

*Recognized as an
American National Standard (ANSI)*

IEEE Std 802.1X-2001

**IEEE Standard for
Local and metropolitan area networks—
Port-Based Network Access Control**

Sponsor

**LAN/MAN Standards Committee
of the
IEEE Computer Society**

Approved 14 June 2001

IEEE-SA Standards Board

Approved 25 October 2001

American National Standards Institute

Abstract: *Port-based network access control makes use of the physical access characteristics of IEEE 802[®] Local Area Networks (LAN) infrastructures in order to provide a means of authenticating and authorizing devices attached to a LAN port that has point-to-point connection characteristics, and of preventing access to that port in cases in which the authentication and authorization process fails.*

Keywords: *authentication, authorization, controlled Port, Local Area Networks, Port Access Control, uncontrolled Port*

*The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA*

*Copyright © 2001 by the Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 13 July 2001. Printed in the United States of America.*

*Print: ISBN 0-7381-2626-7 SH94940
PDF: ISBN 0-7381-2927-5 SS94940*

No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without the prior written permission of the publisher.

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.

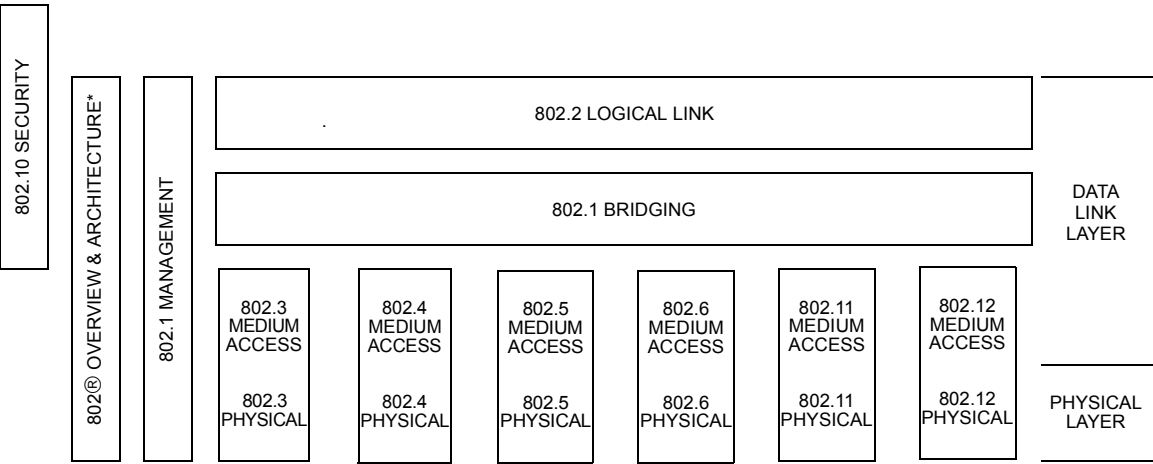
Authorization to photocopy portions of any individual standard for internal or personal use is granted by the Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright 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.

Introduction

(This introduction is not part of IEEE Std 802.1X-2001, IEEE Standards for Local and Metropolitan Area Networks: Port-Based Network Access Control.)

This standard defines a mechanism for Port-based network access control that makes use of the physical access characteristics of IEEE 802 LAN infrastructures in order to provide a means of authenticating and authorizing devices attached to a LAN port that has point-to-point connection characteristics, and of preventing access to that port in cases in which the authentication and authorization process fails.

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.)



* Formerly IEEE Std 802.1A.

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

The standards defining the technologies noted above are as follows:

- IEEE Std 802¹: *Overview and Architecture.* This standard provides an overview to the family of IEEE 802 Standards. This document forms part of the IEEE 802.1 scope of work.
- ANSI/IEEE Std 802.1B and 802.1K [ISO/IEC 15802-2]: *LAN/MAN Management.* Defines an Open Systems Interconnection (OSI) management-compatible architecture, and services and protocol elements for use in a LAN/MAN environment for performing remote management.

¹The 802 Architecture and Overview Specification, originally known as IEEE Std 802.1A, has been renumbered as IEEE Std 802. This has been done to accommodate recognition of the base standard in a family of standards. References to IEEE Std 802.1A should be considered as references to IEEE Std 802.

- ANSI/IEEE Std 802.1D: *Media Access Control (MAC) Bridges*. Specifies an architecture and protocol for the [ISO/IEC 15802-3]: interconnection of IEEE 802 LANs below the MAC service boundary.
- ANSI/IEEE Std 802.1E [ISO/IEC 15802-4]: *System Load Protocol*. Specifies a set of services and protocol for those aspects of management concerned with the loading of systems on IEEE 802 LANs.
- ANSI/IEEE Std 802.1F: *Common Definitions and Procedures for IEEE 802 Management Information*.
- ANSI/IEEE Std 802.1G [ISO/IEC 15802-5]: *Remote Media Access Control (MAC) Bridging*. Specifies extensions for the interconnection, using non-LAN systems communication technologies, of geographically separated IEEE 802 LANs below the level of the logical link control protocol.
- ANSI/IEEE Std 802.1H [ISO/IEC TR 11802-5]: *Recommended Practice for Media Access Control (MAC) Bridging of Ethernet V2.0 in IEEE 802 Local Area Networks*.
- ANSI/IEEE Std 802.1Q: *Virtual Bridged Local Area Networks*. Defines an architecture for Virtual Bridged LANs, the services provided in Virtual Bridged LANs, and the protocols and algorithms involved in the provision of those services.
- ANSI/IEEE Std 802.2 [ISO/IEC 8802-2]: *Logical Link Control*.
- ANSI/IEEE Std 802.3 [ISO/IEC 8802-3]: *CSMA/CD Access Method and Physical Layer Specifications*.
- ANSI/IEEE Std 802.4 [ISO/IEC 8802-4]: *Token Bus Access Method and Physical Layer Specifications*.
- ANSI/IEEE Std 802.5 [ISO/IEC 8802-5]: *Token Ring Access Method and Physical Layer Specifications*.
- ANSI/IEEE Std 802.6 [ISO/IEC 8802-6]: *Distributed Queue Dual Bus Access Method and Physical Layer Specifications*.
- ANSI/IEEE Std 802.10: *Interoperable LAN/MAN Security*. Currently approved: Secure Data Exchange (SDE).
- ANSI/IEEE Std 802.11: [ISO/IEC 8802-11] *Wireless LAN Medium Access Control (MAC) Sublayer and Physical Layer Specifications*.
- ANSI/IEEE Std 802.12: [ISO/IEC 8802-12] *Demand Priority Access Method, Physical Layer and Repeater Specification*.
- IEEE Std 802.15: *Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks*.
- IEEE Std 802.16: *Standard Air Interface for Fixed Broadband Wireless Access Systems*.
- IEEE Std 802.17: *Resilient Packet Ring Access Method and Physical Layer Specifications*.

In addition to the family of standards, the following is a recommended practice for a common physical layer technology:

- IEEE Std 802.7: *IEEE Recommended Practice for Broadband Local Area Networks.*

The reader of this standard is urged to become familiar with the complete family of standards.

Conformance test methodology

An additional standards series, identified by the number IEEE 1802, has been established to identify the conformance test methodology documents for the IEEE 802 family of standards. Thus the conformance test documents for IEEE 802.3 are numbered IEEE 1802.3, the conformance test documents for IEEE 802.5 will be 1802.5, and so on. Similarly, ISO will use ISO/IEC 18802 to number conformance test standards for ISO/IEC 8802 standards.

Participants

At the time this standard was completed, the IEEE 802.1 Working Group had the following membership:

Tony Jeffree, *Chair and Editor*
Neil Jarvis, *Vice-Chair*
Mick Seaman, *Chair, Interworking Task Group*

Les Bell
Alan Chambers
Marc Cochran
Paul Congdon
Hesham El Bakoury
Norman W. Finn
Sharam Hakimi
Bob Hott
Toyoyuki Kato

Hal Keen
Daniel Kelley
Keith Klamm
Joe Laurence
Bill Lidinsky
Yaron Nachman
LeRoy Nash
Satoshi Obara
Luc Pariseau
Anil Rijasinghani

John J. Roesse
Ted Schroeder
Benjamin Schultz
Rosemary V. Slager
Andrew Smith
Michel Soerensen
Robin Tasker
Manoj Wadekar
Robert Williams

The following members of the balloting committee voted on this standard. Balloters may have voted for approval, disapproval, or abstention.

David J. Allred	Simon Harrison	Donal O'Mahony
Jacob Ben Ary	Osamu Ishida	Satoshi Obara
James T. Carlo	Raj Jain	Roger Pandanda
Linda T. Cheng	Kamran Jamal	Vikram Punj
Keith Chow	Neil A. Jarvis	Gary S. Robinson
Guru Dutt Dhingra	Anthony A. Jeffree	Edouard Y. Rocher
Thomas J. Dineen	Stuart J. Kerry	James W. Romlein
Christos Douligeris	Daniel R. Krent	Floyd E. Ross
Sourav K. Dutta	Stephen Barton Kruger	Jaideep Roy
Philip H. Enslow	Joseph Kubler	Leo Sintonen
Changxin Fan	David J. Law	Joseph S. Skorupa
John W. Fendrich	William Lidinsky	Fred J. Strauss
Michael A. Fischer	Randolph S. Little	Jonathan R. Thatcher
Richard A. Froke	Ronald Mahany	Jerry A. Thrasher
Robert J. Gagliano	Peter Martini	Mark-Rene Uchida
Gautam Garai	Bennett Meyer	Scott A. Valcourt
Alireza Ghazizahedi	David S. Millman	John Viaplana
Tim Godfrey	James F. Mollenauer	Paul A. Willis
Robert M. Grow	John E. Montague	Forrest D. Wright
Chris G. Guy	Robert Mortonson	Oren Yuen
	Robert O'Hara	

When the IEEE-SA Standards Board approved this standard on 14 June 2001, it had the following membership:

Donald N. Heirman, *Chair*

James T. Carlo, *Vice Chair*

Judith Gorman, *Secretary*

Chuck Adams	James H. Gurney	Paul J. Menchini
Mark D. Bowman	Raymond Hapeman	Daleep C. Mohla
Clyde R. Camp	Richard J. Holleman	Robert F. Munzner
Richard DeBlasio	Richard H. Hulett	Ronald C. Petersen
Harold E. Epstein	Lowell G. Johnson	Malcolm V. Thaden
H. Landis Floyd	Joseph L. Koepfinger*	Geoffrey O. Thompson
Jay Forster*	Peter H. Lips	Akio Tojo
Howard M. Frazier		Howard L. Wolfman

*Member Emeritus

Also included is the following nonvoting IEEE-SA Standards Board liaisons:

Satish K. Aggarwal, *NRC Representative*
Alan H. Cookson, *NIST Representative*
Donald R. Volzka, *TAB Representative*

Jennifer McClain Longman
IEEE Standards Project Editor

The marks “**IEEE**” and “**802**” are registered trademarks belonging to the IEEE. When using these marks to refer to The Institute of Electrical and Electronics Engineers, **802** standards or other standards, the marks should be in bold typeface and, at least once in text, use the registered trademark symbol “®”.

Contents

1.	Overview.....	1
1.1	Scope.....	1
1.2	Purpose.....	1
2.	References.....	2
3.	Definitions	5
3.1	Definitions	5
4.	Acronyms and abbreviations	5
5.	Conformance.....	5
5.1	Static conformance requirements.....	5
5.2	Options.....	6
6.	Principles of operation	7
6.1	Systems, Ports, and system roles	7
6.2	Port access entity	8
6.3	Controlled and uncontrolled access	8
6.4	Unidirectional and bidirectional control	12
6.5	Use of Port Access Control with IEEE Std 802.3ad-2000	13
7.	EAP encapsulation over LANs (EAPOL)	13
7.1	Transmission and representation of octets.....	13
7.2	EAPOL frame format for 802.3/Ethernet	14
7.3	EAPOL frame format for Token Ring/FDDI	14
7.4	Tagging EAPOL frames	14
7.5	EAPOL PDU field and parameter definitions	15
7.6	Key Descriptor format	17
7.7	EAP packet format—informative	19
7.8	EAPOL addressing	20
7.9	Use of EAPOL in shared media LANs.....	21
8.	Port Access Control	21
8.1	Purpose.....	21
8.2	Scope.....	21
8.3	Overview of Port Access Entity operation	22
8.4	Protocol operation.....	23
8.5	EAPOL state machines	31
9.	Management of port access control	56
9.1	Management functions.....	56
9.2	Managed objects	57
9.3	Data types	58

9.4	Authenticator PAE managed objects.....	58
9.5	Supplicant PAE managed objects	66
9.6	System managed objects	69
10.	Management protocol	70
10.1	Introduction.....	70
10.2	The SNMP Management Framework	70
10.3	Security Considerations	71
10.4	Structure of the MIB	71
10.5	Relationship to other MIBs.....	75
10.6	Definitions for Port Access Control MIB	76
Annex A	PICS Proforma	104
Annex B	Scenarios for the use of Port-Based Network Access Control	112
Annex C	Design considerations and background material for Port-Based Network Access Control	116
Annex D	IEEE 802.1X RADIUS Usage Guidelines.....	122
Annex E	Bibliography	134

IEEE Standard for Local and metropolitan area networks— Port-Based Network Access Control

1. Overview

1.1 Scope

IEEE 802[®] Local Area Networks (or LANs; see 3.4 in IEEE Std 802.1D, 1998 Edition) are often deployed in environments that permit unauthorized devices to be physically attached to the LAN infrastructure, or permit unauthorized users to attempt to access the LAN through equipment already attached. Examples of such environments include corporate LANs that provide LAN connectivity in areas of a building that are accessible to the general public, and LANs that are deployed by one organization in order to offer connectivity services to other organizations (for example, as may occur in a business park or a serviced office building). In such environments, it is desirable to restrict access to the services offered by the LAN to those users and devices that are permitted to make use of those services.

Port-based network access control makes use of the physical access characteristics of IEEE 802 LAN infrastructures in order to provide a means of authenticating and authorizing devices attached to a LAN port that has point-to-point connection characteristics, and of preventing access to that port in cases in which the authentication and authorization process fails. A port in this context is a single point of attachment to the LAN infrastructure. Examples of ports in which the use of authentication can be desirable include the Ports of MAC Bridges (as specified in IEEE 802.1D), the ports used to attach servers or routers to the LAN infrastructure, and associations between stations and access points in IEEE 802.11 Wireless LANs.

1.2 Purpose

For the purpose of providing compatible authentication and authorization mechanisms for devices interconnected by IEEE 802 LANs, this standard specifies a general method for the provision of port-based network access control. To this end, it

- a) Describes the architectural framework within which the authentication, and consequent actions, take place
- b) Defines the principles of operation of the access control mechanisms
- c) Defines the different levels of access control that are supported, and the behavior of the port with respect to the transmission and reception of frames at each level of access control

- d) Establishes the requirements for a protocol between the device that requires the authentication to take place (the Authenticator; see 3.1.1) and the device that is attached to the Authenticator's port (the Supplicant; see 3.1.5)
- e) Establishes the requirements for a protocol between the Authenticator and an Authentication Server (see 3.1.2)
- f) Specifies mechanisms and procedures that support network access control through the use of authentication and authorization protocols
- g) Specifies the encoding of the Protocol Data Units (PDUs) used in authentication and authorization protocol exchanges
- h) Establishes the requirements for management of port-based access control, identifying the managed objects and defining the management operations
- i) Specifies how the management operations are made available to a remote manager using the protocol and architectural description provided by the Simple Network Management Protocol (SNMP)
- j) Specifies the requirements to be satisfied by equipment claiming conformance to this standard

2. References

The following standards contain provisions which, through reference in this text, constitute provisions of the 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 802.1D, 1998 Edition (ISO/IEC 15802-3:1998), IEEE Standard for Information technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Common specifications—Part 3: Media access control (MAC) Bridges.^{2, 3}

IEEE Std 802.1Q, 1998 Edition, IEEE Standards for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks.

IEEE Std 802.1t-2001, IEEE Standard for Information technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Common specifications—Part 3: Media access control (MAC) Bridges—Amendment 1.

IEEE Std 802.3, 2000 Edition, IEEE Standard for Information technology—Local and metropolitan area networks—Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications.

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

²IEEE 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>.

³IEEE [ISO] and IEEE [ISO/IEC] documents are available from ISO Central Secretariat, 1 rue de Varembe, 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. IEEE [ISO] and IEEE [ISO/IEC] documents can be ordered on-line from the IEEE Standards Website: <http://www.standards.ieee.org>.

IEEE Std 802.5, 1998 Edition (ISO/IEC 8802-5-1998), IEEE Standard for Information technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements—Part 5: Token ring access method and physical layer specifications.

IEEE Std 802.11, 1999 Edition (ISO/IEC 8802-11: 1999), IEEE Standard for 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 1155, Structure and Identification of Management Information for TCP/IP-based Internets, Rose, M., and K. McCloghrie, May 1990.⁴

IETF RFC 1157, Simple Network Management Protocol, Case, J., Fedor, M., Schoffstall, M., and Davin, J., SNMP Research, May 1990.

IETF RFC 1212, Concise MIB Definitions, Rose, M., and McCloghrie, K., March 1991.

IETF STD 17, RFC 1213, Management Information Base for Network Management of TCP/IP-based internets, McCloghrie K., and Rose, M., Editors, March 1991.

IETF RFC 1215, A Convention for Defining Traps for use with the SNMP, Rose, M., March 1991.

IETF RFC 1305, Network Time Protocol (Version 3) Specification, Implementation and Analysis, Mills, D. L., March 1992.

IETF RFC 1901, Introduction to Community-based SNMPv2, Case, J., McCloghrie, K., Rose, M., and Waldbusser, S., January 1996.

IETF RFC 1905, Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2), Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, January 1996.

IETF RFC 1906, Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2), Case, J., McCloghrie, K., Rose, M., and Waldbusser, S., January 1996.

IETF RFC 2104, HMAC: Keyed-Hashing for Message Authentication, Krawczyk, H., Bellare, M., and Canetti, R., February 1997.

IETF RFC 2284, PPP Extensible Authentication Protocol (EAP), Blunk, L. and Vollbrecht, J., March 1998.

IETF RFC 2570, Introduction to Version 3 of the Internet-standard Network Management Framework, Case, J., Mundy, R., Partain, D., and Stewart, B., April 1999.

IETF RFC 2571, An Architecture for Describing SNMP Management Frameworks, Harrington, D., Presuhn, R., and Wijnen, B., April 1999.

IETF RFC 2572, Message Processing and Dispatching for the Simple Network Management Protocol (SNMP), Case, J., Harrington D., Presuhn, R., and Wijnen, B., April 1999.

⁴Internet RFCs are retrievable by FTP at ds.internic.net/rfc/rfcnnnn.txt, or by Web browser at <http://www.ietf.org/> (where nnnn is a standard's publication number, such as 1493), or call InterNIC at 1-800-444-4345 for information about receiving copies through the mail.

IETF RFC 2573, SNMP Applications, Levi, D., Meyer, P., and Stewart, B., April 1999.

IETF RFC 2574, The User-Based Security Model (USM) for Version 3 of the Simple Network Management Protocol (SNMPv3), Blumenthal, U. and Wijnen, B., April 1999.

IETF RFC 2575, View-based Access Control Model for the Simple Network Management Protocol (SNMP), Wijnen, B., Presuhn, R., and McCloghrie, K., April 1999.

IETF RFC 2578, STD 58, Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2), McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., Waldbusser, S., April 1999.

IETF RFC 2579, STD 58, Textual Conventions for Version 2 of the Simple Network Management Protocol (SNMPv2), McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., Waldbusser, S., April 1999.

IETF RFC 2580, STD 58, Conformance Statements for SMIV2, McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., Waldbusser, S., April 1999.

IETF RFC 2716, PPP EAP TLS Authentication Protocol, Aboba, B. and Simon, D., October 1999.

IETF RFC 2863, The Interfaces Group MIB using SMIV2, McCloghrie, K. and Kastenholz, F., June 2000.

IETF RFC 2865, Remote Authentication Dial In User Service (RADIUS), Rigney, C., Willens, S., Rubens, A., and Simpson, W., June 2000.

IETF RFC 2866, RADIUS accounting, Rigney, C., June 2000.

IETF RFC 2867, RADIUS Accounting Modifications for Tunnel Protocol Support, Zorn, G., Aboba, B., and Mitton, D., June 2000.

IETF RFC 2868, RADIUS Attributes for Tunnel Protocol Support, Zorn, G., Leifer, D., Rubens, A., Shriver, J., Holdrege, M., and Goyret, I., June 2000.

IETF RFC 2869, RADIUS Extensions, Rigney, C., Willats, W., and Calhoun, P., June 2000.

ISO/IEC 8824:1990, Information technology—Open Systems Interconnection—Specification of Abstract Syntax Notation One (ASN.1) (Provisionally retained edition).⁵

ISO/IEC 8825:1990, Information technology—Open Systems Interconnection—Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1) (Provisionally retained edition).

⁵ISO and ISO/IEC documents are available from the ISO Central Secretariat, 1 rue de Varembe, 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.

3. Definitions

For the purposes of this standard, the following terms, definitions, acronyms, and abbreviations apply. *The Authoritative Dictionary of IEEE Standards Terms*, Seventh Edition [B1],⁶ should be referenced for terms not defined in this clause.

3.1 Definitions

3.1.1 authenticator: An entity at one end of a point-to-point LAN segment that facilitates authentication of the entity attached to the other end of that link.

3.1.2 authentication server: An entity that provides an authentication service to an authenticator. This service determines, from the credentials provided by the supplicant, whether the supplicant is authorized to access the services provided by the authenticator.

NOTE—The authentication server function can be colocated with an authenticator, or it can be accessed remotely via a network to which the authenticator has access.

3.1.3 network access port: A point of attachment of a system to a LAN. It can be a physical port, for example, a single LAN MAC attached to a physical LAN segment, or a logical port, for example, an IEEE 802.11 association between a station and an access point.

NOTE—The term *port* is used in this standard as an abbreviation of network access port (see Clause 4).

3.1.4 port access entity (PAE): The protocol entity associated with a Port. It can support the protocol functionality associated with the authenticator, the supplicant, or both.

3.1.5 supplicant: An entity at one end of a point-to-point LAN segment that is being authenticated by an authenticator attached to the other end of that link.

NOTE—The term *supplicant* is used in this standard in place of the more conventional term, *peer*, used in other access control-related specifications.

3.1.6 system: A device that is attached to a LAN by one or more ports. Examples of systems include end stations, servers, MAC Bridges, and routers.

4. Acronyms and abbreviations

EAP	extensible authentication protocol
EAPOL	EAP over LANs
PAE	port access entity
Port	network access port
RADIUS	remote authentication dial in user service

5. Conformance

5.1 Static conformance requirements

A device for which conformance to this standard is claimed shall, for all Ports for which support is claimed:

⁶The numbers in brackets correspond to those of the bibliography in Annex E.

- a) Support the operation of the Port Access Entity (PAE) over the uncontrolled Port, as a Supplicant PAE, an Authenticator PAE, or both, as defined in Clause 8
- b) Support the system configuration functions as defined in 9.6.1
- c) Where Authenticator PAE operation is supported:
 - 1) Support the ability to configure the operation of the Authenticator as defined in 9.4.1
 - 2) Support the ability to maintain and retrieve the Authenticator statistics as described in 9.4.2
 - 3) Support operation of the controlled Port in a manner consistent with the use of AuthControlledPortControl parameter values of Force Unauthorized, Auto and Force Authorized, as defined in 6.3
 - 4) Support the ability to set the AuthControlledPortControl parameter to the values of Force Unauthorized, Auto and Force Authorized, as defined in 6.3, by management action
 - 5) Support operation of the controlled Port in a manner consistent with the use of AdminControlledDirections and OperControlledDirections parameter values of Both, as defined in 6.4
 - 6) Support regular reauthentication of the Supplicant by means of the Reauthentication Timer state machine, and support the ability to modify the reAuthTimer and reAuthEnabled parameters by management action (8.5.7 and 9.4.1)
- d) Where the Supplicant PAE operation is supported:
 - 1) Support the ability to configure the operation of the Supplicant as defined in 9.5.1
 - 2) Support the ability to maintain and retrieve the Supplicant statistics as described in 9.5.2

5.2 Options

A device for which conformance to this standard is claimed may, for any Port for which support is claimed:

- a) Support the operation of protocol entities other than the PAE over the uncontrolled Port
- b) Where Authenticator PAE operation is supported:
 - 1) Support the ability to maintain and retrieve the Authenticator diagnostics as described in 9.4.3
 - 2) Support the ability to maintain and retrieve the Authenticator session statistics as described in 9.4.4
 - 3) Support operation of the controlled Port in a manner consistent with the use of AdminControlledDirections and OperControlledDirections parameter values of In, and support the ability to set the AdminControlledDirections parameter to the values In and Both by management action, as defined in 6.4
 - 4) Support the ability to transmit key information to the Supplicant following successful authentication, and support the ability to modify the KeyTransmissionEnabled parameter by management action (8.4.9, 8.5.5, and 9.4.1)
- c) Where Supplicant PAE operation is supported:
 - 1) Support the ability to transmit key information to the Authenticator following successful authentication, and support the ability to modify the KeyTransmissionEnabled parameter by management action (8.4.9, 8.5.6, and 9.4.1)

6. Principles of operation

This clause describes the architectural framework of Port-based access control and the relationship between the access control function and the operation of the device(s) within which it is deployed.

6.1 Systems, Ports, and system roles

Devices that attach to a LAN, referred to in this standard as Systems (3.1.6), have one or more points of attachment to the LAN, referred to in this standard as Ports (3.1.3).

NOTE—An end station generally has a single point of attachment in the form of a network interface card (although some end stations, such as servers, often have multiple points of attachment); a MAC Bridge generally has two or more points of attachment in the form of the Bridge Ports.

The Ports of a System provide the means in which it can access services offered by other Systems reachable via the LAN, and provide the means in which it can offer services to other Systems reachable via the LAN. Port-based network access control allows the operation of a System's Port(s) to be controlled in order to ensure that access to its services is only permitted by Systems that are authorized to do so.

NOTE—The services that a System can offer include the relay function of a MAC Bridge, the routing function of a network layer router, file server functionality, and so on.

For the purposes of describing the operation of Port-based access control, a Port of a System (or more correctly, its Port Access Entity; see 6.2) is able to adopt one of two distinct roles within an access control interaction:

- a) **authenticator (3.1.1):** The Port that wishes to enforce authentication before allowing access to services that are accessible via that Port adopts the Authenticator role;
- b) **supplicant (3.1.5):** The Port that wishes to access the services offered by the Authenticator's system adopts the Supplicant role.

A further System role is described as follows:

- c) **authentication server (3.1.2):** The Authentication Server performs the authentication function necessary to check the credentials of the Supplicant on behalf of the Authenticator and indicates whether the Supplicant is authorized to access the Authenticator's services.

As can be seen from these descriptions, all three roles are necessary to complete an authentication exchange. A given System can be capable of adopting one or more of these roles; for example, an Authenticator and an Authentication Server can be colocated within the same System, allowing that System to perform the authentication function without the need for communication with an external server. Similarly, a Port can adopt the Supplicant role in some authentication exchanges, and the Authenticator role in others. An example of the latter may be found in a Bridged LAN, where a new Bridge added to the LAN may need to be successfully authenticated by the Port of the Bridge via which it connects to the LAN before it can authenticate other systems that attach to its Ports.

NOTE—Although colocation of the Authentication Server with an Authenticator is possible, the most common implementation of this mechanism will likely involve the use of an Authentication Server that is external to the Systems that contain the Authenticators.

6.2 Port access entity

The Port Access Entity (PAE) operates the Algorithms and Protocols associated with the authentication mechanisms defined in Clause 8 for a given Port of the System.

In the Supplicant role, the PAE is responsible for responding to requests from an Authenticator for information that will establish its credentials. The PAE that performs the Supplicant role in an authentication exchange is known as the Supplicant PAE.

In the Authenticator role, the PAE is responsible for communication with the Supplicant, and for submitting the information received from the Supplicant to a suitable Authentication Server in order for the credentials to be checked and for the consequent authorization state to be determined. The PAE that performs the Authenticator role in an authentication exchange is known as the Authenticator PAE.

The Authenticator PAE controls the authorized/unauthorized state of its controlled Port (see 6.3) depending on the outcome of the authentication process.

6.3 Controlled and uncontrolled access

Figure 6-1 illustrates that the operation of Port-based access control has the effect of creating two distinct points of access to the Authenticator System's point of attachment to the LAN. One point of access allows the uncontrolled exchange of PDUs between the System and other Systems on the LAN, regardless of the authorization state (the *uncontrolled Port*); the other point of access allows the exchange of PDUs only if the current state of the Port is Authorized (the *controlled Port*). The uncontrolled and controlled Ports are considered to be part of the same point of attachment to the LAN; any frame received on the physical Port is made available at both the controlled and uncontrolled Ports, subject to the authorization state associated with the controlled Port.

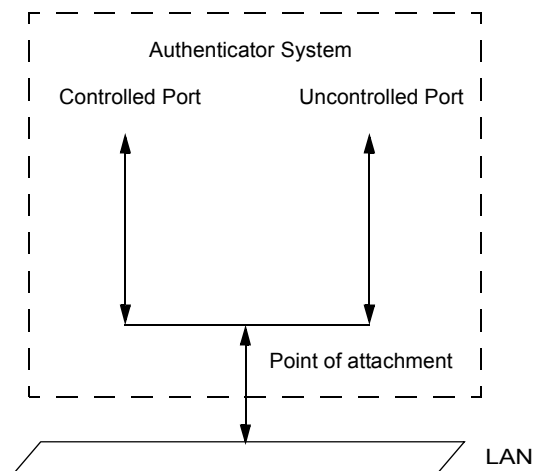


Figure 6-1—Uncontrolled and controlled Ports

NOTE—Arrows are used in this and subsequent diagrams to indicate the connectivity that is available in the various configurations illustrated. For example, in Figure 6-1, the upward pointing arrows indicate that incoming frames can reach users attached to both the controlled and uncontrolled Ports; the downward pointing arrows indicate that outbound frames from either the controlled or uncontrolled Port can reach the LAN.

The point of attachment to the LAN can be provided by any physical or logical Port that can provide a one-to-one connection to a Supplicant System. For example, the point of attachment could be provided by a single LAN MAC in a switched LAN infrastructure. In LAN environments where the MAC method allows

the possibility of a one-to-many relationship between an Authenticator and a Supplicant (for example, in shared media environments), the creation of a distinct association between a single Supplicant and a single Authenticator is a necessary precondition for the access control mechanisms described in this standard to function. An example of such an association would be an IEEE 802.11 association between a station and an access point.

Figure 6-2 illustrates the effect of the *AuthControlledPortStatus* associated with the controlled Port, representing that status as a switch that can be turned on or off, thus allowing or preventing the flow of PDUs via that Port. The figure shows two systems, each with a single Port; the *OperControlledDirections* parameter for each Port is assumed to be set to Both (see 6.4). In Authenticator System 1, the *AuthControlledPortStatus* associated with the controlled Port is *unauthorized* and is therefore disabled (the “switch” is turned off); in Authenticator System 2, the *AuthControlledPortStatus* is *authorized* and is therefore enabled (the “switch” is turned on).

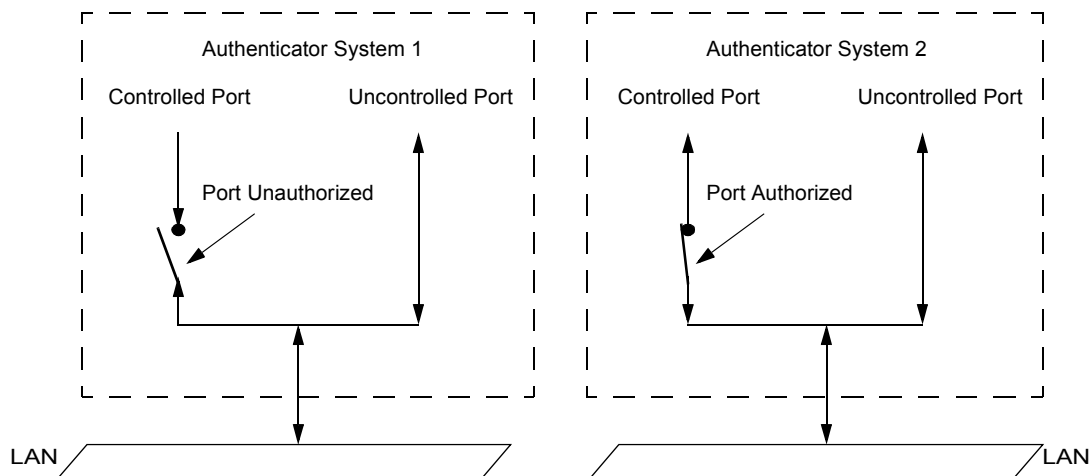


Figure 6-2—Effect of authorization state on controlled Ports

In addition to the *AuthControlledPortStatus*, an *AuthControlledPortControl* parameter associated with the Port allows administrative control over the Port’s authorization status. This parameter can take the values *ForceUnauthorized*, *Auto*, and *ForceAuthorized*; its default value is *Auto*. The relationship between the *AuthControlledPortStatus* and *AuthControlledPortControl* parameters is as follows:

- An *AuthControlledPortControl* value of *ForceUnauthorized* forces the Authenticator PAE state machine (8.5.4) to set the value of *AuthControlledPortStatus* to be *unauthorized*; i.e., the Controlled Port is *unauthorized* unconditionally.
- An *AuthControlledPortControl* value of *ForceAuthorized* forces the Authenticator PAE state machine (8.5.4) to set the value of *AuthControlledPortStatus* to be *Authorized*; i.e., the Controlled Port is *authorized* unconditionally.
- An *AuthControlledPortControl* value of *Auto* allows the Authenticator PAE state machine (8.5.4) to control the value of *AuthControlledPortStatus* to reflect the outcome of the authentication exchanges between Supplicant PAE, Authenticator PAE, and Authentication Server.

In all three cases, the value of *AuthControlledPortStatus* directly reflects the value of the *portStatus* variable maintained by the Authenticator PAE state machine (see 8.5.2.2 and 8.5.4).

The value of the *AuthControlledPortControl* parameter for every Port of a System can be overridden by means of the *SystemAuthControl* parameter for the System. This parameter can take the values *Enabled* and *Disabled*; its default value is *Disabled*. If *SystemAuthControl* is set to *Enabled*, then authentication is

enabled for the System, and each Port's authorization status is controlled in accordance with the value of the Port's AuthControlledPortControl parameter. If SystemAuthControl is set to Disabled, then all Ports behave as if their AuthControlledPortControl parameter is set to ForceAuthorized. In effect, setting the SystemAuthControl parameter to Disabled causes authentication to be disabled on all Ports, and it forces all Ports to be Authorized.

Any access to the LAN is subject to the current administrative and operational state of the MAC (or logical MAC) associated with the Port, in addition to AuthControlledPortStatus. If the MAC is physically or administratively inoperable, then no protocol exchanges of any kind can take place using that MAC on either the controlled or the uncontrolled Port. This is illustrated in Figure 6-3; in system 1, both the controlled and uncontrolled Ports are able to access the LAN, as the controlled Port is authorized, and the MAC providing the point of attachment to the LAN is operable. In system 2, neither the controlled nor the uncontrolled Port can access the LAN, as the MAC providing the point of attachment to the LAN is inoperable. The inoperable state of the MAC has also caused the Authenticator PAE to transition the controlled Port to the Unauthorized state, as shown in the diagram.

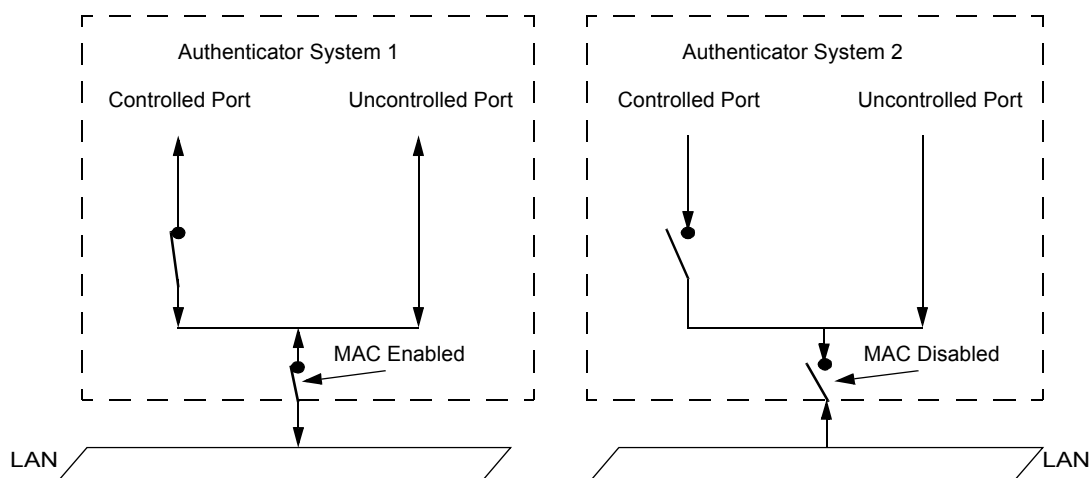


Figure 6-3—Effect of MAC enable/disable states

NOTE—Clause 6 of IEEE Std 802.1t-2001 describes the parameters available in a Bridge Port that indicate the administrative and operational states associated with the Port's MAC. P802.3ad describes similar parameters that define the administrative and operational states associated with the logical MAC offered by an aggregation. The effect of the authorization state on the operability of the Controlled Port is analogous to the effect of the MAC's operational state; however, the effect of the authorization state is modified by the OperControlledDirections parameter associated with the Port, as described in 6-4.

The Authenticator PAE uses the uncontrolled Port for the purposes of exchanging protocol information with the Supplicant.

Protocol exchanges between the Authenticator PAE and the Authentication Server (if the server is not colocated with the Authenticator PAE) can be conducted via one or more of the System's controlled or uncontrolled Ports.

NOTE—The details of communication between the Authenticator and the Authentication Server are outside of the scope of this standard. However, such communication would typically be achieved by means of an EAP connection carried over appropriate higher layer protocols; for example, by means of EAP in RADIUS. Hence, the Authentication Server can be located outside of the confines of the LAN that supports the EAPOL exchanges between Supplicant and Authenticator, and the communication between the Authenticator and Authentication Server need not be subject to the authentication state of the controlled Port(s) of the systems concerned. If a controlled Port is used to achieve communication with the Authentication Server, protocol exchanges can only take place if the controlled Port is in the authorized state.

It is expected that most protocol exchanges conducted by other functions of the System will make use of one or more of the System's controlled Ports. However, a given protocol may need to bypass the authorization function and make use of the uncontrolled Port. Figure 6-4 shows the uses of the controlled and uncontrolled Ports, and the ability of the Authenticator PAE to change the authorization state of its controlled Port depending on the outcome of an authentication exchange; the figure also gives an example of a protocol entity that requires the use of the uncontrolled port in order to conduct its protocol exchanges.

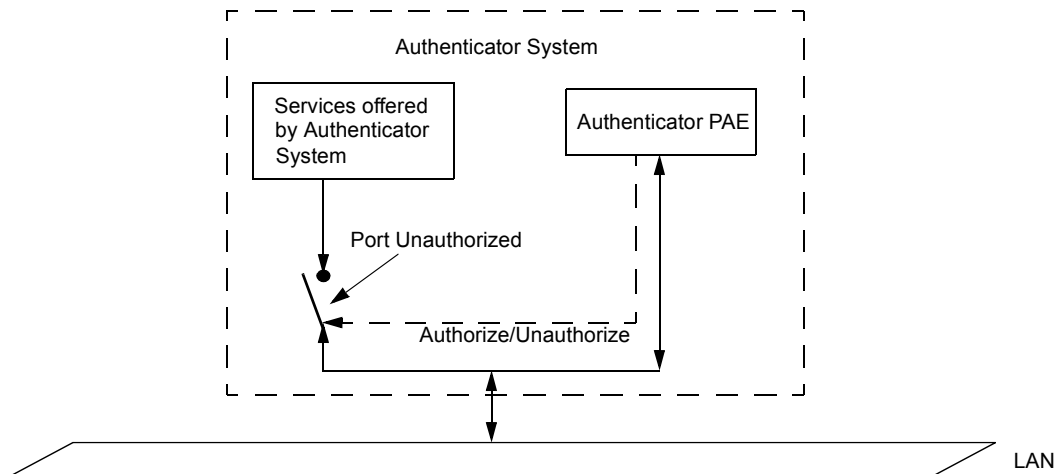


Figure 6-4—Use of the uncontrolled and controlled Ports

Figure 6-5 illustrates the relationship among the Supplicant, Authenticator, and Authentication Server, and the exchange of information among them. In this illustration, the Authenticator's controlled Port is in the unauthorized state and is therefore disabled from the point of view of access to the services offered by the Authenticator's system. The Authenticator PAE makes use of the uncontrolled Port to communicate with the Supplicant PAE, using EAPOL protocol exchanges, and communicates with the Authentication Server using EAP.

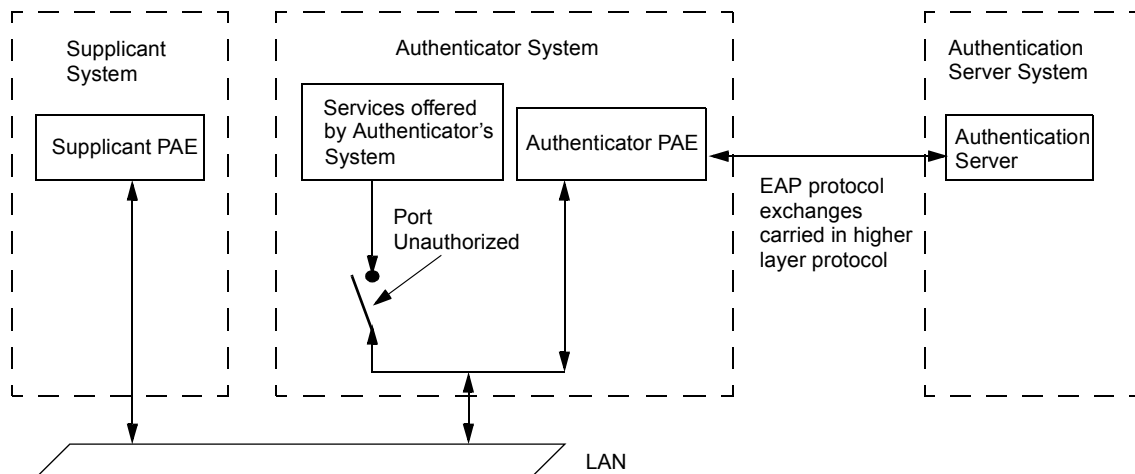


Figure 6-5—Authenticator, Supplicant, and Authentication Server roles

NOTE—The situation shown in Figure 6-5 could be found, for example, where the Supplicant's System was an end station and the Authenticator's System was a Bridge at the edge of a Bridged LAN.

The communication between the Authenticator and the Authentication Server may make use of the services of a LAN, or it may use some other communication channel. In cases in which the Authentication Server is colocated with the Authenticator, EAP protocol exchanges between these two entities are unnecessary.

Figure 6-6 illustrates a situation in which the PAEs associated with the two systems, A and B, are able to adopt either the Supplicant or the Authenticator roles, as necessary. In order for System A to make use of System B's services, System A's PAE must adopt the Supplicant role, and System B's PAE the Authenticator role. For System B to make use of System A's services, the roles are reversed. Note that although the Authentication Server function is shown as residing in two distinct systems in this example, this need not be the case.

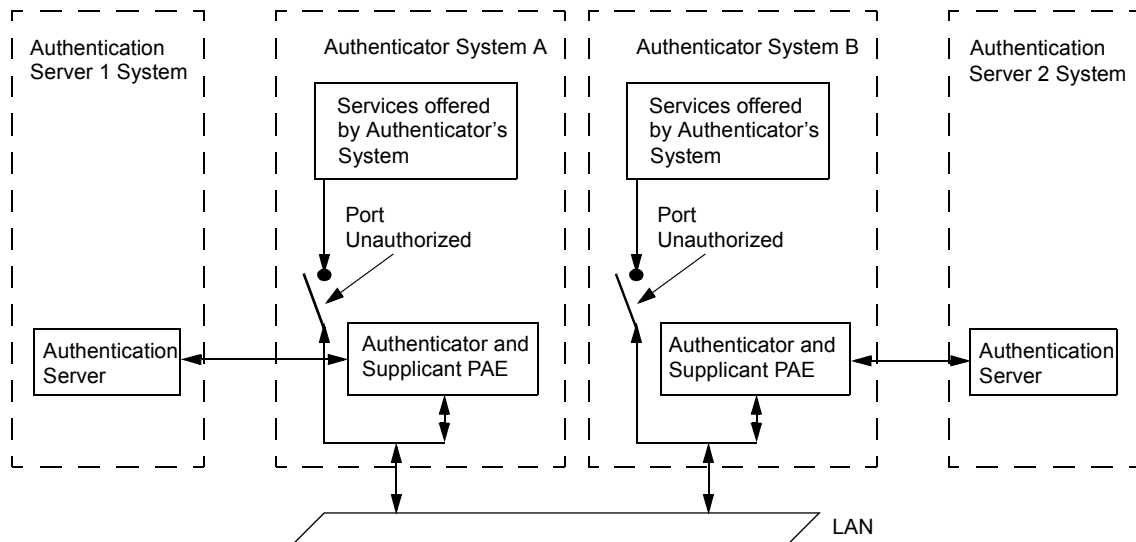


Figure 6-6—Systems adopting both Authenticator and Supplicant roles

NOTE—The situation shown in Figure 6-6 could be found, for example, where System A and System B are both Bridges. When they are initially connected together, each Bridge requires the other Bridge to be authenticated and authorized before it will forward frames on behalf of the other Bridge.

6.4 Unidirectional and bidirectional control

The degree to which protocol exchanges that take place on the controlled Port are affected by the authorization state is determined by two *controlled directions* parameters associated with each controlled port: an *AdminControlledDirections* parameter and an *OperationalControlledDirections* parameter. These parameters determine whether a controlled Port that is unauthorized exerts control over communication in both directions (disabling both reception of incoming frames and transmission of outgoing frames), or just in the incoming direction (disabling only reception of incoming frames). The controlled directions parameters can take one of two possible values, *Both* and *In*. The relationship between these two parameters, and the meaning of their values, is as follows:

- a) **AdminControlledDirections = Both.** This indicates that bidirectional control is required to be exerted; i.e., control is exerted over both incoming and outgoing traffic through the controlled Port. The value of *OperControlledDirections* is unconditionally set equal to *Both* if *AdminControlledDirections* is set equal to *Both*.
- b) **AdminControlledDirections = In.** This indicates that unidirectional control is required to be exerted; i.e., control is only exerted over incoming traffic through the controlled Port. If *AdminControlledDirections* is set equal to *In*, the value of *OperControlledDirections* is set equal to *In* on initialization and when the Port's MAC becomes operable. However, the value of *OperControlledDirections* is set to *Both* if any of the following conditions is true:
 - 1) The Port is a Bridge Port, and the Bridge Detection state machine (Clause 18 of IEEE Std 802.1t-2001) detects the presence of another Bridge connected to the Port.

- 2) The Port is a Bridge Port, and the Edge Port parameter for the Port is FALSE.
- 3) The Port's MAC becomes inoperable.

The value of the AdminControlledDirections parameter can only be modified by management. Implementations of Port-based access control shall support the ability to independently set the AdminControlledDirections parameter for each controlled Port to the value Both. Implementations of Port-based access control may support the ability to independently set the AdminControlledDirections parameter for each controlled Port to the value In.

NOTE—The In setting allows the security provisions of Port-based access control to be relaxed on Ports where an attached device needs to see protocol traffic from a controlled Port to support some forms of startup and initialization function (e.g., Wake-on-LAN), but where it is still desirable to prevent the attached device from transmitting into the controlled Port until authentication has taken place (see B.1). Clearly, the relaxation of the control offered by the controlled Port in this way reduces the effectiveness of Port-based access control in dealing with some types of attack.

The value of OperControlledDirections is determined by the operation of the Controlled Directions state machine (see 8.5.9).

6.5 Use of Port Access Control with IEEE Std 802.3, 2000 Edition

Where Port Access Control is used in conjunction with Ports that can be aggregated in accordance with IEEE Std 802.3ad-2000, the access control mechanisms shall operate on the physical Ports and shall not operate on the aggregated links (i.e., Port Access Control operates below the Link Aggregation sublayer). Ports shall be considered not to be aggregatable from the point of view of the operation of the Link Aggregation Control Protocol while they are in the Unauthorized state. Any Port that has become part of an aggregation, and that subsequently becomes Unauthorized, shall therefore be removed from that aggregation.

NOTE—The above is a requirement for interoperability in LAN environments that support both IEEE Std 802.3ad-2000 and Port access control. The background to this requirement is discussed in C.1.2.

7. EAP encapsulation over LANs (EAPOL)

This clause defines the encapsulation techniques that shall be used to carry EAP packets between Supplicant PAEs and Authenticator PAEs in a LAN environment. The encapsulation is known as *EAP over LANs*, or *EAPOL*. At present, EAPOL encapsulations are described for 802.3/Ethernet MACs and Token Ring/FDDI MACs. The EAPOL encapsulation used with 802.3/Ethernet MACs can be applied to other LAN technologies that share the same basic frame format as Ethernet (for example, IEEE Std 802.12 Demand Priority operating in IEEE Std 802.3 compatibility mode). Similarly, the EAP encapsulation used with Token Ring/FDDI MACs can be applied to other LAN technologies that share the same basic frame format as IEEE Std 802.5 Token Ring (for example, FDDI or IEEE Std 802.12 Demand Priority operating in IEEE Std 802.5 compatibility mode).

NOTE—In this standard, “802.3/Ethernet MACs” is used to refer to IEEE 802 MACs in which the native Link Layer protocol identification mechanism is based on a choice between the Type interpretation and Length interpretation of the Length/Type field. “Token Ring/FDDI MACs” is used to refer to MACs in which the native Link Layer protocol identification mechanism is based on LLC addressing.

7.1 Transmission and representation of octets

All EAPOL PDUs consist of an integral number of octets, numbered starting from 1 and increasing in the order that they are put into a MAC frame. The bits in each octet are numbered from 1 to 8, where 1 is the low-order bit.

When consecutive octets are used to represent a binary number, the lower numbered octet contains the more significant bits of the binary number.

When the encoding of (an element of) an EAPOL PDU is represented using a diagram in this clause, the following representations are used:

- a) Octet 1 is shown toward the top of the page, higher numbered octets being toward the bottom.
- b) Where more than one octet appears on a given line, octets are shown with the lowest numbered octet to the left, higher numbered octets being to the right.
- c) Within an octet, bits are shown with bit 8 to the left and bit 1 to the right.

7.2 EAPOL frame format for 802.3/Ethernet

A summary of the Ethernet form of an EAPOL frame is shown in Figure 7-1, starting with the Length/Type field of the MAC frame. The fields shown in the diagram are defined in 7.5.

	Octet Number
PAE Ethernet Type (7.5.1)	1-2
Protocol Version (7.5.3)	3
Packet Type (7.5.4)	4
Packet Body Length (7.5.5)	5-6
Packet Body (7.5.6)	7-N

Figure 7-1—EAPOL frame format for 802.3/Ethernet

7.3 EAPOL frame format for Token Ring/FDDI

A summary of the Token Ring/FDDI form of an EAPOL frame is shown in Figure 7-2. The fields shown in the diagram are defined in 7.5.

	Octet Number
SNAP-encoded Ethernet Type (7.5.2)	1-8
Protocol Version (7.5.3)	9
Packet Type (7.5.4)	10
Packet Body Length (7.5.5)	11-12
Packet Body (7.5.6)	13-N

Figure 7-2—EAPOL frame format for Token Ring/FDDI

The Routing Information Field (RIF), as defined in IEEE Std 802.5, 1998 Edition, shall not be present in any EAPOL frame.

7.4 Tagging EAPOL frames

EAPOL frames transmitted by a PAE shall not be VLAN tagged, but may optionally be priority tagged. All PAEs shall be capable of receiving both priority tagged and untagged EAPOL frames.

The structure of the tag header used for priority tagging is specified in IEEE Std 802.1Q, 1998 Edition.

7.5 EAPOL PDU field and parameter definitions

7.5.1 PAE Ethernet type

This field is two octets in length, and it shall contain the Ethernet Type value assigned for use by the PAE, as defined in Table 7-1.

7.5.2 SNAP-encoded Ethernet type

This field is eight octets in length, and it shall contain the SNAP-encoded Ethernet Type, encoded in SNAP format, as follows:

- a) Octets numbered 1 through 3 carry the standard SNAP header, consisting of the hexadecimal value AA-AA-03.
- b) Octets numbered 4 through 6 carry the SNAP PID, consisting of the hexadecimal value 00-00-00.
- c) Octets 7 and 8 carry the PAE Ethernet Type value, as defined in 7-1.

7.5.3 Protocol version

This field is one octet in length, taken to represent an unsigned binary number. Its value identifies the version of EAPOL protocol supported by the sender of the EAPOL frame. An implementation conforming to this specification shall use the value 0000 0001 in this field.

7.5.4 Packet type

This field is one octet in length, taken to represent an unsigned binary number. Its value determines the type of packet being transmitted. The following types are defined:

- a) **EAP-Packet.** A value of 0000 0000 indicates that the frame carries an EAP packet.
- b) **EAPOL-Start.** A value of 0000 0001 indicates that the frame is an EAPOL-Start frame.
- c) **EAPOL-Logoff.** A value of 0000 0010 indicates that the frame is an explicit EAPOL-Logoff request frame.
- d) **EAPOL-Key.** A value of 0000 0011 indicates that the frame is an EAPOL-Key frame.
- e) **EAPOL-Encapsulated-ASF-Alert.** A value of 0000 0100 indicates that the frame carries an EAPOL-Encapsulated-ASF-Alert.

All other possible values of this field shall not be used, as they are reserved for use in potential future extensions to this protocol.

NOTE—The EAPOL-Key packet type is used only where the optional ability to transmit key information between the Authenticator and the Supplicant is supported (see 8.4.9).

The EAPOL-Encapsulated-ASF-Alert packet type is provided for use by the Alerting Standards Forum (ASF) as a means of allowing alerts (e.g., specific SNMP traps; see B.1.4) to be forwarded through a Port that is in the Unauthorized state. All EAPOL frames with this packet type that are received on the uncontrolled Port are passed to the protocol entity responsible for handling ASF alerts for validation and further processing in accordance with the relevant ASF protocol specifications. This standard does not

further specify either the syntax or semantics of the alert messages that can be carried in this type of packet, or the protocol actions taken on receipt of a packet of this type.

7.5.5 Packet Body length

This field is two octets in length, taken to represent an unsigned binary number. The value of this field defines the length in octets of the Packet Body field (7.5.6); a value of 0 indicates that there is no Packet Body field present.

7.5.6 Packet Body

The Packet Body field is present if the Packet Type contains the value EAP-Packet, EAPOL-Key, or EAPOL-Encapsulated-ASF-Alert; for all other values of Packet Type, this field is not present.

In a frame carrying a Packet Type of EAP-Packet, this field contains an EAP packet as described in 7.7. Exactly one EAP packet is encapsulated.

In a frame carrying a Packet Type of EAPOL-Key, this field contains a Key Descriptor as described in 7.6. Exactly one Key Descriptor is encapsulated.

In a frame carrying a Packet Type of EAPOL-Encapsulated-ASF-Alert, this field contains an ASF alert frame as specified by the ASF (see B.1.4). Exactly one ASF alert frame is encapsulated.

NOTE—The maximum size of EAP packet that can be carried within an EAPOL frame will depend on the maximum MAC frame size supported by the MAC method by which the frame is transmitted.

7.5.7 Validation of received EAPOL frames and EAPOL protocol version handling

A PAE shall process a received EAPOL PDU as specified in this clause if and only if the following conditions are all true:

- a) The Destination MAC address field contains the PAE group address, as specified in 7.8 (in nonshared media LANs), or the specific MAC address of the PAE (in shared media LANs).
- b) The PAE Ethernet Type field contains the value of the PAE Ethernet Type, as specified in 7.8.
- c) The Packet Type field contains one of the values EAP-Packet, EAPOL-Start, EAPOL-Logoff, or EAPOL-Key, as specified in 7.5.4.

In the case of EAPOL frames that carry the Packet Type values EAPOL-Start and EAPOL-Logoff, any octets that appear in the PDU following the Packet Type field shall be ignored. In the case of EAPOL frames that carry the Packet Type value EAP-Packet and EAPOL-Key, any octets that appear in the PDU following the Packet Body field shall be ignored.

The following rules apply to the validation and interpretation of EAPOL PDUs, in order to ensure that backward compatibility is maintained between versions of this protocol.

For an implementation that supports version A of the protocol, a received EAPOL PDU of a given Packet Type that carries a protocol version number B is interpreted as follows:

- d) Where B is greater than or equal to A, the EAPOL PDU shall be interpreted as if it carried the supported version number, A. Specifically:
 - 1) All EAPOL PDU parameters that are defined in version A shall be interpreted in the manner specified for version A of the protocol for the given EAPOL PDU Packet Type.

- 2) All EAPOL PDU parameters that are undefined in version A for the given EAPOL PDU Packet Type shall be ignored.
- 3) All octets that appear in the EAPOL PDU beyond the largest numbered octet defined for version A for the given EAPOL PDU Packet Type shall be ignored.

NOTE—As a consequence of the rules stated in d) and its sub-bullets, a version 1 implementation may ignore the Protocol Version Identifier. This set of validation rules allows the possibility of future specification of extensions to the EAPOL Protocol, identified as new versions by different values of the identifier. Subsequent versions will be required to check the Protocol Version Identifier in order to correctly interpret the received PDU.

- e) Where B is less than A, the EAPOL PDU shall be interpreted as specified for the version number, B, carried in the EAPOL PDU. Specifically:
 - 1) All EAPOL PDU parameters shall be interpreted in the manner specified for version B of the protocol for the given EAPOL PDU Packet Type.
 - 2) All EAPOL PDU parameters that are undefined in version B for the given EAPOL PDU Packet Type shall be ignored.
 - 3) All octets that appear in the EAPOL PDU beyond the largest numbered octet defined for version B for the given EAPOL PDU Packet Type shall be ignored.

7.6 Key Descriptor format

A summary of the Key Descriptor format is shown in Figure 7-3. The fields shown in the diagram are defined in the following subclauses.

	Octet Number
Descriptor Type (7.6.1)	1
Key Length (7.6.2)	2-3
Replay Counter (7.6.3)	4-11
Key IV (7.6.4)	12-27
Key Index (7.6.5)	28
Key Signature (7.6.6)	29-44
Key (7.6.7)	45-Packet Body Length

Figure 7-3—Key Descriptor format

7.6.1 Descriptor type

This field is one octet in length, taken to represent an unsigned binary number. The value defines the type of the Key Descriptor, which in turn defines how the remaining fields of the Key Descriptor are used and interpreted. In particular, the Descriptor Type value defines how the Replay Counter is generated, how the Key IV is used, how the Key Signature field is generated, and how the Key field is encrypted.

The following values of Descriptor Type are defined:

- **RC4 Key Descriptor.** A value of 1 in the Descriptor Type field indicates that the Key Descriptor is an RC4 Descriptor. The definition of the RC4 Descriptor can be found in 7.6.8.

All other possible values of Descriptor Type are reserved for future standardization.

7.6.2 Key length

This field is two octets in length, taken to represent an unsigned binary number. The value defines the length of the key in octets. For example, a value of 5 in this field indicates a 40-bit key.

7.6.3 Replay counter

This field is 8 octets in length, taken to represent an unsigned binary number. It carries a counter value, used to detect and prevent replay of key messages.

7.6.4 Key IV

This field carries a 16-octet Initialization Vector value, consisting of 128 bits of random data.

7.6.5 Key index

This field is one octet in length, taken to represent a 7-bit unsigned binary number and a flag. The value is generated by the Authenticator specifying the key, and it is used as a key index number if multiple keys are supported. The index number is carried in bits 1 through 7, and it can carry an integer in the range 0-127. Bit 8 is a flag bit. If bit 8 is set to 1, the key is a unicast key; if bit 8 is set to 0, the key is a broadcast key.

7.6.6 Key signature

This field is 16 octets in length. The Key Signature is a signature of all of the fields of the EAPOL packet, from and including the EAPOL protocol version field, to and including the Encrypted Key field, with the signature set to 0.

7.6.7 Key

This field is optional. If it is not present, the Supplicant uses the peer key generated as part of the EAP authentication process as the key material for this message. If the key is longer than the key length specified in this message, then only the first N bytes are used.

7.6.8 RC4 Key Descriptor

Key Descriptors carrying a Descriptor Type of RC4 Key Descriptor are constructed and interpreted as follows:

- a) The Replay Counter field carries an NTP time value (see IETF RFC1305).
- b) The Key IV field carries a random number used to generate an RC4⁷ encryption key.
- c) A signature type of HMAC-MD5 is used to generate the Key Signature (see IETF RFC 2104). The key used for the signature is the server key generated by the EAP authentication (e.g., as defined in IETF RFC 2716).
- d) RC4 is used to encrypt the Key field. The RC4 encryption key is generated by concatenating the Key IV and the session key generated by the EAP authentication process (e.g., as defined in IETF RFC 2716). The key material is then encrypted according to the method specified by Encrypt Type (e.g., RC4 encrypted using the RC4 key).

⁷The mark RC4 is a registered trademark of RSA Security Inc. and may not be used by third parties creating implementations of the algorithm. RSA Security does not hold any patents nor does it have any pending applications on the RC4 algorithm. However, RSA Security does not represent or warrant that implementations of the algorithm will not infringe the intellectual property rights of any third party. Proprietary implementations of the RC4 encryption algorithm are available under license from RSA Security Inc. For licensing information, contact RSA Security Inc., 2955 Campus Drive, Suite 400, San Mateo, CA 94403-2507, USA, or <http://www.rsasecurity.com>.

7.7 EAP packet format—informative

A summary of the EAP packet format is shown in Figure 7-4. The figure and the accompanying field descriptions are included for illustrative purposes only; the normative definition of the EAP packet format and its associated semantics are to be found in the specification of EAP contained in IETF RFC 2284.

Code	Octet Number 1
Identifier	2
Length	3-4
Data	5-N

Figure 7-4—EAP packet format

7.7.1 Code

The Code field is one octet in length and identifies the type of EAP packet. EAP Codes are assigned as follows:

1	Request
2	Response
3	Success
4	Failure

7.7.2 Identifier

The Identifier field is one octet in length and allows matching of responses with requests. The Identifier field and System Port together uniquely identify an authentication exchange. Thus, the use of a single octet identifier field results in a restriction of 256 authentications per System Port.

NOTE—This restriction is not likely to be an issue given that authentication occurs at the local point of access, rather than in the core.

The operation of the Authenticator PAE determines the value of Identifier used in new EAP-Request/Identity frames. The Supplicant PAE uses the same Identifier in subsequent EAP-Response frames responding to that initial request frame. The Authenticator PAE uses the same Identifier value in any retransmissions of the same request.

7.7.3 Length

The Length field is two octets in length and indicates the length of the EAP packet, including the Code, Identifier, Length, and Data fields.

Note—Because in this specification EAP frames are transported directly over the link layer medium, the frame size cannot exceed the maximum permissible on that medium, as fragmentation is not supported.

7.7.4 Data

The Data field is zero or more octets. The format of the Data field is determined by the Code field.

7.8 EAPOL addressing

PAEs (see Clause 8) transmit and receive:

- a) EAPOL frames exchanged between an Authenticator and a Supplicant
- b) EAP frames exchanged between an Authenticator and an Authentication Server

NOTES

1—An Authentication Server may be colocated with an Authenticator; in which case, a communication protocol may not be required.

2—This standard refers to EAP carried in RADIUS as a basis of these exchanges; however, the use of other protocols to achieve these exchanges is permitted.

PAEs use the DL_UNITDATA.request and DL_UNITDATA.indication primitives associated with each active Port to transmit and receive EAPOL frames. In 802.3/Ethernet MACs, frames carry the Port Access Entity Ethernet Type as the data link address; this type value is defined in Table 7-1. In Token Ring/FDDI MACs, frames carry the Port Access Entity Ethernet Type in SNAP-encoded form, using a SNAP header of AA-AA-03 and a SNAP PID of 00-00-00, followed by the Ethernet Type value.

Table 7-1—Standard Ethernet Type assignment

Assignment	Value
Port Access Entity Ethernet Type	88-8E

A group MAC address, the PAE group address (Table 7-2), is assigned by IEEE Std 802.1D, 1998 Edition⁸ for use by PAEs. In MACs where the LAN technology concerned is such that the individual MAC address of the Supplicant is known to the Authenticator, and vice versa (for example, in IEEE 802.11, where the establishment of an association between a station and an access point involves the exchange of MAC addresses), all EAPOL frames transmitted by a PAE shall carry the individual MAC address associated with the destination PAE's point of LAN attachment as the destination MAC address. Otherwise (i.e., in MACs where the LAN technology concerned is such that the individual MAC address of the Supplicant can be unknown to the Authenticator, and vice versa), all EAPOL frames transmitted by a PAE shall carry the PAE group address as the destination MAC address. All EAPOL frames shall carry the individual MAC address associated with the source PAE's point of LAN attachment as the source MAC address.

Table 7-2—PAE group address assignment

Assignment	Value
Port Access Entity group address (PAE group address)	01-80-C2-00-00-03

NOTE—The PAE group address is one of the reserved set of group MAC addresses that are not forwarded by MAC Bridges.

⁸The normative assignment of this address is to be found in Table 7-9 of IEEE Std 802.1D, 1998 Edition. The value is repeated here for informative purposes.

7.9 Use of EAPOL in shared media LANs

The use of individual MAC addresses with EAPOL (7.8) permits the use of EAPOL in shared media LAN environments, and in particular, this has been allowed in order to support the use of Port-based Network Access Control in IEEE 802.11 wireless LAN infrastructures. However, it should be noted that such use can only be made secure if communication between the Supplicant and Authenticator systems takes place using a secure association. Attempting to use EAPOL in a shared medium environment that does not support the use of secure associations renders Port-based network access control highly vulnerable to attack; for example, station A can mount a successful denial of service attack on station B simply by issuing an EAPOL-Logoff packet using station B's individual MAC address.

8. Port Access Control

8.1 Purpose

Port Access Control provides an optional extension to the functionality of a System (6.1) that offers a means of preventing unauthorized access by Supplicants (3.1.5) to the services offered by that System. For example, if the System concerned is a MAC Bridge, control over access to the Bridge and the LAN to which it is connected can be desirable in order to restrict access to publicly accessible Bridge Ports, or within an organization, to restrict access to a departmental LAN to members of that department.

Access control is achieved by the System enforcing authentication of Supplicants that attach to the System's controlled Ports (see 6.3); from the result of the authentication process, the System can determine whether or not the Supplicant is authorized to access its services on that controlled Port. If the Supplicant is not authorized for access, the System sets the controlled Port state to unauthorized. In the unauthorized state, the use of the controlled Port is restricted in accordance with the value of the OperControlledDirections parameter associated with that controlled Port (6.4), preventing unauthorized data transfer between the Supplicant and the services offered by the System.

The mechanisms defined can be applied to allow any System to authenticate another System that is connected to one of its controlled Ports. The Systems concerned include end stations, servers, routers, and MAC Bridges.

NOTE—Given that the Supplicant device may be a Bridge or a router that wished to connect to and thereby extend a Bridged LAN, the process of authentication will determine whether it can be trusted to connect to the Bridged LAN, given that by doing so, it will effectively be extending the authentication boundary to any other Ports of the Supplicant. Similarly, it may be that in this situation, there is a need for each device to authenticate the other before either Port can become authorized; see the example shown in 6-6.

8.2 Scope

The operation of Port-based Access Control assumes that the Ports on which it operates offer a point-to-point connection between a single Supplicant (3.1.5) and a single Authenticator (3.1.1). It is this assumption that allows the authentication decisions to be made on a per-Port basis. The authentication of multiple Supplicant PAEs attached to a single Authenticator PAE is outside of the scope of this standard.

NOTE—For example, the operation of Port Access Control on a classic Token Ring LAN, or on a shared media Ethernet LAN segment, is outside of the scope of this standard.

This standard provides a protocol for communicating authentication information between a Supplicant, attached to a Port of an Authenticator System, and an Authentication Server, and for controlling the state of the Authenticator System's Port, depending on the outcome of the protocol exchange. This standard does not

specify the nature of the authentication information that is exchanged, nor the basis upon which the Authentication Server makes its authentication decisions.

8.3 Overview of Port Access Entity operation

A Port Access Entity (PAE) exists for each Port (3.1.3) of a System that participates in Port-based access control. The PAE is able to operate in the role of a Supplicant or of an Authenticator or both. The following subclauses offer an overview of the operation of the PAE in each of these roles.

8.3.1 Authenticator role

An Authenticator PAE is responsible for enforcing the authentication of a Supplicant PAE that attaches to its controlled Port and for controlling the authorization state of the controlled Port accordingly.

In order to perform the authentication, the Authenticator PAE makes use of an Authentication Server. The Authentication Server may be colocated in the same System as the Authenticator PAE, or it may be located elsewhere, accessible via remote communication mechanisms, LAN-based or otherwise. Communication between the Supplicant PAE and the Authenticator PAE, and between the Authenticator PAE and the Authentication Server (when the Authentication Server is not colocated with the Authenticator), is achieved by means of the protocols and procedures defined in 8.4.

8.3.2 Supplicant role

A Supplicant PAE is responsible for communicating the credentials of the Supplicant to the Authenticator PAE in response to requests from the Authenticator PAE. The Supplicant PAE may also initiate authentication exchanges and perform EAPOL-Logoff exchanges, as further described in this clause.

8.3.3 Port access restrictions

Authentication occurs primarily at System initialization time, or when a Supplicant System is connected to a Port of an Authenticator System. Until authentication has successfully completed, the Supplicant System only has access to the Authenticator System to perform authentication exchanges, or to access any services offered by the Authenticator's System that are not subject to the access control restrictions placed on the Authenticator's controlled Port (see 6.3 and 6.4). Once authentication has successfully completed, the Authenticator System allows full access to the services offered via the Authenticator System's controlled Port.

The operational state of the MAC that supports a controlled Port can be disabled or enabled. If the MAC's operational state is disabled, then the MAC is not available for use, regardless of the authorization state associated with the controlled Port.

NOTE—Clause 6 of IEEE Std 802.1t-2001 describes the parameters available in a Bridge Port that indicate the enable/disable states of the Port's MAC. P802.3ad describes similar parameters that define the enable/disable states of the logical MAC offered by an aggregation.

In a System that implements an Authenticator PAE, the controlled Port is placed in the *unauthorized* state until authentication has taken place, and it is therefore also disabled. Once authentication has succeeded, and it has been determined that the authenticated user is authorized to access the controlled Port, the controlled Port is placed in the *authorized* state; assuming that there is no other reason for it still to be disabled (e.g., the MAC has been disabled for administrative purposes), the controlled Port is then available for use (see 6.3).

NOTE—While in the unauthorized state, DHCP and other initialization traffic may not be transmitted or received via the controlled Port, depending on the current value of the Port's Controlled Directions parameters (see 6.4). As a result,

authentication may need to occur early in the end-station initialization sequence (prior to DHCP and IP initialization for example).

In addition to controlling the controlled Port's authorization state, the operation of the PAE may support ageing out of the authorization state of controlled Ports, and it may request that the Supplicant reauthenticate at any time. Controlled Ports remain authorized during reauthentication and transition to the unauthorized state only if reauthentication fails.

Authentication is configurable on a per-Port basis, because it will be desirable in some configurations not to perform authentication on certain Ports (e.g., on inter-Bridge links, Ports attached to servers, etc.). The management operations available for such configuration are described in Clause 9.

8.3.4 Logoff mechanisms

There are several mechanisms that can result in the Authenticator changing the controlled Port state to unauthorized and thereby controlling access via that Port in accordance with its OperControlledDirections parameter (6.4):

- a) The authentication exchanges between the Supplicant and the Authentication Server may result in failure to authorize the Port.
- b) Management controls may prevent the Port from being authorized, regardless of the credentials of the Supplicant.
- c) The MAC associated with the Port may be nonoperational for any reason (including for hardware failure or administrative reasons).
- d) Connection failure between the Supplicant and the Authenticator may result in the Authenticator timing out the authorization state.
- e) Expiry of a reauthentication timer occurs without successful reauthorization.
- f) The Supplicant PAE fails to respond to a request for authentication information by the Authenticator PAE.
- g) The Supplicant PAE issues an explicit EAPOL-Logoff request.

When a user logs off from an end station, it is possible in some environments for the user (or a different user) to bypass a new login request and thereby gain access to the end station and the network. Providing the explicit logoff mechanism ensures that the session is terminated, not only with respect to the user's access to the end station, but also for the end station's authorization status with the controlled Port of the Authenticator System to which it is connected. An explicit logoff therefore causes the Authenticator PAE to set the controlled Port concerned to the unauthorized state.

8.4 Protocol operation

8.4.1 Overview

The operation of the authentication process makes use of the Extensible Authentication Protocol (EAP, specified in IETF RFC 2284) as the means of communicating authentication information between the Supplicant and the Authentication Server. EAP is a general protocol that supports multiple authentication mechanisms. For example, through the use of EAP, support for a number of authentication schemes may be added, including smart cards, Kerberos, Public Key Encryption, One Time Passwords, and others.

The approach taken in this standard is to define an encapsulation format that allows EAP messages to be carried directly by a LAN MAC service. This encapsulated form of EAP, known as EAP over LANs, or

EAPOL, is used for all communication between the Supplicant PAE and the Authenticator PAE. The Authenticator PAE can then repackage the EAP protocol for onward transmission to the Authentication Server, if the server function is not colocated. RADIUS offers one suitable means of providing this latter aspect of communication; however, this may be achieved by the use of other protocols.

In addition to describing the encapsulation of EAPOL, this standard describes the Authenticator PAE function that relays EAP messages between a Supplicant PAE and an Authentication Server.

The Authenticator PAE controls the operational state of its controlled Port, but it does not interfere with the authentication exchanges between the Supplicant PAE and the Authentication Server. This separation between the Authenticator PAE and the authentication function permits the use of a “backend” Authentication Server that implements the various mechanisms needed to authenticate a Supplicant PAE. The Authenticator PAE simply controls the authorization state of its controlled Port based on the results of the authentication exchange. This separation does not preclude implementation of both Authenticator PAE and Authentication Server in the same item of physical equipment. A full description of the authentication function can be found in IETF RFC 2869, and guidelines for the use of RADIUS in IEEE 802.1X can be found in Annex D.

Rather than only permitting a predetermined authentication method, EAP allows the Authenticator PAE to request more information before determining the specific authentication mechanism. In EAP, the authenticator PAE sends one or more Requests to authenticate the Supplicant PAE. The Request has a type field to indicate what is being requested. Examples of Request types include Identity, MD5-challenge, One-Time Passwords, and Generic Token Card. The MD5-challenge type corresponds closely to the CHAP authentication protocol. Typically, the authenticator will send an initial Identity Request followed by one or more Requests for authentication information. However, an initial Identity Request is not required, and it may be bypassed in cases in which the identity is presumed. The Supplicant PAE sends a Response packet in reply to each Request. As with the request packet, the Response packet contains a type field that corresponds to the type field of the Request.

The authentication exchange ends with an Accept or Reject indication from the Authentication Server. This indication may or may not contain an encapsulated EAP packet. Regardless of whether an EAP packet is encapsulated or not, it is the Accept or Reject indication that signals to the Authenticator whether the Controlled Port is authorized or unauthorized.

8.4.2 Authentication initiation

Authentication can be initiated either by the Supplicant PAE or by the Authenticator PAE. If authentication is enabled on a given Port (i.e., if the Port supports an Authenticator PAE and the `Auth-ControlledPortControl` parameter for the Port is set to Auto and `SystemAuthControl` is set to Enabled; see 6.3), authentication is initiated by the Authenticator PAE on sensing that the operational state of the MAC associated with the Port has transitioned from disabled to enabled. As noted below, if the Authenticator PAE does not receive a response, it will retransmit the authentication request.

A Supplicant PAE may initiate the authentication sequence by sending an EAPOL-Start frame (see 7.5.4).

NOTES

1—On initialization, a Supplicant PAE may not be listening for an authentication initiation during the early portions of the initialization sequence, and consequently, the Supplicant PAE may miss the initiation frame(s) sent by the Authenticator PAE. As a result, if the Supplicant PAE did not initiate authentication itself, an authentication failure could occur purely due to timing issues. A Supplicant PAE that does not receive an authentication initiation frame from the Authenticator PAE on reinitialization may initiate authentication by sending an EAPOL-Start frame.

2—A previously authenticated Supplicant PAE's System may be reinitialized. In this circumstance, an Authenticator PAE might not initiate authentication, because it may not sense a transition in the MAC's operational state. In this case,

it could be argued that authentication is not necessary because the Supplicant PAE has already been authenticated. However, it is possible in some environments to reinitialize a machine, bypass the normal login, and access the Authenticator System's services. To prevent an unauthorized user from accessing the Authenticator by reinitializing an authenticated machine, a Supplicant initiation of authentication is necessary upon reinitialization.

8.4.2.1 Authenticator initiation

The Authenticator PAE will typically initiate the conversation when it receives an indication that the Port has become operable. Before authentication commences, the Port state is forced to the unauthorized state.

If the Supplicant's Identity is not known, then the Authenticator PAE initiates the authentication sequence by sending an EAP-Request/Identity frame. This is typically how an Authenticator PAE will begin the authentication exchange. A Supplicant PAE receiving an EAP-Request frame from the Authenticator PAE responds with an EAP-Response frame.

Authenticator PAEs may support periodic reauthentication, and they may request that a Port reauthenticate at any time. For example, if the Authenticator System reinitializes, the authentication state can be recovered by issuing EAP-Request/Identity frames on all Ports. If a controlled Port is in the authorized state prior to reauthentication, then it will remain in that state during reauthentication. If the authentication fails for a controlled Port that was in the authorized state during reauthentication, then the controlled Port's authorization state is transitioned to unauthorized in order to control external access to that Port in accordance with the current value of the OperControlledDirections parameter (6.4).

8.4.2.2 Supplicant initiation

In order to request that the Authenticator PAE initiate authentication, the Supplicant PAE sends an EAPOL-Start packet (7.5.4). The Authenticator PAE receiving an EAPOL-Start packet responds by sending an EAP-Request/Identity packet.

8.4.3 EAPOL-Logoff

When a Supplicant wishes the Authenticator PAE to perform a logoff (i.e., to set the controlled Port state to unauthorized), the Supplicant PAE originates an EAPOL-Logoff message (7.5.4) to the Authenticator PAE. As a result, the Authenticator PAE immediately places the controlled Port in the unauthorized state.

NOTE—In general, it is advisable for the Supplicant PAE to originate an EAPOL-Logoff in any circumstances in which the user of the Supplicant System has logged off (in the case of an end station), or in which the operation of the Supplicant System has been reconfigured in a manner that would invalidate any previous authentication results (for example, a management change that affects the Supplicant System's identity, or its authorization to use the services of the Authenticator's System).

8.4.4 Timing out authorization state information

Authenticator PAEs can time out the authorization state information on a periodic basis by means of the Reauthentication Timer State Machine (8.5.7). The time period for such timeouts is reAuthPeriod seconds since the last time that the authorization state was confirmed. The state variable reAuthEnabled controls whether periodic reauthentication takes place.

Reauthentication can be enabled and disabled, and the reAuthPeriod modified, by management. The default settings are for the reAuthPeriod to be 3600 s (one hour) and for reauthentication to be disabled.

NOTE—As with Authenticator and Supplicant initiated reauthentication, the implications of setting this to a lower value should be carefully thought out before proceeding. The value chosen will be affected by the reliability with which the MAC associated with the Port can detect and indicate MAC enabled/disabled conditions. If the Port's detection of MAC state is reliable, then longer timeout values may be appropriate.

8.4.5 Retransmission

As noted in the specification of EAP, the Authenticator PAE is responsible for retransmission of messages between the Supplicant PAE and the Authenticator PAE. Thus, if an EAP-Packet message is lost in transit between the Supplicant PAE and the Authenticator PAE (or vice versa), the Authenticator PAE will retransmit. The retransmission strategy takes account of the LAN environment in which this protocol will be used; i.e., there is a low probability of packet loss between Authenticator PAE and Supplicant PAE, and transmission delays are small. A simple timeout is therefore used to drive retransmissions by the Authenticator to the Supplicant and vice-versa, with a default value of 30 s being used to initialize the timer. The exceptions are EAPOL-Start messages, which are retransmitted, if necessary by the Supplicant PAE, and EAP-Failure or EAP-Success messages. Because EAP-Failure and EAP-Success messages are not acknowledged by the Supplicant PAE, they are not retransmitted by the Authenticator PAE. If either an EAP-Failure or EAP-Success message is lost, the Supplicant PAE state machine transitions to the CONNECTING state on authWhile timer expiration.

In implementations in which the backend authentication function is performed by a remote authentication server, retransmissions may be necessary between the Authenticator PAE and the Authentication Server. In this instance, it may be necessary to adopt a retransmission strategy that is more appropriate to the transmission characteristics of the communication path involved.

NOTE—DHCP clients incorporate a randomized exponential backoff algorithm to determine the delay between retransmissions, allowing a retransmission delay of up to 64 s. Assuming that the Authenticator is in blocking mode prior to authentication and initiates authentication on receiving an initial DHCP frame from the end station, the initial DHCP frame will be dropped and will trigger authentication initiation by the Authenticator. Assuming that the authentication can complete in time to avoid a DHCP timeout, the DHCP conversation will complete successfully.

It may be necessary to adjust retransmission strategies and authentication timeouts in certain cases. For example, when a token card is used, additional time may be required to allow the user to find the card and enter the token. If the user takes a particularly long time to find the card, then a DHCP timeout can occur. This problem cannot be ameliorated by enqueueing the initial DHCP frames because DHCP client timers are started when the packets are enqueued, not when they are sent.

8.4.6 Migration considerations

It is desirable that the transition between a nonauthenticated and an authenticated environment be as smooth as possible. For example, when an authentication-capable Supplicant connects to a nonauthentication-capable Bridge, the Supplicant will not receive an EAP-Request/Identity packet. As a result, the Supplicant PAE will initiate an EAPOL-Start frame to the PAE group MAC address. As this address is one of the addresses that are not forwarded by MAC Bridges, the Supplicant PAE will not receive a response. Hence, after a suitable timeout period in which the EAPOL-Start has been retransmitted and no response has been received, the Supplicant can assume that it is authorized to access the Bridged LAN.

As the concept of the controlled and uncontrolled Ports applies only to systems that support Authenticator functionality, a system that only supports Supplicant functionality can attempt to transmit data frames prior to completion of the authentication process. Hence, in the example above, there is no necessity for the Supplicant system to wait for the end of the authentication timeout period before initiating data exchanges via the Bridge Port.

NOTE—DHCP is an example of a process that could take advantage of this approach.

When a nonauthentication-aware Supplicant connects to an authentication enabled Bridge, the Supplicant, having no PAE, will ignore EAP-Request/Identity frames. Consequently, the Port will remain in the unauthorized state. The Supplicant will be able to access the Bridged LAN via the Bridge's controlled Port only in accordance with the value of the OperControlledDirections parameter (6.4); access to any services that are made available via the Bridge's uncontrolled Port are unrestricted.

8.4.7 Relaying EAP frames

The Authenticator PAE is responsible for relaying EAP frames between the Supplicant and the Authentication Server and for performing any repackaging of EAP frames that is necessary in order to convert EAP frames carried as EAPOL between Supplicant PAE and Authenticator PAE into the appropriate format for onward transmission to the Authentication Server, and vice versa. In performing its relay function, the information contained in the EAP frames relayed shall not be modified other than as required to convert the frame format from the EAPOL format into the format necessary for communication with the Authentication Server.

NOTE—EAP in RADIUS (see IETF RFC 2865, IETF RFC 2866, IETF RFC 2869, and Annex D) offers a suitable option for the protocol used between Authenticator and Authentication Server; however, use of RADIUS is not mandated by this standard.

EAPOL-Start and EAPOL-Logoff frames are transmitted by the Supplicant PAE to the Authenticator PAE; EAPOL-Key frames are transmitted by the Authenticator PAE to the Supplicant PAE. These frames shall not be relayed onward by the Authenticator PAE to the Authentication Server.

EAP-Request/Identity frames are transmitted only by the Authenticator PAE to the Supplicant PAE and do not appear in the communication path between the Authentication Server and the Authenticator PAE.

All other EAP frames received from the Supplicant PAE are converted by the Authenticator PAE from their EAPOL format for onward transmission to the Authentication Server in the necessary format for the protocol in use between the Authentication Server and the Authenticator PAE.

All EAP frames received by the Authenticator PAE from the Authentication Server are converted to EAPOL format, as appropriate for the Port concerned, for onward transmission to the Supplicant PAE.

8.4.8 Example EAP exchanges

In these example exchanges, the exchange of EAPOL frames is shown as a solid line; the exchange of EAP frames carried in a higher layer protocol such as RADIUS is shown by a broken line. The diagrams therefore show which PDU exchanges involve the Authenticator PAE in repackaging EAP frames in order to perform protocol translation.

NOTE—For the purposes of these examples, it has been assumed that an EAP-Success is encapsulated in an Accept message from the Authentication Server, and that an EAP-Failure is encapsulated in a Reject message from the Authentication Server. However, it is possible for an Accept or a Reject message to not contain an encapsulated EAP packet, or to contain an EAP packet other than Success or Failure. Because the Backend Authentication state machine is influenced only by the Accept or Reject message, rather than the encapsulated EAP packet, this does not present a problem.

The example in Figure 8-1 shows an Authenticator-initiated conversation for the case of a One Time Password (OTP) authentication. OTP is used only for illustrative purposes; other authentication protocols could also have been used, although they may show somewhat different behavior.

In the case in which the Supplicant fails EAP authentication, the conversation would appear as illustrated in Figure 8-2.

Figure 8-3 illustrates a successful authentication exchange, followed by an explicit logoff requested by the Supplicant.

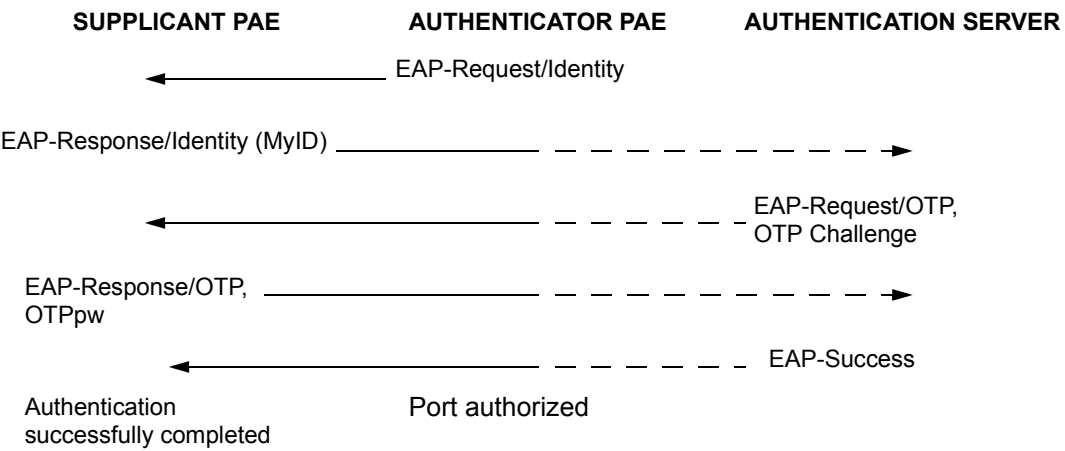


Figure 8-1—Authenticator-initiated, one-time password exchange (success)

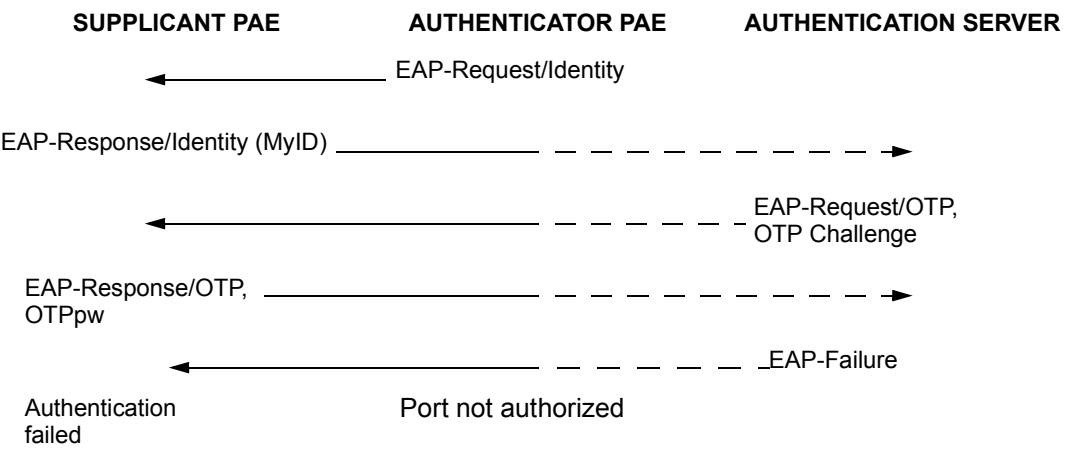


Figure 8-2—Authenticator-initiated, one-time password exchange (failure)

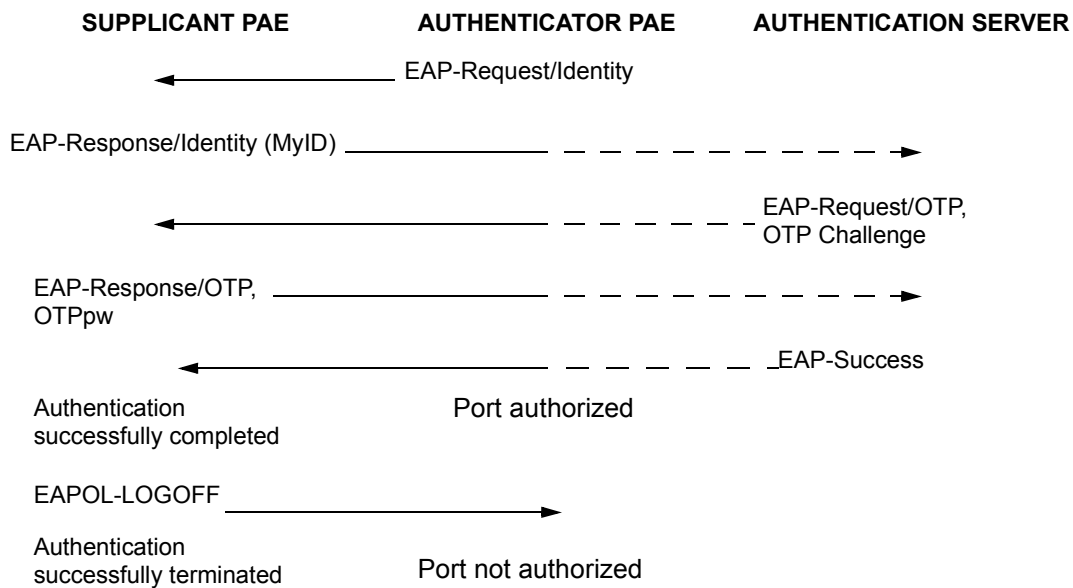


Figure 8-3—Successful authentication followed by Supplicant-initiated logoff

A Supplicant-initiated authentication conversation will appear as illustrated in Figure 8-4.

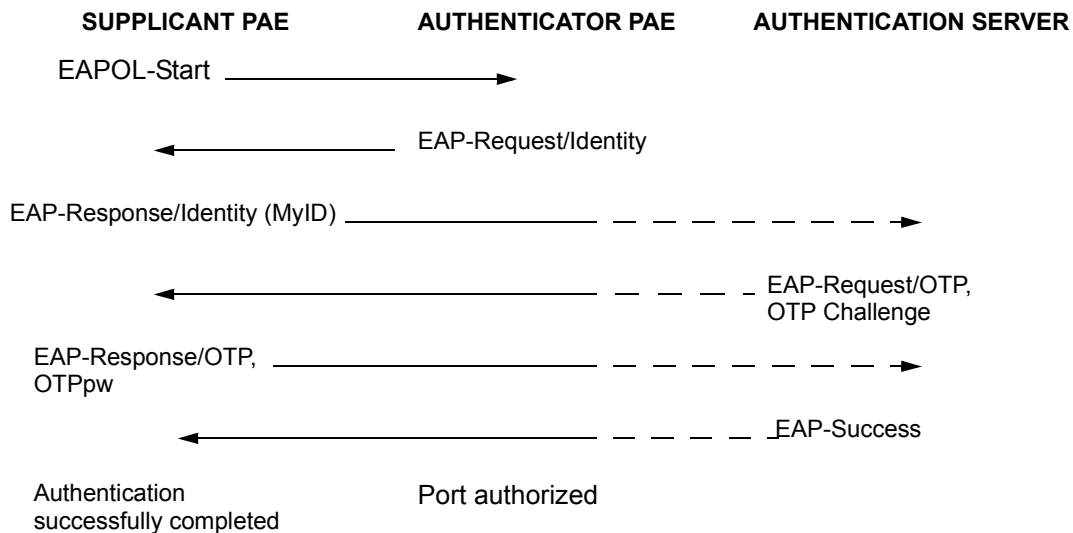


Figure 8-4—Supplicant-initiated, one-time password exchange (success)

In the case in which the Supplicant does not support authentication, but authentication is enabled on that Authenticator, the conversation would appear as illustrated in Figure 8-5.

In the case in which the Authenticator does not support authentication, but authentication is enabled on that Supplicant, the conversation would appear as illustrated in Figure 8-6.

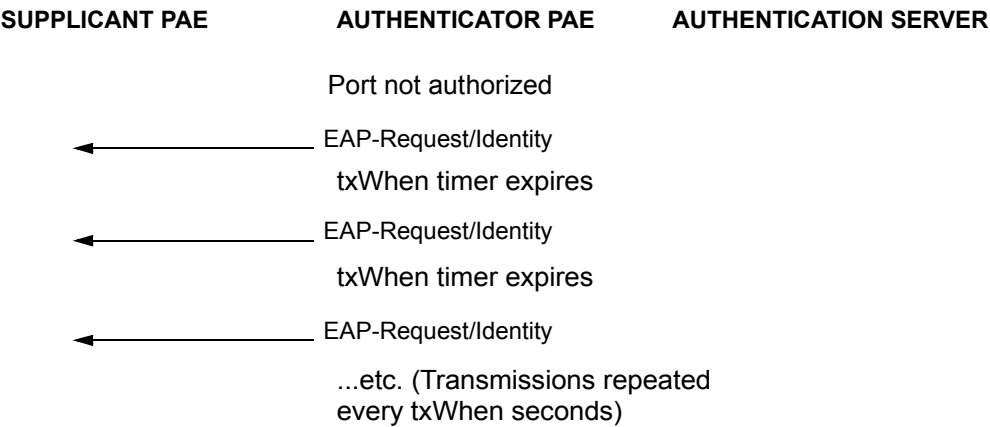


Figure 8-5—Supplicant does not support authentication

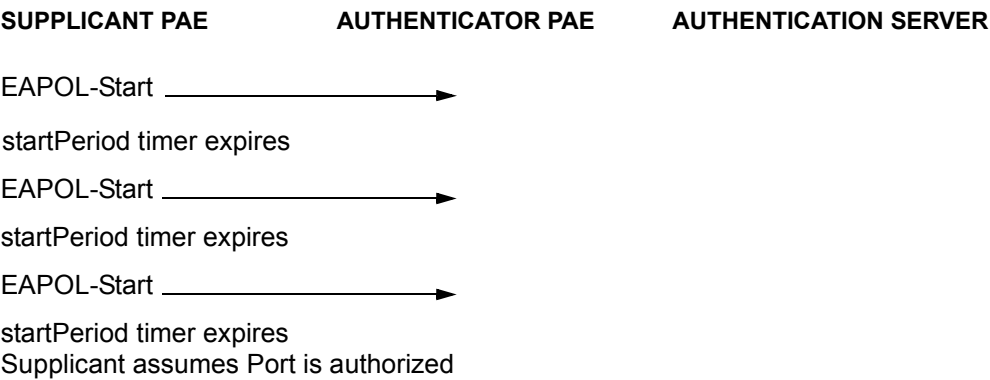


Figure 8-6—Authenticator does not support authentication

8.4.9 Transmission of key information

The EAPOL protocol optionally supports the transmission of global key information from the Authenticator to the Supplicant, or from the Supplicant to the Authenticator, following a successful authentication exchange, in circumstances in which encryption is available between the Supplicant and Authenticator systems (e.g., where encryption is used on an 802.11 association between a station and an access point).

The use of this facility is controlled by the KeyTransmissionEnabled parameter, which may be modified by management. A value of TRUE allows key information to be transmitted following authorization; a value of FALSE disables the key transmission capability. In Supplicant or Authenticator systems that do not support this facility, the KeyTransmissionEnabled parameter is read-only and is set FALSE.

NOTE—This standard provides a mechanism for transmitting global key information; however, it does not specify when, and for what reason, such information is transmitted. The decision as to which key values are transmitted is made externally to the operation of the Authenticator and Supplicant and their associated state machines.

8.5 EAPOL state machines

The following EAPOL state machines are described:

- a) The Port Timers state machine (8.5.3)
- b) The Authenticator PAE state machine (8.5.4)
- c) The Key Transmit state machine (8.5.5)
- d) The Reauthentication Timer state machine (8.5.7)
- e) The Backend Authentication state machine (8.5.8)
- f) The Controlled Directions state machine (8.5.9)
- g) The Supplicant PAE state machine (8.5.10)
- h) The Key Receive state machine (8.5.11)

These state machines are defined on a per-Port basis. Table 8-1 summarizes the state machine support requirements for implementations that support Authenticator functionality, Supplicant functionality, or both. An X marked in a cell of the table indicates that the State Machine named on that row of the table shall be supported in an implementation that claims to support the functionality identified for the column concerned. An O marked in a cell of the table indicates that the State Machine named on that row of the table may be supported in an implementation that claims to support the functionality identified for the column concerned.

Table 8-1—State machine support requirements

State Machine Name	Functionality Supported		
	Authenticator	Supplicant	Both
Port Timers state machine (8.5.3)	X	X	X
Authenticator PAE state machine (8.5.4)	X		X
The Authenticator Key Transmit state machine (8.5.5)	O		O
The Supplicant Key Transmit state machine (8.5.6)		O	O
Reauthentication Timer state machine (8.5.7)	X		X
Backend Authentication state machine (8.5.8)	X		X
Controlled Directions state machine (8.5.9)	X		X
Supplicant PAE state machine (8.5.10)		X	X
The Key Receive state machine (8.5.11)	X	X	X

8.5.1 Notational conventions used in state diagrams

State diagrams are used to represent the operation of a function as a group of connected, mutually exclusive states. Only one state of a function 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 uppercase 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. 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.

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. 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 such a time as one of the conditions is met. All exit conditions are regarded as Boolean expressions that evaluate to True or False; if a condition evaluates to True, then the condition is met. When the condition associated with a global transition is met, it supersedes all other exit conditions, including UCT. The label UCT denotes an unconditional transition (i.e., UCT always evaluates to True). 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).

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.

Where it is necessary to segment 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 segmented 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 8-2; these symbols and operators are derived from the notation of the “C” programming language, ANSI X3.159.

Table 8-2—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.
{statement 1, ... statement N}	Compound statement. Braces are used to group statements that are executed together as if they were a single statement.
!=	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.
<	Less than. Evaluates to TRUE if the value of the expression to the left of the operator is less than the value of the expression to the right.
>	Greater than. Evaluates to TRUE if the value of the expression to the left of the operator is greater than the value of the expression to the right.
>=	Greater than or equal to. Evaluates to TRUE if the value of the expression to the left of the operator is either greater than or equal to the value of the expression to the right.
+	Arithmetic addition operator.
–	Arithmetic subtraction operator.

8.5.2 Timers and global variables used in the definition of the state machines

8.5.2.1 Timers

The timers defined for these state machines are decremented, if their value is nonzero, by the operation of the Port Timers state machine (8.5.3). All timers used by the EAPOL state machines have a resolution of one second; i.e., the initial values used to start the timers are integer values, and they represent the timer period as an integral number of seconds.

NOTE—It is permissible to introduce a degree of jitter into the initialization of these timers; for example, in order to distribute the timing of EAPOL frame transmissions among Ports in multi-Port implementations.

- a) **authWhile.** A timer used by the Supplicant PAE to determine how long to wait for a response from the Authenticator before timing it out. The initial value of this timer is authPeriod.
- b) **aWhile.** A timer used by the Backend Authentication state machine in order to determine timeout conditions in the exchanges between the Authenticator and the Supplicant or Authentication Server. The initial value of this timer is either suppTimeout or serverTimeout, as determined by the operation of the Backend Authentication state machine (see 8.5.8).
- c) **heldWhile.** A timer used by the Supplicant state machine to define periods of time during which it will not attempt to acquire an Authenticator. The initial value of this timer is heldPeriod.
- d) **quietWhile.** A timer used by the Authenticator state machine to define periods of time during which it will not attempt to acquire a Supplicant. The initial value of this timer is quietPeriod.
- e) **reAuthWhen.** A timer used by the Reauthentication Timer state machine to determine when reauthentication of the Supplicant takes place. The initial value of this timer is reAuthPeriod.
- f) **startWhen.** A timer used by the Supplicant PAE state machine to determine when an EAPOL-StartPDU is to be transmitted. The initial value of this timer is startPeriod.
- g) **txWhen.** A timer used by the Authenticator PAE state machine to determine when an EAPOL PDU is to be transmitted. The initial value of this timer is txPeriod.

8.5.2.2 Global variables

Global variables are available for use by more than one state machine and are used to perform interstate-machine communication and initialization functions.

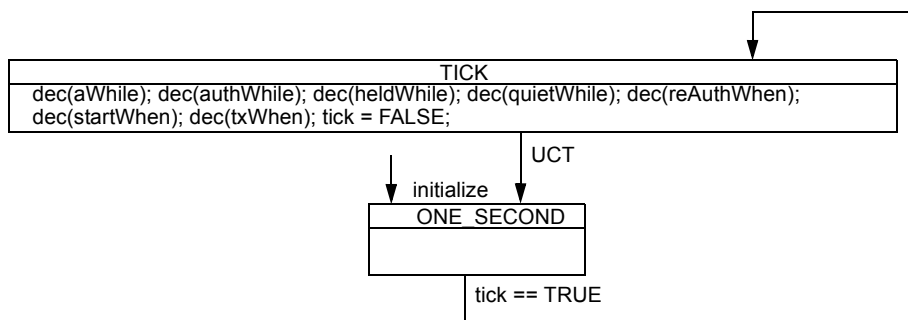
- a) **authAbort.** This variable is set TRUE by the Authenticator PAE state machine in order to signal to the Backend Authentication state machine to abort its authentication procedure. Its value is set FALSE by the Backend Authentication state machine once the authentication procedure has been aborted.
- b) **authFail.** This variable is set TRUE if the authentication process (represented by the Backend Authentication state machine) fails. It is set FALSE by the operation of the Authenticator PAE state machine, prior to initiating authentication.
- c) **authStart.** This variable is set TRUE by the Authenticator PAE state machine in order to signal to the Backend Authentication state machine to start its authentication procedure. Its value is set FALSE by the Backend Authentication state machine once the authentication procedure has been started.
- d) **authTimeout.** This variable is set TRUE if the authentication process (represented by the Backend Authentication state machine) fails to obtain a response from the Supplicant. The variable may be set by management action, or by the operation of a timeout while in the AUTHENTICATED state. This variable is set FALSE by the operation of the Authenticator PAE state machine.
- e) **authSuccess.** This variable is set TRUE if the authentication process (represented by the Backend Authentication state machine) succeeds. It is set FALSE by the operation of the Authenticator PAE state machine, prior to initiating authentication.
- f) **currentId.** The integer in the range 0 through 255 that the Authenticator is using to identify its current authentication session.
- g) **initialize.** This variable is externally controlled. When asserted, it forces all EAPOL state machines to their initial state. The EAPOL state machines are held in their initial state until initialize is deasserted.

- h) **portControl**. This variable is derived from the current values of the *AuthControlledPortControl* and *SystemAuthControl* parameter (6.3) for the Port. This variable can take the following values:
- 1) **ForceUnauthorized**. The controlled Port is required to be held in the Unauthorized state.
 - 2) **ForceAuthorized**. The controlled Port is required to be held in the Authorized state.
 - 3) **Auto**. The controlled Port is set to the Authorized or Unauthorized state in accordance with the outcome of an authentication exchange between the Supplicant and the Authentication Server.
- If SystemAuthControl is set to Enabled, then portControl directly reflects the value of the AuthControlledPortControl parameter. If SystemAuthControl is set to Disabled, then the value of portControl is ForceAuthorized.
- i) **portEnabled**. This variable is externally controlled. Its value reflects the operational state of the MAC service supporting the Port. Its value is TRUE if the MAC service supporting the Port is in an operable condition (see 6.3), and it is otherwise FALSE.
- j) **portStatus**. The current authorization state of the controlled Port. This variable is set to Unauthorized or Authorized by the operation of the Authenticator PAE state machine. The value of portStatus directly determines the value of the *AuthControlledPortStatus* parameter (6.3) for the Port.
- k) **reAuthenticate**. This variable is set TRUE by the Reauthentication Timer state machine on expiry of the reAuthWhen timer. This variable may also be set TRUE by management action. It is set FALSE by the operation of the Authenticator PAE state machine.
- l) **receivedId**. The value of the Identifier field carried in the EAP Request frame most recently received by the Supplicant, in the range 0–255.
- m) **suppStatus**. This variable's value is controlled by the Supplicant PAE state machine; it indicates Authorized or Unauthorized in accordance with the outcome of the most recent authentication exchange that has taken place with the Authenticator. Its value is initialized to the Unauthorized state. This variable is used to control the operation of the Supplicant Key Transmit state machine.

8.5.3 Port Timers state machine

The Port Timers state machine for a given Port is responsible for decrementing the timer variables for that Port each second, in response to an external system clock function. The timer variables are used, and set to their initial values, by the operation of the individual state machines for the Port.

The Port Timers state machine shall implement the function specified by the state diagram contained in Figure 8-7 and the attendant timer definitions contained in 8.5.2.1.



The following abbreviation is used in this diagram
dec(x): {if (x != 0) then x = x-1;}

Figure 8-7—Port Timers state machine