the M_UNITDATA.indication primitive may be supplied in the corresponding M_UNITDATA.request primitive and not recalculated (6.3.7). For frames relayed between LANs of the same MAC type, the Bridge shall not introduce an undetected frame error rate greater than that which would have been achieved by preserving the FCS.

When a frame is forwarded between two MAC Entities of different types, the FCS is recalculated according to the procedures of the transmitting MAC entity if they differ from the FCS calculation procedures of the receiving MAC or the data within the FCS coverage is changed.

NOTE—There are two possibilities for recreating a valid FCS. The first is to generate a new FCS by algorithmically modifying the received FCS, based on knowledge of the FCS algorithm and the transformations that the frame has undergone between reception and transmission. The second is to rely on the normal MAC procedures to recalculate the FCS for the outgoing frame. The former approach can protect against increased levels of undetected frame errors. Annex F (informative) discusses these possibilities in more detail. The frame_check_sequence parameter of the Internal Sublayer Service (6.4) signals the validity, or otherwise, of the FCS; an unspecified value in this parameter in a data request indicates to the transmitting MAC that the FCS is to be recalculated.

8.6 Addressing

<<Some of the very same things that are said about addressing in Bridges apply to SecYs, so some of the text from 802.1D is included here for the editors convenience.>>

All MAC Entities communicating across a Bridged Local Area Network use 48-bit addresses. These can be Universally Administered Addresses or a combination of Universally Administered and Locally Administered Addresses.

8.6.1 End stations

Frames transmitted between end stations using the MAC Service provided by a Bridged Local Area Network carry the MAC Address of the source and destination peer end stations in the source and destination address fields of the frames, respectively. The address, or other means of identification, of a Bridge is not carried in frames transmitted between peer users for the purpose of frame relay in the network.

The broadcast address and other group MAC Addresses apply to the use of the MAC Service provided by a Bridged Local Area Network as a whole. In the absence of explicit filters configured via management as Static Filtering Entries, or via GMRP as Group Registration Entries (Clause 14, Clause 10, 8.9), frames with such destination addresses are relayed throughout the network.

8.6.2 Bridge Ports

The individual MAC Entity associated with each Bridge Port shall have a separate individual MAC Address. This address is used for any MAC procedures required by the particular MAC.

Frames received from or relayed to the LAN to which a Port is attached and which carry a MAC Address for the Port in the destination MAC address field shall be submitted to the MAC Service User (LLC), and to the LLC Service User for the LSAP identified by the destination LLC Address exactly as for an end station.

8.6.3 SecYs

.....

9. MAC Security Protocol (MACsec)

MACsec provides security services on a PDU by PDU basis .....

9.1 Protocol design requirements

MACsec operates in Bridged Local Area Networks and Virtually Bridged Local Area Networks comprising individual point to point or shared media LANs arbitrarily interconnected by MAC Bridges and VLAN Bridges. Each of the end systems and bridges may incorporate MAC Security Enties (SecYs). MACsec supports, preserves, and maintains the quality of the Secure MAC Service in all its aspects as specified by Clause 6, meeting the following requirements:

a) Connectivity is symmetric, i.e. if system A can communicate with system B, B can communicate with A.
b) Connectivity is transitive, i.e. if system A can use the secure MAC service at one MSAP to communicate with system B and system C then system B and system C can communicate using their same MSAPs.

Additionally, the algorithm and protocol meet the following goals, which limit the complexity of SecYs and their configuration:

c) The memory requirements associated with SecY are independent of .....

9.2 Protocol support requirements

In order for the MACsec to operate the following are required

a) A unique MAC Address for each SecY, unique within the Bridged Local Area Network

Values for each of these parameters shall be provided by each Bridge. The unique MAC Address that identifies the Spanning Tree Protocol Entities is the Bridge Group Address ().

To allow management of the ....., means of assigning values to the following are required:

b) Placeholder.

9.3 MACsec overview

MACsec ... secure association ... integrity check value .. keys ... liveness ... relationship to key management ....

9.4 Performance parameter management

Table XXX specifies default values and ranges for timer and transmission rate limiting performance parameters. Defaults are specified to avoid the need to set values prior to operation in most cases, and have been chosen for their wide applicability to maximize ease of operation. Ranges are specified to ensure that the protocol operates correctly, and provide guidance to implementors.

NOTE—Changes to Bridge Forward Delay do not affect reconfiguration times, unless the network includes Bridges that do not conform to this revision of this Standard. Changes to Bridge Max Age can have an effect, as it is possible for old information to persist in loops in the physical topology for a number of ‘hops’ equal to the value of Max Age in seconds, and thus exhaust the Transmit Hold Count in small loops.

Table 9-1—Performance parameters

Parameter	Recommended or Default value	Permitted Range	Compatibility Range

All times are in seconds. —[1] Not applicable, value is fixed.

9.5 MACsec state machines

The behavior of a SecY implementation in a Bridge is specified by a number of cooperating state machines. Figure 9-2 is not itself a state machine, but illustrates the machines, their interrelationships, the principal variables used to communicate between them, their local variables, and performance parameters.

One instance of each of the other state machines shall be implemented.

9.6 Notational conventions used in state diagrams

State diagrams are used to represent the operation of the protocol by a number of cooperating state machines each comprising a group of connected, mutually exclusive states. Only one state of each machine can be active at any given time.

Each state is represented in the state diagram as a rectangular box, divided into two parts by a horizontal line. The upper part contains the state identifier, written in upper case letters. The lower part contains any procedures that are executed on entry to the state.

All permissible transitions between states are represented by arrows, the arrowhead denoting the direction of the possible transition. Labels attached to arrows denote the condition(s) that must be met in order for the transition to take place. All conditions are expressions that evaluate to TRUE or FALSE; if a condition evaluates to TRUE, then the condition is met. The label UCT denotes an unconditional transition (i.e., UCT always evaluates to TRUE). A transition that is global in nature (i.e., a transition that occurs from any of the possible states if the condition attached to the arrow is met) is denoted by an open arrow; i.e., no specific state is identified as the origin of the transition. When the condition associated with a global transition is met, it supersedes all other exit conditions including UCT. The special global condition BEGIN supersedes all other global conditions, and once asserted remains asserted until all state blocks have executed to the point that variable assignments and other consequences of their execution remain unchanged.

On entry to a state, the procedures defined for the state (if any) are executed exactly once, in the order that they appear on the page. Each action is deemed to be atomic; i.e., execution of a procedure completes before the next sequential procedure starts to execute. No procedures execute outside of a state block. The procedures in only one state block execute at a time, even if the conditions for execution of state blocks in different state machines are satisfied, and all procedures in an executing state block complete execution before the transition to and execution of any other state block occurs, i.e. the execution of any state block appears to be atomic with respect to the execution of any other state block and the transition condition to that state from the previous state is TRUE when execution commences. The order of execution of state blocks in different state machines is undefined except as constrained by their transition conditions. A variable that is set to a particular value in a state block retains this value until a subsequent state block executes a procedure that modifies the value.

Figure 9-2—MACsec state machines - overview and interrelationships

On completion of all of the procedures within a state, all exit conditions for the state (including all conditions associated with global transitions) are evaluated continuously until one of the conditions is met. The label ELSE denotes a transition that occurs if none of the other conditions for transitions from the state are met (i.e., ELSE evaluates to TRUE if all other possible exit conditions from the state evaluate to FALSE). Where two or more exit conditions with the same level of precedence become TRUE simultaneously, the choice as to which exit condition causes the state transition to take place is arbitrary.

Where it is necessary to split a state machine description across more than one diagram, a transition between two states that appear on different diagrams is represented by an exit arrow drawn with dashed lines, plus a reference to the diagram that contains the destination state. Similarly, dashed arrows and a dashed state box are used on the destination diagram to show the transition to the destination state. In a state machine that has been split in this way, any global transitions that can cause entry to states defined in one of the diagrams are deemed to be potential exit conditions for all of the states of the state machine, regardless of which diagram the state boxes appear in.

Should a conflict exist between the interpretation of a state diagram and either the corresponding global transition tables or the textual description associated with the state machine, the state diagram takes precedence. The interpretation of the special symbols and operators used in the state diagrams is as defined in Table 9-2; these symbols and operators are derived from the notation of the “C++” programming language, ISO/IEC 14882. If a boolean variable is described in this clause as being set it has or is assigned the value TRUE, if reset or clear the value FALSE.

Table 9-2—State machine symbols

Symbol	Interpretation
( )	Used to force the precedence of operators in Boolean expressions and to delimit the argument(s) of actions within state boxes.
;	Used as a terminating delimiter for actions within state boxes. Where a state box contains multiple actions, the order of execution follows the normal English language conventions for reading text.
=	Assignment action. The value of the expression to the right of the operator is assigned to the variable to the left of the operator. Where this operator is used to define multiple assignments, e.g., a = b = X the action causes the value of the expression following the right-most assignment operator to be assigned to all of the variables that appear to the left of the right-most assignment operator.
!	Logical NOT operator.
&&	Logical AND operator.
||	Logical OR operator.
if...then...	Conditional action. If the Boolean expression following the if evaluates to TRUE, then the action following the then is executed.
!=	Inequality. Evaluates to TRUE if the expression to the left of the operator is not equal in value to the expression to the right.
==	Equality. Evaluates to TRUE if the expression to the left of the operator is equal in value to the expression to the right.
*	Arithmetic multiplication operator.
-	Arithmetic subtraction operator.

9.7 State machine timers

The timer variables declared in this clause, 9.7, are part of the specification of the operation of the SecY. The accompanying descriptions of their meaning and use are provided to aid in the comprehension of the protocol only, and are not part of the specification. A SecY implementation shall implement a single instance of each timer variable.

Each timer variable represents an integral number of seconds before timer expiry.

9.7.1 Timer 1

Timer 1.

9.8 State machine variables

The variables declared in this clause, 9.8, are part of the specification of the operation of the SecY. The accompanying descriptions of their use are provided to aid in the comprehension of the protocol only, and are not part of the specification.

9.8.1 BEGIN

A boolean controlled by the system initialization (9.6). If TRUE causes all state machines, including per Port state machines, to continuously execute their initial state.

9.8.2 Variable 1.

Variable 1.

9.8.3 portEnabled

A boolean. Set if the SecY can use the MAC Service provided by the Port's MAC entity to transmit and receive frames to and from the attached LAN, i.e. portEnabled is TRUE if and only if:

a) MAC_Operational (IEEE Std 802.1D Clause 6.4.2) is TRUE.

9.9 State machine conditions and parameters

The following variable evaluations are defined for notational convenience in the state machines.

9.9.1 Condition 1.

Condition 1.

9.10 State machine procedures

The following naming convention is used for the names of procedures that modify multiple variables (either multiple variables of a single Port or variables of multiple Ports):

a) ***set***: The procedure sets the value of the variables to TRUE.
b) ***clear:*** The procedure clears (resets) the value of the variables to FALSE.
c) ***updt:*** The procedure updates the variables in some other way.

The suffix "Tree" is used for procedures that can modify a variable in all Ports of the Bridge. For example, ***setSyncTree()*** is the name of a procedure that sets a variable TRUE for all Bridge Ports.

Where procedures are used to determine the value of a single variable, the procedure's returned value is explicitly assigned to the variable in the state machine concerned.

9.10.1 proc1()

Returns TRUE if ....

9.11 XXX state machine

The XXX state machine shall implement the function specified by the state diagram in Figure 9-3, the definitions in 9.6, and the variable declarations and procedures specified in 9.7 thru 9.10.

Figure 9-3—XXX state machine

9.12 SecY performance requirements

This clause (9.12) places requirements on the performance of the SecYs to ensure that MACsec operates correctly.

10. Encoding of Secure MAC protocol data units

<<Material borrowed from 802.1D is scattered through this clause as a prompt to the editor and reviewers to supply analagous material for MAC Security, if appropriate.>>

This clause specifies the structure and encoding of the Secure MAC Protocol Data Units (SMPDUs) exchanged between MAC Security Enties (SecYs) for the MAC Security Protocol (MACsec) (Clause 9).

10.1 Structure

10.1.1 Transmission and representation of octets

All BPDUs shall contain an integral number of octets. The octets in a BPDU are numbered starting from 1 and increasing in the order they are put into a Data Link Service Data Unit (DLSDU). The bits in an octet are numbered from 1 to 8, where 1 is the low-order bit.

When consecutive bits within an octet are used to represent a binary number, the higher bit number has the most significant value. When consecutive octets are used to represent a binary number, the lower octet number has the most significant value. All Bridge Protocol Entities respect these bit and octet ordering conventions, thus allowing communications to take place.

10.1.2 Components

A Protocol Identifier is encoded in the initial octets of all SMPDUs. This standard specifies a single Protocol Identifier value for use in SMPDUs. All other Protocol Identifier values are reserved for future standards use. This standard places no further restriction on the structure, encoding, or use of SMPDUs with different values of the Protocol Identifier field, should these exist, by other standard protocols.

10.2 Encoding of parameter types

10.2.1 Encoding of protocol identifiers

A Protocol Identifier shall be encoded in two octets.

10.2.2 Encoding of protocol version identifiers

A Protocol Version Identifier shall be encoded in one octet. If two Protocol Version Identifiers are interpreted as unsigned binary numbers, the greater number identifies the more recently defined Protocol Version.

10.2.3 Encoding of BPDU types

The type of the BPDU shall be encoded as a single octet. The bit pattern contained in the octet merely serves to distinguish the type; no ordering relationship between BPDUs of different types is implied.

10.2.4 Encoding of flags

A flag shall be encoded as a bit in a single octet. A flag is set if the bit takes the value 1. A number of flags may be encoded in a single octet. Bits in the octet that do not correspond to flags defined for the BPDU's type are reset, i.e., shall take the value 0. No additional flags will be defined for a BPDU of given protocol version and type.

10.2.5 Encoding of Security Association Identifiers

.....

10.2.6 Encoding of Data Origin Identifiers

.....

10.3 SMPDU formats and parameters

10.3.1 Single hop point-to-point SMPDUs.

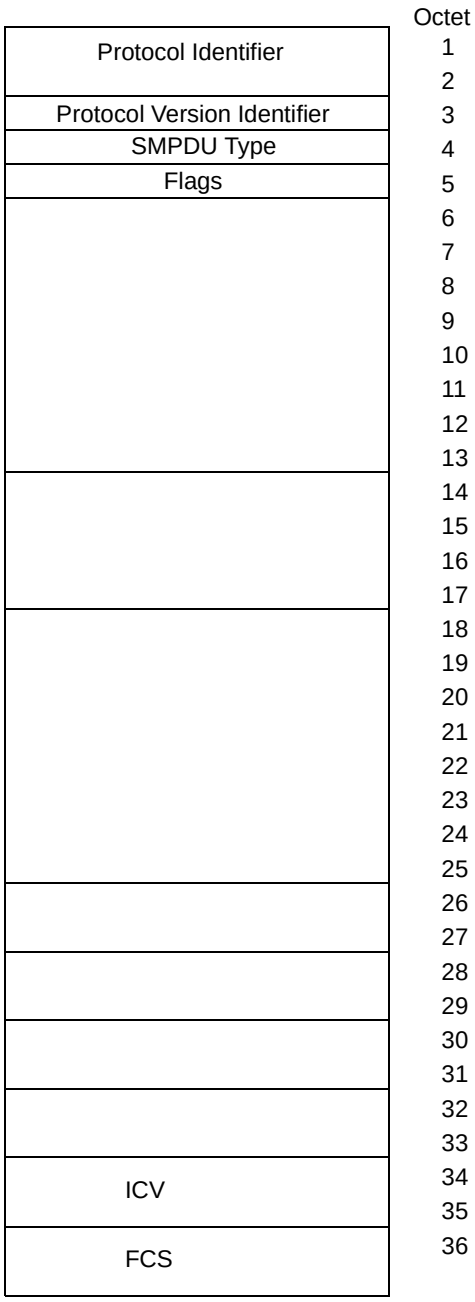


Figure 10-4—SMPDU parameters and format

11. Management of MAC Security Entities

This clause defines the set of managed objects, and their functionality, that allow administrative configuration and monitoring of MAC Security Entities.

This clause

a) Introduces the functions of management to assist in the identification of the requirements placed on MAC Security Entities for the support of management facilities
b) Establishes the correspondence between the state machines used to model the operation of a MAC Security Entity (SECy) () and its managed objects
c) Specifies the management operations supported by each managed object

11.1 Management functions

Management functions relate to the users' needs for facilities that support the planning, organization, supervision, control, protection, and security of communications resources, and account for their use. These facilities may be categorized as supporting the functional areas of Configuration, Fault, Performance, Security, and Accounting Management. Each of these is summarized in 11.1.1 through 11.1.5, together with the facilities commonly required for the management of communication resources, and the particular facilities provided in that functional area by MAC Security Entity Management.

11.1.1 Configuration Management

Configuration Management provides for the identification of communications resources, initialization, reset and close-down, the supply of operational parameters, and the establishment and discovery of the relationship between resources. The facilities provided by MAC Security Entity Management in this functional area are as follows:

a) Configuration of the operational parameters for the SECy (11.4.1.1 and 11.4.1.2)
b) Initialization of the state machines for the SECy ()

11.1.2 Fault Management

Fault Management provides for fault prevention, detection, diagnosis, and correction. The facilities provided by MAC Security Entity Management in this functional area are as follows:

a) Retrieval of SECy statistical information ()
b) Configuration of the operational parameters for the SECy (11.4.1.1 and 11.4.1.2)

11.1.3 Performance Management

Performance Management provides for evaluation of the behavior of communications resources and of the effectiveness of communication activities. The facilities provided by MAC Security Entity Management in this functional area are

a) Retrieval of statistical information ()
b) Configuration of the operational parameters (11.4.1.1 and 11.4.1.2)

11.1.4 Security Management

Security Management provides for the protection of resources. The facilities provided by MAC Security Entity Management in this functional area are as follows:

a) ???

11.1.5 Accounting Management

Accounting Management provides for the identification and distribution of costs and the setting of charges. The facilities provided by MAC Security Entity Management in this functional area is as follows:

a) Retrieval of accounting statistics (11.4.1.3)

11.2 Managed objects

Managed objects model the semantics of management operations. Operations upon a managed object supply information concerning, or facilitate control over, the Process or Entity associated with that managed object.

Management of MAC Security Entity is described in terms of the managed resources that are associated with individual Ports that support MAC Security. The managed resources of a SECy are those of the Processes and Entities established in . Specifically,

a) first sort of resource.....
b) .....

The management of these resources is described in terms of managed objects and operations defined below.

NOTE—The values specified in this clause, as inputs and outputs of management operations, are abstract information elements. Questions of formats or encodings are a matter for particular protocols that convey or otherwise represent this information.

11.3 Data types

This subclause specifies the semantics of operations independent of their encoding in management protocol. The data types of the parameters of operations are defined only as required for that specification.

The following data types are used:

a) Boolean
b) Enumerated, for a collection of named values
c) Unsigned, for all parameters specified as “the number of” some quantity
d) MAC Address
e) Time Interval, an Unsigned value representing a positive integral number of seconds, for all protocol timeout parameters
f) Counter, for all parameters specified as a “count” of some quantity (a counter increments and wraps with a modulus of 2 to the power of 64)

11.4 MAC Security Entity first sort of resource managed objects

The .....are described in .....

The objects that comprise this managed resource are as follows:

a) ....
b) ...

A MAC Security Entity that supports ..... functionality shall support the management functionality defined by the ... managed object. A MAC Security Entity that supports ..... functionality may support the management functionality defined by the ..... managed objects.

The means by which this management functionality is provided (e.g., the management protocol supported) shall be stated in the PICS associated with the implementation.

11.4.1 first resource first object

The .... managed object models the operations that modify, or enquire about, the configuration of the MAC Security Entity's resources. There is a single MAC Security Entity Configuration managed object for each MAC Security Entity.

The management operations that can be performed on the .... managed object are

a) Read .. Configuration (11.4.1.1)
b) Set .. Configuration (11.4.1.2)
c) ...(11.4.1.3)

11.4.1.1 Read ... Configuration

11.4.1.1.1 Purpose

To solicit configuration information regarding the configuration of the MAC Security Entity.

11.4.1.1.2 Inputs

— **Identfier...**.

11.4.1.1.3 Outputs

a) **Identfier...** The identification number assigned to the MAC Security Entity**....**
b) .....

11.4.1.2 Set ... Configuration

11.4.1.2.1 Purpose

To configure the parameters that control the operation of the MAC Security Entity.

11.4.1.2.2 Inputs

Any parameters marked as (optional) may be omitted from the operation to allow selective modification of a subset of the configuration parameters. However, implementations shall support the ability to include all of the parameters identified below.

a) ....
b)
c)

11.4.1.2.3 Outputs

None.

11.4.1.3 .....

11.4.1.3.1 Purpose

...

11.4.1.3.2 Inputs

a) ...

11.4.1.3.3 Outputs

None.

11.4.1.3.4 Effect

This operation ...

12. Management protocol

12.1 Introduction

This clause defines a portion of the Management Information Base (MIB) for use with network management protocols in TCP/IP-based internets. In particular, it defines objects for managing the operation of MAC Security, based on the specification contained in Clause 8 and Clause 1. This clause includes a MIB module that is SNMPv2 SMI compliant.

12.2 The SNMP Management Framework

The SNMP Management Framework presently consists of five major components:

a) An overall architecture, described in IETF RFC 2571.
b) Mechanisms for describing and naming objects and events for the purpose of management. The first version of this Structure of Management Information (SMI) is called SMIv1 and described in IETF RFC 1155, IETF RFC 1212, and IETF RFC 1215. The second version, called SMIv2, is described in IETF RFC 2578, IETF RFC 2579, and IETF RFC 2580.
c) Message protocols for transferring management information. The first version of the SNMP message protocol is called SNMPv1 and described in IETF RFC 1157. A second version of the SNMP message protocol, which is not an Internet standards track protocol, is called SNMPv2c and is described in IETF RFC 1901 and IETF RFC 1906. The third version of the message protocol is called SNMPv3 and is described in IETF RFC 1906, IETF RFC 2572 and IETF RFC 2574.
d) Protocol operations for accessing management information. The first set of protocol operations and associated PDU formats is described in IETF RFC 1157. A second set of protocol operations and associated PDU formats is described in IETF RFC 1905.
e) A set of fundamental applications described in IETF RFC 2573 and the view-based access control mechanism described in IETF RFC 2575.

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the mechanisms defined in the SMI.

This clause specifies a MIB module that is compliant to the SMIv2. A MIB conforming to the SMIv1 can be produced through the appropriate translations. The resulting translated MIB must be semantically equivalent, except where objects or events are omitted because no translation is possible (use of Counter64). Some machine-readable information in SMIv2 will be converted into textual descriptions in SMIv1 during the translation process. However, this loss of machine-readable information is not considered to change the semantics of the MIB.

12.3 Security considerations

A number of management objects are defined in this MIB that have a MAX-ACCESS clause of read-write or read-create. Such objects may be considered sensitive or vulnerable in some network environments. The support for SET operations in a nonsecure environment without proper protection can have a negative effect on network operations.

SNMPv1 by itself is not a secure environment. Even if the network is secure (for example, by using IPSec), there is no control as to who on the secure network is allowed to access (read/change/create/delete) the objects in this MIB.

It is recommended that the implementors consider the security features as provided by the SNMPv3 framework. Specifically, the use of the User-based Security Model, IETF RFC 2574, and the View-based

Access Control Model, IETF RFC 2575, is recommended. It then becomes a user responsibility to ensure that the SNMP entity giving access to an instance of this MIB is properly configured to give access only to those principals (users) that have legitimate rights to access (change/create/delete) them, as appropriate.

12.4 Structure of the MIB

A single MIB module is defined in this clause. Objects in the MIB are arranged into groups. Each group is organized as a set of related objects. The overall structure and assignment of objects to their groups is shown in the following subclauses.

12.4.1 Relationship to the managed objects defined in Clause 11

Table 12-1 contains cross-references between the objects defined in Clause 11 and the MIB objects defined in this clause.

Table 12-1—Managed object cross-reference table

Definition in Clause 11	MIB object(s)
11.4.1 first resource first object	**dot1aeMACsecy**
....	
....	
....	
....	
Initialize SECy	

Table 12-1—Managed object cross-reference table *(continued)*

Definition in Clause 11	MIB object(s)
Statistics	**dot1aeStatsTable**
Diagnostics	**dot1aeDiagTable**

12.4.2 The .....Group

This group of objects provides management functionality that is not specific to the operation of .....

12.4.3 The ... Group

This group of objects provides.....

12.5 Relationship to other MIBs

It is assumed that a system implementing this MIB will also implement (at least) the “system” group defined in MIB-II defined in IETF RFC 1213 and the “interfaces” group defined in IETF RFC 2863.

12.5.1 Relationship to the Interfaces MIB

IETF RFC 2863, the Interface MIB Evolution, requires that any MIB that is an adjunct of the Interface MIB clarify specific areas within the Interface MIB. These areas were intentionally left vague in IETF RFC 2863 to avoid overconstraining the MIB, thereby precluding management of certain media types.

Section 3.3 of IETF RFC 2863 enumerates several areas that a media-specific MIB must clarify. Each of these areas is addressed in a following subsection. The implementor is referred to IETF RFC 2863 in order to understand the general intent of these areas.

In IETF RFC 2863, the “interfaces” group is defined as being mandatory for all systems and contains information on an entity’s interfaces, where each interface is thought of as being attached to a *subnetwork*. (Note that this term is not to be confused with *subnet*, which refers to an addressing partitioning scheme used in the Internet suite of protocols.) The term *segment* is sometimes used to refer to such a subnetwork.

Where Port numbers are used in this standard to identify Ports of a System, these numbers are equal to the ifIndex value for the interface for the corresponding Port.

12.6 Definitions for MAC Security MIB

In the MIB definition below, should any discrepancy between the DESCRIPTION text and the corresponding definition in Clause 11 occur, the definition in Clause 11 shall take precedence.

```
IEEE8021-SECY-MIB DEFINITIONS ::= BEGIN

-- ------------------------------------------------------------ --
-- IEEE 802.1AE MIB
-- ------------------------------------------------------------ --

IMPORTS
    MODULE-IDENTITY, OBJECT-TYPE, Counter32, Counter64,
    Unsigned32, TimeTicks
        FROM SNMPv2-SMI
    MacAddress, TEXTUAL-CONVENTION, TruthValue
        FROM SNMPv2-TC
    MODULE-COMPLIANCE, OBJECT-GROUP
        FROM SNMPv2-CONF
```

```
    SnmpAdminString
        FROM SNMP-FRAMEWORK-MIB
    InterfaceIndex
        FROM IF-MIB
    ;

ieee8021secyMIB MODULE-IDENTITY
    LAST-UPDATED "200307170000Z" -- 17th July 2003
    ORGANIZATION "IEEE 802.1 Working Group"
    CONTACT-INFO
        "http://grouper.ieee.org/groups/802/1/index.html"
    DESCRIPTION
        "The MAC Security Entity"
    REVISION "200307170000Z" -- 17th July 2003
    DESCRIPTION
        "Beginnings of a draft using the 802.1X MIB as a model"
::= { iso(1) std(0) iso8802(8802) ieee802dot1(1)
          ieee802dot1mibs(1) 1 }

paeMIBObjects OBJECT IDENTIFIER ::= { ieee8021paeMIB 1 }

-- ------------------------------------------------------------ --
-- Textual Conventions
-- ------------------------------------------------------------ --

PaeControlledDirections ::= TEXTUAL-CONVENTION
    STATUS      current
    DESCRIPTION
        "The control mode values for the Authenticator PAE."
    SYNTAX      INTEGER {
                    both(0),
                    in(1)
                }

PaeControlledPortStatus ::= TEXTUAL-CONVENTION
    STATUS       current
    DESCRIPTION
        "The status values of the Authenticator PAE controlled
        Port."
    SYNTAX      INTEGER {
                    authorized(1),
                    unauthorized(2)
                }

PaeControlledPortControl ::= TEXTUAL-CONVENTION
    STATUS       current
```

```
    DESCRIPTION
        "The control values of the Authenticator PAE controlled
        Port."
    SYNTAX      INTEGER {
                    forceUnauthorized(1),
                    auto(2),
                    forceAuthorized(3)
                }

-- ----------------------------------------------------------- --

-- ----------------------------------------------------------- --
-- groups in the PAE MIB
-- ----------------------------------------------------------- --

dot1xPaeSystem        OBJECT IDENTIFIER ::= { paeMIBObjects 1 }
dot1xPaeAuthenticator OBJECT IDENTIFIER ::= { paeMIBObjects 2 }
dot1xPaeSupplicant    OBJECT IDENTIFIER ::= { paeMIBObjects 3 }

-- ----------------------------------------------------------- --

-- ----------------------------------------------------------- --
-- The PAE System Group
-- ----------------------------------------------------------- --

dot1xPaeSystemAuthControl OBJECT-TYPE
    SYNTAX      INTEGER { enabled(1), disabled(2) }
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The administrative enable/disable state for
         Port Access Control in a System."
    REFERENCE
        "<clause ref>, SystemAuthControl"
    ::= { dot1xPaeSystem 1 }

-- ----------------------------------------------------------- --
-- The PAE Port Table
-- ----------------------------------------------------------- --

dot1xPaePortTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF Dot1xPaePortEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table of system level information for each port
```

```
        supported by the Port Access Entity.  An entry appears
        in this table for each port of this system."
    REFERENCE
        "<clause ref>"
    ::= { dot1xPaeSystem 2 }

dot1xPaePortEntry OBJECT-TYPE
    SYNTAX      Dot1xPaePortEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The Port number, protocol version, and
        initialization control for a Port."
    INDEX { dot1xPaePortNumber }
    ::= { dot1xPaePortTable 1 }

Dot1xPaePortEntry ::=
    SEQUENCE {
        dot1xPaePortNumber
            InterfaceIndex,
        dot1xPaePortProtocolVersion
            Unsigned32,
        dot1xPaePortCapabilities
            BITS,
        dot1xPaePortInitialize
            TruthValue,
        dot1xPaePortReauthenticate
            TruthValue
        }

dot1xPaePortNumber OBJECT-TYPE
    SYNTAX      InterfaceIndex
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The Port number associated with this Port."
    REFERENCE
        "<clause ref>, Port number"
    ::= { dot1xPaePortEntry 1 }

dot1xPaePortProtocolVersion OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The protocol version associated with this Port."
```

```
    REFERENCE
        "<clause ref>, Protocol version"
    ::= { dot1xPaePortEntry 2 }

dot1xPaePortCapabilities OBJECT-TYPE
    SYNTAX      BITS {
                    dot1xPaePortAuthCapable(0),
                        -- Authenticator functions are supported
                    dot1xPaePortSuppCapable(1)
                        -- Supplicant functions are supported
                }
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicates the PAE functionality that this Port
        supports and that may be managed through this MIB."
    REFERENCE
        "<clause ref>, PAE Capabilities"
    ::= { dot1xPaePortEntry 3 }

dot1xPaePortInitialize OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The initialization control for this Port. Setting this
        attribute TRUE causes the Port to be initialized.
        The attribute value reverts to FALSE once initialization
        has completed."
    REFERENCE
        "<clause ref>, Initialize Port"
    ::= { dot1xPaePortEntry 4 }

dot1xPaePortReauthenticate OBJECT-TYPE
  SYNTAX TruthValue
  MAX-ACCESS read-write
  STATUS current
  DESCRIPTION
    "The reauthentication control for this port. Setting
    this attribute TRUE causes the Authenticator PAE state
    machine for the Port to reauthenticate the Supplicant.
    Setting this attribute FALSE has no effect.
    This attribute always returns FALSE when it is read."
  REFERENCE
    "<clause ref> Reauthenticate"
::= { dot1xPaePortEntry 5 }
```

```
-- ------------------------------------------------------- --
-- The PAE Authenticator Group
-- ------------------------------------------------------- --

-- ------------------------------------------------------- --
-- The Authenticator Configuration Table
-- ------------------------------------------------------- --

dot1xAuthConfigTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF Dot1xAuthConfigEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table that contains the configuration objects for the
        Authenticator PAE associated with each port.
        An entry appears in this table for each port that may
        authenticate access to itself."
    REFERENCE
        "<clause ref> Authenticator Configuration"
    ::= { dot1xPaeAuthenticator 1 }

dot1xAuthConfigEntry OBJECT-TYPE
    SYNTAX      Dot1xAuthConfigEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The configuration information for an Authenticator
        PAE."
    INDEX { dot1xPaePortNumber }
    ::= { dot1xAuthConfigTable 1 }

Dot1xAuthConfigEntry ::=
    SEQUENCE {
        dot1xAuthPaeState
            INTEGER,
        dot1xAuthBackendAuthState
            INTEGER,
        dot1xAuthAdminControlledDirections
            PaeControlledDirections,
        dot1xAuthOperControlledDirections
            PaeControlledDirections,
        dot1xAuthAuthControlledPortStatus
            PaeControlledPortStatus,
        dot1xAuthAuthControlledPortControl
            PaeControlledPortControl,
```

```
        dot1xAuthQuietPeriod
            Unsigned32,
        dot1xAuthTxPeriod
            Unsigned32,
        dot1xAuthSuppTimeout (Deprecated)
            Unsigned32,
        dot1xAuthServerTimeout
            Unsigned32,
        dot1xAuthMaxReq (Deprecated, value is ignored)
            Unsigned32,
        dot1xAuthReAuthPeriod
            Unsigned32,
        dot1xAuthReAuthEnabled
            TruthValue,
        dot1xAuthKeyTxEnabled
            TruthValue,
        dot1xAuthInitEapCode
            INTEGER,
        dot1xAuthInitEapData
            OCTET STRING
        }

dot1xAuthPaeState OBJECT-TYPE
    SYNTAX      INTEGER {
                    initialize(1),
                    disconnected(2),
                    connecting(3),
                    authenticating(4),
                    authenticated(5),
                    aborting(6),
                    held(7),
                    forceAuth(8),
                    forceUnauth(9),
                    restart(10)
                }
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The current value of the Authenticator PAE state
        machine."
    REFERENCE
        "<clause ref>, Authenticator PAE state"
    ::= { dot1xAuthConfigEntry 1 }

dot1xAuthBackendAuthState OBJECT-TYPE
    SYNTAX      INTEGER {
```

```
                    request(1),
                    response(2),
                    success(3),
                    fail(4),
                    timeout(5),
                    idle(6),
                    initialize(7),
                    ignore(8)
                }
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The current state of the Backend Authentication
        state machine."
    REFERENCE
        "<clause ref>, Backend Authentication state"
    ::= { dot1xAuthConfigEntry 2 }

dot1xAuthAdminControlledDirections OBJECT-TYPE
    SYNTAX      PaeControlledDirections
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The current value of the administrative controlled
        directions parameter for the Port."
    REFERENCE
        "<clause ref>, Admin Control Mode"
    ::= { dot1xAuthConfigEntry 3 }

dot1xAuthOperControlledDirections OBJECT-TYPE
    SYNTAX      PaeControlledDirections
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The current value of the operational controlled
        directions parameter for the Port."
    REFERENCE
        "<clause ref>, Oper Control Mode"
    ::= { dot1xAuthConfigEntry 4 }

dot1xAuthAuthControlledPortStatus OBJECT-TYPE
    SYNTAX      PaeControlledPortStatus
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The current value of the controlled Port
```

```
        status parameter for the Port."
    REFERENCE
        "<clause ref>, AuthControlledPortStatus"
    ::= { dot1xAuthConfigEntry 5 }

dot1xAuthAuthControlledPortControl OBJECT-TYPE
    SYNTAX      PaeControlledPortControl
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The current value of the controlled Port
        control parameter for the Port."
    REFERENCE
        "<clause ref>, AuthControlledPortControl"
    ::= { dot1xAuthConfigEntry 6 }

dot1xAuthQuietPeriod OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The value, in seconds, of the quietPeriod constant
        currently in use by the Authenticator PAE state
        machine."
    REFERENCE
        "<clause ref>, quietPeriod"
    DEFVAL { 60 }
    ::= { dot1xAuthConfigEntry 7 }

dot1xAuthSuppTimeout OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-write
    STATUS      deprecated
    DESCRIPTION
        "Use of this valueis deprecated."
    REFERENCE
        "No reference"
    DEFVAL { 30 }
    ::= { dot1xAuthConfigEntry 9 }

dot1xAuthServerTimeout OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The value, in seconds, of the serverTimeout constant
```

```
        currently in use by the Backend Authentication state
        machine."
    REFERENCE
        "<clause ref>, serverTimeout"
    DEFVAL { 30 }
    ::= { dot1xAuthConfigEntry 10 }

dot1xAuthMaxReq OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-write
    STATUS      deprecated
    DESCRIPTION
        "This value is deprecated and is ignored."
    REFERENCE
        "There is no reference, the use of this value is deprecated."
    DEFVAL { 2 }
    ::= { dot1xAuthConfigEntry 11 }

dot1xAuthReAuthPeriod OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The value, in seconds, of the reAuthPeriod constant
        currently in use by the Reauthentication Timer state
        machine."
    REFERENCE
        "<clause ref>, reAuthPeriod"
    DEFVAL { 3600 }
    ::= { dot1xAuthConfigEntry 12 }

dot1xAuthReAuthEnabled OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The enable/disable control used by the Reauthentication
        Timer state machine (<clause ref>)."
    REFERENCE
        "<clause ref>, reAuthEnabled"
    DEFVAL { false }
    ::= { dot1xAuthConfigEntry 13 }

dot1xAuthKeyTxEnabled OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
```

```
    STATUS      current
    DESCRIPTION
        "The value of the keyTransmissionEnabled constant
        currently in use by the Authenticator PAE state
        machine."
    REFERENCE
        "<clause ref>, keyTransmissionEnabled"
    ::= { dot1xAuthConfigEntry 14 }

<<Editor's Note: I think the following object is no longer needed.>>

dot1xAuthInitEapCode OBJECT-TYPE
    SYNTAX      INTEGER {
                    request(1),
                    response(2),
                    success(3),
                    failure(4)
                }
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The value of the EAP Code field to be used by the
        txInitialMsg() function of the Authenticator PAE state
        machine."
    REFERENCE
        "<clause ref>, initialEAPMsg; RFC 2284, section 2.2"
    DEFVAL { request }
    ::= { dot1xAuthConfigEntry 15 }

<<Editor's Note: I think the following object is no longer needed.>>

dot1xAuthInitEapData OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE(0..1477))
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The value used to derive the EAP Data field by the
        txInitialMsg() function of the Authenticator PAE state
        machine.

        The other fields of the initial EAP message generated
        by txInitialMsg() are:

            EAP Code is taken from dot1xAuthInitEapCode

            EAP Identifier is the currentId passed into
```

```
            txInitialMsg().

            EAP Length is derived from the dot1xAuthInitEapData
            value.

        The derived EAP Data field itself depends upon the
        first byte of this object, which corresponds to the EAP
        Type value.  In the simplest case, such as for an EAP
        Identity (EAP Type 1) this may be a direct copy of the
        value of this object and the EAP Length is derived from
        the length of this object, plus 4 octes for the EAP
        Code, Identifier and Length values.  In more complex
        cases, such as an EAP MD5-Challenge (EAP Type 4), this
        may require one-time generated values to be inserted at
        the appropriate place in the data by the txInitialMsg()
        function and the EAP Length must be calculated based on
        the generated data.

        The size of this object is defined to support the
        maximum size that will fit in an EAPOL message.  This
        exceeds the size that may be carried in an SNMP message
        without requiring IP fragmentation on an Ethernet
        network.  It is expected that actual values used in
        this object will be much smaller than the maximum
        allowed.

        The default value for this object is an octet string
        containing a single octet with a value of 01 (hex).
        This corresponds to the EAP Type 'Identity', without
        the optional Type-Data field."
    REFERENCE
        "<clause ref>, initialEAPMsg; RFC 2284, section 3"
    DEFVAL { '01'H }                    -- EAP Type 'Identity'
    ::= { dot1xAuthConfigEntry 16 }

-- ---------------------------------------------------------- --
-- The Authenticator Statistics Table
-- ---------------------------------------------------------- --

dot1xAuthStatsTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF Dot1xAuthStatsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table that contains the statistics objects for the
         Authenticator PAE associated with each Port.
```

```
         An entry appears in this table for each port that may
         authenticate access to itself."
    REFERENCE
        "9.4.1 Authenticator Statistics"
    ::= { dot1xPaeAuthenticator 2 }

dot1xAuthStatsEntry OBJECT-TYPE
    SYNTAX      Dot1xAuthStatsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The statistics information for an Authenticator PAE."
    INDEX { dot1xPaePortNumber }
    ::= { dot1xAuthStatsTable 1 }

Dot1xAuthStatsEntry ::=
    SEQUENCE {
        dot1xAuthEapolFramesRx
            Counter32,
        dot1xAuthEapolFramesTx
            Counter32,
        dot1xAuthEapolStartFramesRx
            Counter32,
        dot1xAuthEapolLogoffFramesRx
            Counter32,
        dot1xAuthEapolRespFramesRx
            Counter32,
        dot1xAuthEapolReqIdFramesTx
            Counter32,
        dot1xAuthEapolReqFramesTx
            Counter32,
        dot1xAuthInvalidEapolFramesRx
            Counter32,
        dot1xAuthEapLengthErrorFramesRx
            Counter32,
        dot1xAuthLastEapolFrameVersion
            Unsigned32,
        dot1xAuthLastEapolFrameSource
            MacAddress
        }

dot1xAuthEapolFramesRx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
```

```
        "The number of valid EAPOL frames of any type
        that have been received by this Authenticator."
    REFERENCE
        "<clause ref>, EAPOL frames received"
    ::= { dot1xAuthStatsEntry 1 }

dot1xAuthEapolFramesTx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of EAPOL frames of any type
        that have been transmitted by this Authenticator."
    REFERENCE
        "<clause ref>, EAPOL frames transmitted"
    ::= { dot1xAuthStatsEntry 2 }

dot1xAuthEapolStartFramesRx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of EAPOL Start frames that have
        been received by this Authenticator."
    REFERENCE
        "<clause ref>, EAPOL Start frames received"
    ::= { dot1xAuthStatsEntry 3 }

dot1xAuthEapolLogoffFramesRx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of EAPOL Logoff frames that have
        been received by this Authenticator."
    REFERENCE
        "<clause ref>, EAPOL Logoff frames received"
    ::= { dot1xAuthStatsEntry 4 }

dot1xAuthEapolRespFramesRx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of valid EAPOL frames
        of type EAP-Packet that have been
```

```
        received by this Authenticator."
    REFERENCE
        "<clause ref>, EAPOL Response frames received"
    ::= { dot1xAuthStatsEntry 6 }

dot1xAuthEapolReqIdFramesTx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of initial EAPOL frames
        of type EAP-Packet that have been
        transmitted by this Authenticator
        prior to the first response from the
        Supplicant. This counts all txReqs carried
        out from the REQUEST state after transitioning
        from the IDLE state but before transitioning
        to the RESPONSE state."
    REFERENCE
        "<clause ref>, EAPOL Initial Request frames transmitted"
    ::= { dot1xAuthStatsEntry 7 }

dot1xAuthEapolReqFramesTx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of EAPOL frames of type EAP-Packet
        (other than initial frames) that have been
        transmitted by this Authenticator. This counts
        all txReqs carried out from the REQUEST state
        after transitioning in from any state other
        than the IDLE state."
    REFERENCE
        "<clause ref>, EAPOL Request frames transmitted"
    ::= { dot1xAuthStatsEntry 8 }

dot1xAuthInvalidEapolFramesRx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of EAPOL frames that have been
        received by this Authenticator in which the
        frame type is not recognized."
    REFERENCE
```

```
        "<clause ref>, Invalid EAPOL frames received"
    ::= { dot1xAuthStatsEntry 9 }

dot1xAuthEapLengthErrorFramesRx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of EAPOL frames that have been received
        by this Authenticator in which the Packet Body
        Length field is invalid."
    REFERENCE
        "<clause ref>, EAPOL length error frames received"
    ::= { dot1xAuthStatsEntry 10 }

dot1xAuthLastEapolFrameVersion OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The protocol version number carried in the
        most recently received EAPOL frame."
    REFERENCE
        "<clause ref>, Last EAPOL frame version"
    ::= { dot1xAuthStatsEntry 11 }

dot1xAuthLastEapolFrameSource OBJECT-TYPE
    SYNTAX      MacAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The source MAC address carried in the
        most recently received EAPOL frame."
    REFERENCE
        "<clause ref>, Last EAPOL frame source"
    ::= { dot1xAuthStatsEntry 12 }

-- ---------------------------------------------------------- --
-- The Authenticator Diagnostics Table
-- ---------------------------------------------------------- --

dot1xAuthDiagTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF Dot1xAuthDiagEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
```

```
        "A table that contains the diagnostics objects for the
         Authenticator PAE associated with each Port.
         An entry appears in this table for each port that may
         authenticate access to itself."
    REFERENCE
        "<clause ref> Authenticator Diagnostics"
    ::= { dot1xPaeAuthenticator 3 }

dot1xAuthDiagEntry OBJECT-TYPE
    SYNTAX      Dot1xAuthDiagEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The diagnostics information for an Authenticator PAE."
    INDEX { dot1xPaePortNumber }
    ::= { dot1xAuthDiagTable 1 }

Dot1xAuthDiagEntry ::=
    SEQUENCE {
        dot1xAuthEntersConnecting
            Counter32,
        dot1xAuthEapLogoffsWhileConnecting
            Counter32,
        dot1xAuthEntersAuthenticating
            Counter32,
        dot1xAuthAuthSuccessWhileAuthenticating
            Counter32,
        dot1xAuthAuthTimeoutsWhileAuthenticating
            Counter32,
        dot1xAuthAuthFailWhileAuthenticating
            Counter32,
        dot1xAuthAuthReauthsWhileAuthenticating
            Counter32,
        dot1xAuthAuthEapStartsWhileAuthenticating
            Counter32,
        dot1xAuthAuthEapLogoffWhileAuthenticating
            Counter32,
        dot1xAuthAuthReauthsWhileAuthenticated (deprecated)
            Counter32,
        dot1xAuthAuthEapStartsWhileAuthenticated
            Counter32,
        dot1xAuthAuthEapLogoffWhileAuthenticated
            Counter32,
        dot1xAuthBackendResponses
            Counter32,
        dot1xAuthBackendAccessChallenges
```

```
            Counter32,
        dot1xAuthBackendOtherRequestsToSupplicant
            Counter32,
        dot1xAuthBackendAuthSuccesses
            Counter32,
        dot1xAuthBackendAuthFails
            Counter32
        }

dot1xAuthEntersConnecting OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Counts the number of times that the state machine
        transitions to the CONNECTING state from any other
        state."
    REFERENCE
        "<clause ref>, <clause ref>"
    ::= { dot1xAuthDiagEntry 1 }

dot1xAuthEapLogoffsWhileConnecting OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Counts the number of times that the state machine
        transitions from CONNECTING to DISCONNECTED as a result
        of receiving an EAPOL-Logoff message."
    REFERENCE
        "<clause ref>, <clause ref>"
    ::= { dot1xAuthDiagEntry 2 }

dot1xAuthEntersAuthenticating OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Counts the number of times that the state machine
        transitions from CONNECTING to AUTHENTICATING, as a
        result of an EAP-Response/Identity message being
        received from the Supplicant."
    REFERENCE
        "<clause ref>, <clause ref>"
    ::= { dot1xAuthDiagEntry 3 }
```

```
dot1xAuthAuthSuccessWhileAuthenticating OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Counts the number of times that the state machine
        transitions from AUTHENTICATING to AUTHENTICATED, as a
        result of the Backend Authentication state machine
        indicating successful authentication of the Supplicant
        (authSuccess = TRUE)."
    REFERENCE
        "<clause ref>, <clause ref>"
    ::= { dot1xAuthDiagEntry 4 }

dot1xAuthAuthTimeoutsWhileAuthenticating OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Counts the number of times that the state machine
        transitions from AUTHENTICATING to ABORTING, as a result
        of the Backend Authentication state machine indicating
        authentication timeout (authTimeout = TRUE)."
    REFERENCE
        "<clause ref>, <clause ref>"
    ::= { dot1xAuthDiagEntry 5 }

dot1xAuthAuthFailWhileAuthenticating OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Counts the number of times that the state machine
        transitions from AUTHENTICATING to HELD, as a result
        of the Backend Authentication state machine indicating
        authentication failure (authFail = TRUE)."
    REFERENCE
        "<clause ref>, <clause ref>"
    ::= { dot1xAuthDiagEntry 6 }

dot1xAuthAuthReauthsWhileAuthenticating OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      deprecated
    DESCRIPTION
        "Use of this value is deprecated."
```

```
    REFERENCE
        "None"
    ::= { dot1xAuthDiagEntry 7 }

dot1xAuthAuthEapStartsWhileAuthenticating OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Counts the number of times that the state machine
        transitions from AUTHENTICATING to ABORTING, as a result
        of an EAPOL-Start message being received
        from the Supplicant."
    REFERENCE
        "<clause ref>, <clause ref>"
    ::= { dot1xAuthDiagEntry 8 }

dot1xAuthAuthEapLogoffWhileAuthenticating OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Counts the number of times that the state machine
        transitions from AUTHENTICATING to ABORTING, as a result
        of an EAPOL-Logoff message being received
        from the Supplicant."
    REFERENCE
        "<clause ref>, <clause ref>"
    ::= { dot1xAuthDiagEntry 9 }

dot1xAuthAuthReauthsWhileAuthenticated OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Counts the number of times that the state machine
        transitions from AUTHENTICATED to CONNECTING, as a
        result of a reauthentication request
        (reAuthenticate = TRUE)."
    REFERENCE
        "<clause ref>, <clause ref>"
    ::= { dot1xAuthDiagEntry 10 }

dot1xAuthAuthEapStartsWhileAuthenticated OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
```

```
    STATUS      current
    DESCRIPTION
        "Counts the number of times that the state machine
        transitions from AUTHENTICATED to CONNECTING, as a
        result of an EAPOL-Start message being received from the
        Supplicant."
    REFERENCE
        "<clause ref>, <clause ref>"
    ::= { dot1xAuthDiagEntry 11 }

dot1xAuthAuthEapLogoffWhileAuthenticated OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Counts the number of times that the state machine
        transitions from AUTHENTICATED to DISCONNECTED, as a
        result of an EAPOL-Logoff message being received from
        the Supplicant."
    REFERENCE
        "<clause ref>, <clause ref>"
    ::= { dot1xAuthDiagEntry 12 }

dot1xAuthBackendResponses OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Counts the number of times that the state machine sends
        a supplicant's first response packet to the EAP layer
        (i.e., executes sendRespToServer on entry to the
        RESPONSE state). Indicates that the Authenticator
        attempted communication with the Authentication Server
        via the EAP layer."
    REFERENCE
        "<clause ref>"
    ::= { dot1xAuthDiagEntry 13 }

dot1xAuthBackendAccessChallenges OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Counts the number of times that the state machine
        receives the first request from the EAP layer following
        the first response from the supplicant (i.e., eapReq
```

```
        becomes TRUE, causing exit from the RESPONSE state).
        Indicates that the Authentication Server has
        communication with the Authenticator via the EAP layer."
    REFERENCE
        "<clause ref>"
    ::= { dot1xAuthDiagEntry 14 }

dot1xAuthBackendOtherRequestsToSupplicant OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Counts the number of times that the state machine
        sends an EAP-Request packet following the first
        to the Supplicant (i.e., executes txReq on entry to the
        REQUEST state). Indicates that the Authentication
        Server chose an EAP-method."
    REFERENCE
        "<clause ref>"
    ::= { dot1xAuthDiagEntry 15 }

dot1xAuthBackendAuthSuccesses OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Counts the number of times that the state machine
        receives a success indication from the EAP layer
        (i.e., aSuccess becomes TRUE, causing a
        transition from RESPONSE to SUCCESS). Indicates that
        the Supplicant has successfully authenticated to
        the Authentication Server."
    REFERENCE
        "<clause ref>"
    ::= { dot1xAuthDiagEntry 17 }

dot1xAuthBackendAuthFails OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Counts the number of times that the state machine
        receives a failure message from the EAP layer
        (i.e., aFail becomes TRUE, causing a transition
        from RESPONSE to FAIL). Indicates that the Supplicant
        has not authenticated to the Authentication Server."
```

```
    REFERENCE
        "<clause ref>"
    ::= { dot1xAuthDiagEntry 18 }

-- ---------------------------------------------------------- --
-- The Authenticator Session Statistics Table
-- ---------------------------------------------------------- --

dot1xAuthSessionStatsTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF Dot1xAuthSessionStatsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table that contains the session statistics objects
        for the Authenticator PAE associated with each Port.
        An entry appears in this table for each port that may
        authenticate access to itself."
    REFERENCE
        "<clause ref>"
    ::= { dot1xPaeAuthenticator 4 }

dot1xAuthSessionStatsEntry OBJECT-TYPE
    SYNTAX      Dot1xAuthSessionStatsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The session statistics information for an Authenticator
        PAE.  This shows the current values being collected for
        each session that is still in progress, or the final
        values for the last valid session on each port where
        there is no session currently active."
    INDEX { dot1xPaePortNumber }
    ::= { dot1xAuthSessionStatsTable 1 }

Dot1xAuthSessionStatsEntry ::=
    SEQUENCE {
        dot1xAuthSessionOctetsRx
            Counter64,
        dot1xAuthSessionOctetsTx
            Counter64,
        dot1xAuthSessionFramesRx
            Counter32,
        dot1xAuthSessionFramesTx
            Counter32,
        dot1xAuthSessionId
            SnmpAdminString,
```

```
        dot1xAuthSessionAuthenticMethod
            INTEGER,
        dot1xAuthSessionTime
            TimeTicks,
        dot1xAuthSessionTerminateCause
            INTEGER,
        dot1xAuthSessionUserName
            SnmpAdminString
    }

dot1xAuthSessionOctetsRx OBJECT-TYPE
    SYNTAX      Counter64
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of octets received in user data
        frames on this Port during the session."
    REFERENCE
        "<clause ref>, Session Octets Received"
    ::= { dot1xAuthSessionStatsEntry 1 }

dot1xAuthSessionOctetsTx OBJECT-TYPE
    SYNTAX      Counter64
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of octets transmitted in user data
        frames on this Port during the session."
    REFERENCE
        "<clause ref>, Session Octets Transmitted"
    ::= { dot1xAuthSessionStatsEntry 2 }

dot1xAuthSessionFramesRx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of user data frames received
        on this Port during the session."
    REFERENCE
        "<clause ref>, Session Frames Received"
    ::= { dot1xAuthSessionStatsEntry 3 }

dot1xAuthSessionFramesTx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
```

```
    STATUS      current
    DESCRIPTION
        "The number of user data frames transmitted
        on this Port during the session."
    REFERENCE
        "<clause ref>, Session Frames Transmitted"
    ::= { dot1xAuthSessionStatsEntry 4 }

dot1xAuthSessionId OBJECT-TYPE
    SYNTAX      SnmpAdminString
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "A unique identifier for the session, in the
        form of a printable ASCII string of at least
        three characters."
    REFERENCE
        "<clause ref>, Session Identifier"
    ::= { dot1xAuthSessionStatsEntry 5 }

dot1xAuthSessionAuthenticMethod OBJECT-TYPE
    SYNTAX      INTEGER {
                    remoteAuthServer(1),
                    localAuthServer(2)
                }
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The authentication method used to establish the
        session."
    REFERENCE
        "<clause ref>, Session Authentication Method"
    ::= { dot1xAuthSessionStatsEntry 6 }

dot1xAuthSessionTime OBJECT-TYPE
    SYNTAX      TimeTicks
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The duration of the session in seconds."
    REFERENCE
        "<clause ref>, Session Time"
    ::= { dot1xAuthSessionStatsEntry 7 }

dot1xAuthSessionTerminateCause OBJECT-TYPE
    SYNTAX      INTEGER {
```

```
                    supplicantLogoff(1),
                    portFailure(2),
                    supplicantRestart(3),
                    reauthFailed(4),
                    authControlForceUnauth(5),
                    portReInit(6),
                    portAdminDisabled(7),
                    notTerminatedYet(999)
                    }
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The reason for the session termination."
    REFERENCE
        "<clause ref>, Session Terminate Cause"
    ::= { dot1xAuthSessionStatsEntry 8 }

dot1xAuthSessionUserName OBJECT-TYPE
    SYNTAX     SnmpAdminString
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The User-Name representing the identity of the
        Supplicant PAE."
    REFERENCE
        "<clause ref>, Session User Name"
    ::= { dot1xAuthSessionStatsEntry 9 }

-- ------------------------------------------------------- --
-- The PAE Supplicant Group
-- ------------------------------------------------------- --

-- ------------------------------------------------------- --
-- The Supplicant Configuration Table
-- ------------------------------------------------------- --

dot1xSuppConfigTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF Dot1xSuppConfigEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table that contains the configuration objects for the
        Supplicant PAE associated with each port.
        An entry appears in this table for each port that may
        authenticate itself when challenged by a remote system."
    REFERENCE
```

```
        "<clause ref>"
    ::= { dot1xPaeSupplicant 1 }

dot1xSuppConfigEntry OBJECT-TYPE
    SYNTAX      Dot1xSuppConfigEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The configuration information for a Supplicant PAE."
    INDEX { dot1xPaePortNumber }
    ::= { dot1xSuppConfigTable 1 }

Dot1xSuppConfigEntry ::=
    SEQUENCE {
        dot1xSuppPaeState
            INTEGER,
        dot1xSuppHeldPeriod
            Unsigned32,
        dot1xSuppAuthPeriod
            Unsigned32,
        dot1xSuppStartPeriod
            Unsigned32,
        dot1xSuppMaxStart
            Unsigned32,
        dot1xSuppBackendPaeState
            Unsigned32
        dot1xSuppSuppControlledPortStatus
            Unsigned32
        }

dot1xSuppPaeState OBJECT-TYPE
    SYNTAX      INTEGER {
                    disconnected(1),
                    logoff(2),
                    connecting(3),
                    authenticating(4),
                    authenticated(5),
                    unused(6),
                    held(7),
                    restart(8)
                }
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The current state of the Supplicant PAE state
        machine (8.5.8)."
```

```
    REFERENCE
        "<clause ref>, Supplicant PAE State"
    ::= { dot1xSuppConfigEntry 1 }

dot1xSuppHeldPeriod OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The value, in seconds, of the heldPeriod
        constant currently in use by the Supplicant
        PAE state machine (8.5.8.1.2)."
    REFERENCE
        "<clause ref>, heldPeriod"
    DEFVAL { 60 }
    ::= { dot1xSuppConfigEntry 2 }

dot1xSuppAuthPeriod OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The value, in seconds, of the serverTimeout
        constant currently in use by the Supplicant
        PAE state machine (8.5.8.1.2)."
    REFERENCE
        "<clause ref>, authPeriod"
    DEFVAL { 30 }
    ::= { dot1xSuppConfigEntry 3 }

dot1xSuppStartPeriod OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The value, in seconds, of the suppTimeout
        constant currently in use by the Supplicant
        PAE state machine (8.5.8.1.2)."
    REFERENCE
        "<clause ref>, startPeriod"
    DEFVAL { 30 }
    ::= { dot1xSuppConfigEntry 4 }

dot1xSuppMaxStart OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-write
```

```
    STATUS      current
    DESCRIPTION
        "The value of the maxStart constant currently in use by
        the Supplicant PAE state machine (8.5.8.1.2)."
    REFERENCE
        "<clause ref>, maxStart"
    DEFVAL { 3}
    ::= { dot1xSuppConfigEntry 5 }

dot1xSuppBackendPaeState OBJECT-TYPE
    SYNTAX INTEGER {
    initialize(1),
    idle(2),
    request(3),
    receive(4),
    response(5),
    fail(6),
    timeout(7),
    success(8)
    }
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "The current state of the Backend Supplicant
         PAE statemachine (8.5.12)."
    REFERENCE
        "<clause ref>, Backend Supplicant PAE State"
    ::= { dot1xSuppConfigEntry 6 }

dot1xSuppSuppControlledPortStatus OBJECT-TYPE
    SYNTAX      PaeControlledPortStatus
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The current value of the controlled Port
        status parameter for the Port."
    REFERENCE
        "<clause ref>, SuppControlledPortStatus"
    ::= { dot1xSuppConfigEntry 7 }

-- ---------------------------------------------------------- --
-- The Supplicant Statistics Table
-- ---------------------------------------------------------- --

dot1xSuppStatsTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF Dot1xSuppStatsEntry
```

```
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table that contains the statistics objects for the
        Supplicant PAE associated with each port.
        An entry appears in this table for each port that may
        authenticate itself when challenged by a remote system."
    REFERENCE
        "<clause ref>"
    ::= { dot1xPaeSupplicant 2 }

dot1xSuppStatsEntry OBJECT-TYPE
    SYNTAX      Dot1xSuppStatsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The statistics information for a Supplicant PAE."
    INDEX { dot1xPaePortNumber }
    ::= { dot1xSuppStatsTable 1 }

Dot1xSuppStatsEntry ::=
    SEQUENCE {
        dot1xSuppEapolFramesRx
            Counter32,
        dot1xSuppEapolFramesTx
            Counter32,
        dot1xSuppEapolStartFramesTx
            Counter32,
        dot1xSuppEapolLogoffFramesTx
            Counter32,
        dot1xSuppEapolRespIdFramesTx
            Counter32,
        dot1xSuppEapolRespFramesTx
            Counter32,
        dot1xSuppEapolReqIdFramesRx
            Counter32,
        dot1xSuppEapolReqFramesRx
            Counter32,
        dot1xSuppInvalidEapolFramesRx
            Counter32,
        dot1xSuppEapLengthErrorFramesRx
            Counter32,
        dot1xSuppLastEapolFrameVersion
            Unsigned32,
        dot1xSuppLastEapolFrameSource
            MacAddress
```

```
        }

dot1xSuppEapolFramesRx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of EAPOL frames of any type
        that have been received by this Supplicant."
    REFERENCE
        "<clause ref>, EAPOL frames received"
    ::= { dot1xSuppStatsEntry 1 }

dot1xSuppEapolFramesTx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of EAPOL frames of any type
        that have been transmitted by this Supplicant."
    REFERENCE
        "<clause ref>, EAPOL frames transmitted"
    ::= { dot1xSuppStatsEntry 2 }

dot1xSuppEapolStartFramesTx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of EAPOL Start frames
        that have been transmitted by this Supplicant."
    REFERENCE
        "<clause ref>, EAPOL Start frames transmitted"
    ::= { dot1xSuppStatsEntry 3 }

dot1xSuppEapolLogoffFramesTx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of EAPOL Logoff frames
        that have been transmitted by this Supplicant."
    REFERENCE
        "<clause ref>, EAPOL Logoff frames transmitted"
    ::= { dot1xSuppStatsEntry 4 }
```

```
dot1xSuppEapolRespFramesTx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of valid EAPOL frames of type EAP-Packet
        that have been transmitted by this Supplicant."
    REFERENCE
        "<clause ref>, EAP Resp frames transmitted"
    ::= { dot1xSuppStatsEntry 6 }

dot1xSuppEapolReqIdFramesRx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of initial EAPOL frames of type EAP-Packet
        that have been received by this Supplicant. This
        counts all txSuppRsp() carried out from the
        RESPONSE state after transitioning from the IDLE
        state but before transitioning to the RECEIVE state."
    REFERENCE
        "<clause ref>, EAP Initial Request frames received"
    ::= { dot1xSuppStatsEntry 7 }

dot1xSuppEapolReqFramesRx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of EAPOL frames of type EAP-Packet
        (other than initial frames) that have been
        received by this Supplicant. This counts all
        txSuppRsp() carried out from the RESPONSE state
        after having passed once through the RECEIVE
        state and having not passed through the IDLE state."
    REFERENCE
        "<clause ref>, EAP Req frames received"
    ::= { dot1xSuppStatsEntry 8 }

dot1xSuppInvalidEapolFramesRx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of EAPOL frames that have been
```

```
        received by this Supplicant in which the
        frame type is not recognized."
    REFERENCE
        "<clause ref>, Invalid EAPOL frames received"
    ::= { dot1xSuppStatsEntry 9 }

dot1xSuppEapLengthErrorFramesRx OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of EAPOL frames that have been
        received by this Supplicant in which the Packet
        Body Length field (7.5.5) is invalid."
    REFERENCE
        "<clause ref>, EAPOL length error frames received"
    ::= { dot1xSuppStatsEntry 10 }

dot1xSuppLastEapolFrameVersion OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The protocol version number carried in the
        most recently received EAPOL frame."
    REFERENCE
        "<clause ref>, Last EAPOL frame version"
    ::= { dot1xSuppStatsEntry 11 }

dot1xSuppLastEapolFrameSource OBJECT-TYPE
    SYNTAX      MacAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The source MAC address carried in the
        most recently received EAPOL frame."
    REFERENCE
        "<clause ref>, Last EAPOL frame source"
    ::= { dot1xSuppStatsEntry 12 }

-- ---------------------------------------------------------- --
-- IEEE 802.1X MIB - Conformance Information
-- ---------------------------------------------------------- --

dot1xPaeConformance OBJECT IDENTIFIER ::= { ieee8021paeMIB 2 }
```

```
dot1xPaeGroups OBJECT IDENTIFIER ::= { dot1xPaeConformance 1 }

dot1xPaeCompliances OBJECT IDENTIFIER
    ::= { dot1xPaeConformance 2 }

-- ------------------------------------------------------- --
-- units of conformance
-- ------------------------------------------------------- --

dot1xPaeSystemGroup OBJECT-GROUP
    OBJECTS {
        dot1xPaeSystemAuthControl,
        dot1xPaePortProtocolVersion,
        dot1xPaePortCapabilities,
        dot1xPaePortInitialize
    }
    STATUS      deprecated
    DESCRIPTION
        "A collection of objects providing system information
        about, and control over, a PAE."
    ::= { dot1xPaeGroups 1 }

dot1xPaeAuthConfigGroup OBJECT-GROUP
    OBJECTS {
        dot1xAuthPaeState,
        dot1xAuthBackendAuthState,
        dot1xAuthAdminControlledDirections,
        dot1xAuthOperControlledDirections,
        dot1xAuthAuthControlledPortStatus,
        dot1xAuthAuthControlledPortControl,
        dot1xAuthQuietPeriod,
        dot1xAuthSuppTimeout,
        dot1xAuthServerTimeout,
        dot1xAuthMaxReq (deprecated),
        dot1xAuthReAuthPeriod,
        dot1xAuthReAuthEnabled,
        dot1xAuthKeyTxEnabled
    }
    STATUS      current
    DESCRIPTION
        "A collection of objects providing configuration
        information about an Authenticator PAE."
    ::= { dot1xPaeGroups 2 }

dot1xPaeAuthStatsGroup OBJECT-GROUP
    OBJECTS {
```

```
        dot1xAuthEapolFramesRx,
        dot1xAuthEapolFramesTx,
        dot1xAuthEapolStartFramesRx,
        dot1xAuthEapolLogoffFramesRx,
        dot1xAuthEapolRespFramesRx,
        dot1xAuthEapolReqIdFramesTx,
        dot1xAuthEapolReqFramesTx,
        dot1xAuthInvalidEapolFramesRx,
        dot1xAuthEapLengthErrorFramesRx,
        dot1xAuthLastEapolFrameVersion,
        dot1xAuthLastEapolFrameSource
    }
    STATUS      current
    DESCRIPTION
        "A collection of objects providing statistics about an
        Authenticator PAE."
    ::= { dot1xPaeGroups 3 }

dot1xPaeAuthDiagGroup OBJECT-GROUP
    OBJECTS {
        dot1xAuthEntersConnecting,
        dot1xAuthEapLogoffsWhileConnecting,
        dot1xAuthEntersAuthenticating,
        dot1xAuthAuthSuccessWhileAuthenticating,
        dot1xAuthAuthTimeoutsWhileAuthenticating,
        dot1xAuthAuthFailWhileAuthenticating,
        dot1xAuthAuthReauthsWhileAuthenticating,
        dot1xAuthAuthEapStartsWhileAuthenticating,
        dot1xAuthAuthEapLogoffWhileAuthenticating,
        dot1xAuthAuthReauthsWhileAuthenticated,
        dot1xAuthAuthEapStartsWhileAuthenticated,
        dot1xAuthAuthEapLogoffWhileAuthenticated,
        dot1xAuthBackendResponses,
        dot1xAuthBackendAccessChallenges,
        dot1xAuthBackendOtherRequestsToSupplicant,
        dot1xAuthBackendAuthSuccesses,
        dot1xAuthBackendAuthFails
    }
    STATUS      current
    DESCRIPTION
        "A collection of objects providing diagnostic statistics
        about an Authenticator PAE."
    ::= { dot1xPaeGroups 4 }

dot1xPaeAuthSessionStatsGroup OBJECT-GROUP
    OBJECTS {
```

```
        dot1xAuthSessionOctetsRx,
        dot1xAuthSessionOctetsTx,
        dot1xAuthSessionFramesRx,
        dot1xAuthSessionFramesTx,
        dot1xAuthSessionId,
        dot1xAuthSessionAuthenticMethod,
        dot1xAuthSessionTime,
        dot1xAuthSessionTerminateCause
    }
    STATUS      deprecated
    DESCRIPTION
        "A collection of objects providing statistics about the
        current, or last session for an Authenticator PAE."
    ::= { dot1xPaeGroups 5 }

dot1xPaeSuppConfigGroup OBJECT-GROUP
    OBJECTS {
        dot1xSuppPaeState,
        dot1xSuppHeldPeriod,
        dot1xSuppAuthPeriod,
        dot1xSuppStartPeriod,
        dot1xSuppMaxStart,
        dot1xSuppSuppControlledPortStatus,
        dot1xSuppBackendPaeState

    }
    STATUS      current
    DESCRIPTION
        "A collection of objects providing configuration
        information about a Supplicant PAE."
    ::= { dot1xPaeGroups 6 }

dot1xPaeSuppStatsGroup OBJECT-GROUP
    OBJECTS {
        dot1xSuppEapolFramesRx,
        dot1xSuppEapolFramesTx,
        dot1xSuppEapolStartFramesTx,
        dot1xSuppEapolLogoffFramesTx,
        dot1xSuppEapolRespFramesTx,
        dot1xSuppEapolReqIdFramesRx,
        dot1xSuppEapolReqFramesRx,
        dot1xSuppInvalidEapolFramesRx,
        dot1xSuppEapLengthErrorFramesRx,
        dot1xSuppLastEapolFrameVersion,
        dot1xSuppLastEapolFrameSource
    }
```

```
    STATUS      current
    DESCRIPTION
        "A collection of objects providing statistics about a
        Supplicant PAE."
    ::= { dot1xPaeGroups 7 }

dot1xPaeAuthSystemGroup OBJECT-GROUP
    OBJECTS {
        dot1xPaeSystemAuthControl,
        dot1xPaePortReauthenticate
    }
    STATUS      current
    DESCRIPTION
        "A collection of objects providing system information
        about, and control over, an Authenticator PAE."
    ::= { dot1xPaeGroups 8 }

dot1xPaeSystemPortGroup OBJECT-GROUP
    OBJECTS {
        dot1xPaePortProtocolVersion,
        dot1xPaePortCapabilities,
        dot1xPaePortInitialize
    }
    STATUS      current
    DESCRIPTION
        "A collection of objects providing system information
        about, and control over, a PAE."
    ::= { dot1xPaeGroups 9 }

dot1xPaeAuthSessionStats2Group OBJECT-GROUP
    OBJECTS {
        dot1xAuthSessionOctetsRx,
        dot1xAuthSessionOctetsTx,
        dot1xAuthSessionFramesRx,
        dot1xAuthSessionFramesTx,
        dot1xAuthSessionId,
        dot1xAuthSessionAuthenticMethod,
        dot1xAuthSessionTime,
        dot1xAuthSessionTerminateCause,
        dot1xAuthSessionUserName
    }
    STATUS      current
    DESCRIPTION
        "A collection of objects providing statistics about the
        current, or last session for an Authenticator PAE."
    ::= { dot1xPaeGroups 10 }
```

```
dot1xPaeAuthInitEapGroup OBJECT-GROUP
    OBJECTS {
        dot1xAuthInitEapCode,
        dot1xAuthInitEapData
    }
    STATUS      current
    DESCRIPTION
        "A collection of objects providing configuration
        information about the initial EAP message generated by
        an Authenticator PAE."
    ::= { dot1xPaeGroups 11 }

-- ---------------------------------------------------------- --
-- compliance statements
-- ---------------------------------------------------------- --

dot1xPaeCompliance MODULE-COMPLIANCE
    STATUS  deprecated
    DESCRIPTION
        "The compliance statement for device support of
        Port Access Control."

    MODULE
        MANDATORY-GROUPS {
            dot1xPaeSystemGroup
        }

        GROUP   dot1xPaeAuthConfigGroup
        DESCRIPTION
            "This group is mandatory for systems that support
            the Authenticator functions of the PAE."

        OBJECT  dot1xAuthAdminControlledDirections
        SYNTAX  INTEGER {
                    both(0)
                }
        MIN-ACCESS read-only
        DESCRIPTION
            "Support for in(1) is optional."

        OBJECT  dot1xAuthOperControlledDirections
        SYNTAX  INTEGER {
                    both(0)
                }
        DESCRIPTION
```

```
            "Support for in(1) is optional."

        OBJECT dot1xAuthKeyTxEnabled
        MIN-ACCESS read-only
        DESCRIPTION
          "An Authenticator PAE that does not support
          EAPOL-Key frames may implement this object as
          read-only, returning a value of FALSE."

        GROUP   dot1xPaeAuthStatsGroup
        DESCRIPTION
            "This group is mandatory for systems that support
            the Authenticator functions of the PAE."

        GROUP   dot1xPaeAuthDiagGroup
        DESCRIPTION
            "This group is optional for systems that support
            the Authenticator functions of the PAE."

        GROUP   dot1xPaeAuthSessionStatsGroup
        DESCRIPTION
            "This group is optional for systems that support
            the Authenticator functions of the PAE."

        GROUP   dot1xPaeSuppConfigGroup
        DESCRIPTION
            "This group is mandatory for systems that support
            the Supplicant functions of the PAE."

        GROUP   dot1xPaeSuppStatsGroup
        DESCRIPTION
            "This group is mandatory for systems that support
            the Supplicant functions of the PAE."

    ::= { dot1xPaeCompliances 1 }

-- ---------------------------------------------------------- --
-- compliance statements for 802.1aa
-- ---------------------------------------------------------- --

dot1xPaeCompliance2 MODULE-COMPLIANCE
    STATUS  current
    DESCRIPTION
        "The compliance statement for device support of
        Port Access Control."
```

```
    MODULE
        MANDATORY-GROUPS {
            dot1xPaeSystemPortGroup
        }

        GROUP   dot1xPaeAuthSystemGroup
        DESCRIPTION
            "This group is mandatory for systems that support
            the Authenticator functions of the PAE."

        GROUP   dot1xPaeAuthConfigGroup
        DESCRIPTION
            "This group is mandatory for systems that support
            the Authenticator functions of the PAE."

        OBJECT  dot1xAuthAdminControlledDirections
        SYNTAX  INTEGER {
                    both(0)
                }
        MIN-ACCESS read-only
        DESCRIPTION
            "Support for in(1) is optional."

        OBJECT  dot1xAuthOperControlledDirections
        SYNTAX  INTEGER {
                    both(0)
                }
        DESCRIPTION
            "Support for in(1) is optional."

        OBJECT dot1xAuthKeyTxEnabled
        MIN-ACCESS read-only
        DESCRIPTION
            "An Authenticator PAE that does not support
            EAPOL-Key frames may implement this object as
            read-only, returning a value of FALSE."

        GROUP   dot1xPaeAuthStatsGroup
        DESCRIPTION
            "This group is mandatory for systems that support
            the Authenticator functions of the PAE."

        GROUP   dot1xPaeAuthDiagGroup
        DESCRIPTION
            "This group is optional for systems that support
            the Authenticator functions of the PAE."
```

```
        GROUP   dot1xPaeAuthSessionStats2Group
        DESCRIPTION
            "This group is optional for systems that support
            the Authenticator functions of the PAE."

        GROUP   dot1xPaeSuppConfigGroup
        DESCRIPTION
            "This group is mandatory for systems that support
            the Supplicant functions of the PAE."

        GROUP   dot1xPaeSuppStatsGroup
        DESCRIPTION
            "This group is mandatory for systems that support
            the Supplicant functions of the PAE."

        GROUP   dot1xPaeAuthInitEapGroup
        DESCRIPTION
            "This group is optional for systems that support
            the Authenticator functions of the PAE."

    ::= { dot1xPaeCompliances 2 }

END
```

Annex A (normative)

PICS Proforma[1]

<<Throughout this document, all notes such as this one, presented between angle braces, are temporary notes inserted by the Editors for a variety of purposes; these notes will all be removed prior to publication and are not part of the normative text.>>

<<Material borrowed from 802.1D is scattered through this clause as a prompt to the editor and reviewers to supply analagous material for MAC Security, if appropriate.>>

A.1 Introduction

The supplier of a protocol implementation which is claimed to conform to this standard shall complete the following Protocol Implementation Conformance Statement (PICS) proforma.

A completed PICS proforma is the PICS for the implementation in question. The PICS is a statement of which capabilities and options of the protocol have been implemented. The PICS can have a number of uses, including use

a) By the protocol implementor, as a checklist to reduce the risk of failure to conform to the standard through oversight;
b) By the supplier and acquirer—or potential acquirer—of the implementation, as a detailed indication of the capabilities of the implementation, stated relative to the common basis for understanding provided by the standard PICS proforma;
c) By the user—or potential user—of the implementation, as a basis for initially checking the possibility of interworking with another implementation (note that, while interworking can never be guaranteed, failure to interwork can often be predicted from incompatible PICSs);
d) By a protocol tester, as the basis for selecting appropriate tests against which to assess the claim for conformance of the implementation.

A.2 Abbreviations and special symbols

A.2.1 Status symbols

M	mandatory
O	optional
O.n	optional, but support of at least one of the group of options labelled by the same numeral *n* is required
X	prohibited
pred:	conditional-item symbol, including predicate identification: see A.3.4
¬	logical negation, applied to a conditional item's predicate

A.2.2 General abbreviations

N/A	not applicable
PICS	Protocol Implementation Conformance Statement

[1]*Copyright release for PICS proformas:* Users of this standard may freely reproduce the PICS proforma in this annex so that it can be used for its intended purpose and may further publish the completed PICS.

A.3 Instructions for completing the PICS proforma

A.3.1 General structure of the PICS proforma

The first part of the PICS proforma, implementation identification and protocol summary, is to be completed as indicated with the information necessary to identify fully both the supplier and the implementation.

The main part of the PICS proforma is a fixed-format questionnaire, divided into several subclauses, each containing a number of individual items. Answers to the questionnaire items are to be provided in the rightmost column, either by simply marking an answer to indicate a restricted choice (usually Yes or No), or by entering a value or a set or range of values. (Note that there are some items where two or more choices from a set of possible answers can apply; all relevant choices are to be marked.)

Each item is identified by an item reference in the first column. The second column contains the question to be answered; the third column records the status of the item—whether support is mandatory, optional, or conditional; see also A.3.4 below. The fourth column contains the reference or references to the material that specifies the item in the main body of this standard, and the fifth column provides the space for the answers.

A supplier may also provide (or be required to provide) further information, categorized as either Additional Information or Exception Information. When present, each kind of further information is to be provided in a further subclause of items labelled *Ai* or *Xi*, respectively, for cross-referencing purposes, where *i* is any unambiguous identification for the item (e.g., simply a numeral). There are no other restrictions on its format and presentation.

A completed PICS proforma, including any Additional Information and Exception Information, is the Protocol Implementation Conformation Statement for the implementation in question.

NOTE—Where an implementation is capable of being configured in more than one way, a single PICS may be able to describe all such configurations. However, the supplier has the choice of providing more than one PICS, each covering some subset of the implementation's configuration capabilities, in case that makes for easier and clearer presentation of the information.

A.3.2 Additional information

Items of Additional Information allow a supplier to provide further information intended to assist the interpretation of the PICS. It is not intended or expected that a large quantity will be supplied, and a PICS can be considered complete without any such information. Examples might be an outline of the ways in which a (single) implementation can be set up to operate in a variety of environments and configurations, or information about aspects of the implementation that are outside the scope of this standard but that have a bearing upon the answers to some items.

References to items of Additional Information may be entered next to any answer in the questionnaire, and may be included in items of Exception Information.

A.3.3 Exception information

It may occasionally happen that a supplier will wish to answer an item with mandatory status (after any conditions have been applied) in a way that conflicts with the indicated requirement. No pre-printed answer will be found in the Support column for this: instead, the supplier shall write the missing answer into the Support column, together with an *Xi* reference to an item of Exception Information, and shall provide the appropriate rationale in the Exception item itself.

An implementation for which an Exception item is required in this way does not conform to this standard.

NOTE—A possible reason for the situation described above is that a defect in this standard has been reported, a correction for which is expected to change the requirement not met by the implementation.

A.3.4 Conditional status

A.3.4.1 Conditional items

The PICS proforma contains a number of conditional items. These are items for which both the applicability of the item itself, and its status if it does apply—mandatory or optional—are dependent upon whether or not certain other items are supported.

Where a group of items is subject to the same condition for applicability, a separate preliminary question about the condition appears at the head of the group, with an instruction to skip to a later point in the questionnaire if the "Not Applicable" answer is selected. Otherwise, individual conditional items are indicated by a conditional symbol in the Status column.

A conditional symbol is of the form "**pred:** S" where **pred** is a predicate as described in A.3.4.2 below, and S is a status symbol, M or 0.

If the value of the predicate is true (see A.3.4.2), the conditional item is applicable, and its status is indicated by the status symbol following the predicate: the answer column is to be marked in the usual way. If the value of the predicate is false, the "Not Applicable" (N/A) answer is to be marked.

A.3.4.2 Predicates

A predicate is one of the following:

a) An item-reference for an item in the PICS proforma: the value of the predicate is true if the item is marked as supported, and is false otherwise;
b) A predicate-name, for a predicate defined as a boolean expression constructed by combining item-references using the boolean operator OR: the value of the predicate is true if one or more of the items is marked as supported;
c) A predicate-name, for a predicate defined as a boolean expression constructed by combining item-references using the boolean operator AND: the value of the predicate is true if all of the items are marked as supported;
d) The logical negation symbol "¬" prefixed to an item-reference or predicate-name: the value of the predicate is true if the value of the predicate formed by omitting the "¬" symbol is false, and vice versa.

Each item whose reference is used in a predicate or predicate definition, or in a preliminary question for grouped conditional items, is indicated by an asterisk in the Item column.

A.4 PICS proforma for IEEE Std 802.1AE

A.4.1 Implementation identification

Supplier	
Contact point for queries about the PICS	
Implementation Name(s) and Version(s)	
Other information necessary for full identification—e.g., name(s) and version(s) of machines and/or operating system names	
NOTE 1—Only the first three items are required for all implementations; other information may be completed as appropriate in meeting the requirement for full identification. NOTE 2—The terms Name and Version should be interpreted appropriately to correspond with a supplier's terminology (e.g., Type, Series, Model).	

A.4.2 Protocol summary, IEEE Std 802.1AE

Identification of protocol specification	IEEE Std 802.1AE, Standards for Local and Metropolitan Area Networks: Media Access Control (MAC) Security
Identification of amendments and corrigenda to the PICS proforma which have been completed as part of the PICS	Amd. : Corr. : Amd. : Corr. :
Have any Exception items been required? (See A.3.3: the answer Yes means that the implementation does not conform to IEEE Std 802.1AE.)	No [ ] Yes [ ]

Date of Statement	

A.5 Major Capabilities

Item	Feature	Status	References	Support
MACP	Does the implementation provide the MAC Service, as specified in <ref>, for use by end system functionality in the containing system?	M	A.6	
EISSP	Does the implementation provide the Extended Internal Sublayer Service as specified in IEEE Std 802.1Q to support the MAC Bridge functionality?	O	A.7	
EISSU	Is each specific MAC Technology used as specified by IEEE Std 802.1Q for the support of the MAC Extended Internal Sublayer Service for that MAC Technology? (The PICS Proforma(s) required by IEEE Std 802.1Q shall also be completed.) (If support of a specific MAC technology is claimed any PICS Proforma(s) required by the Standard specifying that technology shall also be completed.)	M	IEEE Std 802.1Q 6.4, . A.8	Yes [ ]
SECS	Does the implementation support the full range of security services specified in Clause 6 of this standard?	M	<<ref>> A.9	
EX1	Does the implemention provide this mandatory major capability?	M	<<ref>> A.10	Yes [ ]
EX2	Is this major capability supported?	O	<<ref>> A.11	Yes [ ] No [ ]
EX3	Does the implementation do what is supposed to do in respect of this major capability?	**EX2:**M	<<ref>> A.12	Yes [ ] N/A[ ]

A.6 Provision of the MAC Service

Item	Feature	Status	References	Support
MACP-1	Has it been done right?	M		Yes [ ]
MACP-2	First detail?	M		Yes[ ]
MACP-3	Second detail?	M	6.4, .	Yes [ ]
MACP-4	Are the MAC status parameters implemented on all Ports?	M	6.4, .	Yes [ ]

Predicates:
GOOK= GOOK_SPEC[Yes]

A.7 Provision of the Extended Internal Sublayer Service

Item	Feature	Status	References	Support
EISSP-1	Has it been done right?	**EISSP:**M		Yes [ ]
EISSP-2		M		Yes[ ]

A.8 Use of the Extended Internal Sublayer Service

Item	Feature	Status	References	Support
EISSU-1				
EISSU-2				

A.9 Security services

Item	Feature	Status	References	Support
SECS-1				Yes [ ]
SECS-2				Yes [ ]

A.10 Major Capability 1

Item	Feature	Status	References	Support
EX1-1				Yes [ ]
EX1-2				Yes [ ]

A.11 Major Capability 2

Item	Feature	Status	References	Support
EX2-1				Yes [ ]
EX2-2				Yes [ ]

A.12 Major Capability 3

Item	Feature	Status	References	Support
EX3-1				Yes [ ]
EX3-2				Yes [ ]

Annex Y (informative)

Bibiography

<<Items in the References clause should only be those that are definitely referenced by the document, not just useful background reading. The latter should go here.>>

Annex Z (informative)

Commentary

<<Editor's Note: This is a temporary Annex, included as a record of technical issues and their disposition. This annex will be removed prior to Sponsor Ballot, and preserved on the 802.1 website for future reference[1].>>

The order of discussion of issues is intended to help the reader understand first what is the draft, secondly what may be added, and thirdly what has been considered but will not be included. In pursuit of this goal, issues where the proposed disposition is "no change" will be moved to the end. The description of issues is updated to reflect our current understanding[2] of the problem and its solution: where it has been considered useful to retain the original comment, in whole or part, either to ensure that its author does not feel that it has not been sufficiently argued or the editor suspects there may be further aspects to the issue, that has been done as a footnote.

Z.1 Replay protection

Notwithstanding the emphasis on ‘connectionless’ confidentiality and integrity in the PAR, and the implication that not only is the service provided connectionless (i.e. one request primitive has not relationship to any other except for quality of service aspects) but also the service support is connectionless, i.e. there is no relationship between one frame and another, it has been agreed that the discussion, and potential provision, of replay protection falls within the scope of the PAR.

Z.1.1 Disposition

And this is what we have decided so far ....

[1]The footnotes in this annex provide further background to its development. Most of the highly subjective material, who said what and were they were right etc. together with temporary notes on blind alleys will be put into the footnotes so that they can be easily stripped out when the final annex is preserved.

[2]This annex is not intended therefore to be a complete historical record of the development of the draft. The formal record is largely captured in the Disposition of Comments on each ballot.