IEEE STANDARDS ASSOCIATION

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

Amendment 1: Fast Initial Link Setup

IEEE Computer Society

Sponsored by the
LAN/MAN Standards Committee

IEEE
3 Park Avenue
New York, NY 10016-5997
USA

IEEE Std 802.11ai™-2016
(Amendment to
IEEE Std 802.11™-2016)

IEEE Std 802.11ai™-2016
(Amendment to
IEEE Std 802.11™-2016)

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

Amendment 1: Fast Initial Link Setup

SECOND PRINTING: 14 April 2017

Sponsor

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

Approved 7 December 2016

IEEE-SA Standards Board

Abstract: Mechanisms that provide IEEE Std 802.11 networks with fast initial link setup methods that do not degrade the security offered by Robust Security Network Association (RSNA) already defined in IEEE Std 802.11 are defined in this amendment.

Keywords: amendment, Fast Initial Link setup, FILS, IEEE 802.11ai™

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

SECOND PRINTING: 14 April 2017. Erratum included in Clause 9—Table 9-46 and Table 9-47.

PDF: ISBN 978-1-5044-3631-1 STD22359
Print: ISBN 978-1-5044-3632-8 STDPD22359

Important Notices and Disclaimers Concerning IEEE Standards Documents

IEEE documents are made available for use subject to important notices and legal disclaimers. These notices and disclaimers, or a reference to this page, appear in all standards and may be found under the heading “Important Notices and Disclaimers Concerning IEEE Standards Documents.” They can also be obtained on request from IEEE or viewed at http://standards.ieee.org/IPR/disclaimers.html.

Notice and Disclaimer of Liability Concerning the Use of IEEE Standards Documents

IEEE Standards documents (standards, recommended practices, and guides), both full-use and trial-use, are developed within IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (“IEEE-SA”) Standards Board. IEEE (“the Institute”) develops its standards through a consensus development process, approved by the American National Standards Institute (“ANSI”), which brings together volunteers representing varied viewpoints and interests to achieve the final product. IEEE Standards are documents developed through scientific, academic, and industry-based technical working groups. Volunteers in IEEE working groups are not necessarily members of the Institute and participate without compensation from IEEE. While IEEE administers the process and establishes rules to promote fairness in the consensus development process, IEEE does not independently evaluate, test, or verify the accuracy of any of the information or the soundness of any judgments contained in its standards.

IEEE Standards do not guarantee or ensure safety, security, health, or environmental protection, or ensure against interference with or from other devices or networks. Implementers and users of IEEE Standards documents are responsible for determining and complying with all appropriate safety, security, environmental, health, and interference protection practices and all applicable laws and regulations.

IEEE does not warrant or represent the accuracy or content of the material contained in its standards, and expressly disclaims all warranties (express, implied and statutory) not included in this or any other document relating to the standard, including, but not limited to, the warranties of: merchantability; fitness for a particular purpose; non-infringement; and quality, accuracy, effectiveness, currency, or completeness of material. In addition, IEEE disclaims any and all conditions relating to: results; and workmanlike effort. IEEE standards documents are supplied “AS IS” and “WITH ALL FAULTS.”

Use of an IEEE standard is wholly voluntary. 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.

In publishing and making its standards available, IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity nor is IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing any IEEE Standards document, should rely upon his or her own independent judgment in the exercise of reasonable care in any given circumstances or, as appropriate, seek the advice of a competent professional in determining the appropriateness of a given IEEE standard.

IN NO EVENT SHALL IEEE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO: PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE PUBLICATION, USE OF, OR RELIANCE UPON ANY STANDARD, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE AND REGARDLESS OF WHETHER SUCH DAMAGE WAS FORESEEABLE.

Translations

The IEEE consensus development process involves the review of documents in English only. In the event that an IEEE standard is translated, only the English version published by IEEE should be considered the approved IEEE standard.

Official statements

A statement, written or oral, that is not processed in accordance with the IEEE-SA Standards Board Operations Manual shall not be considered or inferred to be the official position of IEEE or any of its committees and shall not be considered to be, or be relied upon as, a formal position of IEEE. At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE standards shall make it clear that his or her views should be considered the personal views of that individual rather than the formal position of IEEE.

Comments on standards

Comments for revision of IEEE Standards documents are welcome from any interested party, regardless of membership affiliation with IEEE. However, IEEE does not provide consulting information or advice pertaining to IEEE Standards documents. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Since IEEE standards represent a consensus of concerned interests, it is important that any responses to comments and questions also receive 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 comments or questions except in those cases where the matter has previously been addressed. For the same reason, IEEE does not respond to interpretation requests. Any person who would like to participate in revisions to an IEEE standard is welcome to join the relevant IEEE working group.

Comments on standards should be submitted to the following address:

Secretary, IEEE-SA Standards Board
445 Hoes Lane
Piscataway, NJ 08854 USA

Laws and regulations

Users of IEEE Standards documents should consult all applicable laws and regulations. Compliance with the provisions of any IEEE Standards document does not imply compliance to any applicable regulatory requirements. Implementers of the standard are responsible for observing or referring to the applicable regulatory requirements. IEEE does not, by the publication of its standards, intend to urge action that is not in compliance with applicable laws, and these documents may not be construed as doing so.

Copyrights

IEEE draft and approved standards are copyrighted by IEEE under U.S. and international copyright laws. They are made available by IEEE and are adopted for a wide variety of both public and private uses. These include both use, by reference, in laws and regulations, and use in private self-regulation, standardization, and the promotion of engineering practices and methods. By making these documents available for use and adoption by public authorities and private users, IEEE does not waive any rights in copyright to the documents.

Photocopies

Subject to payment of the appropriate fee, IEEE will grant users a limited, non-exclusive license to photocopy portions of any individual standard for company or organizational internal use or individual, non-commercial use only. To arrange for payment of licensing fees, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

Updating of IEEE Standards documents

Users of IEEE Standards documents should be aware that these documents may be superseded at any time by the issuance of new editions or may be amended from time to time through the issuance of amendments, corrigenda, or errata. An official IEEE document at any point in time consists of the current edition of the document together with any amendments, corrigenda, or errata then in effect.

Every IEEE standard is subjected to review at least every ten years. When a document is more than ten years old and has not undergone a revision process, 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 order to determine whether a given document is the current edition and whether it has been amended through the issuance of amendments, corrigenda, or errata, visit the IEEE-SA Website at http://ieeexplore.ieee.org or contact IEEE at the address listed previously. For more information about the IEEE SA or IEEE's standards development process, visit the IEEE-SA Website at http://standards.ieee.org.

Errata

Errata, if any, for all IEEE standards can be accessed on the IEEE-SA Website at the following URL: http://standards.ieee.org/findstds/errata/index.html. Users are encouraged to check this URL for errata periodically.

Patents

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 by the IEEE with respect to the existence or validity of any patent rights in connection therewith. If a patent holder or patent applicant has filed a statement of assurance via an Accepted Letter of Assurance, then the statement is listed on the IEEE-SA Website at http://standards.ieee.org/about/sasb/patcom/patents.html. Letters of Assurance may indicate whether the Submitter is willing or unwilling to grant licenses under patent rights without compensation or under reasonable rates, with reasonable terms and conditions that are demonstrably free of any unfair discrimination to applicants desiring to obtain such licenses.

Essential Patent Claims may exist for which a Letter of Assurance has not been received. The IEEE is not responsible for identifying Essential Patent Claims for which a license may be required, for conducting inquiries into the legal validity or scope of Patents Claims, or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. Further information may be obtained from the IEEE Standards Association.

Participants

At the time this standard was completed, the IEEE 802.11 Working Group had the following membership:

Adrian P. Stephens, *Chair*
Jon W. Rosdahl, *1st Vice Chair*
Dorothy V. Stanley, *2nd Vice Chair*
Stephen McCann, *Secretary*

The following were officers of Task Group ai:

Hiroshi Mano, *Chair*
Marc Emmelmann, *Vice Chair*
Hitoshi Morioka, *Secretary*
Lee R. Armstrong, *Co-Technical Editor*
Ping Fang, *Co-Technical Editor*

Osama S. Aboulmagd
Santosh P. Abraham
Roberto Aiello
Thomas Alexander
Peiman Amini
Sirikiat Lek Ariyavisitakul
Yusuke Asai
Alex Ashley
Kwok Shum Au
Vijay Auluck
Stefan Aust
David Bagby
Eugene Baik
Gabor Bajko
Raja Banerjea
Phillip Barber
Anuj Batra
Tuncer Baykas
Alan Berkema
Nehru Bhandaru
Philippe Boucachard
Andre Bourdoux
John Buffington
Lin Cai
George Calcev
Chris Calvert
Radhakrishna Canchi
Laurent Cariou
William Carney
Jaesun Cha
Romana Challans
Kim Chang
Kuor-Hsin Chang
Xin Chang
Clint F. Chaplin
Bin Chen
Jiamin Chen
Jixin Chen
Lidong Chen
Qian Chen
Xi Chen
Minho Cheong
George Cherian
Francois Chin
Rojan Chitrakar
Jinsoo Choi
Sangsung Choi
Li Chia Chia Choo
Sayantan Choudhury
Liwen Chu
Jinyoung Chun
John Coffey
Kenneth Coop
Carlos Cordeiro
Neiyer Correal
Subir Das
Hendricus De Ruijter
Rolf J. de Vegt
Yohannes Demessie
Michael Denson
Ting Dong
Xiandong Dong
Klaus Doppler
Roger P. Durand
Donald E. Eastlake
Peter Ecclesine
Richard Edgar
Amal Ekbal
Vinko Erceg
Yonggang Fang
Qin Fei
Stanislav Filin
Norman Finn
Matthew J. Fischer
George Flammer
Chittabrata Ghosh
James P. K. Gilb
Reinhard Gloger
Daning Gong
David Goodall
Elad Gottlib
Sudheer A. Grandhi
Stephen Grau
Michael Grigat
David Halasz
Mark A. Hamilton
Christopher J. Hansen
Peng Hao
Hiroshi Harada
Daniel N. Harkins
Brian D. Hart
Ahmadreza Hedayat
Robert F. Heile
Jerome Henry
Chin Keong Ho
Anh Tuan Hoang
Dien Hoang
Wei Hong
Ying-Chuan Hsiao
Jing-Rong Hsieh
David Hunter
Yasuhiko Inoue
Mitsuru Iwaoka
Wuncheol Jeong
Yangseok Jeong
Sunggeun Jin
ZhongYi Jin
Nihar Jindal
V. K. Jones
Jari Junell
Padam Kafle
Carl W. Kain
Hyunduk Kang
Mika Kasslin
Richard H. Kennedy
Stuart J. Kerry
Eunkyung Kim
Jeongki Kim
Jinho Kim
Joo Young Kim
Joonsuk Kim
Suhwook Kim
Taejoon Kim
Youhan Kim
Youngsoo Kim
Shoichi Kitazawa
Jarkko Kneckt
Gwangzeen Ko
Fumihide Kojima
Tom Kolze
Timo Koskela
Bruce P. Kraemer
Jin-Sam Kwak
Joseph Kwak
Hyoungjin Kwon

Young Hoon Kwon
Paul Lambert
Zhou Lan
Leonardo Lanante
James Lansford
Jean-Pierre Le Rouzic
Anseok Lee
Donghun Lee
Jae Seung Lee
Wookbong Lee
Zhongding Lei
Wai Kong Leung
Joseph Levy
Feng Li
Huan-Bang Li
Liang Li
Lingjie Li
Yunbo Li
Yunzhou Li
Zhiqiang Li
Erik Lindskog
Jianhan Liu
Pei Liu
Yong Liu
Zongru Liu
Peter Loc
Su Lu
Long Luo
Yi Luo
Zhendong Luo
Kaiying Lv
Michael Lynch
Jouni K. Malinen
Simone Merlin
James Miller
Keiichi Mizutani
Apurva Mody
Michael Montemurro
Kenichi Mori
Ronald Murias
Andrew Myles
Yukimasa Nagai
Yuhei Nagao
Hiroki Nakano
Chiu Ngo
Paul Nikolich
Hiroyo Ogawa
Minseok Oh
Min-seok Oh
David Olson
Satoshi Oyama
Michael J. Paljug
Santos Ghanshyam Pandey
Anna Pantelidou
Giwon Park
Minyoung Park
Seung-Hoon Park
Jaya Shankar Pathmasuntharam
Sandhya Patil
Xiaoming Peng
Eldad Perahia
James E. Petranovich

Albert Petrick
John Petro
Xu Ping
Juho Pirskanen
Khiam Boon Png
Vishakan Ponnampalam
Ron Porat
Henry S. Ptasinski
Rethnakaran Pulikkoonattu
Chang-Woo Chang Pyo
Emily H. Qi
Huyu Qu
Harish Ramamurthy
Jayaram Ramasastry
Ivan Reede
Edward Reuss
Maximilian Riegel
Mark Rison
Zhigang Rong
Jon W. Rosdahl
Cheol Ryu
Kiseon Ryu
Kazuyuki Sakoda
Ruben E. Salazar Cardozo
Hemanth Sampath
Sigurd Schelstraete
Jean Schwoerer
Jonathan Segev
Cristina Seibert
Yongho Seok
Kunal Shah
Huairong Shao
Zhenhai Shao
Stephen J. Shellhammer
Ian Sherlock
Wei Shi
Nobuhiko Shibagaki
Shusaku Shimada
Chang Sub Shin
Thomas M. Siep
Michael Sim
Dwight Smith
Graham Kenneth Smith
Myung Sun Song
Sudhir Srinivasa
Robert Stacey
Dorothy V. Stanley
Lawrence Stefani
Adrian P. Stephens
Rene Struik
Jung Hoon Suh
Chin-sean Sum
Bo Sun
Chen Sun
Sheng Sun
Kazuaki Takahashi
Mineo Takai
Sagar Tamhane
Joseph Teo
Thomas Tetzlaff
Jerry Thrasher
Tong Tian

Jens Tingleff
Fei Tong
Ha Nguyen Tran
Kazuyoshi Tsukada
Masahiro Umehira
Richard D. J. Van Nee
Allert Van Zelst
Prabodh Varshney
Sameer Vermani
Dalton T. Victor
Gabriel Villardi
George A Vlantis
Chao Chun Wang
Haiguang Wang
Haiming Wang
James June Wang
Lei Wang
Lin Wang
Qi Wang
Xiang Wang
Xuehuan Wang
Lisa Ward
Zou Wei-Xia
Lei Wen
Menzo M. Wentink
Harya Wicaksana
Eric Wong
Harry R. Worstell
Tianyu Wu
Zhanji Wu
Zhenyu Xiao
Dongmei Xu
Quanping Xu
Guang-Qi Yang
Lin Yang
Xun Yang
Yunsong Yang
Fan Ye
James Yee
Peter Yee
Wai-Leong Yeow
Kaoru Yokoo
Su Khiong Khiong Yong
Christopher Young
Heejung Yu
Zhan Yu
Tevfik Yucek
Guangrong Yue
Katsuo D. A. Yunoki
Hongyuan Zhang
Hui Zhang
Junjian Zhang
Nianzu Zhang
Xin Zhang
Mu Zhao
Jun Zheng
Shoukang Zheng
Mingtuo Zhou
Yan Zhuang
Lan Zhuo

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

Tomoko Adachi
Thomas Alexander
Richard Alfvin
Nobumitsu Amachi
Carol Ansley
Butch Anton
Yusuke Asai
Alfred Asterjadhi
Stefan Aust
Gabor Bajko
Phillip Barber
Harry Bims
Gennaro Boggia
Nancy Bravin
Jairo Bustos Heredia
William Byrd
Radhakrishna Canchi
Cagatay Capar
William Carney
Juan Carreon
Minho Cheong
Paul Chiuchiolo
Sayantan Choudhury
Keith Chow
Charles Cook
Subir Das
Patrick Diamond
Yezid Donoso
Malcolm Dowse
Sourav Dutta
Richard Edgar
Marc Emmelmann
Michael Fischer
Avraham Freedman
Devon Gayle
Joel Goergen
Randall Groves
Michael Gundlach
Gloria Gwynne
Russell Haines
Mark Hamilton
Daniel Harkins
Jerome Henry
Marco Hernandez
Guido Hiertz
Werner Hoelzl
Russell Housley
Noriyuki Ikeuchi
Yasuhiko Inoue
Akio Iso
Atsushi Ito
Raj Jain
Adri Jovin
Naveen Kakani
Shinkyo Kaku
Hyunjeong Kang
Piotr Karocki
John Kenney
Stuart Kerry
Youhan Kim
Patrick Kinney
Bruce Kraemer
Yasushi Kudoh
Thomas Kurihara
Paul Lambert
Jeremy Landt
Hyeong Ho Lee
Zhongding Lei
James Lepp
Joseph Levy
Arthur H. Light
William Lumpkins
Michael Lynch
Chris Lyttle
Elvis Maculuba
Jouni Malinen
Hiroshi Mano
James Marin
Stephen McCann
Michael McInnis
Filip Mestanov
Michael Montemurro
Jose Morales
Ronald Murias
Rick Murphy
Andrew Myles
Michael Newman
Nick S.A Nikjoo
John Notor
Satoshi Obara
Robert O'Hara
Yoshihiro Ohba
Satoshi Oyama
Stephen Palm
Arumugam Paventhan
Venkatesha Prasad
Karen Randall
Maximilian Riegel
Mark Rison
Robert Robinson
Benjamin Rolfe
Jon W. Rosdahl
Osman Sakr
Shigenobu Sasaki
Naotaka Sato
Bartien Sayogo
Andy Scott
Yongho Seok
Ian Sherlock
Graham Smith
Daniel Smolinski
Ju-Hyung Son
Kapil Sood
Thomas Starai
Adrian P. Stephens
Rene Struik
Walter Struppler
Michael Swearingen
Payam Torab
Kazuyoshi Tsukada
Mark-Rene Uchida
Lorenzo Vangelista
Dmitri Varsanofiev
Prabodh Varshney
Ganesh Venkatesan
George Vlantis
Khurram Waheed
Haiming Wang
James June Wang
Lei Wang
Xiaofei Wang
Hung-Yu Wei
James Yee
Oren Yuen

When the IEEE-SA Standards Board approved this standard on 7 December 2016, it had the following membership:

Jean-Philippe Faure, *Chair*
Ted Burse, *Vice Chair*
John D. Kulick, *Past Chair*
Konstantinos Karachalios, *Secretary*

Chuck Adams
Masayuki Ariyoshi
Stephen Dukes
Jianbin Fan
J. Travis Griffith
Gary Hoffman
Ronald W. Hotchkiss
Michael Janezic
Joseph L. Koepfinger*
Hung Ling
Kevin Lu
Annette D. Reilly
Gary Robinson
Mehmet Ulema
Yingli Wen
Howard Wolfman
Don Wright
Yu Yuan
Daidi Zhong

*Member Emeritus

Introduction

This introduction is not part of IEEE Std 802.11ai-2016, 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—Amendment 1: Fast Initial Link Setup.

This amendment defines mechanisms that provide IEEE 802.11 networks with fast initial link setup methods that do not degrade the security offered by Robust Security Network Association (RSNA) already defined in IEEE Std 802.11.

Contents

Tables

Figures

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

Amendment 1: Fast Initial Link Setup

NOTE—The editing instructions contained in this amendment define how to merge the material contained therein into the existing base standard and its amendments to form the comprehensive standard.

The editing instructions are shown in ***bold italic***. Four editing instructions are used: ***change, delete, insert,*** and ***replace***. ***Change*** is used to make corrections in existing text or tables. The editing instructions specify the location of the change and describes what is being changed by using ~~strikethrough~~ (to remove old material) and underscore (to add new material). ***Delete*** removes existing material. ***Insert*** adds new material without disturbing the existing material. Insertions may require renumbering. If so, renumbering instructions are given in the editing instruction. ***Replace*** is used to make changes in figures or equations by removing the existing figure or equation and replacing it with a new one.

2. Normative references

Insert the following references into Clause 2 in alphanumeric order:

FIPS 186-4, Digital Signature Standard (DSS).

IETF RFC 2131, Dynamic Host Configuration Protocol, March 1997.

IETF RFC 3279, Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, April 2002.

IETF RFC 3315, Dynamic Host Configuration Protocol for IPv6 (DHCPv6), July 2003.

IETF RFC 3447, Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specification Version 2.1, February 2003.

IETF RFC 3490, Internationalizing Domain Names in Applications (IDNA), March 2003.

IETF RFC 4862, IPv6 Stateless Address Autoconfiguration, September 2007.

IETF RFC 5116, An Interface and Algorithms for Authenticated Encryption, January 2008.

IETF RFC 5280, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, May 2008.

IETF RFC 5295, Specification for the Derivation of Root Keys from an Extended Master Session Key (EMSK), August 2008.

IETF RFC 5480, Elliptic Curve Cryptography Subject Public Key Information, March 2009.

IETF RFC 6696, EAP Extensions for the EAP Re-authentication Protocol (ERP), July 2012.

IETF RFC 6942, Diameter Support for the EAP Re-authentication Protocol (ERP), May 2013.

ISO/IEC 14888-3:2006, Information technology - Security techniques-Digital signatures with appendix-Part 3: Discrete logarithm based mechanisms.

NIST Special Publication 800-56A R2, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography, May 2013.

3. Definitions, acronyms, and abbreviations

3.1 Definitions

Insert the following new definitions into 3.1 in alphabetic order:

authenticated encryption with associated data (**AEAD):** A cipher mode that performs authenticated encryption of a plain text, with associated data that is authenticated but not encrypted.

certificate authority (CA): An entity that vouches for the binding between a device's identity, its public key, and associated keying material (such as key validity period and key usage).

perfect forward secrecy (PFS): A property of a key agreement protocol that protects a session key derived from a set of long-term public and private keys from being compromised if one of the (long-term) private keys is compromised in the future.

trusted third party (TTP): An entity that is relied upon to vouch for two parties in a pairwise authentication protocol.

3.2 Definitions specific to IEEE Std 802.11

Insert the following new definitions into 3.2 in alphabetic order:

Extensible Authentication Protocol (EAP) reauthentication protocol (EAP-RP): A protocol, using the EAP framework, that allows single round trip reauthentication with an Authentication Server after an initial EAP authentication.

NOTE—IETF RFC 6696 uses "ERP" for the abbreviation of EAP reauthentication protocol; whereas, IEEE Std 802.11 uses "EAP-RP" because "ERP" stands for "Extended Rate PHY" in IEEE Std 802.11.

fast initial link setup (FILS): A collection of mechanisms that enable IEEE Std 802.11 networks to minimize initial link setup time.

fast initial link setup category (FILSC): A value that indicates the priority category of the station (STA) for fast initial link setup.

fast initial link setup station (FILS STA): A station that implements fast initial link setup (FILS) and for which dot11FILSActivated is true.

fast initial link setup (FILS) association: A type of association used in fast initial link setup.

fast initial link setup (FILS) authentication: A type of authentication used in fast initial link setup.

integrity check key (ICK): A key used to integrity check FILS Authentication frames.

link setup: The process of discovering an extended service set (ESS), (secure) association and authentication, and gaining the ability to send higher layer [e.g., Internet Protocol (IP)] traffic with a valid higher layer address through an access point (AP).

upstream network: An integrated local area network (LAN) to which an access point (AP) is connected through a portal.

Change the following definitions in 3.2:

group temporal key security association (GTKSA): The context resulting from a successful group temporal key (GTK) distribution exchange via either a group key handshake, ~~or~~ a 4-way handshake, or fast initial link setup (FILS) authentication.

pairwise transient key security association (PTKSA): The context resulting from a successful 4-way handshake between a peer and Authenticator or from a successful fast initial link setup (FILS) authentication.

pre-robust security network association (pre-RSNA): The type of association used by a pair of stations (STAs) if the procedure for establishing authentication or association between them did not include the 4-way handshake, was not fast initial link setup (FILS) authentication, and did not use FT protocol.

robust security network association (RSNA): The type of association used by a pair of stations (STAs) if the procedure to establish authentication or association between them includes the 4-way handshake or FT protocol, or is fast initial link setup (FILS) authentication. Note that existence of an RSNA between two STAs does not of itself provide robust security. Robust security is provided when all STAs in the network use RSNAs.

robust security network association (RSNA) key management: Key management that includes the 4-way handshake, the group key handshake, authenticated mesh peering exchange, mesh group key handshake, and the PeerKey handshake. If fast basic service set (BSS) transition (FT) is enabled, the FT 4-way handshake and FT authentication sequence are also included. If fast initial link setup (FILS) is enabled, FILS authentication is also included.

3.4 Abbreviations and acronyms

Insert the following new abbreviations/acronyms into 3.4:

AEAD	authenticated encryption with associated data
ANO	access network options
AP-CSN	AP configuration sequence number
CA	certificate authority
CAG	Common Advertisement Group
EAP-RP	EAP reauthentication protocol
FD	FILS discovery
FILS	fast initial link setup
FILSC	fast initial link setup category
HLP	higher layer protocol
ICK	FILS integrity check key
PFS	perfect forward secrecy
TTP	trusted third party

4. General description

4.5 Overview of the services

4.5.3 Services that support the distribution service and the PCP service

4.5.3.3 Association

Change the last paragraph of 4.5.3.3 as follows:

A STA learns what APs are present and what operational capabilities are available from each of those APs and then invokes the association service to establish an association. A FILS STA is able to discover, authenticate and associate with the AP with reduced number of frame transmissions. For details of how a STA learns about what APs are present, see 11.1.4.

4.5.4 Access control and data confidentiality services

4.5.4.2 Authentication

Change 4.5.4.2 as follows:

IEEE 802.11 authentication operates at the link level between IEEE 802.11 STAs. IEEE Std 802.11 does not provide either end-to-end (MSDU origin to MSDU destination) or user-to-user authentication.

IEEE Std 802.11 attempts to control LAN access via the authentication service. IEEE 802.11 authentication is an SS. This service might be used by all STAs to establish their identity to STAs with which they communicate, in both ESSs and IBSSs. If a mutually acceptable level of authentication has not been established between two STAs, an association is not established.

IEEE Std 802.11 defines five~~four~~ IEEE 802.11 authentication methods: Open System authentication, Shared Key authentication, FT authentication, ~~and~~ simultaneous authentication of equals (SAE), and FILS authentication. Open System authentication admits any STA to the DS. Shared Key authentication relies on WEP to demonstrate knowledge of a WEP encryption key. FT authentication relies on keys derived during the initial mobility domain association to authenticate the stations as defined in Clause 13. SAE authentication uses finite field cryptography to prove knowledge of a shared password. FILS authentication allows for faster connection to the network for FILS non-AP STAs by providing authentication, association, and key confirmation information in an efficient number of frame exchanges (see 4.10.3.6). The IEEE 802.11 authentication mechanism also allows definition of new authentication methods.

An RSNA might support SAE authentication, FILS authentication, or both. An RSNA also supports authentication based on IEEE Std 802.1X-2010, or preshared keys (PSKs) after Open System authentication. IEEE 802.1X authentication utilizes the EAP to authenticate STAs and the AS with one another. This standard does not specify an EAP method that is mandatory to implement. See 12.6.5 for a description of the IEEE 802.1X authentication and PSK usage within an IEEE 802.11 IBSS.

In an RSNA, IEEE 802.1X Supplicants and Authenticators exchange protocol information via the IEEE 802.1X Uncontrolled Port. The IEEE 802.1X Controlled Port is blocked from passing general data traffic between two STAs until an IEEE 802.1X authentication procedure completes successfully over the IEEE 802.1X Uncontrolled Port.

SAE authentication ~~or~~and Open System IEEE 802.11 authentication ~~is~~ are used by non-DMG STAs in an RSN for an infrastructure BSS. FILS authentication may be used by FILS STAs in an RSN for an infrastructure BSS. SAE authentication, Open System IEEE 802.11 authentication, or no IEEE 802.11 authentication is used in an RSN for an IBSS. SAE authentication is used ~~in~~for an MBSS. In a~~A~~n RSN~~A~~, ~~disallows the use of~~ Shared Key authentication is not used. In an RSN for DMG BSS, Open System IEEE 802.11 authentication is not used (12.2.4).

4.5.4.3 Deauthentication

Change 4.5.4.3 as follows:

The deauthentication service is invoked when an existing Open System, Shared Key, FT, ~~or~~ SAE, or FILS authentication is to be terminated. Deauthentication is a SS.

In an ESS, because authentication is a prerequisite for association, the act of deauthentication causes the STA to be disassociated. The deauthentication service can be invoked by either authenticated party (non-AP STA or AP). Deauthentication is not a request; it is a notification. The association at the transmitting STA is terminated when the STA sends a deauthentication notice to an associated STA. Deauthentication, and if associated, disassociation can not be refused by the receiving STA except when management frame protection is negotiated and the message integrity check fails.

~~In an RSN ESS, Open System IEEE Std 802.11 authentication is required.~~ In an RSN ESS, deauthentication results in termination of any association for the deauthenticated STA. It also results in the IEEE 802.1X Controlled Port for that STA, if used for this association, being disabled and deletes the pairwise transient key security association (PTKSA). The deauthentication notification is provided to IEEE Std 802.1X-2010 via the MAC layer.

In an RSNA, deauthentication also destroys any related pairwise transient key security association (PTKSA), group temporal key security association (GTKSA), station-to-station link (STSL) master key security association (SMKSA), STSL transient key security association (STKSA), TPK security association (TPKSA), integrity group temporal key security association (IGTKSA), mesh GTKSA, and mesh TKSA that exist in the STA and closes the associated IEEE 802.1X Controlled Port, if used for this association. If pairwise master key security association (PMKSA) caching is not enabled, deauthentication also deletes the PMKSA or mesh PMKSA.

In an RSN IBSS, Open System authentication is optional, but a STA is required to recognize Deauthentication frames. Deauthentication results in the IEEE 802.1X Controlled Port for that STA being disabled and deletes the PTKSA.

4.5.4.5 Key management

Change 4.5.4.5 as follows:

The enhanced data confidentiality, data authentication, and replay protection mechanisms require fresh cryptographic keys and corresponding security associations. The procedures defined in this standard provide fresh keys by means of various protocols and handshakes.~~called the 4-way handshake, FT 4-way handshake, FT protocol, FT resource request protocol, and group key handshake~~.

4.5.4.8 Fast BSS transition

Change 4.5.4.8 as follows:

The FT mechanism defines a means for a STA to set up security and QoS parameters prior to reassociation to a new AP. This mechanism allows time-consuming operations to be removed from the time-critical reassociation process. When FILS authentication is used during a handover across mobility domains, the overhead incurred during the FT initial mobility domain association in an RSN is further reduced.

4.10 IEEE Std 802.11 and IEEE Std 802.1X-2010

4.10.2 IEEE Std 802.11 usage of IEEE Std 802.1X-2010

Change the second paragraph of 4.10.2 as follows:

IEEE Std 802.11 depends upon IEEE Std 802.1X-2010 and various IEEE 802.11 protocols and handshakes ~~the 4-way handshake, FT 4-way handshake, FT protocol, FT resource request protocol, and group key handshake~~, described in Clause 12 and Clause 13, to establish and change cryptographic keys. Keys are established after authentication has completed. Keys might change for a variety of reasons, including expiration of an IEEE 802.1X authentication timer, key compromise, danger of compromise, or policy.

4.10.3 Infrastructure functional model overview

Insert new subclause 4.10.3.6 as follows:

4.10.3.6 AKM operations using FILS authentication

4.10.3.6.1 General

FILS authentication allows faster link setup to the network by performing authentication, association, and key confirmation using four frames—two Authentication frames, one (Re)Association Request frame, and one (Re)Association Response frame.

The following three FILS authentication methods are defined:

a) FILS Shared Key authentication performed without perfect forward security (PFS)
b) FILS Shared Key authentication performed with PFS
c) FILS Public Key authentication with PFS

FILS Shared Key authentication with and without PFS is described in 4.10.3.6.2.

FILS Public Key authentication exchange is described in 4.10.3.6.3.

4.10.3.6.2 AKM operations using FILS Shared Key authentication

A non-AP STA and an Authentication Server (AS) using FILS Shared Key authentication verify mutual possession of a shared key (rRK) (as defined in IETF RFC 5295 and IETF RFC 6696) using Extensible Authentication Protocol (EAP) reauthentication protocol (EAP-RP) signaling. EAP-RP signaling is encapsulated using FILS wrapped data in an Authentication frame as shown in Figure 4-29a. A valid rRK is derived using a prior full authentication using the full EAP as defined in 4.10.3.2. This rRK can be used for multiple runs of EAP-RP authentications as specified in IETF RFC 5295 and IETF RFC 6696.

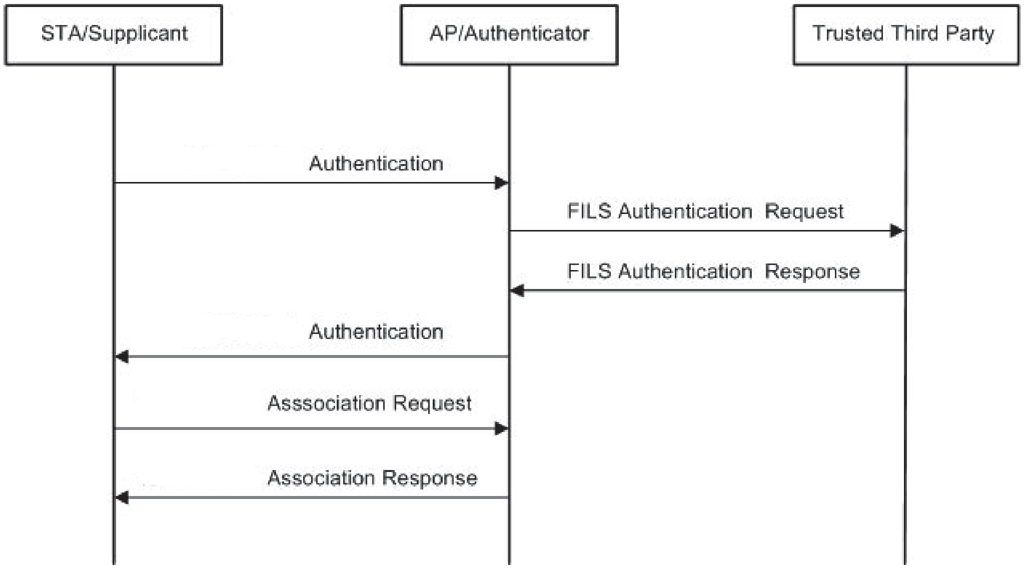


Figure 4-29a—FILS authentication using Authentication Server

4.10.3.6.3 AKM operations using FILS Public Key authentication

When using FILS Public Key authentication, it is assumed that both STAs using FILS have either:

- Obtained a public key certificate from a certificate authority (CA) and are capable of verifying each other's certificate during execution of FILS authentication procedures; or
- A priori knowledge of, and trust in, an uncertified public key.

The manner in which trust is obtained in certificates is outside the scope of this standard.

4.10.7 PMKSA caching

Change 4.10.7 as follows:

The Authenticator and Supplicant can cache PMKSAs, which include the IEEE 802.1X state. A PMKSA can be deleted from the cache for any reason and at any time.

The STA can supply a list of PMK identifiers in the (Re)Association Request frame or first FILS Authentication frame. Each key identifier names a PMKSA; the PMKSA can contain a single PMK. The Authenticator can specify~~ies~~ the selected PMK key identifier in message 1 of the 4-way handshake or the second FILS Authentication frame. The selection of the key identifiers to be included by the STA and Authenticator ~~within the (Re)Association Request frame and message 1 of the 4-Way handshake~~ is out of the scope of this standard.

A FILS STA performing FILS authentication can supply a list of PMK identifiers in its initial Authentication frame. Each PMK identifier names a PMKSA; the PMKSA contains a single PMK. If the AP has retained an identified PMKSA, it can facilitate a faster connection by identifying a single PMKSA in the Authentication frame it transmits. The STA and AP can then use the PMK from the cached PMKSA to authenticate. FILS APs that support PMK caching may identify themselves using a cache identifier. A FILS STA that has successfully established a PMKSA at an AP identifying a particular cache identifier can attempt to use PMK caching in a subsequent attempt with any AP that uses the same cache identifier.

5. MAC service definition

5.2 MAC data service specification

5.2.3 MA-UNITDATA.indication

5.2.3.3 When generated

Change 5.2.3.3 as follows:

The MA-UNITDATA.indication primitive is passed from the MAC sublayer entity to the LLC sublayer entity or entities to indicate the arrival of a frame at the local MAC sublayer entity. Frames are reported only if they are validly formatted at the MAC sublayer, received without error, received with valid (or null) security and integrity information, and their destination address designates the local MAC sublayer entity. The MA-UNITDATA.indication primitive might also be passed from the MAC sublayer entity, in coordination with the MAC sublayer management entity, to the LLC sublayer entity to indicate the arrival of a FILS higher layer protocol (HLP) Container element.

6. Layer management

6.3 MLME SAP interface

6.3.3 Scan

6.3.3.2 MLME-SCAN.request

6.3.3.2.2 Semantics of the service primitive

Change the primitive parameter list in 6.3.3.2.2 as follows:

The primitive parameters are as follows:

```
MLME-SCAN.request(
                              BSSType,
                              BSSID,
                              SSID,
                              ScanType,
                              ProbeDelay,
                              ChannelList,
                              MinChannelTime,
                              MaxChannelTime,
                              RequestInformation,
                              SSID List,
                              ChannelUsage,
                              AccessNetworkType,
                              HESSID,
                              MeshID,
                              DiscoveryMode,
                              FILSRequestParameters,
                              ReportingOption,
                              APConfigurationSequenceNumber,
                              VendorSpecificInfo
                              )
```

Insert the following rows before the VendorSpecificInfo row in the parameter table in 6.3.3.2.2:

Name	Type	Valid range	Description
FILSRequestParameters	As defined in 9.4.2.178	As defined in 9.4.2.178	Used in determining whether to transmit a Probe Response frame. This parameter is optionally present if dot11FILSActivated is true; otherwise not present.
ReportingOption	Enumeration	IMMEDIATE, CHANNEL_SPECIFIC, AT_END	Indicates the result reporting mode as described in 11.1.4. This parameter is optionally present if dot11FILSActivated is true; otherwise not present.
APConfigurationSequenceNumber	Integer	As defined in 9.4.2.182	Indicates the configuration sequence number for the static information fields and elements as described in 11.1.4.3.7. This parameter is optionally present if dot11FILSActivated is true and BSSID is an individual MAC address; otherwise not present.

6.3.3.3 MLME-SCAN.confirm

6.3.3.3.1 Function

Change the first paragraph of 6.3.3.1 as follows:

This primitive returns the descriptions of the set of BSSs detected by the scan process. Multiple MLME-SCAN.confirm primitives can be issued when the value of the ReportingOption parameter in the MLME-SCAN.request primitive is CHANNEL_SPECIFIC or IMMEDIATE. When the value of the ReportingOption parameter is AT_END, or the ReportingOption parameter is not present, a single MLME-SCAN.confirm primitive is issued.

6.3.3.3.2 Semantics of the service primitive

Change the primitive parameter list in 6.3.3.3.2 as follows:

The primitive parameters are as follows:

```
  MLME-SCAN.confirm(
                                    BSSDescriptionSet,
                                    BSSDescriptionFromMeasurementPilotSet,
                                    BSSDescriptionFromFDSet,
                                    ResultCode,
                                    VendorSpecificInfo
                                    )
```

Change the table in 6.3.3.3.2 as follows:

Name	Type	Valid range	Description
BSSDescriptionSet	Set of BSSDescriptions	N/A	The BSSDescriptionSet is returned to indicate the results of the scan request. It is a set containing zero or more instances of a BSSDescription.
BSSDescriptionFrom MeasurementPilotSet	Set of BSS DescriptionFrom MeasurementPilots	N/A	The BSSDescriptionFromMeasurementPilotSet is returned to indicate the results of the scan request derived from measurement pilots. It is a set containing zero or more instances of a BSSDescriptionFrom-MeasurementPilot. Present if dot11RMMeasurementPilotActivated is nonzero; otherwise not present.
BSSDescriptionFromFDSet	Set of BSSDescriptionFromFDs	N/A	The BSSDescriptionFromFDSet is returned to indicate the results of the scan request derived from FILS Discovery frames. It is a set containing zero or more instances of a BSSDescriptionFromFD. Present if dot11FILSActivated is true; otherwise not present.
ResultCode	Enumeration	SUCCESS, INTERMEDIATE_SCAN_RESULT, NOT_SUPPORTED	Indicates the result of the MLME-SCAN.confirm primitive. The INTERMEDIATE_SCAN_RESULT is used to report the discovered BSSs when the value of the ReportingOption parameter in the MLME-SCAN.request primitive is CHANNEL_SPECIFIC or IMMEDIATE and is valid if dot11FILSActivated is true.
VendorSpecificInfo	A set of elements	As defined in 9.4.2.26	Zero or more elements.

Insert the following rows at the end of the second table in 6.3.3.3.2:

Name	Type	Valid range	Description	IBSS adoption
CAG Number	As defined in 9.4.2.177	As defined in 9.4.2.177	One or more Common Advertisement Group (CAG) tuples. Each CAG Tuple describes the CAG Version, Scope, and Partial Advertisement Protocol ID. This parameter is optionally present when dot11FILSActivated is true; otherwise not present.	Do not adopt.
Differentiated Initial Link Setup	As defined in 9.4.2.187	As defined in 9.4.2.187	Contains the values from the Differentiated Initial Link Setup element. It is optionally present when dot11FILSActivated is true and if such an element was present in the Probe Response or Beacon frame; otherwise not present.	Do not adopt.

Insert the following text and table to the end of 6.3.3.3.2:

The BSSDescriptionFromFDSet parameter is present if dot11FILSActivated is true; otherwise, it is not present. Each BSSDescriptionFromFD consists of the parameters shown in the following table:

Name	Type	Valid range	Description
BSSID	MAC address	N/A	The BSSID of the found BSS.
SSID	Octet String	As defined in 9.4.2.2	The SSID of the found BSS. This parameter is present if the Short SSID Indicator in the received FILS Discovery frame is equal to 0.
Short SSID	Integer	As defined in the 9.4.2.171	The Short SSID of the found BSS. This parameter is present if the Short SSID Indicator in the received FILS Discovery frame is equal to 1.
FD Capability	As defined in 9.6.8.36	As defined in 9.6.8.36	The FD Capability contains the capabilities and operational indications of the BSS. This parameter is optional.
Operating class	Integer	As defined in 9.4.1.37	The operating class of the advertised BSS.
Access Network Options	As defined in 8.4.2.91	As defined in 8.4.2.91	The advertised access network options of the BSS. This parameter is optional.

Name	Type	Valid range	Description
Primary Channel	Integer	1–255	The Primary Channel of the advertised BSS. The Primary Channel is defined within the indicated Operating Class as shown in Annex E. This parameter is optional.
FD RSN Information	As defined in 9.6.8.36	As defined in 9.6.8.36	The information for robust security network. This parameter is optional.
Channel Center | Frequency Segment 1	Integer	1–255	The channel frequency index of the 80 MHz channel of frequency segment 1 when the BSS operates on an 80+80 MHz operating channel width. This parameter is optional.
AP-CSN	Integer	As defined in 9.4.2.182	The value of the Configuration Sequence Number in the found BSS. This parameter is optional.
AP's next TBTT Offset	Integer	As defined in 9.6.8.36	The information of next target beacon transmission time of the found BSS. This parameter is optional.
Reduced Neighbor Report	As defined in 9.4.2.171	As defined in 9.4.2.171	The information of the Reduced Neighbor Information field of the received FILS Discovery frame. This parameter is optional.
FILS Indication	As defined in 9.4.2.183	As defined in 9.4.2.183	The information related to FILS authentication and upper layer set up capabilities of the found AP. This parameter is optional.

6.3.3.3.3 When generated

Change 6.3.3.3.3 as follows:

This primitive is generated by the MLME to report the operating environment of the STA. It is issued after receiving an MLME-SCAN.request primitive or, if dot11FILSActivated is true, after receiving an MLME-SCAN-STOP.request primitive following an MLME-SCAN.request primitive. ~~as a result of an MLME-SCAN.request primitive to ascertain the operating environment of the STA.~~

6.3.3.3.4 Effect of receipt

Change 6.3.3.3.4 as follows:

The SME is notified of the results of the scan procedure. If dot11FILSActivated is true, these results might be intermediate results, according to the value of ResultCode.

Insert new subclause 6.3.3.4 as follows:

6.3.3.4 MLME-SCAN-STOP.request

6.3.3.4.1 Function

This primitive terminates any ongoing scan.

6.3.3.4.2 Semantics of the service primitive

The primitive has no parameters.

6.3.3.4.3 When generated

This primitive is generated by the SME in order to stop all ongoing active or passive scan processes in the STA.

6.3.3.4.4 Effect of receipt

This request terminates all ongoing scan procedures.

6.3.5 Authenticate

6.3.5.2 MLME-AUTHENTICATE.request

6.3.5.2.2 Semantics of the service primitive

Change the primitive parameter list in 6.3.5.2.2 as follows:

The primitive parameters are as follows:

```
MLME-AUTHENTICATE.request(
                              PeerSTAAddress,
                              AuthenticationType,
                              AuthenticateFailureTimeout,
                              Content of FT Authentication elements,
                              Content of SAE Authentication frame,
                              Multi-band local,
                              Multi-band peer,
                              Content of FILS Authentication frame,
                              VendorSpecificInfo
                              )
```

Change one row and insert one new row before the VendorSpecificInfo row in the parameter table in 6.3.5.2.2 as follows:

Name	Type	Valid range	Description
AuthenticationType	Enumeration	OPEN_SYSTEM, SHARED_KEY, FAST_BSS_TRANSITION, SAE, FILS_SHARED_KEY WITHOUT_PFS, FILS_SHARED KEY_WITH_PFS, FILS_PUBLIC_KEY	Specifies the type of authentication algorithm to use during the authentication process.
Content of FILS Authentication frame	Sequence of elements and fields	As defined in 9.4.1.43, 9.4.1.41, 9.4.2.25, 9.4.2.180, 9.4.2.188, and 9.4.2.190	The set of elements and fields to be included in the first message of the FILS authentication sequence, as described in 12.12. Present if AuthenticationType indicates FILS SHARED KEY WITHOUT PFS, FILS SHARED KEY WITH PFS, or FILS PUBLIC KEY; otherwise not present.

6.3.5.3 MLME-AUTHENTICATE.confirm

6.3.5.3.2 Semantics of the service primitive

Change the primitive parameter list in 6.3.5.3.2 as follows:

The primitive parameters are as follows:

MLME-AUTHENTICATE.confirm(
PeerSTAAddress,
AuthenticationType,
ResultCode,
Content of FT Authentication elements,
Content of SAE Authentication frame,
Multi-band local,
Multi-band peer,
Content of FILS Authentication frame,
VendorSpecificInfo
)

Change one row and insert a new row before the VendorSpecificInfo row in the parameter table in 6.3.5.3.2 as follows:

Name	Type	Valid range	Description
AuthenticationType	Enumeration	OPEN_SYSTEM, SHARED_KEY, FAST_BSS_TRANSITION, SAE, FILS_SHARED KEY_WITHOUT_PFS, FILS_SHARED_KEY_WITH_PFS, FILS_PUBLIC_KEY	Specifies the type of authentication algorithm to use during the authentication process.
Content of FILS Authentication frame	Sequence of elements and fields	As defined in 9.4.1.43, 9.4.1.41, 9.4.2.25, 9.4.2.176, 9.4.2.180, 9.4.2.188, and 9.4.2.190	The set of elements and fields included in the second message of the FILS authentication sequence, as described in 12.12. Present if AuthenticationType indicates FILS SHARED KEY WITHOUT PFS, FILS SHARED KEY WITH PFS, or FILS PUBLIC KEY; otherwise not present.

6.3.5.4 MLME-AUTHENTICATE.indication

6.3.5.4.2 Semantics of the service primitive

Change the primitive parameter list in 6.3.5.4.2 as follows:

The primitive parameters are as follows:

```
   MLME-AUTHENTICATE.indication(
                                            PeerSTAAddress,
                                            AuthenticationType,
                                            Content of FT Authentication elements,
                                            Content of SAE Authentication frame,
                                            Multi-band local,
                                            Multi-band peer,
                                            Content of FILS Authentication frame,
                                            VendorSpecificInfo
                                            )
```

Change one row and insert a new row before the VendorSpecificInfo row in the parameter table in 6.3.5.4.2 as follows:

Name	Type	Valid range	Description
AuthenticationType	Enumeration	OPEN_SYSTEM, SHARED_KEY, FAST_BSS_TRANSITION, SAE, FILS_SHARED_KEY_WITHOUT_PFS, FILS_SHARED_KEY_WITH_PFS, FILS_PUBLIC_KEY	Specifies the type of authentication algorithm that was used during the authentication process.
Content of FILS Authentication frame	Sequence of elements and fields	As defined in 9.4.1.43, 9.4.1.41, 9.4.2.25, 9.4.2.176, 9.4.2.180, 9.4.2.188, and 9.4.2.190	The set of elements and fields included in the first message of the FILS authentication sequence, as described in 12.12. Present if AuthenticationType indicates FILS SHARED KEY WITHOUT PFS, FILS SHARED KEY WITH PFS, or FILS PUBLIC KEY; otherwise not present.

6.3.5.5 MLME-AUTHENTICATE.response

6.3.5.5.2 Semantics of the service primitive

Change the primitive parameter list in 6.3.5.5.2 as follows:

The primitive parameters are as follows:

```
    MLME-AUTHENTICATE.response(
                                    PeerSTAAddress,
                                    ResultCode,
                                    Content of FT Authentication elements,
                                    Content of SAE Authentication frame,
                                    Multi-band local,
                                    Multi-band peer,
                                    Content of FILS Authentication frame,
                                    VendorSpecificInfo
                                    )
```

Insert a new row before the VendorSpecificInfo row in the parameter table in 6.3.5.5.2 as follows:

Name	Type	Valid range	Description
Content of FILS Authentication frame	Sequence of elements and fields	As defined in 9.4.1.43, 9.4.1.41, 9.4.2.25, 9.4.2.180, 9.4.2.188, and 9.4.2.190	The set of elements and fields to be included in the second message of the FILS authentication sequence, as described in 12.12. Present if AuthenticationType indicates FILS SHARED KEY WITHOUT PFS, FILS SHARED KEY WITH PFS, or FILS PUBLIC KEY; otherwise not present.

6.3.7 Associate

6.3.7.2 MLME-ASSOCIATE.request

6.3.7.2.2 Semantics of the service primitive

Change the primitive parameter list in 6.3.7.2.2 as follows:

The primitive parameters are as follows:

```
MLME-ASSOCIATE.request(
                              PeerSTAAddress,
                              ListenInterval,
                              Supported Channels,
                              RSN,
                              QoSCapability,
                              Content of FT Authentication elements,
                              SupportedOperatingClasses,
                              SM Power Save,
                              QoSTrafficCapability,
                              TIMBroadcastRequest,
                              EmergencyServices,
                              DMG Capabilities,
                              Multi-band local,
                              Multi-band peer,
                              MMS,
                              FILSHLPContainer,
                              FILSIPAddressAssignment,
                              VendorSpecificInfo
                              )
```

Insert two new rows before the VendorSpecificInfo row in the parameter table in 6.3.7.2.2 as follows:

Name	Type	Valid range	Description
FILSHLPContainer	As defined in 9.4.2.184	As defined in 9.4.2.184	A set of elements containing encapsulated data of higher layer protocol frames (e.g., DHCP message) that is transported in FILS association. The parameter is optionally present if dot11FILSActivated is true; otherwise not present.
FILSIPAddressAssignment	As defined in 9.4.2.185	As defined in 9.4.2.185	An explicit request for an IP address. The request is either for a new IP address or for a specified IP address. The parameter is optionally present if dot11FILSActivated is true; otherwise not present.

6.3.7.3 MLME-ASSOCIATE.confirm

6.3.7.3.2 Semantics of the service primitive

Change the primitive parameter list in 6.3.7.3.2 as follows:

The primitive parameters are as follows:
MLME-ASSOCIATE.confirm(
ResultCode,
CapabilityInformation,
AssociationID,
EDCAParameterSet,
RCPI of Request,
RSNI of Request,
RCPI of Response,
RSNI of Response,
RMEnabledCapabilities,
Content of FT Authentication elements,
SupportedOperatingClasses,
Extended Capabilities,
20/40 BSS Coexistence,
TimeoutInterval,
BSSMaxIdlePeriod,
TIMBroadcastResponse,
QoSMapSet,
QMFPolicy,
DMG Capabilities,
Multi-band local,
Multi-band peer,
MMS,
FILSHLPContainer,
FILSIPAddressAssignment,
KeyDelivery,

VendorSpecificInfo
)

Insert three new rows before the VendorSpecificInfo row in the parameter table in 6.3.7.3.2 as follows:

Name	Type	Valid range	Description
FILSHLPContainer	As defined in 9.4.2.184	As defined in 9.4.2.184	A set of elements containing encapsulated data of higher layer protocol (HLP) frames (e.g., DHCP message) that is transported in FILS association. The parameter is optionally present if dot11-FILSActivated is true; otherwise not present.
FILSIPAddressAssignment	As defined in 9.4.2.185	As defined in 9.4.2.185	Contains the IP address of a network layer entity associated with the STA. The parameter is optionally present if dot11-FILSActivated is true; otherwise not present.
KeyDelivery	As defined in 9.4.2.186	As defined in 9.4.2.186	Contains the KDE(s) and the Key RSC. The parameter is present if dot11FILS-Activated is true; otherwise not present.

6.3.7.4 MLME-ASSOCIATE.indication

6.3.7.4.2 Semantics of the service primitive

Change the primitive parameter list in 6.3.7.4.2 as follows:

The primitive parameters are as follows:
MLME-ASSOCIATE.indication(
PeerSTAAddress,
CapabilityInformation,
ListenInterval,
SSID,
OperationalRateSet,
BSSMembershipSelectorSet,
RSN,
QoSCapability,
RCPI,
RSNI,
RMEnabledCapabilities,
Content of FT Authentication elements,
SupportedOperatingClasses,
DSERegisteredLocation,
HT Capabilities,
Extended Capabilities,
20/40 BSS Coexistence,
QoSTrafficCapability,
TIMBroadcastRequest,
EmergencyServices,
DMG Capabilities,

Multi-band local,
Multi-band peer,
MMS,
VHT Capabilities,
FILSHLPContainer,
FILSIPAddressAssignment,
VendorSpecificInfo
)

Insert two new rows before the VendorSpecificInfo row in the parameter table in 6.3.7.4.2 as follows:

Name	Type	Valid range	Description
FILSHLPContainer	As defined in 9.4.2.184	As defined in 9.4.2.184	A set of elements containing encapsulated data of higher layer protocol frames (e.g., DHCP message) that is transported in FILS association. The parameter is optionally present if dot11FILSActivated is true; otherwise not present.
FILSIPAddressAssignment	As defined in 9.4.2.185	As defined in 9.4.2.185	An explicit request for an IP address. The request is for a new IP address or a specified IP address. The parameter is optionally present if dot11FILSActivated is true; otherwise not present.

6.3.7.5 MLME-ASSOCIATE.response

6.3.7.5.2 Semantics of the service primitive

Change the primitive parameter list in 6.3.7.5.2 as follows:

The primitive parameters are as follows:

MLME-ASSOCIATE.response(
PeerSTAAddress,
ResultCode,
AssociationID,
RCPI,
RSNI,
RMEnabledCapabilities,
Content of FT Authentication elements,
SupportedOperatingClasses,
TimeoutInterval,
BSSMaxIdlePeriod,

TIMBroadcastResponse,
QoSMapSet,
Multi-band peer,
FILSHLPContainer,
FILSIPAddressAssignment,
KeyDelivery,
VendorSpecificInfo
)

Insert three new rows before the VendorSpecificInfo row in the parameter table in 6.3.7.5.2 as follows:

Name	Type	Valid range	Description
FILSHLPContainer	As defined in 9.4.2.184	As defined in 9.4.2.184	A set of elements containing encapsulated data of higher layer protocol frames (e.g., a DHCP message) transported in FILS association. The parameter is optionally present if dot11FILSActivated is true; otherwise not present.
FILSIPAddressAssignment	As defined in 9.4.2.185	As defined in 9.4.2.185	Contains the IP address of a network layer entity associated with the STA. The parameter is optionally present if dot11FILSActivated is true; otherwise not present.
KeyDelivery	As defined in 9.4.2.186	As defined in 9.4.2.186	Contains KDE(s) and the current Key RSC. The parameter is present if dot11FILSActivated is true; otherwise not present.

6.3.8 Reassociate

6.3.8.2 MLME-REASSOCIATE.request

6.3.8.2.2 Semantics of the service primitive

Change the primitive parameter list in 6.3.8.2.2 as follows:

The primitive parameters are as follows:
MLME-REASSOCIATE.request(
NewPCPorAPAddress,
ListenInterval,
Supported Channels,
RSN,
QoSCapability,

Content of FT Authentication elements,
SupportedOperatingClasses,
SM Power Save,
QoSTrafficCapability,
TIMBroadcastRequest,
FMSRequest,
DMSRequest,
EmergencyServices,
DMG Capabilities,
Multi-band local,
Multi-band peer,
MMS,
FILSHLPContainer,
FILSIPAddressAssignment,
VendorSpecificInfo
)

Insert two new rows before the VendorSpecificInfo row in the parameter table in 6.3.8.2.2 as follows:

Name	Type	Valid range	Description
FILSHLPContainer	As defined in 9.4.2.184	As defined in 9.4.2.184	A set of elements containing encapsulated data of higher layer protocol frames (e.g., a DHCP message) that is transported in FILS association. The parameter is optionally present if dot11FILSActivated is true; otherwise not present.
FILSIPAddressAssignment	As defined in 9.4.2.185	As defined in 9.4.2.185	An explicit request for an IP address. The request may be for a new IP address or a specified IP address. The parameter is optionally present if dot11FILSActivated is true; otherwise not present.

6.3.8.3 MLME-REASSOCIATE.confirm

6.3.8.3.2 Semantics of the service primitive

Change the primitive parameter list in 6.3.8.3.2 as follows:

The primitive parameters are as follows:
MLME-REASSOCIATE.confirm(
ResultCode,
CapabilityInformation,
AssociationID,
EDCAParameterSet,
RCPI of Request,
RSNI of Request,
RCPI of Response,

RSNI of Response,
RMEnabledCapabilities,
Content of FT Authentication elements,
SupportedOperatingClasses,
Extended Capabilities,
20/40 BSS Coexistence,
TimeoutInterval,
BSSMaxIdlePeriod,
TIMBroadcastResponse,
FMSRespone,
DMSResponse,
QoSMapSet,
QMFPolicy,
DMG Capabilities,
Multi-band local,
Multi-band peer,
MMS,
FILSHLPContainer,
FILSIPAddressAssignment,
KeyDelivery,
VendorSpecificInfo
)

Insert three new rows before the VendorSpecificInfo row in the parameter table in 6.3.8.3.2 as follows:

Name	Type	Valid range	Description
FILSHLPContainer	As defined in 9.4.2.184	As defined in 9.4.2.184	A set of elements containing encapsulated dataset of higher layer protocol frames (e.g., a DHCP message) that is transported in FILS association. The parameter is present if dot11FILSActivated is true; otherwise not present.
FILSIPAddressAssignment	As defined in 9.4.2.185	As defined in 9.4.2.185	Contains the IP address assigned to the STA for higher layer communication. The parameter is present if dot11FILSActivated is true; otherwise not present.
KeyDelivery	As defined in 9.4.2.186	As defined in 9.4.2.186	Contains KDE(s) and the current Key RSC. The parameter is present if dot11FILSActivated is true; otherwise not present.

6.3.8.4 MLME-REASSOCIATE.indication

6.3.8.4.2 Semantics of the service primitive

Change the primitive parameter list in 6.3.8.4.2 as follows:

The primitive parameters are as follows:

MLME-REASSOCIATE.indication(
PeerSTAAddress,
CurrentAPAddress,
CapabilityInformation,
ListenInterval,
SSID,
OperationalRateSet,
BSSMembershipSelectorSet,
RSN,
QoSCapability,
RCPI,
RSNI,
RMEnabledCapabilities,
Content of FT Authentication elements,
SupportedOperatingClasses,
DSERegisteredLocation,
HT Capabilities,
Extended Capabilities,
20/40 BSS Coexistence,
QoSTrafficCapability,
TIMBroadcastRequest,
FMSRequest,
DMSRequest,
EmergencyServices,
DMG Capabilities,
Multi-band local,
Multi-band peer,
MMS,
VHT Capabilities,
FILSHLPContainer,
FILSIPAddressAssignment,
VendorSpecificInfo
)

Insert two new rows before the VendorSpecificInfo row in the parameter table in 6.3.8.4.2 as follows:

Name	Type	Valid range	Description
FILSHLPContainer	As defined in 9.4.2.184	As defined in 9.4.2.184	A set of elements containing encapsulated data of higher layer protocol frames (e.g., a DHCP message) that is transported in FILS reassociation. The parameter is optionally present if dot11FILSActivated is true; otherwise not present.
FILSIPAddressAssignment	As defined in 9.4.2.185	As defined in 9.4.2.185	An explicit request for an IP address. The request may be for a new IP address or a specified IP address. The parameter is optionally present if dot11FILSActivated is true; otherwise not present.

6.3.8.5 MLME-REASSOCIATE.response

6.3.8.5.2 Semantics of the service primitive

Change the primitive parameter list in 6.3.8.5.2 as follows:

The primitive parameters are as follows:

MLME-REASSOCIATE.response(
PeerSTAAddress,
ResultCode,
AssociationID,
RCPI,
RSNI,
RMEnabledCapabilities,
Content of FT Authentication elements,
SupportedOperatingClasses,
TimeoutInterval,
BSSMaxIdlePeriod,
TIMBroadcastResponse,
FMSResponse,
DMSResponse,
QoSMapSet,
Multi-band peer,
FILSHLPContainer,
FILSIPAddressAssignment,
KeyDelivery,
VendorSpecificInfo
)

Insert three new rows before the VendorSpecificInfo row in the parameter table in 6.3.8.5.2 as follows:

Name	Type	Valid range	Description
FILSHLPContainer	As defined in 9.4.2.184	As defined in 9.4.2.184	A set of elements containing encapsulated data of higher layer protocol frames (e.g., a DHCP message) that is transported in FILS reassociation. The parameter is optionally present if dot11FILSActivated is true; otherwise not present.
FILSIPAddressAssignment	As defined in 9.4.2.185	As defined in 9.4.2.185	Contains the IP address of a network layer entity associated with the STA. The parameter is optionally present if dot11FILSActivated is true; otherwise not present.
KeyDelivery	As defined in 9.4.2.186	As defined in 9.4.2.186	Contains KDE(s) and the current Key RSC. The parameter is present if dot11FILSActivated is true; otherwise not present.

6.3.11 Start

6.3.11.2 MLME-START.request

6.3.11.2.2 Semantics of the service primitive

Change the primitive parameter list in 6.3.11.2.2 as follows:

The primitive parameters are as follows:

```
  MLME-START.request(
                                    SSID,
                                    BSSType,
                                    BeaconPeriod,
                                    DTIMPeriod,
                                    CF parameter set,
                                    PHY parameter set,
                                    IBSS parameter set,
                                    NAVSyncDelay,
                                    CapabilityInformation,
                                    BSSBasicRateSet,
                                    OperationalRateSet,
                                    Country,
                                    IBSS DFS Recovery Interval,
                                    EDCAParameterSet,
                                    DSERegisteredLocation,
                                    HT Capabilities,
                                    HT Operation,
```

BSSMembershipSelectorSet,
Extended Capabilities,
20/40 BSS Coexistence,
Overlapping BSS Scan Parameters,
MultipleBSSID,
InterworkingInfo,
AdvertisementProtocolInfo,
RoamingConsortiumInfo,
Mesh ID,
Mesh Configuration,
QMFPolicy,
DMG Capabilities,
Multi-band,
MMS,
DMG Operation,
Clustering Control,
CBAP Only,
PCP Association Ready,
VHT Capabilities,
VHT Operation,
Known OUIs,
VendorSpecificInfo
)

Change the following row in the parameter table in 6.3.11.2.2:

Name	Type	Valid range	Description
MultipleBSSID	As defined in Multiple BSSID Element in 9.4.2.46	As defined in Multiple BSSID Element in 9.4.2.46	This element is optionally present when dot11RMMeasurementPilotActivated is a value between 2 and 7 and the AP is a member of a Multiple BSSID Set (see 11.11.14) with two or more members, or if dot11MultiBSSIDActivated is true. This element is present when dot11FILSActivated is true and the AP is a member of a Multiple BSSID Set with two or more members.

Insert one new row before the VendorSpecificInfo row in the parameter table in 6.3.11.2.2 as follows:

Name	Type	Valid range	Description
Known OUIs	A set of OUIs, each of which is as defined in 9.4.1.32	N/A	Zero or more OUIs that specify the OUIs known by the AP. The AP uses the known OUIs to determine if it should respond to the Probe Request frame as defined in 11.1.4.3.4.

Insert new subclause 6.3.105 as follows:

6.3.105 FILS Container

6.3.105.1 General

This mechanism supports the process of IP address setup with a peer MAC entity.

6.3.105.2 MLME-FILSContainer.request

6.3.105.2.1 Function

This primitive requests transmission of the FILS Container frame with a specified peer MAC entity.

6.3.105.2.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-FILSContainer.request(
                              Peer MAC Address,
                              FILSIPAddressAssignment,
                              VendorSpecificInfo
                              )
```

Name	Type	Valid range	Description
Peer MAC Address	MACAddress	Any valid individual MAC address	Specifies the address of the peer MAC entity.
FILSIPAddressAssignment	FILS IP Address Assignment element	As defined in 9.4.2.185	The request may be for a new IP address or a specified IP address.
VendorSpecificInfo	A set of elements	As defined in 9.4.2.26	Zero or more elements.

6.3.105.2.3 When generated

This primitive is generated by the SME for a STA to request IP Address setup from the AP.

6.3.105.2.4 Effect of receipt

This primitive requests IP Address setup. In the case that a response is received from the AP, the MLME subsequently issues an MLME-FILSContainer.confirm primitive that reflects the results.

6.3.105.3 MLME-FILSContainer.confirm

6.3.105.3.1 Function

This primitive reports the results of an IP Address setup with an AP.

6.3.105.3.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-FILSContainer.confirm(
                                    Peer MAC Address,
                                    FILSIPAddressAssignment,
                                    VendorSpecificInfo
                                    )
```

Name	Type	Valid range	Description
Peer MAC Address	MACAddress	Any valid individual MAC address	Specifies the address of the peer MAC entity.
FILSIPAddressAssignment	FILS IP Address Assignment element	As defined in 9.4.2.185	IP address information.
VendorSpecificInfo	A set of elements	As defined in 9.4.2.26	Zero or more elements.

6.3.105.4 MLME-FILSContainer.indication

6.3.105.4.1 Function

This primitive indicates receipt of a request of IP Address setup.

6.3.105.4.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-FILSContainer.indication(
                                    Peer MAC Address,
                                    FILSIPAddressAssignment,
                                    VendorSpecificInfo
                                    )
```

Name	Type	Valid range	Description
Peer MAC Address	MACAddress	Any valid individual MAC address	Specifies the address of the peer MAC entity.
FILSIPAddressAssignment	FILS IP Address Assignment element	As defined in 9.4.2.185	An explicit request for an IP address. The request may be for a new IP address or a specified IP address.
VendorSpecificInfo	A set of elements	As defined in 9.4.2.26	Zero or more elements.

6.3.105.4.3 When generated

This primitive is generated by the MLME as a result of the receipt of a request to setup IP Addresses from a specific peer MAC entity.

6.3.105.4.4 Effect of receipt

The SME is notified of the receipt of this FILSContainer request.

6.3.105.5 MLME-FILSContainer.response

6.3.105.5.1 Function

This primitive is used to send a response to a specified peer MAC entity that requested IP Address setup with the STA that issued this primitive.

6.3.105.5.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-FILSContainer.response(
                              Peer MAC Address,
                              FILSIPAddressAssignment,
                              VendorSpecificInfo
                              )
```

Name	Type	Valid range	Description
Peer MAC Address	MACAddress	Any valid individual MAC address	Specifies the address of the peer MAC entity.
FILSIPAddressAssignment	FILS IP Address Assignment element	As defined in 9.4.2.185	IP address information.
VendorSpecificInfo	A set of elements	As defined in 9.4.2.26	Zero or more elements.

6.3.105.5.3 When generated

This primitive is generated by the SME of a STA as a response to an MLME-FILSContainer.indication primitive.

6.3.105.5.4 Effect of receipt

This primitive initiates transmission of a response to the specific peer MAC entity that requested IP Address setup.

9. Frame formats

9.3 Format of individual frame types

9.3.3 Management frames

Change 9.3.3.2 as follows:

9.3.3.2 Format of Management frames

A STA uses the contents of the Address 1 field to perform the address matching for receive decisions. In the case where the Address 1 field contains a group address and the frame subtype is other than Beacon or the frame subtype Action, Category Multihop Action (Multihop Action frame), the Address 3 field also is validated to verify that the group addressed frame originated from a STA in the BSS of which the receiving STA is a member or from a mesh STA to which mesh peering is maintained. Details of addressing and forwarding of the group addressed frame in an MBSS are defined in 10.35.4. When the Address 1 field contains a group address and the frame subtype is either Probe Request or Action with Category Public, a wildcard BSSID value matches all receiving STA's BSSIDs. If the frame subtype is Beacon, other address matching rules apply, as specified in 11.1.3.7. Frames of subtype Probe Request ~~with a group address in the Address 1 field~~ are additionally processed as described in 11.1.4.3.2 for non-DMG STAs and 11.1.4.3.3 for DMG STAs. If the frame subtype is Action, the Category is Public, and the Action is 20/40 BSS Coexistence Management, then additional address matching rules for receive decisions apply as specified in 11.16 and 11.18.

9.3.3.3 Beacon frame format

Change the following row in Table 9-27:

Table 9-27—Beacon frame body

Order	Information	Notes
64	Reduced Neighbor Report	The Reduced Neighbor Report element is optionally present if dot11TVHTOptionImplemented or dot11FILSActivated is true; otherwise not present.

Insert new rows/elements before the last order in Table 9-27 as follows:

Table 9-27—Beacon frame body

Order	Information	Notes
68	Common Advertisement Group (CAG) Number	The CAG Number element is optionally present if dot11-FILSActivated is true; otherwise not present.
69	FILS Indication	The FILS Indication element is present if dot11FILSActivated is true; otherwise not present.
70	AP-CSN	The AP Configuration Sequence Number (AP-CSN) element is optionally present if dot11FILSActivated is true; otherwise not present.
71	Differentiated Initial Link Setup	The Differentiated Initial Link Setup element is optionally present if dot11FILSActivated is true; otherwise not present.

9.3.3.6 Association Request frame format

Insert the following new rows before the last row in Table 9-29:

Table 9-29—Association Request frame body

Order	Information	Notes
24	FILS Session	The FILS Session element is optionally present if dot11FILSActivated is true; otherwise not present.
25	FILS Public Key	The FILS Public Key element is present if dot11FILSActivated is true and FILS Public Key authentication is used; otherwise not present.
26	FILS Key Confirmation	The FILS Key Confirmation element is present if dot11FILS-Activated is true and FILS authentication is used; otherwise not present.
27	FILS HLP Container	One or more FILS HLP Container elements are optionally present if dot11FILSActivated is true; otherwise not present.
28	FILS IP Address Assignment	The FILS IP Address Assignment element is optionally present if dot11FILSActivated is true; otherwise not present.

9.3.3.7 Association Response frame format

Insert the following new rows before the last row in Table 9-30:

Table 9-30—Association Response frame body

Order	Information	Notes
31	FILS Session	The FILS Session element is present if dot11FILSActivated is true; otherwise not present.
32	FILS Public Key	The FILS Public Key element is present if dot11FILSActivated is true and FILS Public Key authentication is used; otherwise not present.
33	FILS Key Confirmation	The FILS Key Confirmation element is present if dot11FILSActivated is true and FILS authentication is used; otherwise not present.
34	FILS HLP Container	One or more FILS HLP Container elements are optionally present if dot11FILSActivated is true; otherwise not present.
35	FILS IP Address Assignment	The FILS IP Address Assignment element is optionally present if dot11FILSActivated is true; otherwise not present.
36	Key Delivery	The Key Delivery element is present if dot11FILSActivated is true; otherwise not present.

9.3.3.8 Reassociation Request frame format

Insert the following new rows before the last row in Table 9-31:

Table 9-31—Reassociation Request frame body

Order	Information	Notes
29	FILS Session	The FILS Session element is optionally present if dot11FILSActivated is true; otherwise not present.
30	FILS Public Key	The FILS Public Key element is present if dot11FILSActivated is true and FILS Public Key authentication is used; otherwise not present.
31	FILS Key Confirmation	The FILS Key Confirmation element is present if dot11FILSActivated is true and FILS authentication is used; otherwise not present.
32	FILS HLP Container	One or more FILS HLP Container elements are optionally present if dot11FILSActivated is true; otherwise not present.
33	FILS IP Address Assignment	The FILS IP Address Assignment element is optionally present if dot11FILSActivated is true; otherwise not present.

9.3.3.9 Reassociation Response frame format

Insert the following new rows before the last row in Table 9-32:

Table 9-32—Reassociation Response frame body

Order	Information	Notes
35	FILS Session	The FILS Session element is present if dot11FILSActivated is true and FILS authentication is used; otherwise not present.
36	FILS Public Key	The FILS Public Key element is present if dot11FILSActivated is true and FILS Public Key authentication is used; otherwise not present.
37	FILS Key Confirmation	The FILS Key Confirmation element is present if dot11FILSActivated is true and FILS authentication is used; otherwise not present.
38	FILS HLP Container	One or more FILS HLP Container elements are optionally present if dot11FILSActivated is true; otherwise not present.
39	FILS IP Address Assignment	The FILS IP Address Assignment element is optionally present if dot11FILSActivated is true; otherwise not present.
40	Key Delivery	The Key Delivery element is present if dot11FILSActivated is true and FILS authentication is used; otherwise not present.

9.3.3.10 Probe Request frame format

Insert the following new rows before the last row in Table 9-33:

Table 9-33—Probe Request frame body

Order	Information	Notes
20	FILS Request Parameters	The FILS Request Parameters element is optionally present if dot11FILSActivated is true; otherwise not present.
21	AP-CSN	The AP-CSN element is optionally present if dot11FILSActivated is true; otherwise not present.

9.3.3.11 Probe Response frame format

Change row 66 in Table 9-34 as follows:

Table 9-34—Probe Response frame body

Order	Information	Notes
66	Reduced Neighbor Report	The Reduced Neighbor Report element is optionally present if dot11TVHTOptionImplemented or dot11FILSActivated is true; otherwise not present.

Insert the following new rows before the last row in Table 9-34:

Table 9-34—Probe Response frame body

Order	Information	Notes
70	CAG Number	The CAG Number element is optionally present if dot11FILSActivated is true; otherwise not present.
71	FILS Indication	The FILS Indication element is present if dot11FILSActivated is true; otherwise not present.
72	AP-CSN	The AP-CSN element is optionally present if dot11FILSActivated is true; otherwise not present.
73	Differentiated Initial Link Setup	The Differentiated Initial Link Setup element is optionally present if dot11FILSActivated is true; otherwise not present.

9.3.3.12 Authentication frame format

Change the first paragraph of 9.3.3.12 as follows:

The frame body of an Authentication frame contains the information shown in Table 9-35. FT authentication is used when FT support is advertised by the AP and dot11FastBSSTransitionActivated is true in the STA. SAE authentication is used when dot11MeshActiveAuthenticationProtocol is sae (1). FILS authentication is used if support for FILS authentication is advertised by the AP and dot11FILSActivated is true in the STA.

Change Table 9-35 as follows:

Table 9-35—Authentication frame body

Order	Information	Notes
1	Authentication algorithm number	
2	Authentication transaction sequence number	
3	Status code	The status code information is reserved in certain Authentication frames as defined in Table 9-36.
~~4~~10	Challenge text	The challenge text element is present only in certain Authentication frames as defined in Table 9-36.
~~5~~11	RSN	The RSNE is present only in certain Authentication frames as defined in Table 9-36.
~~6~~12	Mobility Domain	The MDE is present only in certain Authentication frames as defined in Table 9-36.
~~7~~13	Fast BSS Transition	An FTE is present only in certain Authentication frames as defined in Table 9-36.
~~8~~14	Timeout Interval (reassociation deadline)	A TIE containing the reassociation deadline interval is present only in certain Authentication frames as defined in Table 9-36.
~~9~~15	RIC	A resource information container, containing a variable number of elements, is present only in certain Authentication frames as defined in Table 9-36.
~~10~~4	Finite Cyclic Group	An unsigned integer indicating a finite cyclic group as described in 12.4.4. This is present only in certain Authentication frames as defined in Table 9-36.
~~11~~5	Anti-Clogging Token	A random bit- string used for anti-clogging purposes as described in 12.4.6. This is present only in certain Authentication frames as defined in Table 9-36.
~~12~~6	Send-Confirm	A binary encoding of an integer used for anti-replay purposes as described in 12.4.7.5. This is present only in certain Authentication frames as defined in Table 9-36.
~~13~~7	Scalar	An unsigned integer encoded as described in 12.4.7.4. This is present only in certain Authentication frames as defined in Table 9-36.
~~14~~8	Finite field element	A Finite field element field from a finite field encoded as described in 12.4.7.4. This is present only in certain Authentication frames as defined in Table 9-36.

Table 9-35—Authentication frame body *(continued)*

Order	Information	Notes
~~15~~9	Confirm	An unsigned integer encoded as described in 12.4.7.5. This is present only in certain Authentication frames as defined in Table 9-36.
16	Multi-band	The Multi-band element is optionally present if dot11MultibandImplemented is true.
17	Neighbor Report	One or more Neighbor Report elements is present only in certain Authentication frames as defined in Table 9-36.
Last	Vendor Specific	One or more vendor-specific elements are optionally present. These elements follow all other elements.

***Insert the following new rows before the last row in Table 9-35*:**

Table 9-35—Authentication frame body

Order	Information	Notes
18	FILS Nonce	The FILS Nonce element is present in FILS Authentication frames as defined in Table 9-36.
19	FILS Session	The FILS Session element is present in FILS Authentication frames as defined in Table 9-36.
20	FILS Wrapped Data	The FILS Wrapped Data element is present in FILS Authentication frames as defined in Table 9-36.
21	Association Delay Info	The Association Delay Info element is present in FILS Authentication frames as defined in Table 9-36.

Insert new rows at the end of Table 9-36 as follows:

Table 9-36—Presence of fields and elements in Authentication frames

Authentication algorithm	Authentication transaction sequence number	Status code	Presence of fields 4 onwards
FILS Shared Key authentication without PFS	1	Reserved	The RSNE is present. The MDE is present if the FILS authentication is used for FT initial mobility domain association. The FILS Nonce element is present. The FILS Session element is present The FILS Wrapped Data element is present.
FILS Shared Key authentication without PFS	2	Status	The RSNE is present The MDE and the FTE are present if Status Code field is 0 and FILS authentication is used for FT initial mobility domain association. The FILS Nonce element is present if Status Code field is 0. The FILS Session element is present if Status Code field is 0. The FILS Wrapped Data element is present if Status Code field is 0. The Association Delay Info element is present if Status Code field is 0 and the AP expects that the (Re)Association Response frame will be transmitted more than 1 TU after the (Re)Association Request frame.
FILS Shared Key authentication with PFS	1	Reserved	The Finite Cyclic Group field is present. The FFE field is present. The RSNE is present The MDE is present if the FILS authentication is used for FT initial mobility domain association. The FILS Nonce element is present. The FILS Session element is present. The FILS Wrapped Data element is present.

Table 9-36—Presence of fields and elements in Authentication frames *(continued)*

Authentication algorithm	Authentication transaction sequence number	Status code	Presence of fields 4 onwards
FILS Shared Key authentication with PFS	2	Status	The Finite Cyclic Group is present if Status Code field is 0. The FFE field is present if Status Code field is 0. The RSNE is present. The MDE and the FTE are present if Status Code field is 0 and FILS authentication is used for FT initial mobility domain association. The FILS Nonce element is present if Status Code field is 0. The FILS Session element is present if Status Code field is 0. The FILS Wrapped Data element is present if Status Code field is 0. The Association Delay Info element is present if Status Code field is 0 and the AP expects that the (Re)Association Response frame will be transmitted more than 1 TU after the (Re)Association Request frame.
FILS Public Key authentication	1	Reserved	The Finite Cyclic Group field is present. The FFE field is present. The RSNE is present. The MDE is present if the FILS authentication is used for FT initial mobility domain association. The FILS Nonce element is present. The FILS Session element is present.
FILS Public Key authentication	2	Status	The Finite Cyclic Group is present if Status Code field is 0. The FFE field is present if Status Code field is 0. The RSNE is present. The MDE and the FTE are present if Status Code field is 0 and FILS authentication is used for FT initial mobility domain association. The FILS Nonce element is present if Status Code field is 0. The FILS Session element is present if Status Code field is 0. The Association Delay Info element is present if Status Code field is 0 and the AP expects that the (Re)Association Response frame will be transmitted more than 1 TU after the (Re)Association Request frame.

9.4 Management and Extension frame body components

9.4.1 Fields that are not elements

9.4.1.1 Authentication Algorithm Number field

Change 9.4.1.1 as follows:

The Authentication Algorithm Number field indicates a single authentication algorithm. The length of the Authentication Algorithm Number field is 2 octets. The Authentication Algorithm Number field is illustrated in Figure 9-65. The following values are defined for authentication algorithm number:

Authentication algorithm number = 0: Open System
Authentication algorithm number = 1: Shared Key
Authentication algorithm number = 2: Fast BSS Transition
Authentication algorithm number = 3: Simultaneous Authentication of Equals (SAE)
Authentication algorithm number = 4: FILS Shared Key authentication without PFS
Authentication algorithm number = 5: FILS Shared Key authentication with PFS
Authentication algorithm number = 6: FILS Public Key authentication
Authentication algorithm number = 65 535: vendor specific use

NOTE—The use of this value implies that a Vendor Specific element is included with more information.

9.4.1.9 Status Code field

Insert new rows in Table 9-46 and change the reserved row as follows:

Table 9-46—Status codes

Status code	Name	Meaning
112	FILS_AUTHENTICATION_FAILURE	Authentication rejected due to FILS authentication failure.
113	UNKNOWN_AUTHENTICATION_SERVER	Authentication rejected due to unknown Authentication Server.
108–111 114–65 535		Reserved

9.4.1.11 Action field

Insert a new row in Table 9-47 as follows, adjusting the reserved values correspondingly:

Table 9-47—Category values

Code	Meaning	See subclause	Robust	Group addressed privacy
26	FILS	9.6.24	Yes	No

9.4.1.41 Finite field element (FFE) field

Change 9.4.1.41 as follows:

The FFE field is used with SAE authentication and FILS authentication to communicate an element in a finite field as specified in 12.4. See Figure 9-111.

9.4.1.43 Finite Cyclic Group field

Change 9.4.1.43 as follows:

The Finite Cyclic Group is used in SAE to indicate which cryptographic group to use in the SAE exchange as specified in 12.4. This field is also used in FILS to indicate which cryptographic group to use in FILS authentication as specified in 12.12. See Figure 9-113.

9.4.2 Elements

9.4.2.1 General

Insert new column and rows into Table 9-77 as follows (within the new "Fragmentable" column insert "No" for those elements not shown):

Table 9-77— Element IDs

Element	Element ID	Element ID Extension	Extensible	Fragmentable
CAG Number (see 9.4.2.177)	237	N/A	No	No
FILS Public Key (see 9.4.2.181)	255	12		Yes
AP-CSN (see 9.4.2.182)	239	N/A		No
FILS Indication (see 9.4.2.183)	240	N/A		Yes
Differentiated Initial Link Setup (see 9.4.2.187)	241	N/A	Yes	No
Fragment (see 9.4.2.189)	242	N/A		No
Association Delay Info (see 9.4.2.176)	255	1		No
FILS Request Parameters (see 9.4.2.178)	255	2		No
FILS Key Confirmation (see 9.4.2.179)	255	3		Yes
FILS Session (see 9.4.2.180)	255	4		No
FILS HLP Container (see 9.4.2.184)	255	5		Yes
FILS IP Address Assignment (see 9.4.2.185)	255	6		No
Key Delivery (see 9.4.2.186)	255	7		Yes
FILS Wrapped Data (see 9.4.2.188) ID	255	8		Yes
FILS Nonce (see 9.4.2.190)	255	13	No	No

Change the third paragraph of 9.4.2.1 as follows:

The frame body components specified for many management subtypes result in elements ordered by ascending values of the Element ID field and then the Element ID Extension field (when present), with the exception of the MIC Management element (9.4.2.55) and the Fragment element (9.4.2.189). If present, the

MIC Management element appears at the end of the robust management frame body. See 10.27.6 on the parsing of elements. If present, the Fragment element appears immediately after the element that it is fragmenting or after another Fragment element (see 10.27.11 and 10.27.12).

Insert the following new paragraph after the last paragraph of 9.4.2.1:

A “Yes” in the Fragmentable column listed in Table 9-77 indicates that the element may be fragmented (10.27.11). The element is not fragmented otherwise.

9.4.2.25 RSNE

9.4.2.25.3 AKM suites

Insert new rows and change the reserved row in Table 9-133 as follows:

Table 9-133—AKM suite selectors

OUI	Suite type	Meaning		
		Authentication type	Key management type	Key derivation type
00-0F-AC	14	Key management over FILS using SHA-256 and AES-SIV-256	FILS key management defined in 12.12.2.5	Defined in 12.12.2.5 using SHA-256.
00-0F-AC	15	Key management over FILS using SHA-384 and AES-SIV-512	FILS key management defined in 12.12.2.5	Defined in 12.12.2.5 using SHA-384.
00-0F-AC	16	FT authentication over FILS with SHA-256 and AES-SIV-256	FT authentication defined in 12.7.1.7.2	Defined in 12.7.1.7.2 using SHA-256.
00-0F-AC	17	FT authentication over FILS with SHA-384 and AES-SIV-512	FT authentication defined in 12.7.1.7.2	Defined in 12.7.1.7.2 using SHA-384.
00-0F-AC	~~14~~18–255	Reserved	Reserved	Reserved

Insert the following note at the end of 9.4.2.25.3:

NOTE 4—Selector values 00-0F-AC:14, 00-0F-AC:15, 00-0F-AC:16, and 00-0F-AC:17 are used only with FILS authentication (Authentication algorithm number values 4, 5, and 6).

9.4.2.27 Extended Capabilities element

Insert a new row and change the reserved row in Table 9-135 as follows:

Table 9-135—Extended Capabilities field

Bit	**Information**	**Notes**
72	FILS Capability	The FILS Capability field is set to 1 if dot11FILSActivated is true. Otherwise, the FILS Capability field is 0.
~~72,~~ 75–*n*	Reserved	

9.4.2.48 Fast BSS Transition element (FTE)

Change 9.4.2.48 as follows:

The FTE includes information needed to perform the FT authentication sequence or FILS authentication during a fast BSS transition in an RSN. This element is shown in Figure 9-315.

9.4.2.171 Reduced Neighbor Report element

9.4.2.171.1 Neighbor AP Information field

Change 9.4.2.171.1 as follows:

The TBTT Information Field Type subfield is 2 bits in length and defines the structure of the TBTT Information field. ~~Its value is 0.~~ Values ~~1,~~ 2 and 3 are reserved.

The Filtered Neighbor AP subfield is 1 bit in length. When included in the Probe Response frame, ~~I~~it is set to 1 if the SSID of APs in this Neighbor AP Information field matches the specific SSID in the corresponding Probe Request frame. When included in the Beacon frame, it is set to 1 if the SSID of APs in this Neighbor AP Information field matches the specific SSID in the containing Beacon frame. It is set to 0 otherwise.~~This field is valid only in the Reduced Neighbor Report element in a Probe Response frame and is reserved otherwise.~~

The TBTT Information Count subfield is 4 bits in length and contains the number of TBTT Information fields that are included in the Neighbor AP Information field, minus one. A value of 0 indicates one TBTT Information field is present.

The TBTT Information Length subfield is 1 octet in length and contains the length in octets of each TBTT Information field that is included in the Neighbor AP Information field. The TBTT Information Length subfield is 1, 5, 7, or 11 indicating the TBTT Information field contents. Other values are reserved.

The TBTT Information Length subfield is interpreted as shown in Table 9-258a.

Table 9-258a—TBTT Information field

TBTT Information Length subfield value	TBTT Information field contents
1	The Neighbor AP TBTT Offset subfield
5	The Neighbor AP TBTT Offset subfield and the Short-SSID subfield
7	The Neighbor AP TBTT Offset subfield and the BSSID subfield
11	The Neighbor AP TBTT Offset subfield, the BSSID subfield and the Short-SSID subfield
0, 2–4, 6, 8–10, 12–255	Reserved

The Operating Class field is 1 octet in length and indicates a channel starting frequency that, together with the Channel Number field, indicates the primary channel of the BSSs of the APs in this Neighbor AP Information field. Values of Operating Class are shown in Table E-4, of which operating classes that, together with the channel number, indicate the primary channel is valid (see 11.44.8).

NOTE—The Operating Class field and Channel Number tuple indicate the primary channel in order to assist with passive scanning.

The Channel Number field is 1 octet in length and indicates the last known primary channel of the APs in this Neighbor AP Information field. Channel Number is defined within an Operating Class as shown in Table E-4.

The TBTT Information Set field contains one or more TBTT Information fields. The TBTT Information field is defined in Figure 9-583.

	Neighbor AP TBTT Offset	BSSID (conditional)	Short-SSID (conditional)
Octets:	1	0 or 6	0 or 4

Figure 9-583—TBTT Information field format

The Neighbor AP TBTT Offset subfield is 1 octet in length and indicates the offset in TUs, rounded down to nearest TU, to the next TBTT of an AP from the immediately prior TBTT of the AP that transmits this element. The value 254 indicates an offset of 254 TUs or higher. The value 255 indicates an unknown offset value.

The BSSID is defined in 9.2.4.3.4.

The Short-SSID subfield is calculated as given in 9.4.2.171.2.

Insert new subclause 9.4.2.171.2 as follows:

9.4.2.171.2 Calculating the Short-SSID

The Short-SSID field is a 32-bit field. The Short-SSID is calculated over the SSID. The SSID is referred to as the calculation fields.

The Short-SSID is calculated using the following standard generator polynomial of degree 32:

$$G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^{8} + x^{7} + x^{5} + x^{4} + x^{2} + x + 1$$

The Short-SSID is the ones complement of the sum (modulo 2) of the following:

a) The remainder of $x^k \times (x^{31} + x^{30} + x^{29} + \ldots + x^2 + x + 1)$ divided (modulo 2) by $G(x)$, where k is the number of bits in the calculation fields, and

b) The remainder after multiplication of the contents (treated as a polynomial) of the calculation fields by x^{32} and then division by $G(x)$.

The Short-SSID field is transmitted commencing with the coefficient of the highest-order term.

Insert new subclauses as follows:

9.4.2.176 Association Delay Info element

The Association Delay Info element is used to identify the minimum (re)association response timeout to the non-AP STA. The format of the Association Delay Info element is shown in Figure 9-589a.

	Element ID	Length	Element ID Extension	Association Delay Info
Octets:	1	1	1	1

Figure 9-589a—Association Delay Info element format

The Element ID, Length, and Element ID Extension fields are defined in 9.4.2.1.

The Association Delay Info is a 1-octet field whose value is an unsigned integer indicating the minimum association response delay in number of TUs.

9.4.2.177 CAG Number element

The Common Advertisement Group (CAG) is a group of elements that are defined by the same advertisement protocol and that do not change on a rapid basis within an AP. The CAG Number element provides one or more current version numbers of the CAG (CAG Version) associated with the AP, where each version number is associated with a specific advertisement protocol. The CAG Number element is optionally present in the Beacon or Probe Response frame to reduce GAS frame exchanges when dot11InterworkingServiceActivated is true.

The CAG Number element is shown in Figure 9-589b.

	Element ID	Length	CAG Tuple #1	CAG Tuple #2 (optional)	...	CAG Tuple #N (optional)
Octets:	1	1	2	0 or 2		0 or 2

Figure 9-589b—CAG Number element format

The Element ID and Length fields are defined in 9.4.2.1.

One or more 2-octet CAG Tuple fields are used. The format of a CAG Tuple field is shown in Figure 9-589c.

	B0 B7	B8 B15
	CAG Version	Advertisement Protocol ID
Bits:	8	8

Figure 9-589c—CAG Tuple field

The CAG Version subfield is an unsigned integer indicating the current version of the CAG of the associated advertisement protocol. The use of CAG Version is explained in 11.25.3.3.

The Advertisement Protocol ID subfield is a 8-bit subfield and carries a value equal to the Advertisement Protocol ID of the advertisement protocol associated with the CAG Version in the same CAG Tuple field. The Advertisement Protocol ID is defined in Table 9-214 in 9.4.2.93.

9.4.2.178 FILS Request Parameters element

The contents of the FILS Request Parameters element in Probe Request frame are used in determining whether to transmit a Probe Response frame as described in 11.1.4.3.4. The FILS Request Parameters element is defined in Figure 9-589d.

	Element ID	Length	Element ID Extension	Parameter Control Bitmap	Max Channel time
Octets:	1	1	1	1	1

.

	FILS Criteria	Max Delay Limit	Minimum Data Rate	RCPI Limit	OUI Response Criteria
Octets:	1	0 or 1	0 or 3	0 or 1	0 or 2

Figure 9-589d—FILS Request Parameters element format

The Element ID, Length, and Element ID Extension fields are defined in 9.4.2.1.

The Parameter Control Bitmap field is 1 octet in length and illustrated in Figure 9-589e.

	B0	B1	B2	B3
	FILS Criteria Present	Max Delay Limit Present	Minimum Data Rate Present	RCPI Limit Present
Bits:	1	1	1	1

	B4	B5 B7
	OUI Response Criteria Present	Reserved
Bits:	1	2

Figure 9-589e—Parameter Control Bitmap field

Bits 0 to 4 of the Parameter Control Bitmap field correspond to the Parameter fields that are conditionally present in the element. A value of 1 in a bit indicates the corresponding parameter is present, and a value of 0 indicates the corresponding parameter is not present.

The FILS Criteria field is 1 octet in length and is illustrated in Figure 9-589f.

	B0 B2	B3 B5	B6 B7
	BSS Delay Criterion	PHY Support Criterion	Reserved
Bits:	3	3	2

Figure 9-589f—FILS Criteria field

The BSS Delay Criterion subfield indicates the delay type that is applied in the decision to respond to the Probe Request frame as described in 11.1.4.3.4. The delay type is selected as indicated in Table 9-262a.

Table 9-262a—BSS Delay Criterion subfield

Value	Explanation
0	Delay criterion is not in use.
1	Access delay is indicated as Average Access Delay for Background (AC_BK) subfield of the BSS AC Access Delay element as described in 9.4.2.44.
2	Access delay is indicated as Average Access Delay for Best Effort (AC_BE) subfield of the BSS AC Access Delay element as described in 9.4.2.44.
3	Access delay is indicated as Average Access Delay for Video (AC_VI) subfield of the BSS AC Access Delay element as described in 9.4.2.44.
4	Access delay is indicated as Average Access Delay for Voice (AC_VO) subfield of the BSS AC Access Delay element as described in 9.4.2.44.
5	Access Delay is indicated as Average Access Delay as described in 9.4.2.44.
6, 7	Reserved

The PHY Support Criterion subfield indicates the required PHY type of the responding STA.

The indicated PHY type is used in the decision to respond to the Probe Request frame as described in 11.1.4.3.4. The meaning of the values for the PHY Support Criterion is shown in Table 9-262b.

Table 9-262b—PHY Support Criterion subfield

Value	Explanation
0	Indicates that PHY Support Criterion is not in use.
1	Indicates that a responding FILS STA is HT capable.
2	Indicates that a responding FILS STA is VHT capable.
3–7	Reserved

The Max Delay Limit field (see 11.1.4.3.4) is an unsigned integer in units of 400 µs to indicate the value of the maximum access delay as indicated by the BSS Delay Criterion subfield of the FILS Criteria of the FILS Request Parameters element. Value 0 is reserved. The use of the maximum access delay and the delay criterion are explained in 11.1.4.3.4. Max Delay Limit is not present if FILS Criteria is not present or BSS Delay Criterion is not in use.

The Minimum Data Rate field (see 11.1.4.3.4) is 3 octets long and contains an unsigned integer in units of kilobits per second that specifies the lowest total data rate specified at the MAC SAP for transport of MSDUs or A-MSDUs that the STA is going to transmit. The minimum MAC SAP data rate does not include the MAC and PHY overheads incurred in transferring the MSDUs or A-MSDUs.

The RCPI Limit field is an unsigned integer in units of 1 dB. The use of the RCPI Limit field is explained in 11.1.4.3.4.

OUI Response Criteria field (see 11.1.4.3.5) is a bitmap whose bits correspond to the Vendor Specific elements of the Probe Request frame in order of presence. Bit 0 corresponds to the first Vendor Specific element, bit 1 corresponds to the second, etc. A bit value of 1 in the OUI Response Criteria field indicates that the receiver identifies the Organization Identifier field of the corresponding Vendor Specific element in order to respond to the request and otherwise is 0. If the number of the Vendor Specific elements of the Probe Request frame is less than the number of bits of the OUI Response Criteria field, the remaining bits of the OUI Response Criteria field are 0. Value 0 is applied for the Vendor Specific elements that are not in the range of the OUI Response Criteria.

The Max Channel Time field (see 11.1.4.3.5) contains the value of MaxChannelTime parameter of the MLME-SCAN.request primitive represented in units of TUs, as an unsigned integer. A Max Channel Time field value of 255 is used to indicate any duration of more than 254 TUs, or an unspecified or unknown duration.

9.4.2.179 FILS Key Confirmation element

The FILS Key Confirmation element is used to convey a cryptographic proof of authentication between a STA and an AP. The format of the FILS Key Confirmation element is shown in Figure 9-589g.

	Element ID	Length	Element ID Extension	KeyAuth
Octets:	1	1	1	variable

Figure 9-589g—FILS Key Confirmation element format

The Element ID, Length, and Element ID Extension fields are defined in 9.4.2.1.

The KeyAuth field contains the cryptographic authentication information (see 12.12.2.6).

9.4.2.180 FILS Session element

The FILS Session element is used to convey the (unique) identifier of an in-progress FILS authentication protocol session. The format of the FILS Session element is shown in Figure 9-589h.

	Element ID	Length	Element ID Extension	FILS Session
Octets:	1	1	1	8

Figure 9-589h—FILS Session element format

The Element ID, Length, and Element ID Extension fields are defined in 9.4.2.1.

The FILS Session field is chosen randomly by the non-AP STA.

9.4.2.181 FILS Public Key element

The FILS Public Key element is used to communicate the device's (certified) public key for use with the FILS authentication exchange. The format of the FILS Public Key element is shown in Figure 9-589i.

	Element ID	Length	Element ID Extension	Key Type	FILS Public Key
Octets:	1	1	1	1	variable

Figure 9-589i—FILS Public Key element format

The Element ID Length, and Element ID Extension fields are defined in 9.4.2.1.

The Key Type field value determines the contents of the FILS Public Key field as follows:

0: Reserved, FILS Public Key is undefined.
1: FILS Public Key is an X.509v3 certificate encoded according to IETF RFC 5280.
2: FILS Public Key is an uncertified public key encoded according to IETF RFC 5480.
3: FILS Public Key is an uncertified public key encoded according to IETF RFC 3279.

9.4.2.182 AP Configuration Sequence Number (AP-CSN) element

An AP-CSN element indicates the change of system information within a BSS. The format of the AP-CSN element is shown in Figure 9-589j.

	Element ID	Length	AP-CSN
Octets:	1	1	1

Figure 9-589j—AP-CSN element format

The Element ID and Length fields are defined in 9.4.2.1.

The AP-CSN field is 1 octet in length and is defined as an unsigned integer. The AP-CSN contains the version number of the BSS Configuration Parameter Set. This value increments when an update of the non-dynamic information elements (all elements except dynamic elements indicated in 11.1.4.3.7) inside a Beacon frame has occurred, as described in 11.1.4.3.7.

9.4.2.183 FILS Indication element

The FILS Indication element shown in Figure 9-589k contains information related to FILS authentication and higher layer setup capabilities of the AP.

The Element ID and Length fields are defined in 9.4.2.1.

	Element ID	Length	FILS Information	Cache Identifier	HESSID	Realm Identifier	Public Key Identifier
Octets:	1	1	2	0 or 2	0 or 6	variable	variable

Figure 9-589k—FILS Indication element format

The FILS Information field provides information on the presence of the following optional fields in the FILS Indication element and the capability of the FILS authentication algorithms. The format of the FILS Information field is shown in Figure 9-589l.

	B0 B2	B3 B5	B6	B7	B8	B9	B10	B11	B12 B15
	Number of Public Key Identifiers	Number of Realm Identifiers	FILS IP Address Configuration	Cache Identifier Included	HESSID Included	FILS Shared Key Authentication without PFS Supported	FILS Shared Key Authentication with PFS Supported	FILS Public Key Authentication Supported	Reserved
Bits:	3	3	1	1	1	1	1	1	4

Figure 9-589l—FILS Information field definition

The Number of Public Key Identifiers subfield lists the number of Public Key Identifiers that are present in the Public Key Identifiers field in the FILS Indication element. When the Number of Public Key Identifiers subfield is 0, the Public Key Identifier field is not present in the FILS Indication element. Each Public Key Identifier is formatted per Figure 9-589n. Up to seven Public Key Identifiers fields may be carried in a FILS Indication element.

The Number of Realm Identifiers subfield lists the number of realm identifiers that are present in the Realm Identifier field in the FILS Indication element. When the Number of Realm Identifiers subfield is 0, the Realm Identifier field is not present in the FILS Indication element. Each realm identifier is formatted per Figure 9-589m. Up to seven Realm Identifiers fields may be carried in FILS Indication element.

An AP sets the FILS IP Address Configuration bit to 1 if the AP supports FILS IP address configuration and is set to 0 otherwise.

The Cache Identifier Included bit is set in the FILS Information field when PMKSA caching is supported. When the Cache Identifier Included bit is 1, a 2-octet Cache Identifier field is present in the FILS Indication element. When the Cache Identifier Included bit is 0 the Cache Identifier field is not present in the FILS Indication element. The content of the Cache Identifier field is an opaque octet string that identifies the scope in which PMKSAs are cached.The assignment of the cache identifier is outside the scope of the standard but its value must be unique per authenticator within an ESS. On the AP, dot11CacheIdentifier contains the value of the Cache Identifier.

When the HESSID Included bit is 1, a 6-octet HESSID field is present in the FILS Indication element. When the HESSID Included bit is 0, the HESSID field is not present in the FILS Indication element. The HESSID field is set to the value of dot11HESSID.

An AP sets the FILS Shared Key authentication without PFS Supported bit to 1 if the AP supports FILS Shared Key authentication without PFS and sets it to 0 otherwise. An AP sets the FILS Shared Key authentication with PFS Supported bit to 1 if the AP supports FILS Shared Key authentication with PFS and sets it to 0 otherwise. An AP sets the FILS Public Key Authentication Supported bit to 1 if the AP supports FILS Public Key authentication and sets it to 0 otherwise.

	Hashed Realm
Octets:	2

Figure 9-589m—Realm Identifier field

The value of the Hashed Realm subfield of the Realm Identifier field entry is computed from the realm that is compliant with the preferred name syntax defined in IETF RFC 1035 (same as the domain name used in 9.4.5.15). The exact computation method for the hashed realm is given in 11.47.4.

	Key Type	Length	Public Key Indicator
Octets:	1	1	variable

Figure 9-589n—Public Key Identifier field

The Key Type and Public Key Indicator subfields are described in Table 9-262c. The Length subfield indicates the length in octets of the Public Key Indicator subfield.

Table 9-262c—Key Type and Public Key Indicator subfields

Key Type subfield	Public Key Indicator subfield
0	Reserved
1	The Issuer, per IETF RFC 5280, of the AP’s certificate
2	A SHA-256 hash of the AP’s uncertified IETF RFC 5480 public key
3	A SHA-256 hash of the AP's uncertified IETF RFC 3279 public key
4–255	Reserved

9.4.2.184 FILS HLP Container element

The FILS HLP Container element contains higher layer protocol (HLP) packets transported during FILS association. If dot11FILSActivated is true, one or more FILS HLP Container elements may be included in a (Re)Association Request or Response frame. This element is used for higher layer protocol packet encapsulation (11.47.3.2). The format of the FILS HLP Container element is shown in Figure 9-589o.

	Element ID	Length	Element ID Extension	Destination MAC Address	Source MAC Address	HLP Packet
Octets:	1	1	1	6	6	variable

Figure 9-589o—FILS HLP Container element format

The Element ID, Length, and Element ID Extension fields are defined in 9.4.2.1.

The Destination MAC Address field and the Source MAC Address field are set as described in 11.47.3.2.

The HLP Packet field contains the HLP packet.

9.4.2.185 FILS IP Address Assignment element

9.4.2.185.1 General

The FILS IP Address Assignment element is used by a STA to request or to assign an IP address using FILS IP Address Configuration (11.47.3.3). If dot11FILSActivated is true, a FILS IP Address Assignment element may be sent in a (Re)Association Request/Response frame or a FILS Container frame. The format of the FILS IP Address Assignment element is shown in Figure 9-589p.

	Element ID	Length	Element ID Extension	IP Address Data
Octets:	1	1	1	variable

Figure 9-589p—FILS IP Address Assignment element format

The Element ID, Length, and Element ID Extension fields are defined in 9.4.2.1.

The value of the IP Address Data field in (Re)Association Request and FILS Container frames from a non-AP STA to an AP is described in 9.4.2.185.2. The value of the IP Address Data field in (Re)Association Response frame and FILS Container frames from an AP to a non-AP STA is described in 9.4.2.185.3.

9.4.2.185.2 IP Address Data field for request

The format of the IP Address Data field for request is shown in Figure 9-589q.

	IP Address Request Control	Requested IPv4 Address (optional)	Requested IPv6 Address (optional)
Octets:	1	0 or 4	0 or 16

Figure 9-589q— IP Address Data field for request format

The format of the IP Address Request Control subfield is shown in Figure 9-589r.

	B0 B1	B2 B3	B4	B5 B7
	IPv4	IPv6	DNS Server Address Request	Reserved
Bits:	2	2	1	3

Figure 9-589r—IP Address Request Control subfield format

The IPv4 subfields are set as shown in Table 9-262d. The IPv6 subfields are set as shown in Table 9-262e.

Table 9-262d—IPv4 subfield settings

IPv4 subfield	Explanation
0	STA is not requesting an IPv4 address
1	Reserved
2	STA is requesting a new IPv4 address
3	STA is requesting the IPv4 address present in the element

Table 9-262e—IPv6 subfield settings

IPv6 subfield	Explanation
0	STA is not requesting an IPv6 address
1	Reserved
2	STA is requesting a new IPv6 address
3	STA is requesting the IPv6 address present in the element

The DNS Server Address Request subfield is 1 if the STA is requesting DNS server(s) address(es). The type of DNS server requested corresponds to the type of the IP address requested. If both IPv4 and IPv6 are requested, then DNS server addresses for both types are also requested by setting this bit to 1.

The Requested IPv4 Address field (4 octets) carries the specific IPv4 address that the non-AP STA is requesting, when the IPv4 field indicates the STA is requesting a specific IPv4 address.

The Requested IPv6 Address field (16 octets) carries the specific IPv6 address that the non-AP STA is requesting, when the IPv6 field indicates the STA is requesting a specific IPv6 address.

9.4.2.185.3 IP Address Data field for response

The format of the IP Address Data field for response is shown in Figure 9-589s.

NOTE—IPv4 addressing is described in IETF RFC 791. IP Subnet and Subnet Mask is described IETF RFC 917. IPv6 addressing, IPv6 prefix and prefix-length is described in IETF RFC 4291. A default gateway in computer networking is a device that is assumed to know how to forward packets on to other networks or the Internet. IPv4 Gateway Address is the IPv4 address of the default gateway when IPv4 is used. IPv6 Gateway Address is the IPv6 address of the default gateway when IPv6 is used. The IPv6 Gateway MAC address is the MAC address of the IPv6 default gateway.

	IP Address Response Control	DNS Info Control	Assigned IPv4 Address (optional)	Subnet Mask (optional)	IPv4 Gateway Address (optional)
Octets:	1	1	0 or 4	0 or 4	0 or 4

	IPv4 Gateway MAC Address (optional)	Assigned IPv6 Address (optional)	IPv6 Prefix Length (optional)	IPv6 Gateway Address (optional)
Octets:	0 or 6	0 or 16	0 or 1	0 or 16

	IPv6 Gateway MAC Address (optional)	Lifetime of the Assigned IPv4 Address (optional)	Lifetime of the Assigned IPv6 Address (optional)	DNS Server IPv4 Address (optional)
Octets:	0 or 6	0 or 1	0 or 1	0 or 4

	DNS Server IPv6 Address (optional)	IPv4 DNS Server MAC Address (optional)	IPv6 DNS Server MAC Address (optional)
Octets:	0 or 16	0 or 6	0 or 6

Figure 9-589s—IP Address Data field format for response

Subfields of the IP Address Response Control field (8 bits) are interpreted dependent on the value of B0 (IP Address pending) as defined in Table 9-262f and Table 9-262g.

Table 9-262f—IP Address Response Control subfield with B0 = 0

Bit	Function of the subfield	Explanation
B0	IP address pending	Set to 0 if an IP address assignment is included in the frame. B1 to B6 are set as shown below in this table when B0 = 0.
B1	IPv4 assigned	Set to 1 if the Assigned IPv4 Address subfields are included in the element, and set to 0 otherwise.
B2	IPv4 gateway included	Set to 1 if IPv4 Gateway Address and IPv4 Gateway MAC Address subfields are included in the element and set to 0 otherwise.
B3	IPv6 assigned	Set to 1 if Assigned IPv6 Address and IPv6 Prefix Length subfields are included in the element and set to 0 otherwise.
B4	IPv6 gateway included	Sets to 1 if IPv6 Gateway Address and IPv6 Gateway MAC Address subfields are included in the element and set to 0 otherwise.
B5	Lifetime of the assigned IPv4 address included	Set to 1 if Assigned IPv4 Address subfield is present and the Lifetime of the Assigned IPv4 Address is included in the element. If this bit is 0, and if Assigned IPv4 Address subfield is present, then the assigned IPv4 address is valid during the entire time of association with the AP.
B6	Lifetime of the assigned IPv6 address included	Set to 1 if Assigned IPv6 Address subfield is present and the Lifetime of the Assigned IPv6 Address is included in the element. If this bit is 0, and if Assigned IPv6 Address subfield is present, then the assigned IPv6 address is valid during the entire time of association with the AP.
B7	Reserved	

Table 9-262g—IP Address Response Control subfield with B0 = 1

Bit	Function of the subfield	Explanation
B0	IP address pending	Set to 1 if an IP address is sent in a later transmission. An IP address is NOT present in the frame when set to 1. B1 to B6 are set as shown below in this table when B0 = 1.
B1–B6	IP address request timeout	IP address request timeout value is the maximum estimated time in the unit of seconds within which an IP address may be assigned and a DNS server address maybe provided to the requesting STA. The value of 0 is reserved.
B7	Reserved	

The format of the DNS Info Control subfield is shown in Figure 9-589t and the DNS Info Control subfield settings are shown in Table 9-262h.

	B0	B1	B2	B3	B4 B7
	DNS Server IPv4 Address Present	DNS Server IPv6 Address Present	IPv4 DNS Server MAC Address Present	IPv6 DNS Server MAC Address Present	Reserved
Bits:	1	1	1	1	4

Figure 9-589t—DNS Info Control subfield format

Table 9-262h—DNS Info Control subfield settings

Bit subfield	Function of the subfield	Explanation
B0	DNS server IPv4 address included	Set to 1 if the DNS Server IPv4 Address subfield is present in the element and set to 0 otherwise. The value of the DNS Server IPv4 Address subfield is the IPv4 address of the DNS server if the DNS Server IPv4 address Present bit of the DNS Info Control subfield is 1.
B1	DNS server IPv6 address included	Set to 1 if the DNS Server IPv6 Address subfield is present in the element and set to 0 otherwise. The value of the DNS Server IPv6Address is the IPv6 address of the DNS server if the DNS Server IPv6 address Present bit of the DNS Info Control is 1.
B2	IPv4 DNS server MAC address included	Set to 1 if the MAC address to which IPv4 based DNS queries can be sent is present in the element and set to 0 otherwise. The value of the IPv4 DNS Server MAC Address subfield is the MAC address of the IPv4 DNS server if the IPv4 DNS Server MAC Address Present bit of the DNS Info Control subfield is 1.
B3	IPv6 DNS server MAC address included	Set to 1 if the MAC address to which IPv6 based DNS queries can be sent is present in the element and set to 0 otherwise. The value of the IPv6 DNS Server MAC Address subfield is the MAC address of the IPv6 DNS server if the IPv6 DNS Server MAC Address Present bit of the DNS Info Control subfield is 1.

9.4.2.186 Key Delivery element

Key Delivery element contains the current Key RSC and one or more KDEs. This is used to communicate the Key RSC and one or more KDEs in a FILS authentication exchange. The format of the Key Delivery element is shown in Figure 9-589u.

	Element ID	Length	Element ID Extension	Key RSC	KDE List
Octets:	1	1	1	8	variable

Figure 9-589u—Key Delivery element format

The Element ID, Length, and Element ID Extension fields are defined in 9.4.2.1.

The Key RSC field contains the receive sequence counter for the GTK being installed.

The KDE List field contains one or more KDEs (see Figure 12-34) onwards in 12.7.2.

9.4.2.187 Differentiated Initial Link Setup element

The Differentiated Initial Link Setup element includes the conditions for a STA to determine whether it is allowed to attempt fast initial link setup for the duration specified in the element. The Differentiated Initial Link Setup element is optionally present in the Beacon and Probe Response frames. The Differentiated Initial Link Setup element is defined in Figure 9-589v.

	Element ID	Length	Differentiated FILS Time	FILSC Type	FILS User Priority (optional)	MAC Address Filter (optional)
Octets:	1	1	1	1	0 or 1	0 or 1

Figure 9-589v—Differentiated Initial Link Setup element format

The Element ID and Length fields are defined in 9.4.2.1.

The Differentiated FILS Time field contains an unsigned integer that specifies the time duration for the validity of fast initial link setup category (FILSC) Information priority condition, expressed in units of 10 ms, starting from the beginning of the frame transmission of the Differentiated Initial Link Setup element.

The FILS category (FILSC) Type field (see Figure 9-589w) indicates the presence of optional fields following it.

	B0	B1	B2 B7
	FILS User Priority Present	MAC Address Filter Present	Reserved
Bits:	1	1	6

Figure 9-589w—FILSC Type field format

A value of 1 in the FILS User Priority Present or MAC Address Filter Present subfields indicates that the corresponding field is present. At least one of the bits in FILSC Type field is set to 1.

The FILS User Priority field is defined in Figure 9-589x.

	B0	B1	B2	B3 B7
	FILS User Priority Bit 0	FILS User Priority Bit 1	FILS User Priority Bit 2	Reserved
Bits:	1	1	1	5

Figure 9-589x—FILS User Priority field format

FILS User Priority Bit 0 subfield of 1 indicates high priority link setup without additional delays for the STAs that have frames with User Priority 4–7 in their transmission queues, FILS User Priority Bit 1 subfield of 1 indicates high priority for the STAs that have frames with User Priority 0–3 in their transmission queues. FILS User Priority Bit 2 subfield of 1 indicates high priority for the STAs that have no frames in their transmission queues. See 11.47.5.2.

The MAC Address Filter field is 1 octet in length as shown in Figure 9-589y.

	B0 B2	B3 B7
	Bit Pattern Length	Bit Pattern
Bits:	3	5

Figure 9-589y—MAC Address Filter field

The usage of the Bit Pattern Length subfield and Bit Pattern subfield is defined in Table 9-262i. The Bit Pattern Length subfield specifies the number of bits and the position of the bits in the Bit Pattern subfield that are used for MAC address filtering. The Bit Pattern subfield specifies the MAC addresses of the STAs that are allowed to attempt fast initial link setup. The details of MAC address filtering is described in 11.47.5.3.

Table 9-262i—MAC Address Filter field

Bit Pattern Length value B0 B1 B2	Bit Pattern				
	B3	B4	B5	B6	B7
1	Reserved	Reserved	Reserved	Reserved	Used for MAC address filtering
2	Reserved	Reserved	Reserved	Used for MAC address filtering	
3	Reserved	Reserved	Used for MAC address filtering		
4	Reserved	Used for MAC address filtering			
5	Used for MAC address filtering				
0	Reserved				
6–7	Reserved				

9.4.2.188 FILS Wrapped Data element

The FILS Wrapped Data element is used for the STA and AP to communicate data used by the FILS authentication algorithm. The format of the FILS Wrapped Data element is defined in Figure 9-589z.

	Element ID	Length	Element ID Extension	FILS Wrapped Data
Octets:	1	1	1	variable

Figure 9-589z—FILS Wrapped Data element format

The Element ID, Length, and Element ID Extension fields are defined in 9.4.2.1.

The FILS Wrapped Data field is the data used by the FILS authentication algorithm (see 12.12).

9.4.2.189 Fragment element

The payload of each element is limited to a maximum of 254 or 255 octets since the Length field is a single octet and the possible Element ID Extension field is one octet in length (see Figure 9-121). If information to be represented in an element is too large, it is necessary to fragment the information as described in 10.27.11 and 10.27.12. The format of the Fragment element is indicated in Figure 9-589aa.

	Element ID	Length	Fragmented Data
Octets:	1	1	variable

Figure 9-589aa—Fragment element format

The Element ID and Length fields are defined in 9.4.2.1.

The Fragmented Data field contains the data of the element that is fragmented as described in 10.27.11.

For the information being fragmented, each Fragment element contains 255 octets, except the final Fragment element that contains 1 to 255 octets.

9.4.2.190 FILS Nonce element

The FILS Nonce element is used for exchanging an additional source of randomness in the FILS authentication exchange. The format of the FILS Nonce element is shown in Figure 9-589ab.

	Element ID	Length	Element ID Extension	FILS Nonce
Octets:	1	1	1	16

Figure 9-589ab—FILS Nonce element format

The Element ID, Length, and Element ID Extension fields are defined in 9.4.2.1.

The FILS Nonce field contains randomly generated data.

9.4.5 Access network query protocol (ANQP) elements

9.4.5.1 General

Insert four new rows after "Neighbor Report" and change the reserved row in Table 9-271 as follows:

Table 9-271—ANQP-element definitions

ANQP-element name	Info ID	ANQP-element (subclause)
Query AP List	273	9.4.5.24
AP List Response	274	9.4.5.25
FILS Realm Information	275	9.4.5.26
CAG	276	9.4.5.27
Reserved	~~273–276,~~281–56 796	n/a

Insert new subclause 9.4.5.24 as follows:

9.4.5.24 Query AP List ANQP-element

The Query AP List ANQP-element provides a list of APs and a list of identifiers of ANQP-elements that the requesting STA is querying. The Query AP List ANQP-element declares that the STA performing the ANQP query is requesting the ANQP-element corresponding to that Info ID be returned in the ANQP query response. This element allows an optimization of the ANQP query procedure by having multiple queries in a single ANQP query list thus reducing the time necessary for network discovery and selection. Each ANQP-element can be returned in response to Query AP List ANQP-element using the procedures in 11.25.3.2.14.

The format of the Query AP List ANQP-element is provided in Figure 9-628a.

	Info ID	Length	AP List	ANQP Query ID #1	...	ANQP Query ID #N (optional)
Octets:	2	2	variable	2		0 or 2

Figure 9-628a—Query AP List ANQP-element format

The Info ID and Length fields are defined in 9.4.5.1.

The ANQP Query IDs are ordered by increasing info ID value.The AP List field is a variable length field defined in Figure 9-628b that contains the list of BSSIDs for requested information.

	AP List Length	BSSID	...	BSSID
Octets:	1	6		6

Figure 9-628b—AP List field format

The AP List Length subfield is a 1-octet field whose value indicates the total number of the subsequent BSSID subfields.

Each BSSID subfield takes 6 octets to indicate the BSSID of an AP that the requesting STA wants to query.

Each ANQP Query ID field value is drawn from Table 9-271.

Insert the new subclause 9.4.5.25 as follows:

9.4.5.25 AP List Response ANQP-element

The AP List Response ANQP-element provides the response to the Query AP List ANQP-element request. The ANQP query response is defined in 11.25.3.3. The frame format of the response frame is defined in Figure 9-628c.

	Info ID	Length	AP 1 Identifier	AP 1 Response Length	AP 1 Query Response	...	AP N Identifier	AP N Response Length	AP N Query Response
Octets:	2	2	6	2	variable		6	2	variable

Figure 9-628c—AP List Response ANQP-element format

The Info ID and Length fields are defined in 9.4.5.1.

Each AP N Identifier field takes 6 octets to indicate the BSSID of the AP that the requesting STA queries.

Each AP N Response Length field is a 2-octet field whose value is the number of octets in the following AP Query Response field(s).

Each AP N Query Response field is a container that contains one or multiple ANQP-elements (Figure 9-588) that correspond to the ANQP response to the received Query AP List ANQP-element as specified in 9.4.5.24. This field is also formatted in accordance with ANQP. This field can contain one or more values of the ANQP attributes that are specific for a particular AP and were requested by a STA via the Query AP List ANQP-element.

Insert new subclause 9.4.5.26 as follows:

9.4.5.26 FILS Realm Information ANQP-element

The FILS Realm Information ANQP-element provides a list of information about the domains.

	Info ID	Length	Realm Identifier #1	...	Realm Identifier #N
Octets:	2	2	2		2

Figure 9-628d—FILS Realm Information ANQP-element format

The Info ID field and Length fields are defined in 9.4.5.1.

The Realm Identifier field is defined in Figure 9-589m.

Insert new subclause 9.4.5.27 as follows:

9.4.5.27 CAG ANQP-element

The CAG ANQP-element provides the info IDs for the ANQP-elements contained within a CAG associated with ANQP and the current value of the ANQP CAG version, indicating the version of information within the CAG associated with ANQP. The selection of the specific number of info IDs and the specific values of info IDs in a CAG associated with ANQP is left to the implementation and is beyond the scope of this document.

	Info ID	Length	ANQP CAG Version	Info ID 1	...	Info ID N (optional)
Octets:	2	2	1	2		0 or 2

Figure 9-628e—CAG ANQP-element format

The Info ID and Length fields are defined in 9.4.5.1.

The ANQP CAG Version field indicates the current version of the CAG associated with ANQP.

The Info ID field represents info ID of an ANQP-element Info ID specified in Table 9-271.

9.6 Action frame format details

9.6.8 Public Action details

9.6.8.1 Public Action frames

Insert a new row into Table 9-307 and change the reserved row as follows:

Table 9-307—Public Action field values

Public Action field value	Description
34	FILS Discovery
~~34~~35–255	Reserved

Insert the new subclause 9.6.8.36 as follows:

9.6.8.36 FILS Discovery frame format

The FILS Discovery frame is a Public Action frame. The format of its Action field is shown in Table 9-325a.

Table 9-325a—FILS Discovery frame format

Order	Information	Notes
1	Category	
2	Public Action	
3	FILS Discovery Information field	
4	Reduced Neighbor Report element	The Reduced Neighbor Report element is optionally present.
5	FILS Indication element	The FILS Indication element is optionally present.
6	Roaming Consortium element	The Roaming Consortium element is optionally present.
6	Vendor Specific element	One or more Vendor Specific elements are optionally present.

The Category field indicates the public category specified in Table 9-47.

The Public Action field indicates the value of the FILS Discovery frame, as specified in Table 9-307 in 9.6.8.1.

The FILS Discovery Information field is shown in Figure 9-687a.

	FILS Discovery Frame Control	Timestamp	Beacon Interval	SSID/Short SSID	Length	FD Capability
Octets:	2	8	2	1-32	0 or 1	0 or 2

	Operating Class	Primary Channel	AP Configuration Sequence Number	Access Network Options	FD RSN Information	Channel Center Frequency Segment 1	Mobility Domain
Octets:	0 or 1	0 or 1	0 or 1	0 or 1	0 or 5	0 or 1	0 or 3

Figure 9-687a—FILS Discovery Information field format

The format of the FILS Discovery Frame Control subfield is shown in Figure 9-687b.

	B0 B4	B5	B6	B7
	SSID Length	Capability Presence Indicator	Short SSID Indicator	AP-CSN Presence Indicator
Bits:	5	1	1	1

	B8	B9	B10	B11	B12	B13	B14 B15
	ANO Presence Indicator	Channel Center Frequency Segment 1 Presence Indicator	Primary Channel Presence Indicator	RSN Info Presence Indicator	Length Presence Indicator	MD Presence Indicator	Reserved
Bits:	1	1	1	1	1	1	2

Figure 9-687b—FILS Discovery Frame Control subfield format

The SSID Length subfield of the FILS Discovery Frame Control subfield indicates the length, in octets, of the SSID/Short SSID subfield in the FILS Discovery frame. This subfield is equal to the length of the SSID/

Short SSID subfield in octets minus 1. When the Short SSID Indicator subfield is equal to 1, the SSID Length subfield is equal to 3 (the length of the Short SSID in octets minus 1).

When the Capability Presence Indicator subfield has a value of 1, it indicates that the FILS discovery (FD) Capability subfield is present in the FILS Discovery frame. A value of 0 indicates that the FD capability subfield is not present in the FILS Discovery frame.

The Short SSID Indicator subfield of 1 indicates that a Short SSID is contained in the SSID/Short SSID field of the FILS Discovery frame. A value of 0 indicates that a SSID is contained in the SSID/Short SSID field of the FILS Discovery frame.

The AP-CSN Presence Indicator subfield of 1 indicates that the AP Configuration Sequence Number subfield is present in the FILS Discovery frame. A value of 0 indicates that the AP-CSN subfield is not present in the FILS Discovery frame.

An Access Network Options (ANO) Presence Indicator subfield of 1 indicates that the Access Network Options subfield is present in the FILS Discovery frame. A value of 0 indicates that the ANO subfield is not present in the FILS Discovery frame.

The Channel Center Frequency Segment 1 Presence Indicator subfield of 1 indicates that the Channel Center Frequency Segment 1 subfield is present in the FILS Discovery frame. A value of 0 indicates that Channel Center Frequency Segment 1 subfield is not present.

The Primary Channel Presence Indicator subfield of 1 indicates that the Primary Channel and the Operating Class subfields are present in the FILS Discovery frame. A value of 0 indicates that the Primary Channel and the Operating Class subfields are not present in the FILS Discovery frame.

The RSN Information Presence Indicator subfield of 1 indicates that the FD RSN Information subfield is present in the FILS Discovery frame. A value of 0 indicates that the FD RSN Information subfield is not present in the FILS Discovery frame.

The Length Presence Indicator subfield of 1 indicates that the Length field is present in the FILS Discovery frame. A value of 0 indicates that the Length field is not present in the FILS Discovery frame.

A value of 1 for the MD Presence Indicator subfield indicates that the Mobility Domain subfield is present in the FILS Discovery frame. A value of 0 indicates that the Mobility Domain subfield is not present in the FILS Discovery frame.

The Timestamp subfield carries the value of the TSF timer at the frame source.

The Beacon Interval subfield carries the beacon interval in TUs.

The SSID/Short SSID subfield is variable length between 1 and 32 octets.When the Short SSID Indicator subfield is 1, the SSID/Short SSID field contains the 4-byte Short SSID (see 9.4.2.171). Otherwise, the SSID/Short SSID field contains the SSID, whose length is specified by the 5-bit SSID Length subfield in the FILS Discovery Frame Control of the FILS Discovery frame (see 9.4.2.2).

The Length subfield is 1 octet in length and indicates the length of the remaining fields in the FILS Discovery Information field in octets. Its value is variable.

NOTE—The Length field is used to facilitate STAs parsing the FILS Discovery frame in case of future expansions of the FILS Discovery Information field; STAs can determine the end of the FILS Discovery Information field using the value indicated in the Length field even if they do not recognize one or more subfields in the FILS Discovery Information field.

The FD Capability subfield contains the information that advertises the capabilities and operational indications of the STA transmitting the FILS Discovery frame. Its presence is indicated by the Capability Presence Indicator subfield in the FILS Discovery Frame Control subfield being equal to 1. The format of the FD Capability subfield is shown in Figure 9-687c.

	B0	B1	B2 B4	B5 B7
	ESS	Privacy	BSS Operating Channel Width	Maximum Number of Spatial Streams
Bits:	1	1	3	3

	B8	B9	B10 B12	B13 B15
	Reserved	Multiple BSSIDs Presence Indicator	PHY Index	FILS Minimum Rate
Bits:	1	1	3	3

Figure 9-687c—FD Capability subfield format

The ESS and Privacy subfields are interpreted as specified in 9.4.1.4.

The BSS Operating Channel Width subfield indicates the BSS operating channel width as defined in Table 9-325b.

Table 9-325b—BSS Operating Channel Width

BSS Operating Channel Width field	HR/DSSS, OFDM, ERP, HT, or VHT BSS operating channel width	TVHT BSS operating channel width
0	20 MHz or 22 MHz	TVHT_W
1	40 MHz	TVHT_W+W
2	80 MHz	TVHT_2W
3	160 MHz or 80+80 MHz	TVHT_4W or TVHT_2W+2W
4–7	Reserved	Reserved

The Maximum Number of Spatial Streams subfield indicates the number of the spatial streams supported by the AP. The Maximum Number of Spatial Streams subfield is coded per Table 9-325c.

Table 9-325c—Maximum Number of Spatial Streams

Number of Spatial Streams subfield	Maximum Number of Spatial Streams subfield
0	1
1	2
2	3
3	4
4	5–8
5–7	Reserved

The Multiple BSSIDs Presence Indicator subfield is set to 1 to indicate that the Multiple BSSID element is present in the Beacon and Probe Response frames. It is set to 0 to indicate that the Multiple BSSID element is not present in the Beacon and Probe Response frames.

The PHY Index subfield is defined as in Table 9-325d.

Table 9-325d—PHY Index subfield

PHY Index subfield	PHY
0	HR/DSSS (see Clause 17)
1	ERP-OFDM (see Clause 18 and Clause 19)
2	HT (see Clause 20)
3	VHT (see Clause 22) Or TVHT (see Clause 23)
4–7	Reserved

The FILS Minimum Rate subfield indicates the minimum rate to be used by the AP transmitting the FILS Discovery frame and by FILS STAs in subsequent transmissions between the AP and FILS STAs. Depending on the PHY Index subfield of the received FILS Discovery frame, the minimum rate is represented as a bit rate value or as an MCS value as shown in Table 9-325e. If an MCS value is provided, then the minimum rate is derived from the MCS value and the PHY Index in the FD Capability subfield.

Table 9-325e—FILS Minimum Rate

FILS Minimum Rate subfield	PHY Index subfield is 0 (HR/DSSS)	PHY Index subfield is 1 (ERP-OFDM)	PHY Index subfield is 2 (HT)	PHY Index subfield is 3 (VHT or TVHT)
0	1 Mbps	6 Mbps	MCS 0	MCS 0
1	2 Mbps	9 Mbps	MCS 1	MCS 1
2	5.5 Mbps	12 Mbps	MCS 2	MCS 2
3	11 Mbps	18 Mbps	MCS 3	MCS 3
4	Reserved	24 Mbps	MCS 4	MCS 4
5–7	Reserved	Reserved	Reserved	Reserved

The Operating Class subfield is 1 octet in length. It specifies the operating class of the Primary Channel of the transmitting AP (see 9.4.1.37).

Primary Channel subfield is set to the channel number of the primary channel (see 11.16.2) if the FILS Discovery frame is transmitted as a non-HT duplicate PPDU; otherwise, the subfield is not present.

AP Configuration Sequence Number (AP-CSN) subfield format is defined in 9.4.2.182.

Access Network Options (ANO) subfield format is specified in Figure 9-440 in 9.4.2.92.

Primary Channel subfield is set to the channel number of the primary channel (see 11.16.2) if the FILS Discovery frame is transmitted as a non-HT duplicate PPDU; otherwise, the subfield is not present.

The FD RSN Information subfield contains the RSN information, including: RSN capability, an authentication suite selector, a pairwise cipher suite selector, a group data cipher suite selector, and a group management cipher suite selector. Its format is defined in Figure 9-687d.

	B0 B15	B16 B21	B22 B27	B28 B33	B34 B39
	RSN Capability	Group Data Cipher Suite Selector	Group Management Cipher Suite Selector	Pairwise Cipher Suite Selector	AKM Suite Selector
Bits:	16	6	6	6	6

Figure 9-687d—Format of the FD RSN Information subfield

The FD RSN Information subfield contains an RSN Capability subfield, as specified in Figure 9-256 in 9.4.2.25.4.

The FD RSN Information subfield also contains three cipher suite selectors. The Group Data Cipher Suite Selector subfield contains one group data cipher suite selector. The Group Management Cipher Suite Selector subfield contains group management cipher suite selector. The Pairwise Cipher Suite Selector subfield contains one pairwise cipher suite selector. Each of these subfields contains a code identifying a Cipher Suite Type as specified in Table 9-130. The definition of the cipher suite selectors is shown in Table 9-325f.

Table 9-325f—Cipher suite selector definitions

Cipher suite selector	Cipher suite type
0–13	Cipher suite type 0 to 13 in Table 9-130
14–61	Reserved
62	Vendor Specific
63	No cipher suite selected

The RSN Information subfield contains one AKM suite selector. An AKM suite selector is a code identifying a AKM suite type as specified in Table 9-133. The definition of the AKM suite selectors is shown in Table 9-325g.

Table 9-325g—AKM suite selector definitions

AKM suite selector	AKM suite type
0	Use AKM from RSN IE Beacon/Probe Response frame
1	Set AKM suite to 14 of Table 9-133
2	Set AKM suite to 15 of Table 9-133
3	Set AKM suite to either 14 or 15 of Table 9-133
4	Set AKM Suite to 17 of Table 9-133
5–61	Reserved
62	Vendor Specific
63	No AKM suite selected

Channel Center Frequency Segment 1 subfield is set to the index of the channel center frequency of the frequency segment 1 for an 80+80 MHz VHT BSS, if the FILS Discovery frame is transmitted as a non-HT duplicate PPDUs at an 80+80 MHz channel bandwidth; otherwise, the subfield is not present.

The format of the Mobility Domain subfield is shown in Figure 9-687e.

	MDID	FT Capability and Policy
Octets:	2	1

Figure 9-687e—Mobility Domain subfield format

The MDID field is 2 octets in length and is defined in 9.4.2.47.

The FT Capability and Policy field is 1 octet in length and is defined in 9.4.2.47.

The Reduced Neighbor Report Element is defined in 9.4.2.171.

The FILS Indication element is defined in 9.4.2.183.

The FILS Discovery frame may include one or more Vendor Specific elements. The Vendor Specific element is defined in 9.4.2.26.

Insert new subclause 9.6.24 as follows:

9.6.24 FILS Action frame details

9.6.24.1 General

The FILS Action frame is used for FILS operation after the non-AP STA has associated with the AP. The defined FILS Action frames are listed in Table 9-421a.

Table 9-421a—FILS Action frame values

Action field value	Description
0	FILS Container frame
1–255	Reserved

9.6.24.2 FILS Container frame

The FILS Container frame is used to exchange FILS IP Address Assignment elements (see 9.4.2.185).

	Category	FILS Action	FILS IP Address Assignment elements (defined in 9.4.2.185)
Octets:	1	1	variable

Figure 9-740a—FILS Container frame format

The Category field is set to the value for FILS defined in Table 9-47.

The FILS Action field is set to the value given in Table 9-421a for FILS Container frame.

The FILS IP Address Assignment element carries the FILS parameters for IP address assignment and DNS server information.

10. MAC sublayer functional description

Change the title of 10.5 as follows:

10.5 MPDU ~~F~~fragmentation

Change the title of 10.6 as follows:

10.6 MPDU ~~D~~defragmentation

10.27 MAC frame processing

Insert new subclauses 10.27.11 and 10.27.12 as follows:

10.27.11 Element fragmentation

The general format of elements limits the size of the information to 255 octets in an element without Element ID Extension field or 254 octets in an element with Element ID Extension field. Information that is too large to fit in a single element is fragmented into a series of elements consisting of the element that the information does not fit, immediately followed by one or more Fragment elements as illustrated in Figure 10-40a. Information that fits in a single element is not fragmented. All the information for a fragmented element appears in the same MMPDU.

The information to be fragmented is divided into M + N portions, where the following define each variable:

For Element without Element ID Extension

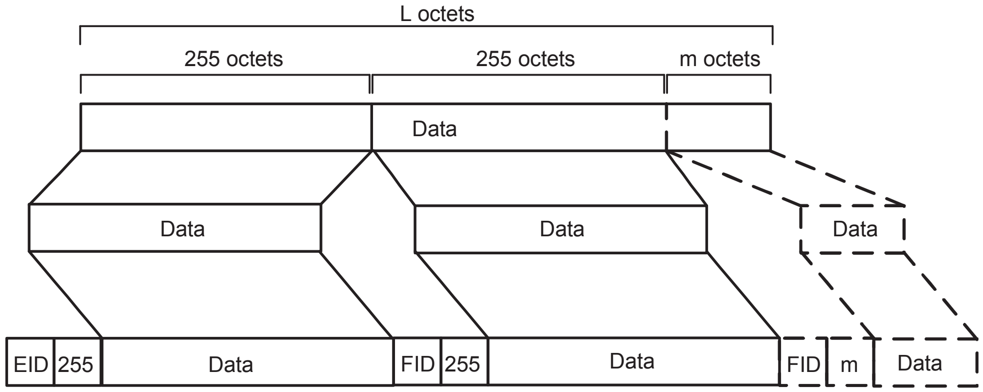
- L is the size of the information in octets.
- M is Floor (L/255).
- N is equal to 1 if L mod 255 > 0 and equal to 0 otherwise.

For Element with Element ID Extension

- L is the size of the information in octets.
- M is Floor ((L+1)/255).
- N is equal to 1 if (L-254) mod 255 > 0 and equal to 0 otherwise.

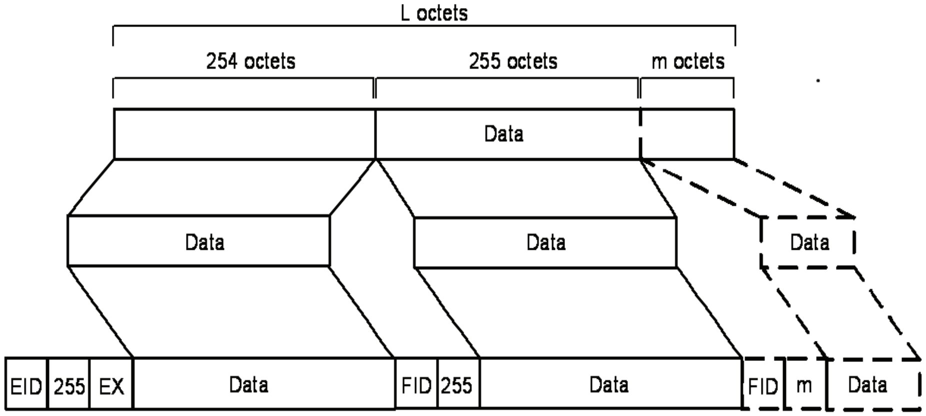
The element into which the information does not fit is filled with the first portion of information and is termed the leading element. The leading element contains 255 octets of information in case of the element without Element ID Extension, or 254 octets of information in case of the element with Element ID Extension. This element is immediately followed by M – 1 Fragment elements, each containing the next portion of 255 octets of information. If N = 1 these elements are immediately followed by the last portion of information.

NOTE—A Fragment element never follows an element with fewer than 255 octets of information without Element ID Extension, and an element with fewer than 254 octets of information with Element ID Extension. A Fragment element is never fragmented.



EID: The element ID of the fragmented element
FID: The Fragment element ID
m: L mod 255

Figure 10-40a—Example of the element fragmentation without Element ID Extension



EID: The element ID of the fragmented element
EX: the element ID extension of the fragmented element
FID: The Fragment element ID
m: (L-254) mod 255 (The dashed Fragment element is present when (L-254) mod 255 is not equal to 0)

Figure 10-40b—Example of the element fragmentation with Element ID Extension

10.27.12 Element defragmentation

Elements that have had their information fragmented are followed by one or more Fragment elements. To reconstruct the original information, the portion of information from the leading element is concatenated, in order, with the portions of information from the series of Fragment elements that follow it. The defragmentation procedure completes when any element other than a Fragment element is encountered, or the end of the MMPDU is reached.

11. MLME

11.1 Synchronization

11.1.3 Maintaining synchronization

11.1.3.8 Multiple BSSID procedure

Change 11.1.3.8 as follows:

Implementation of the Multiple BSSID capability is optional for a WNM STA and for a DMG STA. Implementation of the Multiple BSSID capability is mandatory for a FILS STA. A STA that implements the Multiple BSSID capability has dot11MultiBSSIDImplemented equal to true. When dot11MultiBSSIDImplemented is true, dot11WirelessManagementImplemented shall be equal to true except for a DMG STA, in which case it may be equal to false. A STA in which dot11MultiBSSIDActivated is true is defined as a STA that supports the Multiple BSSID capability. The STA shall set to 1 the Multiple BSSID field of the Extended Capabilities elements that it transmits.

11.1.4 Acquiring synchronization, scanning

11.1.4.1 General

Change the third, fourth, and fifth paragraphs in 11.1.4.1 as follows:

Upon receipt of the MLME-SCAN.request primitive, a STA shall perform scanning procedures according to the parameters given in the primitive. The SSID parameter indicates the SSID for which to scan. The SSID List parameter indicates one or more SSIDs for which to scan.

To become a member of a particular ESS using passive scanning, a STA shall scan for Beacon and DMG Beacon frames containing that ESS's SSID, returning all Beacon and DMG Beacon frames matching the desired SSID in the BSSDescriptionSet parameter of the corresponding MLME-SCAN.confirm primitive with the appropriate bits in the Capability Information field or DMG Capabilities field indicating whether the Beacon frame or DMG Beacon frame came from an infrastructure BSS, PBSS, or IBSS. If dot11RM-MeasurementPilotActivated is greater than 1, the STA shall additionally scan for Measurement Pilot frames, returning in the BSSDescriptionFromMeasurementPilotSet parameter all Measurement Pilot frames that equal the requested BSSID of the corresponding MLME-SCAN.request primitive and are not already members of the BSSDescriptionSet. A FILS STA shall additionally scan for FILS Discovery frames, returning in the BSSDescriptionFromFDSet parameter all FILS Discovery frames of the scanned ESSs. The STA is not required to return a BSSDescriptionFromFDSet parameter for any BSS that is already a member of the BSS-DescriptionSet.

To actively scan, the STA shall transmit Probe Request frames containing a wildcard SSID (see 9.4.2.2), the desired SSID, or one or more SSID List elements, but a DMG STA might also have to transmit DMG Beacon frames or perform beamforming training prior to the transmission of Probe Request frames. When the SSID List element is present in the Probe Request frame, one or more of the SSID elements may include a wildcard SSID (see 9.4.2.2). The exact procedure for determining the SSID or SSID List values in the MLME-SCAN.request primitive is not specified in this standard. When a STA scans for a BSS whose AP does not support the SSID List element, or for a BSS for which AP support of the SSID List element is unknown, the SSID element with an SSID or wildcard SSID shall be included in the MLME-SCAN.request primitive. During FILS scanning, the scanning STA may optimize the scanning process by using intermediate results, including the Reduced Neighbor Report element. Details of how to optimize scanning is out of scope of this standard. An MLME-SCAN.confirm primitive is issued each time that a suitable BSS is discovered when the value of the ReportingOption parameter in the MLME-SCAN.request primitive is present and is equal to IMMEDIATE. An MLME-SCAN.confirm primitive is issued each time that the scanning STA has completed the scanning of a channel when the value of the ReportingOption parameter in the MLME-SCAN.request primitive is present and is equal to CHANNEL_SPECIFIC.

Change the seventh paragraph in 11.1.4.1 as follows:

Upon receipt of an MLME-SCAN.request primitive with the SSID parameter equal to the wildcard SSID, the STA shall passively scan for any Beacon, DMG Beacon, FILS Discovery, ~~or~~ Measurement Pilot frames, or Probe Response frames containing Address 1 field equal to the broadcast address, or actively transmit Probe Request or DMG Beacon frames containing the wildcard SSID, as appropriate depending upon the value of ScanMode.

11.1.4.3 Active scanning

11.1.4.3.2 Active scanning procedure for a non-DMG STA

Change 11.1.4.3.2, relettering as follows:

Upon receipt of the MLME-SCAN.request primitive with ScanType indicating an active scan, a STA shall use the following procedure:

For each channel to be scanned:

a) Wait until the ProbeDelay time has expired or a PHY-RXSTART.indication primitive has been received.

b) If the STA is a FILS STA, set the FILSProbeTimer to 0 and starts the FILSProbeTimer. While the FILSProbeTimer is less than dot11FILSProbeDelay the STA may skip a probe request transmission and proceed to step i) after setting the ActiveScanningTimer to 0 and starting the ActiveScanningTimer, if one of the following conditions matches:

   1) The STA receives a broadcast addressed Probe Request frame that the SME considers to be suitable to discover a candidate AP for association.

   2) The STA receives one or more of Probe Response, Beacon, Measurement Pilot, or FILS Discovery frame that identify an AP that the SME considers a suitable candidate for association.

NOTE—The logic how an SME considers a probe request suitable or the AP as a suitable candidate for association is out of the scope of this standard.

c) Perform the Basic Access procedure as defined in 10.3.4.2.

d) Send a probe request to the broadcast destination address. The probe request is sent with the SSID and BSSID from the received MLME-SCAN.request primitive. When the SSID List is present in the MLME-SCAN.request primitive, send one or more Probe Request frames, each with an SSID indicated in the SSID List and the BSSID from the MLME-SCAN.request primitive.

e) When the SSID List is present in the invocation of the MLME-SCAN.request primitive, send zero or more Probe Request frames, to the broadcast destination address. Each probe request is sent with an SSID indicated in the SSID List and the BSSID from the MLME-SCAN.request primitive. The basic access procedure (10.3.4.2) is performed prior to each probe request transmission.

f) Initialize the ~~a timer~~ ActiveScanningTimer to 0 and start ~~it running~~the ActiveScanningTimer.

g) If a PHY-CCA.indication (BUSY) primitive is not received before the ~~timer~~ActiveScanningTimer reaches MinChannelTime, then proceed to step ~~h~~l).

h) If the STA is a non-FILS STA, receive all Probe Response and Beacon frames while the ActiveScanningTimer is less than MaxChannelTime.

i) If the STA is a FILS STA and while the ActiveScanningTimer is less than MaxChannelTime:

   1) Receive Probe Response, FILS Discovery, and Beacon frames regardless of the receiver address. Process any received FILS Discovery, Probe Response, and Beacon frames.

   2) If the ReportingOption parameter of the MLME-SCAN.request primitive is IMMEDIATE, and the scanning FILS STA detects a BSS whose MLME-SCAN.confirm primitive has not been issued during the ongoing scan, then an MLME-SCAN.confirm primitive with the ResultCode equal to INTERMEDIATE_SCAN_RESULT and one or more of BSSDescriptionSet, BSSDescriptionFromFDSet, or BSSDescriptionFromMeasurementPilotSet containing information of the detected BSS is immediately issued.

j) If the ReportingOption parameter of the MLME-SCAN.request primitive is CHANNEL_SPECIFIC, do the following:

   1) If the ActiveScanningTimer has not reached MaxChannelTime, wait until the ActiveScanningTimer reaches the MaxChannelTime and then proceed to item 2); otherwise, proceed directly to item 2).

   2) Issue an MLME-SCAN.confirm primitive, with the ResultCode equal to INTERMEDIATE_SCAN_RESULT and one BSSDescriptionSet, BSSDescriptionFromFDSet, or BSSDescriptionFromMeasurementPilotSet containing information of all BSSs that have been discovered from the scanned channel.

k) Process all probe responses received until the timer reaches MaxChannelTime, constructing BSSDescriptions corresponding to the probe responses that match the criteria specified in the MLME-SCAN.request primitive.

l) Set the NAV to 0 and scan the next channel.

m) When all channels in the ChannelList have been scanned, and the ReportingOption parameter of the MLME-SCAN. request primitive is AT_END or not present, the MLME shall issue an MLME-SCAN.confirm primitive with ~~the~~ one or more of BSSDescriptionSet, BSSDescriptionFromFDSet, or BSSDescriptionFromMeasurementPilotSet containing all of the information gathered during the scan.

See Figure 11-4a and Figure 11-4b ~~Figure 11-4 (Probe response) for non-DMG STAs~~.

~~When all channels in the ChannelList have been scanned, the MLME shall issue an MLME-SCAN.confirm primitive with the BSSDescriptionSet containing all of the information gathered during the scan.~~

If the MLME receives an MLME-SCAN-STOP.request primitive, the STA shall stop scanning. The STA should discard any Probe Request frame queued for transmission. If the STA is transmitting a Probe Request frame, the STA shall complete the transmission of the Probe Request frame. The STA shall not continue the active scanning process on unscanned channels listed in the ChannelList parameter of the MLME-SCAN.request primitive. If the ReportingOption parameter of the MLME-SCAN.request primitive is AT_END or not present, then the MLME shall issue an MLME-SCAN.confirm primitive with the ResultCode set to SUCCESS and with one or more of BSSDescriptionSet, BSSDescriptionFromFDSet, or BSSDescriptionFromMeasurementPilotSet containing all of the information gathered during the scan.

A FILS STA shall indicate its MaxChannelTime in the Max Channel Time field of the FILS Request Parameters element of the Probe Request frame to prevent the responding STA from transmitting the Probe Response frame after the time indicated by the MaxChannelTime has elapsed.

The Max Channel Time field shall be set to the MaxChannelTime of the MLME-SCAN.request primitive as defined in 9.4.2.178.

Delete Figure 11-4.

Insert two new figures, Figure 11-4a and Figure 11-4b as follows:

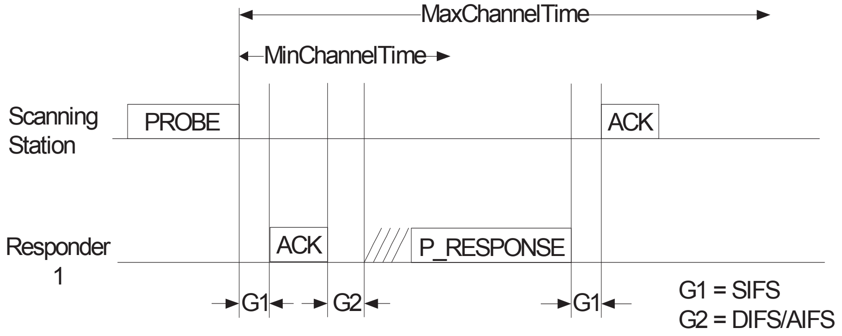


Figure 11-4a—Active scanning by a non-DMG STA with a probe request addressed to an individual address

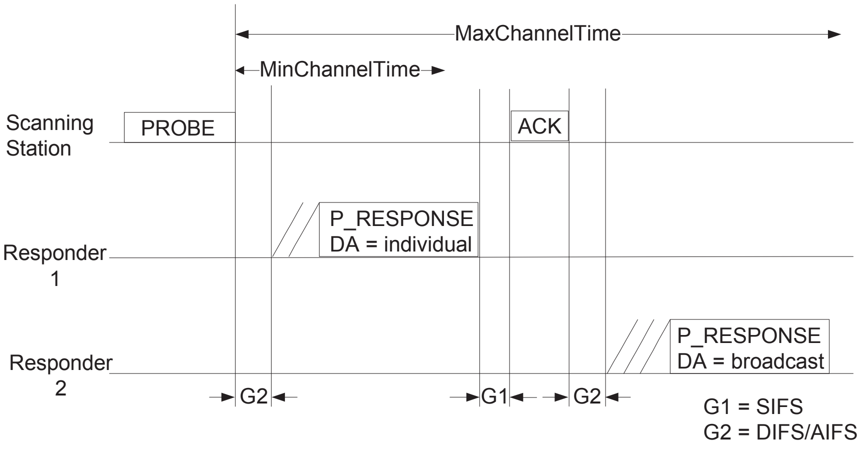


Figure 11-4b — Active scanning by a non-DMG STA with a probe request addressed to wildcard BSSID

Change subclause title of 11.1.4.3.4 as follows:

11.1.4.3.4 Criteria for sending a ~~probe~~ response

Insert the following text after item k) of 11.1.4.3.4:

A FILS STA shall not respond to a Probe Request frame if any of the following criteria is met for a FILS Request Parameters element contained in the Probe Request frame:

1) If the FILS Criteria field is present in the FILS Requests Parameters element and the Max Delay Limit field of the FILS Request Parameters indicates a delay shorter than the selected average access delay of the responding STA. The BSS Delay Criterion field of the FILS Criteria field of the FILS

Request Parameters element indicates the selected average access delay for the comparison as defined in Table 9-262a. The Max Delay Limit field indicates the maximum value of the selected average access delay. If the compared Average Access Delay indicates Measurement not available, the STA shall respond and the response shall include a BSS AC Access Delay element as described in 9.4.2.44 and Average Access Delay as described in 9.4.2.39 or Average Access Delay as described in 9.4.2.39 that was requested in the Probe Request frame. If the compared Average Access Delay indicates Service unable to access channel, the response shall not be transmitted.

2) If the FILS Criteria field is present in the FILS Requests Parameters element and the PHY Support Criterion of the FILS Criteria field of the FILS Request Parameters element is 1 and the responding STA is not HT capable.

3) If the FILS Criteria field is present in the FILS Requests Parameters element and the PHY Support Criterion of the FILS Criteria field of the FILS Request Parameters element is 2 and the responding STA is not VHT capable.

4) If the Minimum Data Rate is present in the FILS Request Parameters element and the Minimum Data Rate field of the FILS Request Parameters element indicates a data rate higher than the one that is provided over the MAC SAP.

5) If the RCPI Limit field is present in the FILS Request Parameters element and either of the following conditions is true:

   — The RCPI of the Probe Request frame > –90 dBm + the value of the RCPI Limit field of the FILS Request Parameters element.

   — The RCPI Limit field of the FILS Request Parameters element contains value 255.

6) If the OUI Response Criteria field is present in the FILS Request Parameters element and if any OUIs specified by the OUI Response Criteria field are not known to the AP (see Known OUIs, 6.3.5.2.2).

If the FILS Request Parameters element is present in the Probe Request frame, the responding FILS STA should discard any Probe Response frame that has not been transmitted as a response to the Probe Request frame when the elapsed time measured from the end of the reception of the Probe Request frame by the MAC entity of the responding STA exceeds the time indicated by value of the Max Channel Time field of the FILS Request Parameters element of the Probe Request frame. If the FILS Request Parameter element is not present in the Probe Request frame, transmission time of the Probe Response frame to the Probe Request frame by the responding STA is only limited by the retransmission procedure in 10.22.2.11.

NOTE—It is possible for the STA to leave the channel on which it sent the Probe Request frame prior to MaxChannelTime. Should this occur the STA might not receive some of the Probe Response frames transmitted.

If the Multiple BSSID bit is set in the Extended Capabilities element in the Probe Request frame, the FILS STA shall not respond to the Probe Request if its BSS information is present as a Nontransmitted BSSID Profile of a Multiple BSSID element in the response generated from another FILS STA.

An individually addressed Probe Response frame, subject to the criteria above, shall be transmitted to all non-FILS STAs from which a Probe Request frame is received.

If a FILS STA has dot11FILSOmitReplicateProbeResponses equal to false, an individually addressed Probe Response frame, subject to the criteria above, shall be transmitted to all STAs from which a Probe Request frame is received.

If a FILS STA receives one or more Probe Request frame(s), subject to the criteria above, and the STA has dot11FILSOmitFILSReplicateProbeResponses equal to true, the responding STA shall select the response with the next Beacon frame or one or more Probe Response frames as a response to all Probe Request frames.

The FILS STA shall respond with the next Beacon frame, as described in 11.1.3, to Probe Request frames addressed to individual or broadcast address if all of the following conditions are met:

- The STA is queuing a Beacon frame for transmission;
- The next TBTT of the responding STA is within dot11FILSBeaconResponseWindow;
- The next TBTT is no later than any deadline of Max Channel Time indicated in the FILS Request Parameter element of the Probe Request frame(s), if present; and
- The Beacon frame contains all elements requested by the Request element.

If the next Beacon is not used as a response, a Probe Response frame is transmitted. The Probe Response frame shall be addressed to the broadcast or the address of the transmitter of the Probe Request frame. The Probe Response frame may be transmitted to all or some of the Probe Request frames received from FILS STAs. A FILS STA may choose not to respond to Probe Request frames from a FILS STA addressed to broadcast address if the responding STA receives an acknowledged probe response addressed to the requesting STA containing the SSID of the responding STA.

Change 11.1.4.3.5 as follows:

11.1.4.3.5 Contents of a probe response

A STA that responds to a Probe Request frame according to 11.1.4.3.4 shall transmit a Probe Response frame or a Beacon frame ~~individually addressed~~ to the STA that transmitted the Probe Request frame.

A FILS STA that transmits a Probe Response frame shall either set the Address 1 field to the address of the STA that generated the probe request or to the broadcast address if the STA that generated the probe request indicated FILS Capability. A non-FILS STA that transmits a Probe Response frame shall set the Address 1 field to the address of the STA that generated the probe request.

When a FILS AP responds to a Probe Request frame containing a FILS Capability field in the Extended Capabilities element equal to 1, and when both the FILS AP and the FILS STA that transmitted the Probe Request frame support other than DSSS/CCK rates (see Clause 15 or Clause 16), the AP shall transmit a Probe Response frame in a PPDU using a rate other than a DSSS/CCK.

A STA having dot11InterworkingServiceActivated true may include in the Probe Response frame a CAG Number element containing one or more current version numbers of the CAG information associated with the AP, where each version number is associated with an advertisement protocol. The current CAG information associated with the AP can be acquired by the GAS query mechanism as described in 11.25.3 using the associated advertisement protocol.

If there was a Request element or Extended Request element in the Probe Request frame, then:

- Each element that is listed by a non-FILS STA in the Request element or Extended Request element(s) and that is supported by the STA shall be included in the Probe Response frame. An element that is listed in a Request element or Extended Request element and that is not supported by the STA shall not be included. Each element requested by a FILS STA in a Request element shall be included in the Probe Response frame or a Beacon frame if the responding FILS STA supports that element.
- Elements that would not have been included otherwise shall be included after all of the elements that would have been included even in the absence of the Request element or Extended Request element.
- Elements that would have been included even in the absence of the Request element or Extended Request element shall be included in their normal position (see Table 9-34), and may be included again after all of the elements that would have been included even in the absence of the Request

element.

NOTE—An element that would necessarily be included anyway is not expected to be requested.

- Elements after all of the elements that would have been included even in the absence of the Request element or Extended Request element shall be included in the order that they appear in the (Extended) Request element(s) of the Probe Request frame.
- If dot11RadioMeasurementActivated is true and the RCPI element was requested, an RCPI element containing the RCPI of the Probe Request frame shall be included. If no measurement result is available, the RCPI value shall be set to indicate "Measurement not available" (see Table 9-154).
- If dot11RadioMeasurementActivated is true and the RSNI element was requested, an RSNI element containing the RSNI of the Probe Request frame shall be included. If no measurement result is available, the RSNI value shall be set to indicate that a measurement is not available (see 9.4.2.41).
- If a Probe Request frame includes a Request element that the element ID of the Reduced Neighbor Report Request element, a Probe Response frame or a Beacon frame if transmitted may include the Reduced Neighbor Report element if the criteria as defined in 11.1.4.3.4 are met for the included BSS. The reported BSSs may have different primary channels to the responding STA.

Insert the new subclause 11.1.4.3.7 as follows:

11.1.4.3.7 Enhanced FILS active scanning to preferred AP

A FILS non-AP STA may maintain one or more BSS Configuration Parameter Sets. A BSS Configuration Parameter Set is obtained from a preferred AP by using a preferred AP determination process that is out of scope of this standard. Each BSS Configuration Parameter Set may be different according to the preferred AP's capabilities. A BSS Configuration Parameter Set is a set of elements of the Beacon frame or the Probe Response frame. The following dynamic information elements are excluded from a BSS Configuration Parameter Set:

- TIM element
- Quiet element
- BSS Load element
- EDCA Parameter element
- BSS Average Access Delay element
- BSS Available Admission Capacity element
- BSS AC Access Delay element
- Time Advertisement element
- Emergency Alert Identifier element
- Beacon Timing element
- QLoad Report element
- Extended BSS Load element
- Quiet Channel element
- Reduced Neighbor Report element (see Note 1)
- CAG Number element
- AP-CSN element
- Differentiated Initial Link Setup element
- Fragment element (see Note 2)
- Vendor Specific element

If a vendor-specific subelement is included in an element within the BSS Configuration Parameter Set, the AP-CSN does not provide any indication regarding if that vendor-specific subelement has changed or not, and AP-CSN is not increased if the only change within the BSS Configuration Parameter Set is due to the change to a vendor-specific subelement embedded in an element within the BSS Configuration Parameter Set.

NOTE 1—The Reduced Neighbor Report element is excluded from the BSS Configuration Parameter Set based on the principle that an element is excluded from the BSS Configuration Parameter Set if that element has no impact on a FILS STA's ability of using AP-CSN to make a decision of initiating an association procedure with an AP without receiving Beacon or Probe Response frame from the AP.

NOTE 2—Any change in a Fragment element is considered under the context of the element being fragmented by the Fragment element.

A FILS AP maintains an AP-CSN List consisting of the current AP-CSN value and zero or more previous AP-CSN values.The AP initializes the AP-CSN to a random integer value in the range of 0 to 255. For each maintained previous AP-CSN value, the AP also maintains the identifiers of the changed elements. The AP may maintain the AP-CSN values in the AP-CSN List for a duration whose value is out of the scope of this standard.

An AP maintaining an AP-CSN list shall increase the current AP-CSN value (modulo 256) by one if an update occurs to any of the fields or elements within the BSS Configuration Parameter Set.

A FILS AP may provide FILS STAs its AP-CSN value by sending a Beacon frame or a Probe Response frame including an AP-CSN element (as defined in 9.4.2.182).

A FILS non-AP STA identifies an BSS Configuration Parameter Set by its associated AP-CSN value and the AP's BSSID.

A FILS non-AP STA may send a Probe Request frame including an AP-CSN element (as defined in 9.4.2.182), if the STA has the BSS Configuration Parameter Set associated with the AP-CSN of the AP. When sending a Probe Request frame including an AP-CSN element, the FILS non-AP STA shall set the Address 1 and Address 3 fields in the Probe Request frame to the BSSID of the AP, of which the AP-CSN is being sent.

When a FILS AP receives a Probe Request frame with AP-CSN element, an individually addressed BSSID matches this AP, and the criteria for responding to a Probe Request (11.1.4.3.4) are met, the AP sends a Probe Response frame according to comparison result, as follows:

a) If the AP does not maintain AP-CSN List, the AP sends a Probe Response.

b) If the received AP-CSN value matches with the current AP-CSN value of the AP, the AP sends an optimized Probe Response frame including mandatory fields (i.e., Timestamp, Capability, and Beacon Interval), the current AP-CSN element, and one or more elements among dynamic elements defined in this subclause.

c) If the received AP-CSN value matches with one of the previous AP-CSN values in the AP-CSN List, the AP sends an optimized Probe Response frame including mandatory fields, the current AP-CSN element, the information elements that need to be updated at the STA, and one or more elements among dynamic elements defined in this subclause.

If the received AP-CSN value does not match with any of AP-CSN values in the AP-CSN List, the AP shall send a Probe Response frame with its current AP-CSN, the information fields, and elements as defined in 9.3.3.11.

11.3 STA authentication and association

11.3.2 State transition diagram for nonmesh STAs

Replace Figure 11-13 with the following:

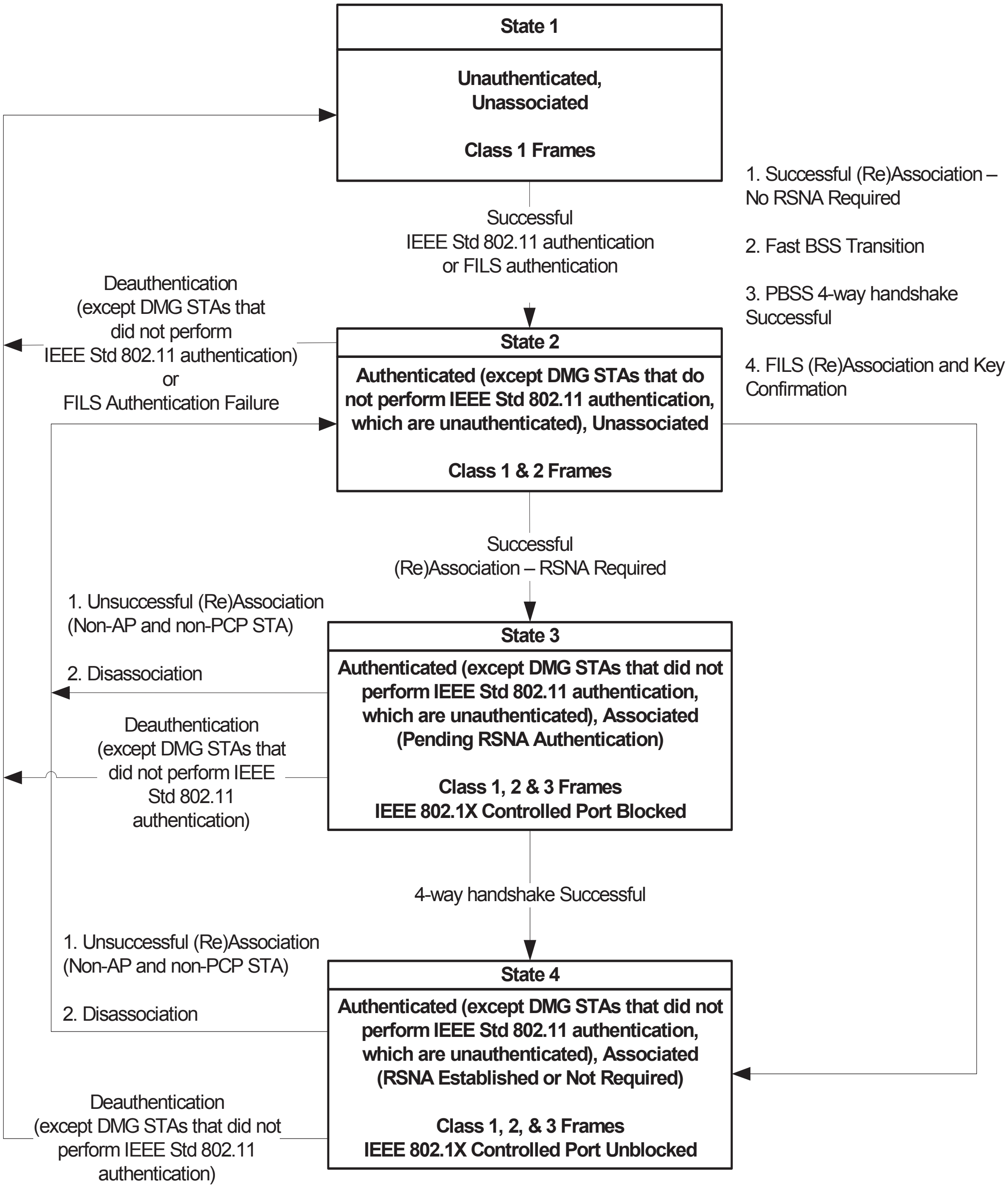

State 1
Unauthenticated,
Unassociated
Class 1 Frames
Successful
IEEE Std 802.11 authentication
or FILS authentication
Deauthentication
(except DMG STAs that
did not perform
IEEE Std 802.11 authentication)
or
FILS Authentication Failure
1. Successful (Re)Association – No RSNA Required
2. Fast BSS Transition
3. PBSS 4-way handshake Successful
4. FILS (Re)Association and Key Confirmation
State 2
Authenticated (except DMG STAs that do not perform IEEE Std 802.11 authentication, which are unauthenticated), Unassociated
Class 1 & 2 Frames
Successful
(Re)Association – RSNA Required
1. Unsuccessful (Re)Association
(Non-AP and non-PCP STA)
2. Disassociation
Deauthentication
(except DMG STAs that
did not perform IEEE
Std 802.11
authentication)
State 3
Authenticated (except DMG STAs that did not perform IEEE Std 802.11 authentication, which are unauthenticated), Associated
(Pending RSNA Authentication)
Class 1, 2 & 3 Frames
IEEE 802.1X Controlled Port Blocked
4-way handshake Successful
1. Unsuccessful (Re)Association
(Non-AP and non-PCP STA)
2. Disassociation
State 4
Authenticated (except DMG STAs that did not perform IEEE Std 802.11 authentication, which are unauthenticated), Associated
(RSNA Established or Not Required)
Class 1, 2, & 3 Frames
IEEE 802.1X Controlled Port Unblocked
Deauthentication
(except DMG STAs that did not
perform IEEE Std 802.11
authentication)

Figure 11-13—Relationship between state and services between a given pair of nonmesh STAs

11.3.3 Frame filtering based on STA state

Change 11.3.3 as follows:

The current state existing between the transmitter and receiver STAs determines the IEEE 802.11 frame types that may be exchanged between that pair of STAs (see Clause 9). A unique state exists for each pair of transmitter and receiver STAs. The allowed frame types are grouped into classes and the classes correspond to the STA state. In State 1, only Class 1 frames are allowed. In State 2, ~~either~~ only Class 1 or Class 2 frames are allowed. In State 3 and State 4, all frames are allowed (Classes 1, 2, and 3). In the definition of frame classes, the following terms are used:

11.3.4 Authentication and deauthentication

11.3.4.2 Authentication—originating STA

Change 11.3.4 as follows:

Upon receipt of an MLME-AUTHENTICATE.request primitive that is part of an on-channel tunneling (see 10.33.4), the originating STA shall follow the rules in 11.33.4 in addition to the authentication procedure described below.

Upon receipt of an MLME-AUTHENTICATE.request primitive, the originating STA shall authenticate with the indicated STA using the following procedure:

a) If the STA is in an IBSS the SME shall delete any PTKSA,GTKSA, IGTKSA, and temporal keys held for communication with the indicated STA by using the MLME-DELETEKEYS.request primitive (see 12.6.18).

b) The STA shall execute one of the following:

   1) For the Open System or Shared Key authentication algorithm, the authentication mechanism described in 12.3.3.2 or 12.3.3.3, respectively.

   2) For the fast BSS transition (FT) authentication algorithm in an ESS, the authentication mechanism described in 13.5, or, if resource requests are included, 13.6.

   3) For SAE authentication in an infrastructure BSS, IBSS, or MBSS, the authentication mechanism described in 12.4.

   4) For FILS authentication, the authentication mechanism described in 12.12. An AP may provide estimated association response latency to a non-AP STA using the Association Delay Info field in the Association Delay Info element. (9.4.2.175). The value of the Association Delay Info field shall be larger than dot11HLPWaitTime.

c) If the authentication was successful within the AuthenticateFailureTimeout, the state for the indicated STA shall be set to State 2 if it was State 1; the state shall remain unchanged if it was other than State 1.

d) The MLME shall issue an MLME-AUTHENTICATE.confirm primitive to inform the SME of the result of the authentication.

11.3.4.3 Authentication—destination STA

Change 11.3.4.3 as follows (with updates to list numbering as shown):

Upon receipt of an Authentication frame with authentication transaction sequence number equal to 1, the destination STA shall authenticate with the originating STA using the following procedure:

a) If Open System or Shared Key authentication algorithm is being used, the STA shall execute the procedure described in 12.3.3.2 or 12.3.3.3, respectively. These result in the generation of an MLME-AUTHENTICATE.indication primitive to inform the SME of the authentication request.

b) If FT authentication is being used, the MLME shall issue an MLME-AUTHENTICATE.indication primitive to inform the SME of the authentication request, including the FT Authentication Elements, and the SME shall execute the procedure as described in 13.5 (or 13.6).

c) If SAE authentication is being used in an infrastructure BSS, IBSS, or MBSS, the MLME shall issue an MLME-AUTHENTICATE. indication primitive to inform the SME of the authentication request, including the SAE Authentication Elements, and the SME shall execute the procedure as described in 12.4.

d) If FILS authentication is being used, the MLME shall issue an MLME-AUTHENTICATE.indication primitive to inform the SME of the authentication request, and the SME shall execute the procedure described in 12.12.

e) If the STA is in an IBSS and management frame protection was not negotiated when the PTKSA(s) were created, the SME shall delete any PTKSA,GTKSA, IGTKSA and temporal keys held for communication with the originating STA by using the MLME-DELETEKEYS.request primitive (see 12.6.18).

f) Upon receipt of an MLME-AUTHENTICATE.response primitive, if the ResultCode is not SUCCESS, the MLME shall transmit an Authentication frame with the corresponding status code, as defined in 9.4.1.9, and the state for the originating STA shall be left unchanged. The Authentication frame is constructed using the appropriate procedure in 12.3.3.2, 12.3.3.3, 13.5, or 13.6.

g) Upon receipt of an MLME-AUTHENTICATE.response primitive, if the ResultCode is SUCCESS, the MLME shall transmit an Authentication frame that is constructed using the appropriate procedure in 12.3.3.2, 12.3.3.3, 13.5, or 13.6, with a status code of SUCCESS, and the state for the originating STA shall be set to State 2 if it was in State 1.

If the STA is in an IBSS, if the SME decides to initiate an RSNA, and if the SME does not know the security policy of the peer, it may issue an individually addressed Probe Request frame to the peer by invoking an MLME-SCAN.request primitive to discover the peer's security policy. When a non-AP STA receives an Authentication frame that includes an Association Delay Info element, the non-AP STA sets the dot11AssociationResponseTimeOutequal to or larger than the value of the Association Delay Info field.

11.3.5 Association, reassociation, and disassociation

11.3.5.1 General

Change 11.3.5.1 as follows:

Subclause 11.3.5 describes the procedures used for IEEE 802.11 association, reassociation, and disassociation.

The states used in this description are defined in 11.3.1.

Successful association enables a STA to exchange Class 3 frames. Successful association sets the non-FILS STA's state to State 3 or State 4. Successful association sets the state for FILS STAs to State 4.

Successful reassociation enables a STA to exchange Class 3 frames. Unsuccessful reassociation when not in State 1 leaves the STA's state unchanged [with respect to the AP or PCP that was sent the Reassociation Request (which may be the current STA)]. Successful reassociation sets the non-FILS STA's state to State 3 or State 4 (with respect to the AP or PCP that was sent the Reassociation Request frame). Successful reassociation when not in State 1 sets the STA's state to State 2 (with respect to the current AP or PCP, if this is not the AP or PCP that was sent the Reassociation Request frame). Successful reassociation sets a FILS STA's state to State 4 (with respect to the AP or PCP that was sent the Reassociation Request frame) and enables it to exchange Class 3 frames. Reassociation shall be performed only if the originating STA is already associated in the same ESS.

Disassociation notification when not in State 1 sets ~~the~~ a non-FILS STA's state to State 2. Disassociation notification when not in State 1 sets a FILS STA's state to State 1. The STA shall become associated again prior to sending Class 3 frames. A STA may disassociate a peer STA at any time, for any reason.

If non-DMG STA A in an infrastructure BSS receives a Class 3 frame from STA B that is authenticated but not associated with STA A (i.e., the state for STA B is State 2), STA A shall discard the frame. If the frame has an individual address in the Address 1 field, the MLME of STA A shall send a Disassociation frame to STA B.

If DMG STA A in an infrastructure BSS receives a Class 3 frame from STA B that is not associated with STA A (i.e., the state for STA B is State 2), STA A shall discard the frame. If the frame has an individual address in the Address 1 field, the MLME of STA A shall send a Disassociation frame to STA B.

If an MM-SME coordinated STA receives an Association Response frame with a result code equal to SUCCESS and with the value of the Single AID field within MMS element equal to 1, then

— For each of its MAC entities advertised within the MMS element and for which dot11RSNAActivated is true, the state is set to State 3. Progress from State 3 to State 4 occurs independently in each such MAC entity.

— For each of its MAC entities advertised within the MMS element and for which dot11RSNAActivated is false, the state is set to State 4.

If the MM-SME coordinated STA in State 3 is assigned an AID for only the MAC entity identified by the RA field of the Association Response frame with result code equal to SUCCESS, the MM-SME may repeat the association procedure for any other MAC entity coordinated by the MM-SME.

Association is not applicable in an IBSS. In an infrastructure BSS, association is required. In a PBSS, association is optional. APs do not initiate association.

11.3.5.2 Non-AP and non-PCP STA association initiation procedures

Change 11.3.5.2 as follows:

g) If an MLME-ASSOCIATE.confirm primitive is received with a ResultCode of SUCCESS, and RSNA is required, and FILS authentication was not used, then the SME shall perform a 4-way handshake to establish an RSNA. As a part of a successful 4-way handshake, the SME shall enable~~s~~ protection by generating an MLME-SETPROTECTION.request(Rx_Tx) primitive. If an MLME-ASSOCIATE.confirm primitive is received with a ResultCode of SUCCESS, and FILS authentication was used, then the SME shall enable protection by generating an MLME-SETPROTECTION.request(Rx_Tx) primitive.

11.3.5.3 AP or PCP association receipt procedures

Change 11.3.5.3 as follows:

n) If the ResultCode in the MLME-ASSOCIATE.response primitive is SUCCESS and RSNA establishment is required, and FILS authentication was not used, the SME shall attempt a 4-way handshake. Upon a successful completion of ~~a~~the 4-way handshake, the SME shall enable protection by issuing an MLME-SETPROTECTION.request(Rx_Tx) primitive. ~~and the state for the STA shall be set to State 4.~~ If FILS authentication was used, the SME shall enable protection by generating an MLME-SETPROTECTION.request(Rx_Tx) primitive. In either case, upon receipt of the MLME-SETPROTECTION.request(Rx_Tx), the MLME shall set the state for the STA to State 4.

11.3.5.4 Non-AP and non-PCP STA reassociation initiation procedures

Change 11.3.5.4 as follows:

g) If an MLME-REASSOCIATE.confirm primitive is received with a ResultCode of SUCCESS, and RSNA is required, and FILS authentication was not used, and the STA is in State 3, then the SME shall perform a 4-way handshake to establish an RSNA. As a part of a successful 4-way handshake, the SME shall enable protection by ~~generation~~generating an MLME-SETPROTECTION.request(Rx_Tx) primitive. If an MLME-REASSOCIATE.confirm primitive is received with a ResultCode of SUCCESS, and FILS authentication was used, and the STA is in State 3, then the SME shall enable protection by generating an MLME-SETPROTECTION.request(Rx_Tx) primitive.

11.3.5.5 AP or PCP reassociation receipt procedures

Change 11.3.5.5 as follows:

n) If the ResultCode in the MLME-REASSOCIATE.response primitive is SUCCESS, RSNA establishment is required, and the reassociation is not part of a fast BSS transition, and FILS is not in use, the SME shall attempt a 4-way handshake. Upon a successful completion of a 4-way handshake, the SME shall enable protection by issuing an MLME-SETPROTECTION.request(Rx_Tx) primitive ~~and the state for the STA shall be set to State 4~~. If FILS authentication was used, the SME shall enable protection by generating an MLME-SETPROTECTION.request(Rx_Tx) primitive. In either case, upon receipt of the MLME-SETPROTECTION.request(Rx_Tx), the MLME shall set the state for the STA to State 4.

11.25 WLAN interworking with external networks procedures

11.25.3.3 ANQP procedures

11.25.3.3.1 General

Insert the following new paragraphs after the fourth paragraph of 11.25.3.3.1:

A STA receiving a CAG ANQP-element in an ANQP response should cache the ANQP-element contents, for later use. The STA stores the ANQP CAG Version in the received CAG ANQP-element, the ANQP attributes, and information corresponding to that CAG Version with the values of BSSID, or HESSID and the corresponding SSID associated with the responding AP within the cache.

The AP obtains the current value of a CAG Version from the associated advertisement server. How the AP obtains the value of the CAG Version from the associated advertisement server is out of the scope of this document.

When dot11InterworkingServiceActivated is true, an AP may provide one or more current version numbers of the CAG (CAG Version) associated with the AP, where each version number is associated with a specific advertisement protocol by including the CAG Number element in the Beacon or Probe Response frame.

In order to reduce the network discovery delay at an AP, the BSSID, HESSID, or the corresponding SSID of that AP are used as a key index in the STA’s cache to determine if an ANQP CAG Version received from an AP matches previously cached information.

A FILS STA that makes use of the CAG Version is subject to the following:

- The STA may obtain the CAG Version from the CAG element received in Beacon or Probe Response frames.
- The STA searches its cached information using either the BSSID, HESSID, or SSID of the AP as an index to obtain the correct cached ANQP CAG Version entry.
- If the cached ANQP CAG Version does not match the CAG version received in Beacon or Probe Response frames, the STA should transmit an ANQP query request for any of the ANQP-elements contained within the CAG. If the CAG Versions do match, the STA should use the stored ANQP attributes and information of the CAG for network discovery.

The ANQP server assignment of the ANQP CAG Version associated with the to ANQP attributes and information is out of the scope of this document.

Insert the following new rows into the Table 11-15:

Table 11-15—ANQP usage

			BSS		IBSS
ANQP-element name	**ANQP-element (subclause)**	**ANQP-element type**	**AP**	**Non-AP STA**	**STA**
Query AP List	9.4.5.24	Q	R	T	—
AP List Response	9.4.5.25	S	T	R	—
FILS Realm Information	9.4.5.26	S	T	R	—
Common Advertisement Group	9.4.5.27	S	T	R	—

Insert the new subclause 11.25.3.2.14 as follows:

11.25.3.2.14 Query AP List procedure

The Query AP List ANQP-element is used by a requesting STA to perform a single ANQP query to an AP, using the procedures defined in 11.25.3.3.1 requesting the ANQP information on each AP indicated in the AP list. The requesting STA shall only include Info IDs in the Query List ANQP-element that have the sole ANQP-element type of S as shown in Table 11-15. Info IDs that have an ANQP-element type of Q shall not be included in the Query AP List ANQP-element (e.g., the Info ID for Vendor Specific ANQP-element shall not be included).

A responding STA that encounters an unknown or reserved ANQP Info ID value in a Query AP List ANQP-element shall ignore that ANQP Info ID and shall parse any remaining ANQP Info IDs.

In the response to a Query AP List, the Query Response field contains an AP List Response ANQP-element. The AP List Response contains multiple query response reports for the APs indicated in the Query AP List. If the responding AP does not provide the requested information for all APs specified in the Query AP List, or for all the requested Info IDs, the STA interprets it as that information not available. How the information obtained from the AP list is used by the STA is left to the implementation and is beyond the scope of this specification.

Insert the new subclause 11.25.3.2.15 as follows:

11.25.3.2.15 CAG procedure

The CAG ANQP-element is used by a requesting STA to perform an ANQP query to retrieve the Info IDs contained within the CAG associated with ANQP and the current ANQP CAG Version associated with these Info IDs. For this purpose, a requesting STA shall use the procedures defined in 11.25.3.3.1 and 11.25.3.2.2 and shall include in the ANQP query the Info ID of the CAG ANQP-element as shown in Table 11-15. When a responding AP receives a Query List ANQP-element that contains the Info ID of CAG ANQP-element, the responding STA shall include in the ANQP Query Response frame a CAG ANQP-element containing the ANQP CAG Version and the Info IDs of the ANQP-elements that are in the CAG associated with ANQP in the increasing order of the Info ID values. The ANQP query response also includes the (other) ANQP-elements that a STA requested in the ANQP query list in the increasing order of the Info ID values.

The ANQP CAG Version is an unsigned number and it is incremented when there is any change in the CAG associated with ANQP, including a change of the Info ID of the ANQP-elements of the CAG or a change in the values of the ANQP-elements included in the CAG. If the ANQP CAG Version exceeds 255, it is reset to 0.

11.44 Operation under the control of a GDB

11.44.8 Reduced neighbor report

Change 11.44.8 as follows:

In Beacon and Probe Response frames, a Reduced Neighbor Report element may be transmitted by an AP with dot11TVHTOptionImplemented or dot11FILSActivated true. In FILS Discovery frames, a Reduced Neighbor Report element is optionally sent by a FILS AP. A Reduced Neighbor Report element contains information on neighbor APs.A Reduced Neighbor Report element might not be exhaustive either by choice or by the fact that there may be neighbor APs not known to the AP.

Insert new subclause 11.47 as follows:

11.47 Fast Initial Link Setup (FILS) procedures

11.47.1 General

This subclause describes the Fast Initial Link Setup (FILS) procedures that are used for FILS STAs. FILS is only supported in non-DMG infrastructure BSS.

A FILS STA is a QoS STA and shall set dot11QosOptionImplemented to true.

A FILS non-AP STA shall set the FILS Capability field to 1 in the Extended Capabilities element included in Probe Request and (Re)Association Request frames. A FILS non-AP STA may include FILS Request Parameters or AP-CSN elements in Probe Request frames. A FILS non-AP STA may set the Authentication Algorithm Number field to the value of Fast Initial Link Setup (FILS) authentication in the Authentication frame with the authentication transaction sequence number set to 1 (Authentication Request). A FILS AP advertises its FILS authentication and FILS higher layer setup capabilities by including a FILS Indication element, as specified in 9.4.2.183 and 11.47.4, in Beacon, Probe Response, and/or FILS Discovery frames.

A FILS AP shall set the FILS Capability field to 1 in the Extended Capabilities element and shall include the FILS Indication element in Beacon frames, Probe Response frames, and (Re)Association Response frames. A FILS AP may transmit FILS Discovery frames.

11.47.2 FILS Discovery frame generation and usage

11.47.2.1 FILS Discovery frame transmission

A FILS AP supporting FILS discovery may generate and transmit FILS Discovery frames. The FILS Discovery frame shall not be transmitted in a DSSS or HR/DSSS PPDU.

An AP may transmit a FILS Discovery frame as a non-HT duplicate PPDU. When a FILS Discovery frame is transmitted as a non-HT duplicate PPDU, its primary channel shall be indicated by its Primary Channel field.

If an AP transmits a FILS Discovery frame as a non-HT duplicate PPDU in an 80+80 MHz channel bandwidth, the Channel Center Frequency Segment 1 field shall be present in the FILS Discovery frame and shall be set to the channel center frequency of the frequency segment 1 for an 80+80 MHz VHT operating channel.

An AP transmitting a FILS Discovery frame may transmit the FILS Discovery frame between Beacon frames. The interval between the transmission of a Beacon frame and a subsequent FILS Discovery frame shall be no less than the interval indicated in dot11FILSFDFrameBeaconMinimumInterval. The transmission interval between subsequent FILS Discovery frames by an AP in a beacon interval shall be no less than the interval indicated in dot11FILSFDFrameBeaconMinimumInterval. If dot11FILSFDFrameBeaconMaximumInteval is not equal to 0, and if a Beacon frame or FD frame has not been transmitted by an AP for a period that is equal to dot11FILSFDFrameBeaconMaximumInterval, that AP shall queue for transmission a FD frame or a Beacon frame unless the next TBTT is within a duration indicated by the value of dot11-FILSFDFrameBeaconMinimumInterval.

The transmitted FILS Discovery frame shall contain the FILS Discovery Information field.

An AP may use the FILS Minimum Rate subfield in the FILS Discovery frame to indicate the minimum rate to be used by the AP and FILS STAs in subsequent transmissions between the AP and FILS STAs.

An AP may include its RSN information in the FD RSN subfield of the FILS Discovery frame as described in 12.12.2.2.

11.47.2.2 FILS Discovery frame reception

If a FILS STA has the ReportingOption parameter present in the MLME-SCAN.request primitive and it is not equal to IMMEDIATE or CHANNEL_SPECIFIC, then the STA shall follow the procedures indicated in 11.1.4.1 and not the procedures provided in this clause.

If an AP has indicated the FILS Minimum Rate in the FILS Minimum Rate subfield of a FILS Discovery frame, a scanning FILS STA that receives such a FILS Discovery frame shall use a data rate that is equal or higher than the indicated FILS Minimum Rate in subsequent transmissions between the AP and the FILS STA.

A scanning FILS STA that receives a FILS Discovery frame compares the received SSID or Short SSID in the FILS Discovery frame with the SSID parameter or SSID list provided to the STA previously in a MLME-SCAN.request primitive. If the STA has the ReportingOption parameter present in the MLME-SCAN.request primitive and equal to IMMEDIATE and if the SSID in the FILS Discovery frame matches the SSID parameter or one of the SSIDs in the SSID list, the STA shall issue an MLME-SCAN.confirm primitive with the information obtained from the received FILS Discovery frame immediately after the reception of the FILS Discovery frame, with the ResultCode equal to INTERMEDIATE_SCAN_RESULT.If the STA has the ReportingOption parameter in the MLME-SCAN. Request primitive equal to CHANNEL_SPECIFIC and if the SSID in the FILS Discovery frame matches the SSID parameter or one of the SSIDs in the SSID list, the STA shall issue an MLME-SCAN.confirm primitive with the information obtained from the received FILS Discovery frame after the STA has completed the scanning of the current channel, with the ResultCode equal to INTERMEDIATE_SCAN_RESULT.

If the received FILS Discovery frame contains the AP-CSN subfield as defined in 11.1.4.3.7 and the non-AP STA maintains previously obtained BSS Configuration Parameter Sets, the non-AP STA shall use the received FD AP-CSN information as follows:

— The STA shall check if the BSSID in the received FILS Discovery frame is equal to a BSSID in the previously obtained BSS Configuration Parameter Sets.

— If an equal BSSID is found, the STA compares the AP-CSN value in the received FILS Discovery frame to the AP-CSN value associated with the BSSID in the BSS Configuration Parameter Sets.

— If the AP-CSN values are not equal, a FILS non-AP STA may send a Probe Request frame including an AP-CSN element, with the value of the AP-CSN associated with the BSSID in the BSS Configuration Parameter set in the non-AP STA. When sending a Probe Request frame including an AP-CSN element, the FILS non-AP STA shall set the Address 1 and Address 3 fields in the Probe Request frame to the BSSID of the AP, of which the AP-CSN is being sent.

— If the AP-CSN values are equal, then the non-AP STA may use the information contained in the BSS Configuration Parameter Set to initiate one or more FILS procedures (as defined in 11.47.3, 11.47.4, and 11.47.5), without waiting for next Beacon frame or Probe Response frame.

— If the non-AP STA has not successfully associated with an AP using the above procedures, it shall follow the procedures specified in 11.3.4.2 and 11.3.4.3.

If a received FILS Discovery frame contains RSN information in the FD RSN subfield, a FILS STA may conduct FILS authentication with the AP that transmitted the FILS Discovery frame as described in 12.12.2.

A scanning FILS STA that receives a FILS discovery frame can compute the next TBTT based on the Timestamp subfield of the FILS Discovery Information field (see Figure 9-687b) and beacon interval as follows:

$$\text{Next TBTT} = \text{Ceil}\,(\text{Timestamp}/(\text{Beacon Interval} \times 1024)) \times (\text{Beacon Interval} \times 1024)$$

11.47.3 Higher layer setup during (re)association procedure

11.47.3.1 General

Higher layer setup, such as IP layer setup, may be performed during a STA's FILS (re)association procedure. Two mechanisms are provided for higher layer setup. One is the higher layer protocol (HLP) encapsulation. The HLP encapsulation, described in 11.47.3.2, shall be supported by all FILS STAs. The other mechanism is the FILS IP address configuration. This is optional for FILS STAs and described in 11.47.3.3. FILS IP address configuration is used to reduce the overhead caused by using the DHCP for the IP address assignment. However, FILS IP Address Assignment method provides a subset of features supported by the DHCP. The AP advertises whether it supports the FILS IP address configuration or not by the FILS IP address configuration in the FILS Indication element (9.4.2.183) in Beacon and Probe Response frames.

NOTE—The non-AP STA can use the following methods to obtain an IP address: 1) FILS IP address configuration, if supported by the AP. 2) Encapsulating higher layer protocols during association.

Higher layer setup information in (Re)Association Request/Response frames shall be protected by the authenticated encryption with associated data (AEAD) scheme (12.12.2.7).

11.47.3.2 Higher layer protocol encapsulation

The FILS HLP Container element (9.4.2.184) is used for encapsulating higher layer protocol (HLP) packets.

If a non-AP STA uses higher layer protocol encapsulation, the non-AP STA shall construct a FILS HLP Container element for each HLP packet. The non-AP STA may put multiple FILS HLP Container elements into a (Re)Association Request frame as long as they fit in the MMPDU size limit. If the size of the (Re)Association Request frame exceeds the maximum MMPDU size, the non-AP STA shall remove the last FILS HLP Container element from the (Re)Association Request frame until the size of the (Re)Association Request frame does not exceed the maximum MMPDU size. The non-AP STA shall send the HLP packets contained in the removed FILS HLP Container elements as Data frames after association. Then the non-AP STA transmits a (Re)Association Request frame including all of the subjected FILS HLP Container elements. The HLP packet in the FILS HLP Container element can contain any MSDU format defined in 5.1.4. The FILS HLP Container element may be fragmented as described in 10.27.11 if required. The encapsulation procedure is as follows:

a) The non-AP STA fills FILS HLP Container element(s) with the destination MAC address, the source MAC address of the HLP packet and the HLP packet in MSDU format (see 5.1.4). The source MAC address shall be the MAC address of the non-AP STA.
b) The non-AP STA includes the FILS HLP Container element(s) into the (Re)Association Request frame.

If the AP receives a (Re)Association Request frame including FILS HLP Container element(s), the AP decapsulates the HLP packet(s) but shall not transfer the HLP packet(s) until the key confirmation (see 12.12.2.6) is successfully completed. After successful key confirmation, the AP forwards the HLP packet(s) to the upstream network or BSS according to the destination MAC address of the HLP packet(s). The order of forwarding the HLP packets shall be the same as the order of the FILS HLP Container elements in the (Re)Association Request frame. If the key confirmation fails, the AP discards the HLP packet(s). The AP may filter HLP packets based on rules that are out of scope for this standard. The packet decapsulation procedure for each FILS HLP Container element is as follows:

1) The AP extracts the destination MAC address, the source MAC address and the HLP packet from the FILS HLP Container element;
2) The AP verifies that the extracted source MAC address is equal to the source MAC address of the (Re)Association frame. If these are different, the AP shall discard the FILS HLP Container element;
3) The AP constructs the frame in appropriate format to deliver the HLP packet to the upstream network or BSS by using the extracted destination MAC address, source MAC address, and HLP packet.

The AP should wait to a transmit (Re)Association Response frame until dot11HLPWaitTime has elapsed after receiving a (Re)Association Request. If, before it transmits a (Re)Association Response frame, the AP receives one or more HLP packets that have the non-AP STA’s MAC address or a group address as the destination address, from the upstream network or BSS. The order of the FILS HLP Container elements in the (Re)Association Response frame is same as the order of receiving the HLP packets. If the AP receives HLP packets for the non-AP STA after transmitting a (Re)Association Response frame, the AP transmits the HLP packets as Data frames. If, before it transmits a (Re)Association Response frame, the AP does not receive any HLP packets that have the non-AP STA’s MAC address or a group address as the destination address, from the upstream network or BSS. The status code in the (Re)Association Response frame is not effected by the presence or absence of a FILS HLP Container element. The packet encapsulation procedure for each FILS HLP Container element is as follows:

- The AP fills the FILS HLP Container element with the destination MAC address, the source MAC address and the HLP packet. The source MAC address shall be the source MAC address of the received HLP packet. The destination MAC address shall be the destination MAC address of the received HLP packet. It is the MAC address of the non-AP STA or a group address. The HLP packet shall be in MSDU format (see 5.1.4).
- The AP includes the FILS HLP Container elements into the (Re)Association Response frame.

If the non-AP STA receives a (Re)Association Response frame with one or more FILS HLP Container elements, the non-AP STA performs key confirmation (12.12.2.6) first. After successful key confirmation, the non-AP STA shall generate an MA-UNITDATA.indication primitive for each HLP packet. The order of generating MA-UNITDATA.indicate primitive of the HLP packets shall be the same as the order of the FILS HLP Container elements in the (Re)Association Response frame. If the key confirmation fails, the non-AP STA shall discard the HLP packet(s). The packet decapsulation procedure for each FILS HLP Container element is as follows:

- The non-AP STA extracts the destination MAC address, source MAC address, and the HLP packet.
- The non-AP STA shall verify that the extracted destination MAC address is equal to the MAC address of the non-AP STA or group addresses. If the destination MAC address is not for the non-AP STA, the non-AP STA shall discard the FILS HLP Container element.
- The non-AP STA shall generate an MA-UNITDATA.indication primitive with the following parameters:
    1) source address: The extracted source MAC address
    2) destination address: The extracted destination MAC address
    3) routing information: null
    4) data: The extracted HLP packet

5) reception status: success
6) priority: Contention
7) service class: QoSAck when the destination address is a unicast address. QoSNoAck when the destination address is not a unicast address

For example, this mechanism can be used for IP address configuration. If the network uses DHCPv4/v6 (IETF RFC 2131, IETF RFC 3315), the FILS HLP Container element can carry DHCPv4/v6 packets. If the network uses IPv6 stateless address autoconfiguration (IETF RFC 4862), it can carry Router Solicitation and Router Advertisement packets (IETF RFC 4861).

11.47.3.3 FILS IP address configuration

In order to request an IP address, a STA may include a FILS IP Address Assignment element in the (Re)Association Request frame or FILS Container frame that it sends to the AP.

The AP may send the IP address assigned to the STA in a FILS IP Address Assignment element (9.4.2.186) that is included in a (Re)Association Response frame or a FILS Container frame. Methods for determining the IP address to be assigned to a STA are out of scope in this document.

When the AP receives a (Re)Association Request frame including a FILS IP Address Assignment element or a FILS Container frame, the AP initiates a procedure to assign an IP address for the STA using a mechanism that is outside the scope of this standard.

If the STA has included a FILS IP Address Assignment element in the (Re)Association Request frame, then the AP may respond to the STA in one of the following ways:

— If the AP is able to assign an IP address in the (Re)Association Response frame, then the AP sets the IP address assignment pending flag in the IP Address Response Control field of the FILS IP Address Assignment element to 0 and includes the IP Address Data field as defined in 9.4.2.185 in the (Re)Association Response frame. For IPv6 addresses, an AP performs Duplicate Address Detection (IETF RFC 4862) before assigning an IPv6 address for the STA.

— If the AP is unable to assign an IP address in the (Re)Association Response frame, then the AP sets the IP address assignment pending flag in the IP Address Response Control field of the FILS IP Address Assignment element to 1 and sets the IP address request timeout to 0 in (Re)Association Response frame.

— If the AP needs more time to assign an IP address, the AP sets the IP address assignment pending flag in the IP Address Response Control field of the FILS IP Address Assignment element to 1 and sets the IP address request timeout to the maximum estimated time in the unit of seconds, within which the AP tries to assign an IP address to the requesting STA in the (Re)Association Response frame. When the AP is ready with an IP address within IP address request timeout period, then AP shall send the IP address to the STA using a FILS Container frame. If the STA does not receive the FILS Container frame containing IP assignment within IP address request timeout period, then the STA may initiate IP address assignment procedure using a FILS Container frame or mechanisms that are out of scope of this specification. If an STA has initiated an IP address assignment procedure (using mechanisms that are out of scope) due to the expiry of the timeout period, and subsequently receives an FILS container frame containing an IP assignment, it shall discard the IP address assignment received through the FILS container frame.

The STA may use the MLME-FILSContainer.request primitive to re-request its IP address to extend its lifetime and include the requested IP address in an IP Address Assignment element in a FILS Container frame. If the STA has included an IP Address Assignment element in the FILS Container frame, then the AP may respond to the STA using the MLME-FILSContainer.response primitive in one of the following ways:

- If the AP is able to assign an IP address immediately, then the AP sets the IP address assignment pending flag in the IP Address Response Control field of the FILS IP Address Assignment element to 0 and includes the IP Address Data field as defined in 9.4.2.185 in the FILS Container frame.
- If the AP is unable to assign an IP address, then the AP sets the IP address assignment pending flag in the IP Address Response Control field of the FILS IP Address Assignment element to 1 and sets the IP address request timeout to 0 in the FILS Container frame.
- If the AP needs more time to assign an IP address, then the AP sets the IP address assignment pending flag in the IP Address Response Control field of the FILS IP Address Assignment element to 1 and sets the IP address request timeout to the maximum estimated time in the unit of seconds within which it (AP) tries to assign an IP address to the requesting STA in FILS Container frame. When the AP is ready to assign an IP address within IP address request timeout period, then the AP shall send the IP address to the STA using a FILS Container frame. If the STA does not receive the FILS Container frame containing an IP assignment within the IP address request timeout period, then the STA may initiate an IP address assignment procedure using mechanisms that are out of scope of this specification. If an STA has initiated an IP address assignment procedure (using mechanisms that are out of scope) due to the expiry of the timeout period, and subsequently receives an FILS container frame containing an IP assignment, it shall discard the IP address assignment received through the FILS container frame.

If a non-AP STA determines a duplicate IP address assignment (through means that are out of scope for this standard), it may discard the assigned IP address and request a new IP address.

11.47.4 FILS authentication and higher layer setup capability indications

A FILS AP shall include a FILS Indication element in Beacon and Probe Response frames, and may include a FILS Indication element in FILS Discovery frames. The FILS Indication element indicates properties of the FILS authentication protocol used and whether the AP performs IP address assignment.

An AP can indicate up to 7 realms that indicate the domain names of the server that the AP is capable of participating in an EAP-RP exchange with (see IETF RFC 6696). The realm of an EAP-RP server is the realm portion of the keyName-NA as defined in IETF RFC 6696. For each of the realms, the FILS Indication element carries a 2-octet hash of the network realm. The hash of the realm (IETF RFC 1035 compliant) is computed as follows:

NOTE—Internationalized domain names are first converted to an IETF RFC 1035 compliant ASCII form using the operations defined in IETF RFC 3490.

$$H = L(SHA256(ToLowerCase(D),0,16))$$

where

H	is the hashed realm name
L	is defined in 11.6.1
ToLowerCase	is the function that converts upper case characters to lower case
D for a non-AP STA	is NAI Realm used of the EAP-RP server used in EAP-RP authentication

11.47.5 Differentiated initial link setup

11.47.5.1 General

To limit the number of STAs that attempt link setup concurrently, the differentiated link setup procedure provides a method for an AP to moderate the rate that non-AP STAs transmit Authentication and (Re)Association frames to the AP.

11.47.5.2 AP procedures for differentiated initial link setup

If dot11DILSImplemented is true, a FILS AP may include the Differentiated Initial Link Setup element in Beacon and Probe Response frames, and set the Differentiated FILS Time and the optional fields to limit the number of STAs that are allowed to attempt link setup concurrently.

The AP may set a Differentiated FILS Time reserved for high priority link setup, and may set the FILS User Priority field, MAC Address Filter field, and/or Vendor Specific field to specify a subset of STAs that may attempt fast initial link setup during the reserved Differentiated FILS Time specified in the element.

An AP may set the FILS User Priority B0, B1, and B2 to 1 to indicate high priority link setup without additional delays for the STAs based on the queued type of traffic as specified in 9.4.2.188.

An AP may set the Bit Pattern Length subfield in the MAC Address Filter field to decide the number of bits used for MAC address filtering, and specify the bit pattern in the Bit Pattern subfield to allow STAs with specific MAC addresses to transmit fast initial link setup frames immediately. The more bits used for MAC address filtering, the fewer STAs are allowed to transmit a fast initial link setup frame immediately. How an AP sets the bit pattern in the Bit Pattern subfield is beyond the scope of this specification.

An AP may set one or more vendor specific criteria in a Vendor Specific field to allow a set of STAs that satisfy the specified criteria to transmit fast initial link setup frames to the AP without additional delays.

11.47.5.3 Non-AP STA procedures for differentiated initial link setup

When a FILS non-AP STA with dot11DILSImplemented value of true receives a Beacon or Probe Response frame that includes a Differentiated Initial Link Setup element, the non-AP STA shall check the FILSC Type field to determine if it satisfies the condition specified in each and every optional field that is present. If the non-AP STA satisfies all of the conditions specified in the present optional fields, the non-AP STA has a FILSC of 1 and it proceeds with a FILS procedure with the AP without additional delays. Otherwise, the non-AP STA shall have a FILSC of 0 and shall postpone the link setup with the AP until the time specified in the last received Differentiated FILS Time field elapses. Each time the non-AP STA receives a Beacon and/or Probe Response frame including a Differentiated Initial Link Setup element, the non-AP STA shall check the FILSC Type field and update its FILSC.

When the FILS User Priority field is present, the FILS User Priority condition is satisfied if the non-AP STA has frames with user priority 4–7 in the transmission queue(s) and the FILS User Priority B0 is 1, or if the non-AP STA has frames with user priority 0–3 in their transmission queue(s) and the FILS User Priority B1 is 1, or if the non-AP STA has no frame in their transmission queue(s) and the FILS User Priority B2 is 1. If a STA has frames in multiple queues with different priorities, the STA attempts to associate with the AP based on its highest priority queue.

Any combination of bit values for B0, B1, and B2 is allowed. For instance, B2B1B0=011 indicates that only STAs with traffic are allowed for high priority link setup and those STAs with no data frames in their queues are not allowed. A value of B2B1B0=000 indicates that no FILS station is allowed to join the AP in the next specified time interval.

If a MAC Address Filter field is present, the non-AP STA shall compare its MAC address to the Bit Pattern subfield in the MAC Address Filter field. If the value of the last *n* LSBs of the non-AP STA's MAC address matches the value of the bits used for MAC address filtering in the Bit Pattern subfield, where *n* is specified in the Bit Pattern Length subfield, the MAC address condition is satisfied.

12. Security

12.5 RSNA confidentiality and integrity protocols

12.5.4 Broadcast/multicast integrity protocol (BIP)

12.5.4.4 BIP replay protection

Change 12.5.4.4 as follows:

The MME Sequence Number field represents a sequence number whose length is 6 octets.

When management frame protection is negotiated, the receiver shall maintain a 48-bit replay counter for each IGTK. The receiver shall set the receive replay counter to the value of the IPN in the IGTK key data encapsulation (KDE) (see 12.7.2) provided by the Authenticator in ~~either~~the 4-way handshake, FT 4-way handshake, FT handshake, ~~or~~group key handshake, or FILS authentication. The transmitter shall maintain a single IPN for each IGTK. The IPN shall be implemented as a 48-bit strictly increasing integer, initialized to 1 when the corresponding IGTK is initialized. The transmitter may reinitialize the sequence counter when the IGTK is refreshed. See 12.5.4.5 and 12.5.4.6 for per packet BIP processing.

12.6 RSNA security association management

12.6.1 Security associations

12.6.1.1 Security association definitions

12.6.1.1.1 General

Change 12.6.1.1.1 as follows:

- PMKSA: A result of a successful IEEE 802.1X exchange, SAE authentication, FILS authentication, or preshared PMK information. A PMSKA can be cached.
- PMK-R0 security association: A result of a successful FT initial mobility domain association.
- PMK-R1 security association: A result of a successful FT initial mobility domain association or FT authentication sequence.
- Mesh PMKSA: A result of successful completion of the active authentication protocol.
- PTKSA: A result of a successful 4-way handshake, FT 4-way handshake, ~~or~~FT authentication sequence, or FILS authentication.
- Mesh TKSA: A result of a successful authenticated mesh peering exchange (AMPE).
- GTKSA: A result of a successful group key handshake, 4-way handshake, FT 4-way handshake, ~~or~~ FT authentication sequence, or FILS authentication.
- IGTKSA: A result of a successful group key handshake, successful 4-way handshake, successful FT 4-way handshake, ~~or~~ the Reassociation Response frame of the fast BSS transition protocol, or successful FILS authentication.
- Mesh GTKSA: A result of a successful AMPE or mesh group key handshake.
- SMKSA: A result of a successful initial SMK handshake.
- STKSA: A result of a successful 4-way STK handshake following the initial SMK handshake or subsequent rekeying.

12.6.1.1.2 PMKSA

Change 12.6.1.1.2 as follows:

When the PMKSA is the result of a successful IEEE 802.1X authentication, it is derived from the EAP authentication and authorization parameters provided by the AS. When the PMKSA is the result of a successful SAE authentication, it is generated as a result of the successful completion of the SAE exchange. ~~This security~~A PMKSA association is bidirectional. In other words, both parties use the information in the security association for both sending and receiving. The PMKSA is created by the Supplicant's SME when the EAP authentication or FILS authentication completes successfully or the PSK is configured. The PMKSA is created by the Authenticator's SME when the PMK is created from the keying information transferred from the AS~~, when ~~in an IEEE 802.1X authentication ~~is utilized~~exchange, when the FILS authentication completes successfully, when the SAE exchange successfully completes, or when the PSK is configured. The PMKSA is used to create the PTKSA. PMKSAs have a certain lifetime. The PMKSA consists of the following:

— PMKID, as defined in 12.7.1.3. The PMKID identifies the security association.

— Authenticator's or peer's MAC address. For multiband RSNA, the MAC address is associated with the operating band in use when the PMKSA is established.

— PMK.

— Lifetime, as defined in 12.7.1.3.

— AKMP.

— All authorization parameters specified by the AS or local configuration. This might include parameters such as the STA's authorized SSID.

— Cache Identifier, if advertised by the AP in FILS Indication element.

12.6.1.1.6 PTKSA

Change 12.6.1.1.6 as follows:

The PTKSA ~~is a~~ results from a successful ~~of the~~4-way handshake, FT 4-way handshake, FT protocol, ~~or~~ FT resource request protocol, or FILS authentication. This security association is also bidirectional. PTKSAs have the same lifetime as the PMKSA or PMK-R1 security association, whichever comes first. Because the PTKSA is tied to the PMKSA or to a PMK-R1 security association, it only has the additional information from the 4-way handshake, or FT Protocol authentication, or FILS authentication. For the PTKSA derived as a result of the 4-way handshake, there shall be only one PTKSA per band (see 12.6.19) with the same Supplicant and Authenticator MAC addresses. For the PTKSA derived as a result of an initial mobility domain association or fast BSS transition, there shall be only one PTKSA with the same STA's MAC address and BSSID.

During the 4-way handshake defined in 12.7.6.5 and the FT 4-way handshake defined in 13.4.2, there is state created between message 1 and message 3 of the Handshake. This does not create a PTKSA until message 3 is validated by the Supplicant and message 4 is validated by the Authenticator.

During the FT authentication sequence defined in 13.8, the PTKSA is validated when message 3 is validated by the R1KH and message 4 is validated by the S1KH.

During the FILS authentication sequence defined in 12.12.2, the PTKSA is validated by key confirmation using (Re)Association Request and (Re)Association Response frames.

The PTKSA consists of the following:

- PTK
- Pairwise cipher suite selector
- Supplicant MAC address or STA's MAC address
- Authenticator MAC address or BSSID
- Key ID
- If FT key hierarchy is used,
  - R1KH-ID
  - S1KH-ID
  - PTKName

12.6.1.1.8 GTKSA

Change 12.6.1.1.8 as follows:

The GTKSA results from a successful 4-way handshake, FT 4-way handshake, FT protocol, FT resource request protocol, ~~or the~~group key handshake, or FILS authentication, and is unidirectional. In an infrastructure BSS, there is one GTKSA, used exclusively for encrypting group addressed MPDUs that are transmitted by the AP and for decrypting group addressed transmissions that are received by the STAs. In an IBSS or in a PBSS, each STA defines its own GTKSA, which is used to encrypt its group addressed transmissions, and stores a separate GTKSA for each peer STA so that encrypted group addressed traffic received from other STAs may be decrypted. A GTKSA is created by the Supplicant's SME when message 3 of the 4-way handshake is received, when message 1 of the group key handshake is received, ~~or~~ when a Reassociation Response frame of the FT handshake is received, or when the FILS authentication with a status code indicating success is received. The GTKSA is created by the Authenticator's SME when the SME changes the GTK and has sent the GTK to all STAs with which it has a PTKSA. A GTKSA consists of the following:

12.6.1.1.9 IGTKSA

Change 12.6.1.1.9 as follows:

When management frame protection is enabled, a non-AP STA's SME creates an IGTKSA when it receives a valid Message 3 of the 4-way handshake or FT 4-way handshake, the Reassociation Response frame of the fast BSS transition protocol with a status code indicating success, a Mesh Peering Open Message of the Authenticated Mesh Peering Exchange (AMPE) protocol, ~~or~~ a valid Message 1 of the group key handshake, or the (Re)Association Response frame of FILS authentication with a status code indicating success. The Authenticator's SME creates an IGTKSA when it establishes or changes the IGTK with all STAs to which it has a valid PTKSA or MTKSA.

12.6.1.3.2 Security association in an ESS

Change 12.6.1.3.2 as follows:

A STA and AP establish an initial security association via the following steps:

a) The STA selects an authorized ESS by selecting among APs that advertise an appropriate SSID.

b) The STA then performs IEEE 802.11 authentication followed by association to the chosen AP. Confirmation of security parameters takes place during association. A STA performing IEEE 802.1X authentication uses Open System authentication. A STA performing secure password-based authentication can use SAE authentication. A STA performing FILS uses FILS authentication.

NOTE 1—It is possible for more than one PMKSA to exist. As an example, a second PMKSA might come into existence through PMKSA caching. A STA might leave the ESS and flush its cache. Before its PMKSA expires in the AP's cache, the STA returns to the ESS and establishes a second PMKSA from the AP's perspective.

NOTE 2—An attack altering the security parameters is detected by the key derivation procedure.

NOTE 3—IEEE 802.11 Open System authentication provides no security, but is included to maintain backward compatibility with the IEEE 802.11 state machine (see 11.3).

c) SAE authentication and FILS authentication provide~~s~~ mutual authentication and derivation of a PMK. If Open System authentication is chosen instead, the Authenticator or the Supplicant initiates IEEE 802.1X authentication. The EAP method used by IEEE Std 802.1X-2010 needs to support mutual authentication, as the STA needs assurance that the AP is a legitimate AP.

NOTE 4—Prior to the completion of IEEE 802.1X authentication and the installation of keys, the IEEE 802.1X Controlled Port in the AP blocks all data frames. The IEEE 802.1X Controlled Port returns to the unauthorized state and blocks all Data frames before invocation of an MLME-DELETEKEYS.request primitive. The IEEE 802.1X Uncontrolled Port allows IEEE 802.1X frames to pass between the Supplicant and Authenticator. Although IEEE Std 802.1X-2010 does not require a Supplicant Controlled Port, this standard assumes that the Supplicant has a Controlled Port in order to provide the needed level of security. Supplicants without a Controlled Port compromise RSN security and are not used.

NOTE 5—Any secure network cannot support promiscuous association, e.g., an unsecured operation of IEEE Std 802.11. A trust relationship is needed between the STA and the AS of the targeted SSID prior to association and secure operation, in order for the association to be trustworthy. The reason is that an attacker can deploy a rogue AP just as easily as a legitimate network provider can deploy a legitimate AP, so some sort of prior relationship is necessary to establish credentials between the ESS and the STA.

d) The last step is key management. The authentication process, whether SAE authentication or FILS authentication utilizing Authentication frames or IEEE 802.1X authentication utilizing Data frames post association, creates cryptographic keys shared between the cryptographic endpoints-the AP and STA, or the IEEE 802.1X AS and the STA, when using SAE/FILS or IEEE Std 802.1X, respectively. When using IEEE Std 802.1X, the AS transfers these keys to the AP, and the AP and STA uses one of the key confirmation handshakes, e.g., the 4-way handshake or FT 4-way handshake, to complete security association establishment. When using SAE authentication there is no AS and therefore no key transfer; the 4-way handshake is performed directly between the AP and STA. The key confirmation handshake indicates when the link has been secured by the keys and is ready to allow normal data traffic and protected robust management frames. When FILS authentication is performed, the key confirmation is performed as part of the FILS exchange using association frames. Hence, no additional handshake is necessary.

When FT is not enabled, a STA roaming within an ESS establishes a new PMKSA by one of the ~~four~~five schemes:

— In the case of (re)association followed by IEEE 802.1X or PSK authentication, the STA repeats the same actions as for an initial contact association, but its Supplicant also deletes the PTKSA when it roams from the old AP. The Supplicant also deletes the PTKSA when it disassociates/deauthenticates from all BSSIDs in the ESS.

— In the case of SAE authentication followed by (re)association, the STA repeats the same actions as for initial contact association, but the non-AP STA also deletes the PTKSA when it roams from the old AP. Note that a STA can take advantage of the fact that it can perform SAE authentication to multiple APs while maintaining a single association with one AP, and then use any of the PMKSAs created during authentication to effect a fast BSS transition.

- A STA (AP) can cache PMKSAs for APs (STAs) in the ESS to which it has previously performed a full IEEE 802.1X authentication or SAE authentication. If a STA wishes to roam to an AP for which it has cached one or more PMKSAs, it can include one or more PMKIDs in the RSNE of its (Re)Association Request frame. An AP that has retained the PMK for one or more of the PMKIDs can proceed with the 4-way handshake. The AP shall include the PMKID of the selected PMKSA in message 1 of the 4-way handshake. If none of the PMKIDs of the cached PMKSAs matches any of the supplied PMKIDs, or if the AKM of the cached PMKSA differs from that offered in the (Re)Association Request, then the Authenticator, in the case of Open System authentication, shall perform another IEEE 802.1X authentication and, in the case of SAE authentication, shall transmit a Deauthentication frame to the STA. Similarly, if the STA fails to send a PMKID, the STA and AP need to perform a full IEEE 802.1X authentication.
- A STA already associated with the ESS can request its IEEE 802.1X Supplicant to authenticate with a new AP before associating to that new AP. The normal operation of the DS via the old AP provides the communication between the STA and the new AP. The SME delays reassociation with the new AP until IEEE 802.1X authentication completes via the DS. If IEEE 802.1X authentication completes successfully, then PMKSAs shared between the new AP and the STA are cached, thereby enabling the possible usage of reassociation without requiring a subsequent full IEEE 802.1X authentication procedure.
- In the case of FILS authentication, the STA may repeat the same actions as an initial contact and authentication. The STA may also use a cached PMKSA to authenticate. A STA already associated with the ESS can initiate FILS authentication to multiple other APs while associated.

The MLME-DELETEKEYS.request primitive destroys the temporal key(s) established for the security association so that they cannot be used to protect subsequent IEEE 802.11 traffic. An SME uses this primitive when it deletes a PTKSA, GTKSA, or IGTKSA.

12.6.3 RSNA policy selection in an infrastructure BSS

Change 12.6.3 as follows:

An RSNA-enabled AP shall use Table 12-2 and the values of the Management Frame Protection Capable (MFPC) and Management Frame Protection Required (MFPR) bits advertised in the RSNEs to determine if it may associate with a non-AP STA. An RSNA-enabled non-AP STA shall use Table 12-2 and the values of the Management Frame Protection Capable and Management Frame Protection Required bits advertised in the RSNEs to determine if it may associate with an AP. Management frame protection is enabled when dot11RSNAProtectedManagementFramesActivated is set to 1. Management frame protection is negotiated when an AP and non-AP STA set the Management Frame Protection Capable field to 1 in their respective RSNEs in the (re)association procedure, and both parties confirm the Management Frame Protection Capable bit set to 1 in the 4-way handshake, FT 4-way handshake, ~~or the~~ FT fast BSS transition protocol, or the (Re)Association Request and (Re)Association Response frames of FILS authentication.

12.6.10 RSNA authentication in an infrastructure BSS

12.6.10.1 General

Change 12.6.10.1 as follows:

When establishing an RSNA in a non-FT environment or during an FT initial mobility domain association, a STA shall use IEEE 802.11 SAE authentication, FILS authentication, or Open System authentication prior to (re)association.

SAE authentication is initiated when a STA's MLME-SCAN.confirm primitive finds another AP within the current ESS that advertises support for SAE in its RSNE.

FILS authentication is initiated when a STA's MLME-SCAN.confirm primitive finds an AP that advertises support for FILS authentication in its RSNE.

IEEE 802.1X authentication is initiated by any one of the following mechanisms:

— If a STA negotiates to use IEEE 802.1X authentication during (re)association, the STA's management entity may respond to the MLME-ASSOCIATE.confirm (or indication) or MLME-REASSOCIATE.confirm primitive by requesting the Supplicant (or Authenticator) to initiate IEEE 802.1X authentication. Thus, in this case, authentication is driven by the STA's decision to associate and the AP's decision to accept the association.

— If a STA's MLME-SCAN.confirm primitive finds another AP within the current ESS, a STA may signal its Supplicant to use IEEE Std 802.1X-2010 to preauthenticate with that AP.

NOTE—A roaming STA's IEEE 802.1X Supplicant can initiate preauthentication by sending an EAPOL-Start message via its old AP, through the DS, to a new AP.

— If a STA receives an IEEE 802.1X message, it delivers this to its Supplicant or Authenticator, which may initiate a new IEEE 802.1X authentication.

12.6.10.3 Cached PMKSAs and RSNA key management

Change 12.6.10.3 as follows:

In a non-FT environment, a STA might cache PMKSAs it establishes as a result of previous authentication. The PMKSA cannot be changed while cached. The PMK~~SA~~ in the PMKSA is used with the 4-way handshake or FILS authentication to establish fresh PTKs.

If a STA in an infrastructure BSS has determined it has a valid PMKSA with an AP to which it is about to (re)associate, it performs Open System authentication to the AP, and then it includes the PMKID for the PMKSA in the RSNE in the (Re)Association Request. When the PMKSA was not created using preauthentication, the AKM indicated in the RSNE by the STA in the (Re)Association Request shall be identical to the AKM used to establish the cached PMKSA in the first place.

Upon receipt of a (Re)Association Request frame with one or more PMKIDs, an AP checks whether its Authenticator has cached a PMKSA for the PMKIDs, whether the AKM in the cached PMKSA matches the AKM in the (Re)Association Request, and if so, it shall assert possession of that PMKSA by beginning the 4-way handshake after association has completed. If the Authenticator does not have a PMKSA for the PMKIDs in the (Re)Association Request, its behavior depends on how the PMKSA was established. If SAE authentication was used to establish the PMKSA, then the AP shall reject (re)association by sending a (Re)Association Response frame with status code STATUS_INVALID_PMKID. Note that this allows the non-AP STA to fall back to full SAE authentication to establish another PMKSA. If IEEE 802.1X authentication was used to establish the PMKSA, the AP begins a full IEEE 802.1X authentication after association has completed.

Upon receipt of a FILS Authentication frame with one or more PMKIDs, an AP checks whether its Authenticator has cached a PMKSA for the PMKIDs, whether the AKM in the cached PMKSA matches the AKM in the FILS Authentication frame, and whether the PMK is still valid; and if so, it shall assert possession of that PMK by including the PMKID in the FILS Authentication frame sent in response. If the Authenticator does not have a PMK for the PMKIDs in the FILS Authentication frame, the AP may either reply with EAP-Finish/Re-auth to continue FILS Shared Key authentication option if the non-AP STA included sufficient information for that, or the AP rejects the authentication.

If both sides assert possession of a cached PMKSA, but the 4-way handshake or FILS authentication fails, both sides may delete the cached PMKSA for the selected PMKID.

If the lifetime of a cached PMKSA expires, the STA shall delete the expired PMKSA.

If a STA roams to an AP with which it is preauthenticating and the STA does not have a PMKSA for that AP, the STA needs to initiate a full IEEE 802.1X EAP authentication.

12.6.14 RSNA key management in an infrastructure BSS

Change 12.6.14 as follows:

When the IEEE 802.1X authentication completes successfully, this standard assumes that the STA's IEEE 802.1X Supplicant and the IEEE 802.1X AS share a secret, called a PMK. In a non-FT environment, the AS transfers the PMK, within the MSK, to the AP, using a technique that is outside the scope of this standard; the derivation of the PMK from the MSK is EAP-method-specific. With the PMK in place, the AP initiates a key confirmation handshake with the STA. The key confirmation handshake sets the IEEE 802.1X state variable port Valid (as described in IEEE Std 802.1X-2010) to true.

When SAE authentication completes, both STAs share a PMK. With this PMK in place, the AP initiates the key confirmation handshake with the STA.

Key confirmation is part of the FILS authentication exchange and no further handshakes are needed to satisfy key management requirements.

When FILS authentication is not used, t~~T~~he key confirmation handshake is implemented by the 4-way handshake. The purposes of the 4-way handshake are as follows:

12.6.21 RSNA rekeying

Change 12.6.21 as follows:

When a PTKSA is deleted, a non-AP and non-PCP STA may reassociate with the same AP or PCP and/or establish a new RSNA with the AP or PCP. If the non-AP and non-PCP STA has cached one or more PMKSAs, it may skip the PMKSA establishment and proceed with the creation of a new PTKSA by using 4-way handshake or FILS authentication.

12.7 Keys and key distribution

12.7.1 Key hierarchy

12.7.1.2 PRF

Insert the following in 12.7.1.2 after the paragraph regarding "AKM is 00-0F-AC:13":

When the negotiated AKM is 00-0F-AC:14 or 00-0F-AC:16, the KDF specified in 12.7.1.7.2 shall be used instead of the PRF construction defined here. In this case, A is used as the KDF label and B as the KDF Context, and the PRF functions are defined as follows:

PRF-384(K, A, B) = KDF-SHA-256-384(K, A, B)
PRF-512(K, A, B) = KDF-SHA-256-512(K, A, B)
PRF-640(K, A, B) = KDF-SHA-256-640(K, A, B)
PRF-768(K, A, B) = KDF-SHA-256-768(K, A, B)
PRF-896(K, A, B) = KDF-SHA-256-896(K, A, B)
PRF-1024(K, A, B) = KDF-SHA-256-1024(K, A, B)

When the negotiated AKM is 00-0F-AC:15 or 00-0F-AC:17, the KDF specified in 12.7.1.7.2 shall be used instead of the PRF construction defined here. In this case, A is used as the KDF label and B as the KDF Context, and the PRF functions are defined as follows:

PRF-640(K, A, B) = KDF-SHA-384-640(K, A, B)
PRF-768(K, A, B) = KDF-SHA-384-768(K, A, B)
PRF-1024(K, A, B) = KDF-SHA-384-1024(K, A, B)
PRF-1152(K, A, B) = KDF-SHA-384-1152(K, A, B)
PRF-1408(K, A, B) = KDF-SHA-384-1408(K, A, B)
PRF-1536(K, A, B) = KDF-SHA-384-1536(K, A, B)

12.7.1.3 Pairwise key hierarchy

Change 12.7.1.3 as follows:

Except when preauthentication or FILS authentication is used, the pairwise key hierarchy utilizes PRF-384, PRF-512, or PRF-704 to derive session-specific keys from a PMK, as depicted in Figure 12-28. When using AKM suite selector 00-0F-AC:12, the length of the PMK, PMK_bits, shall be 384 bits. With all other AKM suite selectors, the length of the PMK, PMK_bits, shall be 256 bits. The pairwise key hierarchy takes a PMK and generates a PTK. The PTK is partitioned into KCK, KEK, and a temporal key, which is used by the MAC to protect individually addressed communication between the Authenticator's and Supplicant's respective STAs. PTKs are used between a single Supplicant and a single Authenticator.

Change Note 4 of 12.7.1.3 as follows:

NOTE 4—The Authenticator and Supplicant normally derive a PTK only once per association. A Supplicant or an Authenticator use the 4-way handshake or FILS authentication to derive a new PTK. Both the Authenticator and Supplicant create a new nonce value for each 4-way handshake or FILS authentication instance.

12.7.1.7 FT key hierarchy

Change 12.7.1.7 as follows:

12.7.1.7.1 Overview

This subclause describes the FT key hierarchy and its supporting architecture. The FT key hierarchy is designed to allow a STA to make fast BSS transitions between APs without the need to perform an SAE or IEEE 802.1X authentication at every AP within the mobility domain.

The FT key hierarchy can be used with SAE, IEEE 802.1X authentication, ~~or~~ PSK authentication, or FILS authentication.

A three-level key hierarchy provides key separation between the key holders. The FT key hierarchy for the Authenticator is shown in Figure 12-31. An identical key hierarchy exists for the Supplicant, and identical functions are performed by the corresponding S0KH and S1KH.

The FT key hierarchy shown in Figure 12-31 consists of three levels whose keys are derived using the key derivation function (KDF) described in 12.7.1.7.2 as follows:

a) PMK-R0 – the first-level key of the FT key hierarchy. This key is derived as a function of the master session key (MSK) or PSK. It is stored by the PMK-R0 key holders, R0KH and S0KH.

b) PMK-R1 – the second-level key of the FT key hierarchy. This key is mutually derived by the S0KH and R0KH.

c) PTK – the third-level key of the FT key hierarchy that defines the IEEE 802.11 and IEEE 802.1X protection keys. The PTK is mutually derived by the PMK-R1 key holders, R1KH and S1KH.

As shown in Figure 12-31, the R0KH computes the PMK-R0 from the key obtained from SAE authentication (for the purposes of FT this key is identified as the Master PMK, or MPMK), from the PSK, ~~or~~ from the MSK resulting (per IETF RFC 3748 [B41]) from a successful IEEE 802.1X authentication between the AS and the Supplicant, or from the PMK (see 12.12.2.5.2) resulting from a successful FILS authentication. Upon a successful authentication, the R0KH shall delete any prior PMK-R0 security association for this mobility domain pertaining to this S0KH. The R0KH shall also delete all PMK-R1 security associations derived from that prior PMK-R0 security association. The PMK-R1s are generated by the R0KH and are assumed to be delivered from the R0KH to the R1KHs within the same mobility domain. The PMK-R1s are used for PTK generation. Upon receiving a new PMK-R1 for an S0KH, an R1KH deletes the prior PMK-R1 security association and PTKSAs derived from the prior PMK-R1.

Replace Figure 12-31 with the following:

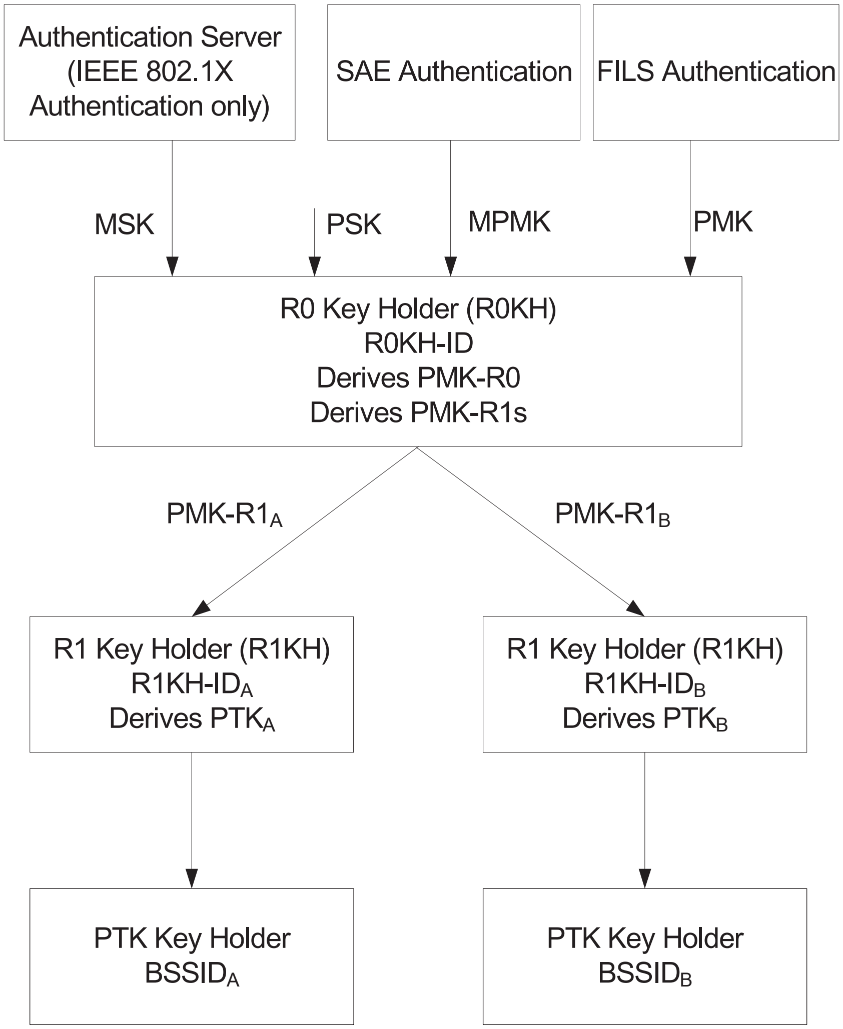


Figure 12-31—FT key hierarchy at an Authenticator

Change 12.7.1.7.3 as follows:

12.7.1.7.3 PMK-R0

The first-level key in the FT key hierarchy, PMK-R0, is derived using the KDF defined in 12.7.1.7.2. The PMK-R0 is the first level keying material used to derive the next level keys (PMK-R1s):

R0-Key-Data = KDF-Hash-Length(XXKey, "FT-R0", SSIDlength || SSID || MDID || R0KHlength || R0KH-ID || S0KH-ID)
PMK-R0 = L(R0-Key-Data, 0, Q)
PMK-R0Name-Salt = L(R0-Key-Data, Q, 128)
Length = Q + 128

where

- KDF-Hash-Length is the KDF as defined in 12.7.1.7.2 (Key derivation function (KDF)) using the hash algorithm identified by the AKM suite selector (see Table 9-133).
- If the AKM negotiated is 00-0F-AC:3, then Q shall be 256, and XXKey shall be the second 256 bits of the MSK (which is derived from the IEEE 802.1X authentication), i.e., XXKey = L(MSK, 256, 256). If the AKM negotiated is 00-0F-AC:4, then Q shall be 256, and XXKey shall be the PSK. If the AKM negotiated is 00-0F-AC:9, then Q shall be 256, and XXKey shall be the MPMK generated as the result of SAE authentication. If the AKM negotiated is 00-0F-AC:13, then Q shall be 384, and XXKey shall be the first 384 bits of the MSK (which is derived from the IEEE 802.1X authentication), i.e., XXKey = L(MSK, 0, 384). If the AKM negotiated is 00-0F-AC:16, then Q shall be 256, and XXKey shall be the FILS-FT described in 12.12.2.5.3. If the AKM negotiated is 00-0F-AC:17, then Q shall be 384, and XXKey shall be the FILS-FT described in 12.12.2.5.3.

Change 12.7.1.7.5 as follows:

12.7.1.7.5 PTK

The third-level key in the FT key hierarchy is the PTK. When FILS authentication is used to establish the FT key hierarchy, PTK for the initial mobility domain association is derived as part of the FILS authentication as defined in 12.12.2.5.3. Otherwise, ~~T~~this key is mutually derived by the S1KH and the R1KH used by the target AP, with the key length being a function of the negotiated cipher suite as defined by Table 12-4 in 12.7.2.

12.7.2 EAPOL-Key frames

Change the parts shown of 12.7.2 as follows:

b) **Key Information.** This field is 2 octets and specifies characteristics of the key. See Figure 12-33.

1) Key MIC (bit 8) is set to 1 if not using an AEAD cipher and a MIC is in this EAPOL-Key frame, and is set to 0 otherwise~~ if this message contains no MIC~~.

Encrypted Key Data (bit 12) is set to 1 if the Key Data field is encrypted and is set to 0 if the Key Data field is not encrypted. This subfield shall be set to 1, and the Key Data field shall be encrypted, if any key material (e.g., GTK or SMK) is included in the frame. When using an AEAD cipher and having PTK, this subfield is set to 1.

h) **Key MIC.** When AKM negotiated is not 00-0F-AC:14, 00-0F-AC:15, 00-0F-AC:16, or 00-0F-AC:17, ~~T~~the EAPOL Key MIC is a MIC of the EAPOL-Key frames, from and including the EAPOL protocol version field to and including the Key Data field, calculated with the Key MIC field set to 0. If the Encrypted Key Data subfield (of the Key Information field) is 1, the Key Data field is encrypted prior to computing the MIC. When using an AEAD cipher, the EAPOL Key MIC is not present. The length of this field depends on the negotiated AKM as defined in 12.7.3.

j) **Key Data**.

If the Encrypted Key Data subfield (of the Key Information field) is 1, the entire Key Data field shall be encrypted. If the Key Data field uses the NIST AES key wrap, then the Key Data field shall be padded before encrypting if the key data length is less than 16 octets or if it is not a multiple of 8. The padding consists of appending a single octet 0xdd followed by zero or more 0x00 octets. When processing a received EAPOL-Key frame, the receiver shall ignore this trailing padding. If the Key Data field uses an AEAD cipher, then the Key Data field shall not be padded and the AAD for the encipherment operation shall be the data of the EAPOL-Key frame from the EAPOL protocol version field (inclusive) to the Key Data field (exclusive). Key Data fields that are encrypted, but do not contain the GroupKey or SMK KDE, shall be accepted.

12.7.4 EAPOL-Key frame notation

Insert the following new rows at the end of Table 12-8.

Table 12-8—Integrity and key-wrap algorithms

AKM	Integrity algorithm	KCK_bits	Size of MIC	Key-wrap algorithm	KEK_bits
00-0F-AC:14	AES-SIV-256	0	0	AES-SIV-256	256
00-0F-AC:15	AES-SIV-512	0	0	AES-SIV-512	512
00-0F-AC:16	AES-SIV-256	0	0	AES-SIV-256	256
00-0F-AC:17	AES-SIV-512	0	0	AES-SIV-512	512

Change the parts shown of 12.7.4 as follows:

12.7.4 EAPOL-Key frame notation

The following notation is used throughout the remainder of 12.7 and 13.4 to represent EAPOL-Key frames:

EAPOL-Key(S, M, A, I, K, SM, KeyRSC, ANonce/SNonce, MIC, DataKDs)

where

S means the initial key exchange is complete. This is the Secure bit of the Key Information field.

M means the MIC is available in message. This should be set in all messages except Message 1 of a 4-way handshake. This is the Key MIC bit of the Key Information field. When the negotiated AKM is 00-0F-AC:14, 00-0F-AC:15, 00-0F-AC:16, or 00-0F-AC:17, this Key MIC bit is set to 0 regardless of the M parameter value.

MIC is the integrity check, which is generated using the KCK. This is the Key MIC field. When the negotiated AKM is 00-0F-AC:14, 00-0F-AC:15, 00-0F-AC:16, or 00-0F-AC:17, the Key MIC field is not included regardless of the MIC parameter value.

12.7.6 4-way handshake

12.7.6.3 4-way handshake message 2

Change the parts shown of 12.7.6.3 as follows:

Key Information:

Key Descriptor Version = 1 (ARC4 encryption with HMAC-MD5) or 2 (NIST AES key wrap with HMAC-SHA-1-128) or 3 (NIST AES key wrap with AES-128-CMAC), in all other cases 0 – same as message 1

Key Type = 1 (Pairwise) – same as message 1

SMK Message = 0 – same as message 1

Install = 0

Key Ack = 0

Key MIC = ~~1~~0 when using an AEAD cipher or 1 otherwise

Key MIC = Not present when using an AEAD cipher; otherwise, MIC(KCK, EAPOL) – MIC computed over the body of this EAPOL-Key frame with the Key MIC field first initialized to 0

On reception of message 2, the Authenticator checks that the key replay counter corresponds to the outstanding message 1. If not, it silently discards the message. Otherwise, the Authenticator:

a) Derives PTK.

b) Verifies the message 2 MIC or AEAD decryption operation result.

  1) If the calculated MIC does not match the MIC that the Supplicant included in the EAPOL-Key frame or the AEAD decryption operation returns failure, the Authenticator silently discards message 2.

  2) If the MIC or AEAD decryption is valid and this message 2 is part of a fast BSS transition initial mobility domain association or an association started through the FT protocol, the Authenticator checks that all fields of the RSNE other than the PMKID field bitwise matches the fields from the (Re)Association Request frame and that the FTE and MDE are the same as those provided in the AP’s (Re)Association Response frame.If the MIC or AEAD decryption is valid and this message 2 is not part of an association started through the FT protocol, the Authenticator checks that the RSNE bitwise matches that from the (Re)Association Request frame.

    i) If these are not exactly the same, the Authenticator uses MLME-DEAUTHENTICATE.request primitive to terminate the association.

    ii) If they do match bitwise, the Authenticator constructs message 3.

12.7.6.4 4-way handshake message 3

Change the parts shown of 12.7.6.4 as follows:

Key Information:

Key Descriptor Version = 1 (ARC4 encryption with HMAC-MD5) or 2 (NIST AES key wrap with HMAC-SHA-1-128) or 3 (NIST AES key wrap with AES-128-CMAC), in all other cases 0 – same as message 1

Key Type = 1 (Pairwise) – same as message 1

SMK Message = 0 – same as message 1

Install = 0/1 – For PTK generation, 0 only if the AP does not support key mapping keys, or if the STA has the No Pairwise bit (in the RSN Capabilities field) equal to 1 and only the group key is used. For STK generation, this bit is set to 1.

Key Ack = 1

Key MIC = ~~1~~0 when using an AEAD cipher or 1 otherwise

Key MIC = Not present when using AEAD cipher; otherwise, MIC(KCK, EAPOL) or MIC(SKCK, EAPOL) – MIC computed over the body of this EAPOL-Key frame with the Key MIC field first initialized to 0

On reception of message 3, the Supplicant silently discards the message if the Key Replay Counter field value has already been used or if the ANonce value in message 3 differs from the ANonce value in message 1. The Supplicant also:

a) Verifies the RSNE. If this message 3 is part of a fast BSS transition initial mobility domain association or an association started through the FT protocol, the Supplicant verifies that the PMKR1name in the PMKID field of the RSNE is identical to the value it sent in message 2 and verifies that all other fields of the RSNE are identical to the fields in the RSNE present in the Beacon or Probe Response frames and verifies that the FTD and MDE are the same as in the (Re)Association Response frame. Otherwise, the Supplicant verifies that the RSNE isidentical to that the STA received in the Beacon or Probe Response frame. If any of these verification steps indicates a mismatch, the STA shall disassociate or deauthenticate. If a second RSNE is provided in the message, the Supplicant uses the pairwise cipher suite specified in the second RSNE or deauthenticates.

b) Verifies the message 3 MIC or AEAD decryption operation result. If the calculated MIC does not match the MIC that the Authenticator included in the EAPOL-Key frame or AEAD decryption operation returns failure, the Supplicant silently discards message 3.

12.7.6.5 4-way handshake message 4

Change the parts shown of 12.7.6.5 as follows:

Key Information:

Key Descriptor Version = 1 (ARC4 encryption with HMAC-MD5) or 2 (NIST AES key wrap with HMAC-SHA-1-128) or 3 (NIST AES key wrap with AES-128-CMAC), in all other cases 0 – same as message 1

Key Type = 1 (Pairwise) – same as message 1

SMK Message = 0 – same as message 1

Install = 0

Key Ack = 0 – this is the last message

Key MIC = ~~1~~0 when using AEAD cipher or 1 otherwise

Key MIC = Not present when using an AEAD cipher: otherwise, MIC(KCK, EAPOL) or MIC(SKCK, EAPOL) – MIC computed over the body of this EAPOL-Key frame with the Key MIC field first initialized to 0

On reception of message 4, the Authenticator verifies that the Key Replay Counter field value is one that it used on this 4-way handshake; if it is not, it silently discards the message. Otherwise:

a) The Authenticator checks the MIC or AEAD decryption operation result. If the calculated MIC does not match the MIC that the Supplicant included in the EAPOL-Key frame or AEAD decryption operation returns failure, the Authenticator silently discards message 4.

b) If the MIC is valid, the Authenticator uses the MLME-SETKEYS.request primitive to configure the IEEE 802.11 MAC to send and, if the receive key has not yet been installed, to receive protected, individually addressed MPDUs using for the new PTK.

12.7.7.2 Group key handshake message 1

Change the parts shown of 12.7.7.2 as follows:

Key Information:

Key Descriptor Version = 1 (ARC4 encryption with HMAC-MD5) or 2 (NIST AES key wrap with HMAC-SHA-1-128) or 3 (NIST AES key wrap with AES-128-CMAC), in all other cases 0

Key Type = 0 (Group/SMK)

SMK Message = 0

Install = 0

Key Ack = 1

Key MIC = ~~1~~0 when using AEAD cipher or 1 otherwise

Key MIC = Not present when using AEAD cipher; otherwise, MIC(KCK, EAPOL)

On reception of message 1, the Supplicant:

a) Verifies that the Key Replay Counter field value has not yet been seen before, i.e., its value is strictly larger than that in any other EAPOL-Key frame received thus far during this session.

b) Verifies that the MIC is valid, i.e., it uses the KCK that is part of the PTK to verify that there is no data integrity error, or that the AEAD decryption steps succeed.

12.7.7.3 Group key handshake message 2

Change the parts shown of 12.7.7.3 as follows:

Key Information:
Key Descriptor Version = 1 (ARC4 encryption with HMAC-MD5) or 2 (NIST AES key wrap with HMAC-SHA-1-128) or 3 (NIST AES key wrap with AES-128-CMAC), in all other cases 0 – same as message 1
Key Type = 0 (Group/SMK) – same as message 1
Install = 0
Key Ack = 0
Key MIC = ~~1~~0 when using AEAD cipher or 1 otherwise

Key MIC = Not present when using AEAD cipher; otherwise, MIC(KCK, EAPOL)

On reception of message 2, the Authenticator:

a) Verifies that the Key Replay Counter field value matches one it has used in the group key handshake.
b) Verifies that the MIC is valid, i.e., it uses the KCK that is part of the PTK to verify that there is no data integrity error, or that the AEAD decryption steps succeed.

12.7.8 PeerKey handshake

12.7.8.2.2 SMK handshake message 1

Change the parts shown of 12.7.8.2.2 as follows:

Key Information:
Key Descriptor Version = 1 (ARC4 encryption with HMAC-MD5) or 2 (NIST AES key wrap with HMAC-SHA-1-128) or 3 (NIST AES key wrap with AES-128-CMAC), in all other cases 0
Key Type = 0 (Group/SMK)
SMK Message = 1 (SMK)
Install = 0
Key Ack = 0
Key MIC = ~~1~~0 when using AEAD cipher or 1 otherwise

Key MIC = Not present when using an AEAD cipher; otherwise, MIC (initiating STA's KCK, EAPOL)

On receipt of message 1, the AP checks that the key replay counter corresponds to message 1. If not, it silently discards the message. Otherwise:

a) The AP verifies the Message 1 MIC using the STA_I PTKSA if an AEAD cipher is not used. If the calculated MIC does not match the MIC that the STA_I included in the EAPOL-Key frame or AEAD decryption operation returns failure, the AP silently discards Message 1.

b) If the MIC is correct or the AEAD decryption steps succeed, the AP checks if the STA_P is reachable. If it is not reachable, the AP shall send an error EAPOL-Key frame to STA_I per 12.7.8.5.2. After sending the message, AP silently discards Message 1.

12.7.8.2.3 SMK handshake message 2

Change the parts shown of 12.7.8.2.3 as follows:

Key Information:

Key Descriptor Version = 1 (ARC4 encryption with HMAC-MD5) or 2 (NIST AES key wrap with HMAC-SHA-1-128) or 3 (NIST AES key wrap with AES-128-CMAC), in all other cases 0

Key Type = 0 (Group/SMK)

SMK Message = 1 (SMK)

Install = 0

Key Ack = 1

Key MIC = ~~1~~0 when using AEAD cipher or 1 otherwise

Key MIC = Not present when using an AEAD cipher; otherwise, MIC (KCK of the STA_P, EAPOL)

The AP sends message 2 to the STA_P. On receipt of message 2, the STA_P checks that the key replay counter corresponds to message 2. If not, it silently discards the message. Otherwise,

a) The STA_P verifies the message 2 MIC using the STA_P PTKSA if an AEAD cipher is not used. If the calculated MIC does not match the MIC that the AP included in the EAPOL-Key, the STA_P silently discards message 2.

b) If the MIC is correct or the AEAD decryption steps succeed, the STA_P checks if it supports at least one cipher suites proposed by the STA_I. If it does not, the STA_P shall send an error EAPOL-Key frame to STA_I through the AP per 12.7.8.5.4. After sending the error message, the STA_P silently discards message 2.

12.7.8.2.4 SMK handshake message 3

Change the parts shown of 12.7.8.2.4 as follows:

Key Information:

Key Descriptor Version = 1 (ARC4 encryption with HMAC-MD5) or 2 (NIST AES key wrap with HMAC-SHA-1-128) or 3 (NIST AES key wrap with AES-128-CMAC), in all other cases 0

Key Type = 0 (Group/SMK)

SMK Message = 1 (SMK)

Install = 0

Key Ack = 0

Key MIC = ~~1~~0 when using AEAD cipher or 1 otherwise

Key MIC = Not present when using an AEAD cipher; otherwise, MIC (KCK of STA_I, EAPOL)

The STA_P sends message 3 to the AP. On receipt of message 3, the AP checks that the key replay counter corresponds to message 3. If not, it silently discards the message. Otherwise,

a) The AP verifies the message ~~1~~3 MIC using the STA_I PTKSA if an AEAD cipher is not used. If the calculated MIC does not match the MIC that the STA_P included in the EAPOL-Key frame or AEAD decryption operation returns failure, the AP silently discards message ~~1~~3.

b) If MIC is correct or the AEAD decryption steps succeed, the AP checks if the STA_I is reachable. If it is not reachable, the AP shall send an error EAPOL-Key frame to the STA_P per 12.7.8.5.2. After sending the message, the AP silently discards message 3.

12.7.8.2.5 SMK handshake message 4

Change the parts shown of 12.7.8.2.5 as follows:

Key Information:

Key Descriptor Version = 1 (ARC4 encryption with HMAC-MD5) or 2 (NIST AES key wrap with HMAC-SHA-1-128) or 3 (NIST AES key wrap with AES-128-CMAC), in all other cases 0

Key Type = 0 (Group/SMK)

SMK Message = 1 (SMK)

Install = 1

Key Ack = 0

Key MIC = ~~1~~0 when using AEAD cipher or 1 otherwise

Key MIC = Not present when using AEAD cipher; otherwise, MIC (KCK of the STA_~~I~~P, EAPOL)

The AP sends message 4 to the STA_P. On receipt of message 4, the STA_P checks that the key replay counter corresponds to message 4. If it does not, STA_P silently discards the message. Otherwise,

a) The STA_P verifies the message 4 MIC using STA_P PTKSA if and AEAD cipher is not used. If the calculated MIC does not match the MIC that the AP included in the EAPOL-Key frame or AEAD decryption operation returns failure, the STA_P silently discards message 4.

b) If the MIC is correct or the AEAD decryption steps succeed, STA_P identifies the PeerKey session using the PNonce sent as part of the Key Nonce field of message 4. If STA_P has an existing PeerKey state for this session, i.e., STA_P has received message 2 and this message is a follow-up to that. If STA_P has an existing PeerKey state for this session, STA_P silently discards message 4.

12.7.8.2.6 SMK handshake message 5

Change the parts shown of 12.7.8.2.6 as follows:

Key Information:
Key Descriptor Version = 1 (ARC4 encryption with HMAC-MD5) or 2 (NIST AES key wrap with HMAC-SHA-1-128) or 3 (NIST AES key wrap with AES-128-CMAC), in all other cases 0
Key Type = 0 (Group/SMK)
SMK Message = 1 (SMK)
Install = 0
Key Ack = 0
Key MIC = ~~1~~0 when using AEAD cipher or 1 otherwise

Key MIC = Not present when using AEAD cipher; otherwise, MIC (KCK of the STA_I, EAPOL)

The AP sends message 5 to the STA_I. On receipt of message 5, the STA_I checks that the key replay counter corresponds to message 5. If it does not, the STA_I silently discards the message. Otherwise,

a) STA_I verifies the message ~~4~~5 MIC using STA I~~P~~ PTKSA if an AEAD cipher is not used. If the calculated MIC does not match the MIC that the AP included in the EAPOL-Key frame or AEAD decryption operation returns failure, the STA_I silently discards message 5.
b) If the MIC is correct or the AEAD decryption steps succeed, the STA_I identifies the PeerKey session using the INonce sent as part of the Key Nonce field of message 5. If STA_I has an existing PeerKey state for this session, i.e., STA_I has initiated this message exchange using message 1 and this message is a follow-up to that. If STA_I has an existing PeerKey state for this session, STA_I shall silently discard message 5.

12.7.8.5 Error reporting

12.7.8.5.1 General

Change the parts shown of 12.7.8.5.1 as follows:

Key Information:
Key Descriptor Version = 1 (ARC4 encryption with HMAC-MD5) or 2 (NIST AES key wrap with HMAC-SHA-1-128) or 3 (NIST AES key wrap with AES-128-CMAC), in all other cases 0
Key Type = 0 (Group/SMK)
SMK Message = 1 (SMK)
Install = 0
Key Ack = 0
Key MIC = ~~1~~0 when using AEAD cipher or 1 otherwise

Key MIC = Not present when using AEAD cipher; otherwise, MIC computed over the body of this EAPOL-Key frame

12.7.9 TDLS PeerKey (TPK) security protocol

12.7.9.7 Supplicant PeerKey state machine variables

Change the following list item in 12.7.9.7:

- *MICVerified* **–** This variable is set to true if the MIC on the received EAPOL-Key frame is verified and is correct or if an AEAD cipher is used and the AEAD decryption steps succeed. Any EAPOL-Key frames with an invalid MIC are dropped and ignored.

12.7.10 RSNA Supplicant key management state machine

12.7.10.3 Supplicant state machine variables

Change the following list item in 12.7.10.3:

- *MICVerified* **–** The Supplicant sets this variable to true if the MIC on the received EAPOL-Key frame verifies as correct or if the AEAD cipher is used and the AEAD decryption steps succeed. The Supplicant silently discards any EAPOL-Key frame received with an invalid MIC.

12.7.11 RSNA Authenticator key management state machine

12.7.11.3 Authenticator state machine variables

Change the following list item in 12.7.11.3:

- *MICVerified* – This variable is set to true if the MIC on the received EAPOL-Key frame is verified and is correct or if AEAD cipher is used and AEAD decryption steps succeed. Any EAPOL-Key frames with an invalid MIC are dropped and ignored.

12.7.11.4 Authenticator state machine procedures

Change the following list item in 12.7.11.4:

- **MIC**(x) – Computes a MIC over the plaintext data. When an AEAD cipher is used, returns an empty string.

Insert new subclause 12.12 as follows:

12.12 Authentication for FILS

12.12.1 General

FILS authentication is an RSNA authentication protocol. The FILS authentication protocol authenticates STAs to each other, using either a shared key or a public key. When FILS Shared Key authentication is used, the authentication exchange can optionally be performed with PFS. When FILS Public Key authentication is used, PFS is always used. When the FILS authentication protocol is performed with PFS, the STA and AP derive ephemeral public and private keys with respect to a particular set of domain parameters that define a finite cyclic group and then exchange public keys. The result of the FILS authentication includes a PTKSA.

The security of FILS authentication depends on the following assumptions:

— When FILS Shared Key authentication is used, each STA shares either a valid rRK as defined in IETF RFC 6696 with a trusted third party (TTP) that is capable of being used with EAP-RP, or a PMK cached from a previous authenticated connection.

— When FILS Public Key authentication is used, each STA has a means to trust the public key of the other STA.

— When PFS is used, a finite cyclic group is negotiated where solving the discrete logarithm problem is computationally infeasible.

— When PFS is used, both the STA and AP have in common at least one finite cyclic group from the dot11RSNAConfigDLCGroupTable.

12.12.2 FILS authentication protocol

12.12.2.1 General

The STA and AP perform key establishment using Authentication frames and perform key confirmation using (Re)Association Request and (Re)Association Response frames.

After exchanging Authentication frames, the STA and AP derive a shared and secret key that will be used to derive a set of secret keys (as defined in 12.12.2.5.2) that are authenticated after exchanging (Re)Association Request and (Re)Association Response frames.

When a shared key is used for FILS authentication, and if the STA shares a valid rRK with the TTP, then EAP-RP as defined in IETF RFC 5295 and IETF RFC 6696 shall be used.

12.12.2.2 Discovery of a FILS capable AP

An AP indicates that it is capable of performing FILS authentication by indicating support for a FILS AKM in an RSN element and including it, and the FILS Indication element, in Beacon and Probe Response frames.

An AP may indicate that it is capable of performing FILS authentication by indicating support for a FILS AKM in the FD RSN subfield in a FILS Discovery frame.

An AP indicates support for FILS Shared Key authentication without PFS by setting the FILS Shared Key authentication without PFS supported bit to 1 in the FILS Information field of the FILS Indication element. An AP indicates support for FILS Shared Key authentication with PFS by setting the FILS Shared Key authentication with PFS Supported bit to 1 in the FILS Information field of the FILS Indication element. An AP may advertise between zero and seven realms using the Realm Identifier subfield(s) of the FILS Indication element that is part of Beacon, Probe Response, and FILS Discovery frames. If the STA believes it shares a valid rRK as defined in IETF RFC 6696 with the AP through, e.g., a hashed domain name that matches an AP-advertised realm, a HESSID, or other ANQP information, the STA may begin FILS Shared Key authentication with the AP using EAP-RP. Domain name hashing is specified in 11.47.4. If a STA discovers a FILS-capable AP and the STA believes it shares a PMKSA with the AP, it may begin the FILS authentication protocol with the AP using PMKSA caching.

An AP indicates support for FILS Public Key authentication by setting the FILS Public Key authentication Supported bit to 1 in the FILS Information field of the FILS Indication element. An AP may advertise up to seven public key indicators in the FILS Indication element that is part of Beacon, Probe Response, and FILS Discovery frames. If the STA discovers that it trusts the issuer of an AP’s X.509v3 certificate, or that it trusts its uncertified public key identified by matching its hash, the STA may begin the FILS authentication protocol to the AP and perform mutual authentication using trusted public keys.

12.12.2.3 Key establishment with FILS Shared Key authentication

12.12.2.3.1 Overview

This subclause defines the procedure for establishing a shared key between a FILS capable STA and AP using FILS Shared Key authentication that uses shared symmetric keys between the STA and the Authentication Server.

A STA may initiate FILS Shared Key authentication either with a FILS capable AP that is connected to a TTP Authentication Server that shares a valid key, called an rRK, as defined in IETF RFC 6696 with the STA, or with a FILS capable AP with whom it shares a cached PMKSA. If neither of these cases applies, a full EAP exchange may be performed via IEEE 802.1X authentication to establish rRK as defined in IETF RFC 6696 or another form of FILS authentication may be used to establish a shared PMKSA. When performing a full EAP exchange using RSNA to establish rRK, the Authentication algorithm number 0 (Open System) is used.

EAP-RP signaling as defined in IETF RFC 5295 and IETF RFC 6696 is used to validate the mutual possession of rRK between the STA and the Authentication Server. EAP-RP signaling is encapsulated using a FILS Wrapped Data element in the Authentication frame. The AP unwraps the encapsulated EAP-RP packet received from the STA in the FILS Wrapped Data element and forwards the EAP-RP packet to the Authentication Server using a transport that is out of scope of this specification. When the AP receives an EAP-RP packet from the Authentication Server, the AP forwards the packet to the STA by encapsulating the EAP-RP packet in the FILS Wrapped Data element of the Authentication frame.

The message sequence is depicted in Figure 12-54.

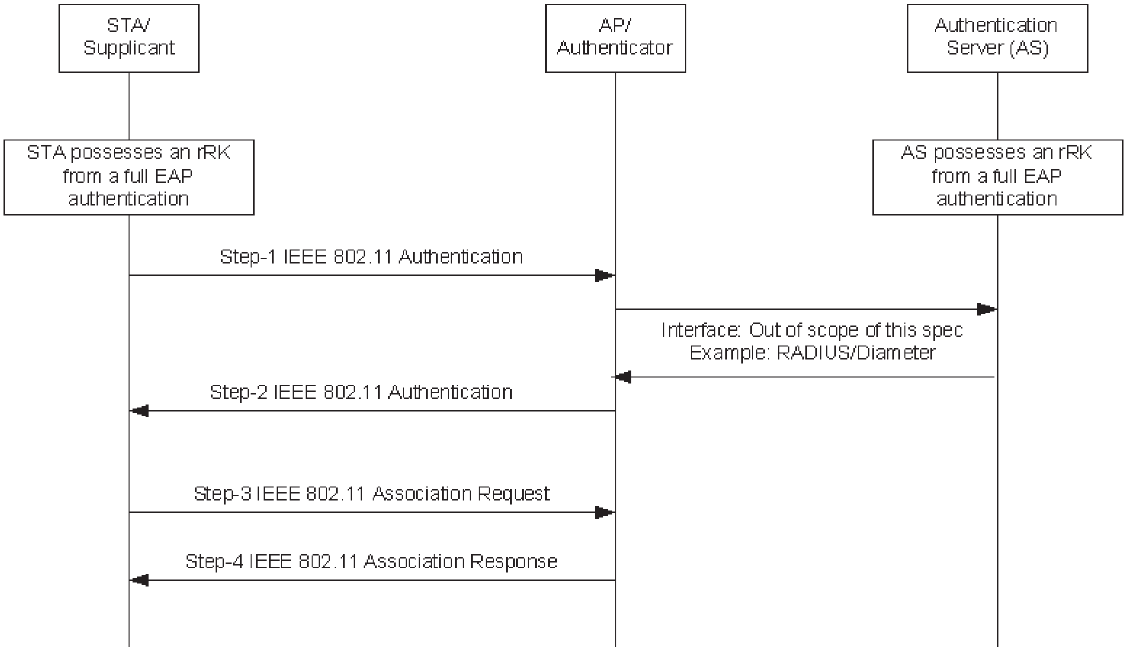

STA/
Supplicant
AP/
Authenticator
Authentication
Server (AS)
STA possesses an rRK
from a full EAP
authentication
AS possesses an rRK
from a full EAP
authentication
Step-1 IEEE 802.11 Authentication
Interface: Out of scope of this spec
Example: RADIUS/Diameter
Step-2 IEEE 802.11 Authentication
Step-3 IEEE 802.11 Association Request
Step-4 IEEE 802.11 Association Response

Figure 12-54—FILS Shared Key authentication

The following subclauses are organized per each step as shown in Figure 12-54.

12.12.2.3.2 Non-AP STA construction of Authentication frame

If the STA chooses to initiate FILS Shared Key authentication, it shall first choose a random 16-octet nonce and then determine whether to attempt PMKSA caching. If PMKSA caching is attempted, it shall generate a list of PMKSA identifiers. If the STA attempts to initiate EAP-RP, it shall construct an EAP-Initiate/Re-auth packet per IETF RFC 6696, with the following clarifications:

- Regarding EAP-RP Flags:
  - The B flag shall be set to 0, indicating that this is not an EAP-RP bootstrap message.
  - The L flag shall be set to 1, indicating that the TTP with whom the STA shares the rRK is to provide the lifetimes of rRK and rMSK in the EAP-Finish/Re-auth Packet.
  - EAP Identifier is set to 0.
- The Cryptosuite field shall not be set to 1.

If PFS is desired, the STA selects a finite cyclic group from the dot11RSNAConfigDLGGroupTable, generates an ephemeral private key, and performs the group's scalar-op (see 11.3.4.1) with its random ephemeral private key and the generator from the selected finite cyclic group to compute an ephemeral public key.

The STA then constructs an Authentication frame with the Authentication algorithm number set to 4 (FILS Shared Key authentication without PFS) or 5 (FILS Shared Key authentication with PFS) (see 9.4.1.1) depending on whether PFS is used, and the Authentication transaction sequence number set to 1. The random nonce shall be encoded in the FILS Nonce element (see 9.4.2.190). If a list of PMKSA identifiers was generated, it shall be used to construct the PMKID List field in RSNE. The random FILS Session shall be encoded in the FILS Session element (see 9.4.2.180). The EAP-Initiate/Re-auth packet, if generated, shall be copied into the FILS Wrapped Data field (see 9.4.2.188). If PFS is desired, the chosen finite cyclic group shall be encoded in the Finite Cyclic Group field (see 9.4.1.43) and the ephemeral public key shall be encoded in the FFE field (see 9.4.1.41) according to the element to octet-string conversion in 12.4.7.2.4.

The STA transmits the Authentication frame to the AP.

12.12.2.3.3 AP processing of Authentication frame

Upon reception of the Authentication frame with the Authentication algorithm number equal to 4 or 5, the AP shall perform the following procedure:

a) If Authentication frame includes a Finite Cyclic Group field, then the AP shall first determine whether the indicated finite cyclic group in the received FILS Authentication frame is supported.

b) If the indicated finite cyclic group in the received FILS Authentication frame is not supported, the AP shall respond with an Authentication frame with the Authentication algorithm number set to 5 (FILS Shared Key authentication with PFS) (see 9.4.1.1) and the Status Code field set to 77 (Authentication is rejected because the offered finite cyclic group is not supported) and shall terminate the exchange.

c) If PFS is being used, the STA’s public key shall be converted from an octet string to an element according to the conversion in 12.4.7.2.5. Then the AP shall validate the STA’s public key in a group-specific fashion as described in 5.6.2.3 of NIST SP800-56A R2. If validation fails, the AP shall terminate the exchange.

d) The AP shall check whether PMKSA caching is being attempted by the presence of the PMKID List field in RSNE.

   1) If the PMKID List field is present in RSNE, the AP checks whether any PMKSA identifier offered in the PMKID List matches an identifier for a cached PMKSA. If so, the AP selects a PMKID that matches and continues the FILS Shared Key authentication protocol using the PMK from the identified PMKSA.

   2) If a PMKID List field is not present in RSNE or if no PMKSA identifier offered in the PMKID list matches any identifier for a cached PMKSA, the AP checks whether an EAP-Initiate/Re-auth packet was included. If not, the AP shall respond with an Authentication frame with the Authentication algorithm number set to 4 or 5 depending on whether PFS is used and the Status Code field set to 53 (invalid PMKID) and shall terminate the exchange.

   3) If an EAP-Initiate/Re-auth packet is included and PMKSA caching is not used, the AP shall extract the EAP-Initiate/Re-auth data from the FILS Wrapped Data field (see 9.4.2.188) and shall forward it to the Authentication Server. When applicable, the AP communicates with the Authentication Server using the same protocols it uses when authenticating with EAP. Suitable protocols include, but are not limited to, remote authentication dial-in user service RADIUS (as specified in IETF RFC 2865) and Diameter (as specified in IETF RFC 6942).

If PFS is being used, the AP shall also generate an ephemeral private key and perform the group’s scalar-op (see 12.4.4.1) to produce its own ephemeral public key. The AP may delay the generation of its ephemeral public/private key pair until after receiving a response from the Authentication Server, if applicable. The Authentication Server processes the EAP-Initiate/Re-auth packet as specified in IETF RFC 6696 and returns an EAP-Finish/Re-auth packet to the AP. In the case of successful authentication by the Authentication Server, the Authentication Server returns the associated EAP-RP rMSK with the EAP-Finish/Re-auth packet. If the Authentication Server responds with a failure indication, then the AP shall produce an Authentication frame with the Authentication Algorithm Number field set to 4 (FILS Shared Key authentication without PFS) or 5 (FILS Shared Key authentication with PFS) (see 9.4.1.1), and the Status Code field set to 15 (Authentication rejected because of challenge failure). In the case of successful authentication by the Authentication Server, the Authentication Server returns the associated EAP-RP rMSK with the EAP-Finish/Re-auth packet and processing terminates.

The AP proceeds by constructing an Authentication frame.

12.12.2.3.4 AP construction of Authentication frame

If PMKSA caching is not used and the AP is not connected to, or does not recognize the Authentication Server identified by the STA using the realm in the key Name-NAI field of the EAP-Initiate/Re-auth packet, then the AP shall send Authentication frame with Status Code field set to 113, “Authentication rejected due to unknown Authentication Server” to the non-AP STA.

Otherwise, the AP shall generate its own nonce and construct an Authentication frame for the STA.The AP shall copy the FILS Session element from the Authentication frame sent by the non-AP STA to this response Authentication frame. If PMKSA caching is not used, this frame shall contain the FILS wrapped data that encapsulates EAP-Finish/Re-auth packet received from the Authentication Server. In addition, if PFS is used, the FFE field of the Authentication frame sent by the AP contains the AP’s ephemeral public key. In this frame, the AP shall set the Authentication algorithm number to 4 or 5 depending on whether PFS is used, and the Authentication sequence number to 2. If PMKSA caching is used, the AP indicates the selected PMKID in the PMKID List.

If PFS is being used for the exchange, the AP shall perform the group’s scalar-op (see 12.4.4.1) with the STA’s ephemeral public key and its own ephemeral private key to produce an ephemeral Diffie-Hellman shared secret, DHss.

The AP transmits the Authentication frame to the STA. Upon transmission of the FILS Authentication frame, the AP proceeds to key establishment per 12.12.2.5.

12.12.2.3.5 Non-AP STA processing of Authentication frame

The STA processes the received Authentication frame as follows:

a) The STA shall abandon FILS authentication if any of the following conditions occur:
   1) The received Authentication frame does not include the Authentication Algorithm Number equal to 4 (FILS Shared Key authentication without PFS) or 5 (FILS Shared Key authentication with PFS) (see 9.4.1.1).
   2) PMKSA caching was attempted and the received Authentication frame includes a PMKID that does not match a PMKID in the Authentication frame sent by the STA.
   3) The received Authentication frame does not include either a PMKID or an EAP-Finish/Re-auth packet.
   4) The received Authentication frame does not include the FILS Session element.
   5) The received FILS Session value does not match the one in the Authentication frame sent by the STA.
b) If the received Authentication frame includes the Status Code field equal to 15 (Authentication rejected because of challenge failure) or 53 (invalid PMKID), then the STA shall abandon the FILS authentication.
c) The STA verifies that the AP transmitted PFS parameters are consistent with the STA’s previous transmissions (indicated by whether or not the STA transmitted an ephemeral public key):
   1) If the STA transmitted an ephemeral public key, and the received Authentication frame does not include an ephemeral public key, then the STA shall abandon the FILS authentication.
   2) If the STA did not transmit an ephemeral public key, and the received Authentication frame includes an ephemeral public key, then the STA shall abandon the FILS authentication.
d) If applicable, the STA processes the EAP-Finish/Re-auth packet as per IETF RFC 6696:

1) If the 'R' flag = 0, indicating success, then the STA shall derive rMSK.
2) If the 'R' flag = 1, indicating failure, then the STA shall abandon the FILS authentication.

e) If PFS is being used for the exchange, the AP's public key shall be converted from an octet string to an element according to the conversion in 12.4.7.2.5. Then the STA shall validate the AP's public key in a group-specific fashion as described in 5.6.2.3 of NIST SP 800-56A R2. If validation fails, the STA shall terminate the FILS authentication protocol. Otherwise, the STA shall perform the group's scalar-op (see 12.4.4.1) with the AP's ephemeral public key and its own ephemeral private key to produce an ephemeral Diffie-Hellman shared secret, DHss.
f) The STA shall perform key derivation per 12.12.2.5 and key confirmation per 12.12.2.6.

If the STA was attempting EAP-RP Authentication and did not successfully receive an Authentication frame within the time of dot11AuthenticationResponseTimeout, then the STA should perform retransmission procedure as defined in IETF RFC 6696. If the retransmission procedure fails, then the STA shall abandon the FILS authentication and should perform full EAP authentication via IEEE 802.1X authentication.

If the STA was attempting PMKSA caching and did not receive an Authentication frame from the AP, the STA may attempt to use an alternate authentication method.

Upon successful processing of the Authentication frame, the STA proceeds with key establishment per 12.12.2.4.

12.12.2.4 Key establishment with FILS Public Key authentication

12.12.2.4.1 General

This subclause defines the procedure for establishing a shared key between a FILS capable STA and AP using FILS Public Key authentication.

12.12.2.4.2 Prior to exchange

FILS Public Key authentication performs key establishment with a Diffie-Hellman exchange. Prior to beginning the exchange, the non-AP STA performs the following:

a) Selects a finite cyclic group from the dot11RSNConfigDLCGroup table to perform the Diffie-Hellman exchange.
b) Generates a random nonce, generates an ephemeral private key, and uses the selected group's scalar-op (see 12.4.4.1) with its private key to generate its ephemeral public key.
c) Constructs an Authentication frame (see 9.3.3.12) as follows:
    1) The Authentication algorithm number is set to 6 (FILS Public Key authentication) (see 9.4.1.1) and the Authentication transaction sequence number is set to 1.
    2) The random nonce is encoded in the FILS Nonce element (see 9.4.2.190).
    3) The chosen finite cyclic group is encoded in the Finite Cyclic Group field (see 9.4.1.43).
    4) The STA's public key is encoded into the FFE field (see 9.4.1.41) according to the element to octet-string conversion in 12.4.7.2.4.
    5) The random FILS Session is encoded in the FILS Session element (see 9.4.2.180).

The STA then transmits the Authentication frame to the AP.

12.12.2.4.3 Processing after receipt

Upon receipt, the AP processes the STA's Authentication frame as follows:

a) If the finite cyclic group indicated by the Finite Cyclic Group field is not acceptable, the AP shall respond with an Authentication frame with the status code of 77 ("Authentication is rejected because the offered finite cyclic group is not supported") and terminate the FILS authentication protocol.

b) If the finite cyclic group is acceptable, the AP verifies the validity of the STA's public key:

   1) The public key is converted from an octet string to an element according to the conversion in 12.4.7.2.5.

   2) The public key, as a group element, is verified in a group-specific fashion as described in 5.6.2.3 of NIST SP 800-56A R2. If verification fails, the AP shall terminate the FILS authentication protocol.

c) The STA's nonce and validated public key are extracted from the Authentication frame.

12.12.2.4.4 Post processing

Next, the AP shall

a) Generate a nonce and a random ephemeral private key, and then uses the agreed-upon group's scalar-op (see 12.4.4.1) with its private key to generate its ephemeral public key.

b) Construct an Authentication frame (see 9.3.3.12) as follows:

   1) The Authentication algorithm number is set to 6 (FILS Public Key authentication) (see 9.4.1.1), and the Authentication transaction sequence number is set to 2.

   2) The random nonce is encoded in the FILS Nonce element (see 9.4.2.190).

   3) The finite cyclic group is encoded in the Finite Cyclic Group field (see 9.4.1.43).

   4) The AP's public key is encoded in the FFE field (see 9.4.1.41) according to the element to octet-string conversion in 12.4.7.2.4.

   5) The AP copies the FILS Session element from the Authentication frame received from the STA.

c) Transmit the Authentication frame to the STA.

d) Compute the Diffie-Hellman shared secret, DHss, based on the STA's ephemeral public key and its own private key with the chosen group's scalar-op.

e) Perform key derivation (see 12.12.2.5).

12.12.2.4.5 Upon receipt

Upon receipt, the STA

a) Verifies that the finite cyclic group in the AP's response is equal to the group selected by the STA and that the FILS Session element received from the AP is equal to the FILS Session selected by the STA. If these differ, the STA shall terminate the authentication exchange.

b) Verifies the validity of the AP's public key:

   1) The public key is converted from an octet string to an element according to the conversion in 12.4.7.2.5.

   2) The public key, as a group element, is verified in a group-specific fashion according to 5.6.2.3 of NIST SP 800-56A R2. If public key validation fails the STA shall terminate the authentication exchange.

c) Extracts the AP's nonce and verified public key from the Authentication frame.

d) Compute the Diffie-Hellman shared secret, DHss, based on the AP's ephemeral public key and its own private key with the chosen group's scalar-op to derive DHss.

e) Performs key derivation (see 12.12.2.5) and begins key confirmation (see 12.12.2.6).

12.12.2.5 Key establishment with FILS authentication

12.12.2.5.1 General

When not using PMKSA caching, a PMK is created according to 12.12.2.5.2. When using PMKSA caching, a new PMKSA is not created. Instead, the PMKSA used for PMKSA caching remains and continues to be identified by the appropriate PMKID. Regardless of whether PMKSA caching is used or not, a PTKSA shall be generated with each FILS authentication exchange.

PTKSA creation uses the KDF from 12.7.1.7.2 to derive the following keys from the PMK: an integrity check key (ICK), a key encryption key (KEK), and a temporal key (TK).

PTKSA key establishment shall immediately be followed by key confirmation per 12.12.2.6.

12.12.2.5.2 PMKSA key derivation with FILS authentication

The PMK is derived using the two nonces and the secret(s) from FILS Key establishment. A PMKID used to identify the PMKSA is generated using the hash algorithm from the negotiated AKM on input data specific to the FILS Key Establishment step. The length of the PMK shall be either 256 bits or 384 bits depending on the negotiated AKM, and the length of the PMKID shall be 128 bits. If FILS Shared Key authentication was used to generate input keying material, the PMK and PMKID are derived as follows:

$$\text{PMK} = \text{HMAC-Hash}(\text{SNonce} \parallel \text{ANonce}, \text{rMSK} [\parallel \text{DHss}])$$

$$\text{PMKID} = \text{Truncate-128}(\text{Hash}(\text{EAP-Initiate/Reauth}))$$

When FILS Public Key authentication is used to generate input keying material, the PMK and PMKID are derived as follows:

$$\text{PMK} = \text{HMAC-Hash}(\text{SNonce} \parallel \text{ANonce}, \text{DHss})$$

$$\text{PMKID} = \text{Truncate-128}(\text{Hash}(\text{gSTA} \parallel \text{gAP}))$$

where

SNonce is the STA nonce and ANonce is the AP nonce
rMSK is the shared secret from the EAP-RP exchange
DHss is the shared secret derived from the Diffie-Hellman exchange, when performed
Brackets indicate the inclusion of the shared secret when doing a Diffie-Hellman exchange; there is no shared secret to include otherwise
EAP-Initiate/Reauth is the EAP-RP packet sent by the STA during key establishment with FILS Shared Key authentication
gSTA is the STA’s Diffie-Hellman value and gAP is the AP’s Diffie-Hellman value
Hash is the AKM-specific hash function

Upon completion of PMK and PMKID generation the shared secret, DHss, and rMSK, if applicable, shall be irretrievably deleted.

12.12.2.5.3 PTKSA key derivation with FILS authentication

For PTKSA key generation, the inputs to the PRF are the PMK of the PMKSA, a constant label, and a concatenation of the STA's MAC address, the AP's BSSID, the STA's nonce, and the AP's nonce. When the AKM negotiated is 00-0F-AC:14 or 00-0F-AC:16, the length of KEK shall be 256 bits, and the length of the ICK shall be 256 bits. When the AKM negotiated is 00-0F-AC:15 or 00-0F-AC:17, the length of the KEK shall be 512 bits, and the length of ICK shall be 384 bits. When the AKM negotiated is 00-0F-AC:16, FILS-FT is 256 bits; when AKM negotiated if 00-0F-AC:17, FILS-FT is 384 bits; otherwise, FILS-FT is not derived. The total amount of bits extracted from the KDF shall therefore be 512+TK bits, 896+TK bits, or 1280+TK bits depending on the AKM negotiated, where TK_bits are determined from Table 12-4:

FILS-Key-Data = PRF-X(PMK, "FILS PTK Derivation", SPA || AA || SNonce || ANonce)
ICK = L(FILS-Key-Data, 0, ICK_bits)
KEK = L(FILS-Key-Data, ICK_bits, KEK_bits)
TK = L(FILS-Key-Data, ICK_bits + KEK_bits, TK_bits)

When doing FT initial mobility domain association using FILS authentication,

FILS-FT = L(FILS-Key-Data, ICK_bits + KEK_bits + TK_bits, FILS-FT_bits)

where

ICK_bits is the length of ICK in bits
KEK_bits is the length of KEK in bits
FILS-FT_bits is the length of FILS-FT in bits when doing FT initial mobility domain association using FILS authentication
X is 512+TK_bits, 768+TK bits, 896+TK bits, or 1280+TK bits from Table 12-4 depending on the AKM negotiated
PMK is the PMK from the PMKSA, either created from an initial FILS connection or from a cached PMKSA, when PMKSA caching is used
SPA is the STA's MAC address and the AA is the AP's BSSID
SNonce is the STA's nonce and ANonce is the AP's nonce

12.12.2.6 Key confirmation with FILS authentication

12.12.2.6.1 General

Key confirmation for FILS authentication is a (Re)Association Request frame followed by a (Re)Association Response frame. Components of the (Re)Association Request and (Re)Association Response frames shall be protected using KEK.

12.12.2.6.2 (Re)Association Request for FILS key confirmation

The STA constructs a (Re)Association Request frame for FILS authentication per 9.3.3.6 and 9.3.3.8. Hash functions are used to generate the FILS Key Confirmation element and the specific hash function depends on the AKM negotiated (9.4.2.25.3).

For FILS Shared Key authentication, the KeyAuth field of the FILS Key Confirmation element is constructed by using the HMAC mode of the negotiated hash function with a key of ICK on a concatenation of the STA's nonce, the AP's nonce, the STA's MAC address, the AP's BSSID, and conditionally the STA's public Diffie-Hellman value and the AP's public Diffie-Hellman value, in that order:

Key-Auth = HMAC-Hash(ICK, SNonce || ANonce || STA-MAC || AP-BSSID [|| gSTA || gAP]) where

Hash is the hash function specific to the negotiated AKM

SNonce is the STA's nonce, ANonce is the AP's nonce

STA-MAC is the MAC address of the STA and AP-BSSID is the BSSID of the AP

gSTA is the STA's Diffie-Hellman public value and gAP is the AP's Diffie-Hellman public value

Brackets indicate the inclusion of the Diffie-Hellman public values when doing PFS with FILS Shared Key authentication; there are no Diffie-Hellman public values to include otherwise

For FILS Public Key authentication, the KeyAuth field of the FILS Key Confirmation element is a digital signature using the STA's private key, of the negotiated hash function on a concatenation of the STA's public Diffie-Hellman value, the AP's public Diffie-Hellman value, the STA's nonce, the AP's nonce, the STA's MAC address, and the AP's BSSID, in that order:

$$\text{Key-Auth} = \text{Sig-}_{\text{STA}}(\text{gSTA} \,||\, \text{gAP} \,||\, \text{SNonce} \,||\, \text{ANonce} \,||\, \text{STA-MAC} \,||\, \text{AP-BSSID})$$

where

Sig-$_{\text{STA}}$( ) indicates a digital signature using the STA's private key, analog to the STA's trusted public key

The form of signature depends on the type of public key used by the STA (IETF RFC 3447 for RSA, FIPS 186-4 for DSA, and ISO/IEC 14888-3 for ECDSA). The data to be signed is first hashed and the hash algorithm used with the appropriate digital signature algorithm shall be specific to the negotiated AKM.

The (Re)Association Request frame shall be encrypted using the AEAD algorithm as defined in 12.12.2.7 with the KEK as the key. The AAD used with the AEAD algorithm for the Association Request frame consists of the following data passed as separate components in the following order:

- STA's MAC address
- AP's BSSID
- STA's nonce
- AP's nonce
- The contents of the (Re)Association Request frame from the Capability Information field (inclusive) to the FILS Session element (inclusive)

The plaintext passed to the AEAD algorithm is the data that would follow the FILS Session element in an unencrypted frame. The output of the AEAD algorithm becomes the data that follows the FILS Session element in the encrypted and authenticated (Re)Association Request frame. The output of the algorithm is as specified in IETF RFC 5116. The resulting (Re)Association Request frame shall be transmitted to the AP.

The AP decrypts and verifies the received (Re)Association Request frame with the AEAD algorithm as defined in 12.12.2.7 with the KEK as the key. The AAD is reconstructed as defined above and is passed, along with the ciphertext of the received frame, to the AEAD decryption operation.

If the output from the AEAD decryption operation returns a failure, the authentication exchange fails. If the output does not return failure, the output plaintext replaces the ciphertext as portion of the frame that follows the FILS Session element and processing of the received frame continues by checking the value of the FILS Key Confirmation element.

The AP verifies that the RSNE received in the (Re)Association Request frame has identical AKM suite and cipher suites and RSN capabilities as were included in the RSNE in the Authentication frame from the STA. If these fields differ, the authentication exchange fails.

For FILS Shared Key authentication, the AP constructs a verifier, Key-Auth', in an identical manner as the STA constructed its Key-Auth above.

The AP compares Key-Auth' with the KeyAuth field in the FILS Key Confirmation element of the received frame. If they differ, authentication fails.

For FILS Public Key authentication, the AP uses the STA's (certified) public key from the FILS Public Key element to verify that the signature contained in the KeyAuth field corresponds to the purported signature by the STA over the concatenation of the following:

- STA's public Diffie-Hellman value gSTA,
- AP's public Diffie-Hellman value gAP,
- STA's nonce SNonce, the AP's nonce ANonce,
- STA's MAC address STA-MAC,
- AP's BSSID AP-BSSID,

in that order, according to the signature scheme used. Furthermore, the AP checks all certificates in the certificate chain, both cryptographically and from a security policy perspective, according to the procedures for checking certificates and certificate chains in IETF RFC 5280. If any of these verifications fail, authentication fails.

If authentication is deemed a failure, ICK, KEK, TK, and the PTKSA shall be irretrievably deleted and the AP shall return an Authentication frame with a status code set to 112 (Authentication rejected due to FILS authentication failure). If PMKSA caching was not being employed for this failed authentication attempt, the PMKSA shall also be deleted. If PMKSA caching was being used, the cached PMKSA may not be deleted.

12.12.2.6.3 (Re)Association Response for FILS key confirmation

The AP constructs a (Re)Association Response frame for FILS authentication per 9.3.3.7 and 9.3.3.9. As with the (Re)Association Request frame, hash functions are used to generate the FILS Key Confirmation element and the specific hash function depends on the AKM negotiated (see 9.4.2.25.3).

The AP constructs a Key Delivery element indicating the current GTK and Key RSC, the current IGTK and IPN if management frame protection is enabled. The GTK is carried in a GTK KDE with Tx subfield equal to 0. The IGTK and IPN are carried in an IGTK KDE. The AP puts this element into the (Re)Association Response frame.

For FILS Shared Key authentication, the KeyAuth field of the FILS Key Confirmation element is constructed by using the HMAC mode of the negotiated hash function with a key of ICK on a concatenation of the AP's nonce, the STA's nonce, the AP's BSSID, the STA's MAC address, and conditionally the AP's public Diffie-Hellman value and the STA's public Diffie-Hellman value, in that order:

$$\text{Key-Auth} = \text{HMAC-Hash}(\text{ICK}, \text{ANonce} \,\|\, \text{SNonce} \,\|\, \text{AP-BSSID} \,\|\, \text{STA-MAC} \;[\; \|\, \text{gAP} \,\|\, \text{gSTA} \;])$$

where

Hash is the hash function specific to the negotiated AKM

ANonce is the AP's nonce and SNonce is the STA's nonce

AP-BSSID is the BSSID of the AP and STA-MAC is the MAC address of the STA

gAP is the AP's Diffie-Hellman public value and gSTA is the STA's Diffie-Hellman public value

Brackets indicate the inclusion of the Diffie-Hellman public values when doing PFS with FILS Shared Key authentication; there are no Diffie-Hellman public values to include otherwise

For FILS Public Key authentication, the KeyAuth field of the FILS Key Confirmation element is a digital signature using the AP's private key of the output from the negotiated hash function on a concatenation of the AP's public Diffie-Hellman value, the STA's public Diffie-Hellman value, the AP's nonce, the STA's nonce, AP's BSSID, and the STA's MAC address, in that order. The specific construction of the digital signature depends on the crypto-system of the public/private keypair:

$$\text{Key-Auth} = \text{Sig-}_{AP}(\text{gAP} \parallel \text{gSTA} \parallel \text{ANonce} \parallel \text{SNonce} \parallel \text{AP-BSSID} \parallel \text{STA-MAC}\ )$$

where

Sig-$_{AP}$() indicates a digital signature using the AP's private key analog to the AP's trusted public key

The form of signature depends on the type of public key used by the AP (IETF RFC 3447 for RSA, FIPS 186-4 for DSA, and ISO/IEC 14888-3 for ECDSA). The data to be signed is first hashed and the hash algorithm used with the appropriate digital signature algorithm shall be specific to the negotiated AKM.

The (Re)Association Response frame shall be encrypted using the AEAD algorithm as defined in 12.12.2.7 with the KEK as the key. The AAD used with the AEAD algorithm for the (Re)Association Response frame consists of the following data passed as separate components in the following order:

- AP's BSSID
- STA's MAC address
- AP's nonce
- STA's nonce
- The contents of the (Re)Association Response frame from the Capability Information field (inclusive) to the FILS Session element (inclusive)

The plaintext passed to the AEAD algorithm is the data that would follow the FILS Session element in an unencrypted frame. The output of the AEAD algorithm becomes the data that follows the FILS Session element in the encrypted and authenticated (Re)Association Request frame. The output of the algorithm is as specified in IETF RFC 5116. The resulting (Re)Association Response frame shall be transmitted to the STA.

The STA decrypts and verifies the received (Re)Association Response frame with the AEAD algorithm as defined in 12.12.2.5 with the KEK as the key. The AAD is reconstructed as defined in this subclause above and is passed with the ciphertext of the received frame to the AEAD decryption operation.

The STA compares FILS Session of the received frame with the FILS Session it selected to identify the FILS session. If they differ, authentication fails.

If the output from the AEAD decryption operation returns failure, the authentication exchange fails. If the output does not return failure, the output plaintext replaces the ciphertext as portion of the frame that follows the FILS Session element and processing of the received frame continues by checking the value of the FILS Key Confirmation element.

The STA verifies that the RSNE received in the (Re)Association Response frame has identical AKM suites and cipher suites and RSN capabilities as were included in the RSNE in the Beacon, Probe Response, and Authentication frames from the AP. If these fields differ, authentication fails.

For FILS Shared Key authentication, the STA constructs a verifier, Key-Auth', in an identical manner as the AP constructed its Key-Auth above.

The STA compares Key-Auth' with the KeyAuth field in the FILS Key Confirmation element of the received frame. If they differ, authentication fails.

For FILS Public Key authentication, the STA uses the AP's (certified) public key from the FILS Public Key element to verify that the signature contained in the KeyAuth field corresponds to the purported signature by the AP over the concatenation of the following:

- AP's public Diffie-Hellman value gAP,
- STA's public Diffie-Hellman value gSTA,
- AP's nonce ANonce,
- STA's nonce SNonce,
- AP's BSSID AP-BSSID,
- STA's MAC address STA-MAC,

in that order, according to the signature scheme used. Furthermore, the AP checks all certificates in the certificate chain, both cryptographically and from a security policy perspective, according to the procedures for checking certificates and certificate chains in IETF RFC 5280. If any of these verifications fail, authentication fails.

If authentication is deemed a failure, the ICK, KEK, PMK, and TK shall be irretrievably deleted and the STA shall abandon the exchange. Otherwise authentication succeeds and the STA and AP shall irretrievably delete the nonpersistent secret keying material that is created by executing the key establishment with FILS Shared Key authentication scheme (12.12.2.3) or the key establishment with FILS Public Key authentication scheme (12.12.2.4). The KEK and PMK shall be used for subsequent key management as specified in 12.6. If the lifetime of the rMSK is known, the STA and AP shall set the lifetime of the PMKSA to the lifetime of the rMSK. Otherwise, the STA and AP shall set the lifetime of the PMKSA to the value dot11RSNAConfig-PMKLifetime.

Upon successful completion of the FILS authentication procedure, the STA shall process the Key Delivery element in the (Re)Association Response frame. The STA installs the GTK and key RSC, and IGTK and IPN if management frame protection is enabled.

12.12.2.7 AEAD cipher mode for FILS

FILS authentication uses an AEAD cipher mode to protect (Re)Association Request/Response and EAPOL-Key frames. The AEAD cipher mode is determined by the specific FILS AKM negotiated.

AES-SIV-256 is used when the AKM negotiated is 00-0F-AC:14 or 00-0F-AC:16 and AES-SIV-512 is used when the AKM negotiated is 00-0F-AC:15 or 00-0F-AC:17.

13. Fast BSS transition

13.2 Key holders

13.2.2 Authenticator key holders

Change the paragraphs shown of 13.2.2 as follows:

The R0KH and R1KH are responsible for the derivation of keys in the FT key hierarchy. For fast BSS transition, the functions of the IEEE 802.1X Authenticator are distributed among the R0KH and R1KHs.

The R0KH interacts with the IEEE 802.1X Authenticator to receive the MSK resulting from an EAP authentication. The R1KH interacts with the IEEE 802.1X Authenticator to open the Controlled Port. Both the R0KH and R1KH interactions with the IEEE 802.1X Authenticator occur within the SME.

The R0KH derives the PMK-R0 for use in the mobility domain utilizing the MSK (when the AKM negotiated is 00-0F-AC:3), the PSK (when the AKM negotiated is 00-0F-AC:4), or the PMK (when the AKM negotiated is 00-0F-AC:9), or the IKM (when the AKM negotiated is 00-0F-AC:16 or 00-0F-AC:17). The R0KH shall be responsible for deriving a PMK-R1 for each R1KH within the mobility domain.

13.2.3 Supplicant key holders

Change the paragraphs shown of 13.2.3 as follows:

The S0KH and S1KH are responsible for the derivation of keys in the FT key hierarchy. The S0KH and S1KH are entities that are assumed to physically reside in the Supplicant.

The S0KH interacts with the IEEE 802.1X functional block (see Figure 4-19) in 4.9 to receive the MSK resulting from an EAP authentication or the IKM resulting from a FILS authentication. The S1KH interacts with the IEEE 802.1X entity to open the Controlled Port. Both the S0KH and S1KH interactions with the IEEE 802.1X entity occur within the SME of a STA.

The S0KH derives the PMK-R0 for use in the mobility domain utilizing the MSK (when the AKM negotiated is 00-0F-AC:3), the PSK (when the AKM negotiated is 00-0F-AC:4), ~~or~~ the PMK (when the AKM negotiated is 00-0F-AC:9), or the IKM (when the AKM negotiated is 00-0F-AC:16 or 00-0F-AC:17).

Insert new subclause 13.2.4 as follows:

13.2.4 FT initial mobility domain association over FILS in an RSN

A STA may perform FT initial mobility domain association with an AP using FILS Authentication as specified in this clause if it receives an MDE and FILS Indication element in the Beacon or Probe Response frame from the AP.

A STA indicates its support for the FT procedures by including the MDE in the Authentication frame and indicates its support of security in the RSNE. To establish FT key hierarchy, the AP responds by including the FTE, MDE, and RSNE in the Authentication frame. At the end of the sequence, the FT key hierarchy has been established. The message flow is shown in Figure 13-1a.

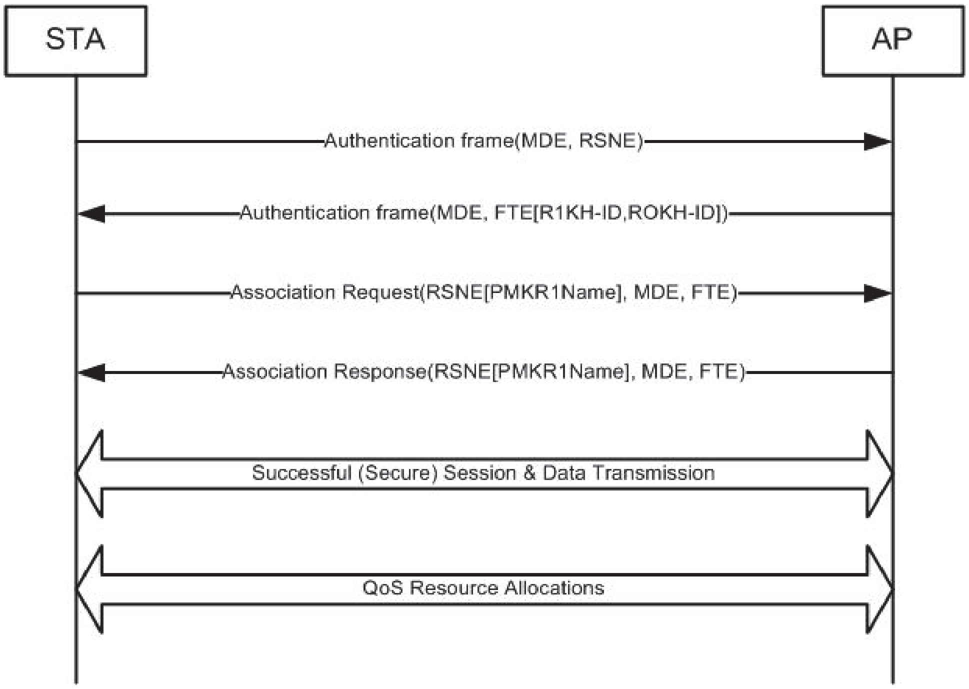

STA
AP
Authentication frame(MDE, RSNE)
Authentication frame(MDE, FTE[R1KH-ID,ROKH-ID])
Association Request(RSNE[PMKR1Name], MDE, FTE)
Association Response(RSNE[PMKR1Name], MDE, FTE)
Successful (Secure) Session & Data Transmission
QoS Resource Allocations

Figure 13-1a—FT initial mobility domain association using FILS authentication in an RSN

To establish the FT key hierarchy, the STA shall send an Authentication frame with the MDE and Authentication algorithm number 4, 5, or 6 to the AP. The contents of the MDE shall be the values advertised by the AP in its Beacon or Probe Response frames. Additionally, the STA includes its security capabilities in the RSNE.

If the contents of the MDE received by the AP do not match the contents advertised in the Beacon and Probe Response frames, the AP shall reject the Authentication frame with status code 54 (“Invalid MDE”). If an MDE is present in the Authentication frame and the contents of the RSNE do not indicate a negotiated AKM of Fast BSS Transition over FILS (suite type 00-0F-AC:16 or 00-0F-AC:17), the AP shall reject the Authentication frame with status code 43 (“Invalid AKMP”).

Upon successful completion of FILS authentication request processing, the R0KH on the AP uses the IKM (see 12.11.2.3.1) to establish key hierarchy. If a key hierarchy already exists for this STA belonging to the same mobility domain (i.e., having the same MDID), the R0KH shall delete the existing PMK-R0 security association and PMK-R1 security associations. It then calculates the PMK-R0, PMKR0Name, and PMK-R1 and makes the PMK-R1 available to the R1KH of the AP to which the STA is associated.

Next, the AP shall construct an Authentication frame. The Authentication frame shall contain an MDE, with contents as presented in Beacon and Probe Response frames. The FTE shall include the key holder identities of the AP, the R0KH-ID and R1KH-ID, set to the values of dot11FTR0KeyHolderID and dot11FTR1KeyHolderID, respectively. The FTE shall have a MIC element count of zero (i.e., no MIC present) and have ANonce, SNonce, and MIC fields set to 0.

The S1KH on STA provides the PMKR1Name in the PMKID field of the RSNE to be included in the (Re)Association Request frame. The PMKR1Name shall be as calculated by the S1KH according to the procedures of 12.7.1.7.4; all other fields of the RSNE shall be identical to the RSNE present in the

Authentication frame. The S1KH shall provide the FTE and MDE; the FTE and MDE shall be the same as those provided in the AP's Authentication frame.

Finally, the R1KH provides the PMKR1Name in the PMKID field of the RSNE to be included in (Re)Association Response frame. The PMKR1Name shall be as calculated by the R1KH according to the procedures of 12.7.1.7.4 and shall be the same as the PMKR1Name in the Association Request frame; all other fields of the RSNE shall be identical to the RSNE present in the Beacon or Probe Response frames. The R1KH shall also provide the FTE and the MDE. The FTE and MDE shall be the same as in the Authentication frame.

When FILS authentication is used to establish the FT key hierarchy, PTK for the initial mobility domain association is derived as part of the FILS authentication as defined in 12.11.2.3.2.

Annex B

(normative)

Protocol Implementation Conformance Statement (PICS) proforma

B.2.2 General abbreviations for Item and Support columns

Insert the following entry in the appropriate place in B.2.2:

FILS	Fast Initial Link Setup

B.4.3 IUT configuration

Insert the following new row at the end of the table in B.4.3:

* CFFILS	Fast Initial Link Setup (FILS)	11.47	O	Yes □ No □

Insert new subclause B.4.27 as follows:

B.4.27 FILS features

Item	Protocol capability	References	Status	Support
FILS1.1	FILS Discovery frame	11.47.2	(CFAP OR CFSTAof AP) AND CFFILS: M	Yes □ No □ N/A □
FILS1.2	FILS Discovery frame	9.6.8.36	(CFAP OR CFSTAof AP) AND CFFILS: M	Yes □ No □ N/A □
FILS2.1	Differentiated Initial Link Setup	11.47.5, 9.4.2.187	CFFILS: O	Yes □ No □ N/A □
* FILS3	ANQP Procedure for FILS	9.4.5.24	(CFAP OR CFSTAof AP) AND CFIW AND CFFILS: M	Yes □ No □ N/A □
FILS3.1	Query AP List	9.4.5.25	FILS3: M	Yes □ No □ N/A □
FILS3.2	FILS Realm Information	9.4.5.26	FILS3: M	Yes □ No □ N/A □
FILS3.3	Common ANQP Group	9.4.5.27 9.4.2.177	FILS3: M	Yes □ No □ N/A □

Item	Protocol capability	References	Status	Support
* FILS4	FILS authentication	12.12 9.4 9.4.2.179 9.4.2.180 9.4.2.183 9.4.2.190	(CFAP OR CFSTAof AP) AND CFFILS: M	Yes □ No □ N/A □
FILS4.1	FILS Shared Key authentication without PFS	9.4.2.188 12.12.2.3	FILS4:O.1	Yes □ No □ N/A □
FILS4.2	FILS Shared Key authentication with PFS	9.4.2.188 12.12.2.3	FILS4: O	Yes □ No □ N/A □
FILS4.3	FILS Public Key authentication	9.4.2.181 9.4.2.189 12.12.2.4	FILS4: O.1	Yes □ No □ N/A □
* FILS5	Higher Layer Setup During (re)association procedure	11.47.3	(CFAP OR CFSTAof AP) AND CFFILS: M	Yes □ No □ N/A □
FILS5.1	HLP Packet Encapsulation	11.47.3.2 9.4.2.184	FILS5: M	Yes □ No □ N/A □
FILS5.2	FILS IP Address Configuration	11.47.3.3 9.4.2.185	FILS5: O	Yes □ No □ N/A □
* FILS6	Scanning Enhancement	11.1.4	(CFAP OR CFSTAof AP) AND CFFILS: M	Yes □ No □ N/A □
FILS6.1	Probe Request Reduction	11.1.4.3.2	FILS6: M	Yes □ No □ N/A □
FILS6.2	Probe Response Reduction	11.1.4.3.4 9.4.2.178	FILS6: M	Yes □ No □ N/A □
FILS6.3	OmitReplicateProbeResponses	11.1.4.3.5	FILS6: M	Yes □ No □ N/A □
FILS6.3	AP Configuration Sequence Number	9.4.2.182	FILS6: O	Yes □ No □ N/A □
FILS6.4	Reduced Neighbor Report	11.44.8 9.4.2.171	FILS6: M	Yes □ No □ N/A □

Annex C

(normative)

ASN.1 encoding of the MAC and PHY MIB

C.3 MIB detail

In the major section of C.3, insert the following text before the entry of “`-- dot11STALCIConfigTable ::= { dot11smt 36 }`*” in the dot11smt attribute:*

```
    --  dot11FILSConfigTable::= { dot11smt 35 }
```

Insert the last entry of Dot11StationConfigEntry as follows:

```
Dot11StationConfigEntry ::= SEQUENCE
    {
        dot11FILSActivated                          TruthValue
    }
```

Change the text as follows:

```
dot11AuthenticationAlgorithm OBJECT-TYPE
    SYNTAX INTEGER {
        openSystem(1),
        sharedKey(2),
        fastBSSTransition(3),
        simultaneousAuthEquals(4),
        FILSSharedKeyWithoutPFS (5),
        FILSSharedKeyWithPFS (6),
        FILSPublicKey (7),
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "This is a control variable.
        It is written by an external management entity.
        Changes take effect as soon as practical in the implementation.

        This attribute is the authentication algorithm described by this entry in
        the table. The following values can be used here
        Value = 1: Open system
        Value = 2: Shared key
        Value = 3: Fast BSS transition (FT)
        Value = 4: Simultaneous authentication of equals (SAE)
        Value = 5: FILS Shared Key authentication without PFS
        Value = 6: FILS Shared Key authentication with PFS
        Value = 7: FILS Public Key authentication
        A given value shall not be used more than once."
    ::= { dot11AuthenticationAlgorithmsEntry 2 }
```

Insert the following new group at the end of "Groups - units of compliance - RSN" section:

```
dot11FILSComplianceGroup OBJECT-GROUP
    OBJECTS {
        dot11FILSActivated,
        dot11FILSFDFrameBeaconMinimumInterval,
        dot11FILSFDFrameBeaconMaximumInterval,
        dot11FILSBeaconResponseWindow,
        dot11FILSOmitReplicateProbeResponses,
        dot11DILSImplemented,
        dot11FILSProbeDelay,
        dot11HLPWaitTime,
        dot11CacheIdentifier
    }
    STATUS current
    DESCRIPTION
        "The FILS Compliance group defines those objects that provide fast initial
        link setup for IEEE Std 802.11."
    ::= { dot11Groups 91 }
```

Insert the following new statement after dot11BSSStatisticsGroup statement in "Compliance Statements" section:

```
    GROUP dot11FILSComplianceGroup
    DESCRIPTION
        "FILS Compliance Group"
```

Change the last line of OPTIONAL-GROUPS in "Compliance Statements" section as follows:

```
    -- dot11TVWSComplianceGroup,
    -- dot11FILSComplianceGroup }
```

Insert the following new element at the end of theDot11StationConfigTable element definition:

```
dot11FILSActivated OBJECT-TYPE
    SYNTAX TruthValue
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "This is a control variable. It is written by an external management
        entity.
        Changes take effect for the next MLME-START.request primitive or MLME-
        JOIN.request primitive.
        This attribute, when true, indicates that FILS is enabled."
    DEFVAL {false}
    ::= { dot11StationConfigEntry 160 }
```

After the definition of the dot11TVHTStationConfigTable, insert the dot11FILSConfigTable as defined next:

```
-- **********************************************************************
-- * dot11FILSConfigTable TABLE
-- **********************************************************************
dot11FILSConfigTable OBJECT-TYPE
   SYNTAX SEQUENCE OF Dot11FILSConfigEntry
   MAX-ACCESS not-accessible
   STATUS current
   DESCRIPTION
      "The table containing fast initial link setup configuration objects."
   ::= { dot11smt 35 }

dot11FILSConfigEntry OBJECT-TYPE
   SYNTAX Dot11FILSConfigEntry
   MAX-ACCESS not-accessible
   STATUS current
   DESCRIPTION
      "An entry in the dot11FILSConfigTable."
   INDEX { ifIndex }
   ::= { dot11FILSConfigTable 1 }

Dot11FILSConfigEntry ::=
   SEQUENCE {
      dot11FILSFDFrameBeaconMinimumInterval             Unsigned32,
      dot11FILSBeaconResponseWindow                     Unsigned32,
      dot11FILSFDFrameBeaconMaximumInterval             Unsigned32,
      dot11FILSOmitReplicateProbeResponses              TruthValue,
      dot11DILSImplemented                              TruthValue,
      dot11FILSProbeDelay                               Unsigned32,
      dot11HLPWaitTime                                  Unsigned32,
      dot11CacheIdentifier                              OCTET STRING
   }

dot11FILSFDFrameBeaconMinimumInterval OBJECT-TYPE
   SYNTAX Unsigned32(0..10000)
   UNITS "TUs"
   MAX-ACCESS read-write
   STATUS current
   DESCRIPTION
      "This is a control variable.
      It is written by an external management entity.
      The STA is allowed to transmit a FILS Discovery frame if a duration
      defined by this value has elapsed since the previous Beacon or FILS Dis-
      covery frame transmission and the next TBTT is later than a duration
      defined by this value."
   DEFVAL {20}
   ::= { dot11FILSConfigEntry 1 }

dot11FILSFDFrameBeaconMaximumInterval OBJECT-TYPE
   SYNTAX Unsigned32 (0..10000)
   UNITS "TUs"
   MAX-ACCESS read-write
   STATUS current
   DESCRIPTION
      "This is a control variable. It is written by an external management
      entity. Changes take effect as soon as practical in the implementation. If
      this value is not equal to 0, the STA queues for transmission a FILS Dis-
      covery frame or a Beacon frame if a duration defined by this value has
      elapsed since the previous Beacon or FILS Discovery frame transmission
      unless the next TBTT is within a duration defined by the value of dot11-
      FILSFDFrameBeaconMinimumInterval."
DEFVAL {0}
::= { dot11FILSConfigEntry 2 }
```

```
dot11FILSBeaconResponseWindow OBJECT-TYPE
   SYNTAX Unsigned32(0..1000000)
   UNITS "0.1 milliseconds"
   MAX-ACCESS read-write
   STATUS current
   DESCRIPTION
      "This is a control variable.
      It is written by an external management entity.
      Changes take effect as soon as practical in the implementation.
      If the duration from the reception of the Probe Request frame to the TBTT
      is less than the value, the STA does not transmit a Probe Response frame
      as response to the Probe Request frame."
   DEFVAL {50}
   ::= { dot11FILSConfigEntry 3 }

dot11FILSOmitReplicateProbeResponses OBJECT-TYPE
   SYNTAX  TruthValue
   MAX-ACCESS read-write
   STATUS current
   DESCRIPTION
      "This is a control variable.
      It is written by an external management entity. Changes take effect for
      the next Probe Response frame.
      This attribute, when true, indicates that the station may respond to one
      or more received Probe Request frames with a single Probe Response frame
      addressed to the broadcast address or alternatively, by not transmitting a
      Probe Response frame and instead letting the next Beacon frame be the
      response to the Probe Request frame(s)."
   DEFVAL { false }
   ::= { dot11FILSConfigEntry 4 }


dot11DILSImplemented OBJECT-TYPE
   SYNTAX TruthValue
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "This is a capability variable.
      Its value is determined by device capabilities.
      This attribute, when true, indicates that the station implementation is
      capable of supporting differentiated initial link setup category.
      The capability is disabled, otherwise."
   DEFVAL{false}
   ::= { dot11FILSConfigEntry 5 }

dot11FILSProbeDelay OBJECT-TYPE
   SYNTAX Unsigned32(0..100000)
   UNITS "microseconds"
   MAX-ACCESS read-write
   STATUS current
   DESCRIPTION
      "This is a control variable.
      It is written by an external management entity.
      Changes take effect as soon as practical in the implementation.
      If a FILS STA receives a suitable Probe Request, Probe Response, Beacon,
      FILS Discovery or Measurement Pilot frame within this duration of the
      start of active scanning on a given channel, it does not transmit a Probe
      Request frame."
   DEFVAL {5000}
   ::= { dot11FILSConfigEntry 6 }
```

```
dot11HLPWaitTime OBJECT-TYPE
   SYNTAX Unsigned32
   UNITS "TUs"
   MAX-ACCESS read-write
   STATUS current
   DESCRIPTION
      "This is a control variable.
      It is written by an external management entity.
      Changes take effect as soon as practical in the implementation.
      This value specifies a time that the FILS AP waits for incoming HLP pack-
      ets after receiving a (Re)Association Request frame."
   DEFVAL {30}
   ::= { dot11FILSConfigEntry 7 }

dot11CacheIdentifier OBJECT-TYPE
   SYNTAX OCTET STRING (SIZE (2))
   MAX-ACCESS read-write
   STATUS current
   DESCRIPTION
      "This is a control variable. It is written by an external management
      entity. Changes take effect as soon as practical in the implementation.
      This value specifies the Cache Identifier that the FILS AP advertises in
      FILS Indication elements."
   ::= { dot11FILSConfigEntry 8 }


-- **********************************************************************
-- * End of dot11FILSConfigTable TABLE
-- **********************************************************************
```

Insert the following new statement before the "End of 802.11 MIB":

```
-- **********************************************************************
-- * Compliance Statements - FILS
-- **********************************************************************
dot11FILSCompliance MODULE-COMPLIANCE
   STATUS current
   DESCRIPTION
      "The compliance statement for SNMPv2 entities that implement the IEEE
      802.11 MIB for FILS operation."
   MODULE -- this module
   MANDATORY-GROUPS { dot11FILSComplianceGroup }
   OPTIONAL-GROUPS { }
   ::= { dot11Compliances 17 }
```

Consensus

WE BUILD IT.

Connect with us on:

Facebook: https://www.facebook.com/ieeesa

Twitter: @ieeesa

LinkedIn: http://www.linkedin.com/groups/IEEESA-Official-IEEE-Standards-Association-1791118

IEEE-SA Standards Insight blog: http://standardsinsight.com

YouTube: IEEE-SA Channel

IEEE
standards.ieee.org
Phone: +1 732 981 0060   Fax: +1 732 562 1571