WiMedia Logical Link Control Protocol

Making High-Speed Wireless a Reality...

WLP Specification: Approved Draft 1.0

August 13, 2007

Notice

The WiMedia Alliance, Inc. (WiMedia) disclaims any and all warranties, whether expressed or implied, including (without limitation) any implied warranties of merchantability or fitness for a particular purpose. WiMedia reserves the right to make changes to the document without further notice.

This is a draft specification under development by the WLP Technical Working Group of WiMedia. As such, this is not a completed specification and has not been accepted. The Technical Working Group may modify this document as a result of comments received during ballot and its subsequent acceptance as a WiMedia Alliance Specification.

WiMedia Limited Copyright License Agreement

By receiving, installing, copying, reviewing or otherwise using the WiMedia Logical Link Control Protocol Specification (the "***Specification***"), you (the "***Specification Recipient***") agree to the terms and conditions of this WiMedia Limited Copyright License Agreement (the "***Agreement***") by and between the WiMedia Alliance, Inc. ("***WiMedia***") and Specification Recipient.

NO WIMEDIA PROMOTER, CONTRIBUTOR OR ADOPTER MEMBER SHALL BE BOUND TO THE TERMS OR CONDITIONS OF THIS AGREEMENT WHILE IT IS A PROMOTER, CONTRIBUTOR OR ADOPTER MEMBER. THIS AGREEMENT DOES BIND ALL WIMEDIA SUPPORTER MEMBERS AND NON-MEMBERS.

1. The Specification. "***Specification***" shall mean this WiMedia Logical Link Control Protocol Specification document. WiMedia reserves the right to change the Specification at any time without notice to Specification Recipient.

2. Limited Copyright Grant. Provided Specification Recipient complies with all terms and conditions of this Agreement, WiMedia grants Specification Recipient a non-exclusive, revocable, temporary, royalty-free, personal copyright license to copy, display and distribute the Specification solely for the purpose of reviewing such Specification. The scope of this limited license does not permit Specification Recipient or Specification Recipient's organization to create any products based on the Specification, in whole or in part, or to use the Specification in whole or in part for any commercial use.

3. Other Restrictions. WiMedia reserves all rights not expressly granted to Specification Recipient herein. Without limiting the generality of the foregoing, Specification Recipient shall not: (i) disclose the Specification outside the Specification Recipient's corporation or organization; (ii) disclose the Specification without prominently displaying the terms of this Agreement with the Specification and binding each recipient to the terms of this Agreement; (iii) modify or create derivative works based upon the Specification; (iv) commercially distribute products based on the Specification, in whole or in part; (v) sublicense, resell or otherwise transfer the Specification, in whole or in part; (vi) embed the Specification, in whole or in part, or any derivation thereof in any product; or (vii) rent, lease, lend, or use the Specification for commercial use.

4. Ownership of the Specification. All title and intellectual property in and to the Specification are owned by WiMedia and its licensor(s), if any.

5. Termination. This Agreement and any and all rights hereunder may be terminated by WiMedia upon notice for any reason or no reason at all. Upon termination of this Agreement, Specification Recipient shall immediately cease any and all use of the Specification and destroy all copies of the Specification within its control.

6. No Warranties. SPECIFICATION RECIPIENT ACKNOWLEDGES AND AGREES THAT THE SPECIFICATION IS PROVIDED "AS IS" AND WITH NO WARRANTIES WHATSOEVER, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING, BUT NOT LIMITED TO ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, TITLE, FITNESS OF ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF THE SPECIFICATION AND/OR THIS AGREEMENT. THE SPECIFICATION RECIPIENT'S USE OF THE SPECIFICATION IS SOLELY AT THE SPECIFICATION RECIPIENT'S OWN RISK.

7. Limitation of Liability. IN NO EVENT SHALL WIMEDIA OR ANY WIMEDIA MEMBER BE LIABLE OR OBLIGATED TO THE SPECIFICATION RECIPIENT OR ANY THIRD PARTY IN ANY MANNER FOR ANY DIRECT, SPECIAL, NON-COMPENSATORY, CONSEQUENTIAL, INDIRECT, INCIDENTAL, STATUTORY OR PUNITIVE DAMAGES OF ANY KIND, INCLUDING, WITHOUT LIMITATION, LOST PROFITS AND LOST REVENUE, REGARDLESS OF THE FORM OF ACTION, WHETHER IN CONTRACT, TORT, NEGLIGENCE,

STRICT PRODUCT LIABILITY, OR OTHERWISE, EVEN IF WIMEDIA OR ANY WIMEDIA MEMBER HAS BEEN INFORMED OF OR IS AWARE OF THE POSSIBILITY OF ANY SUCH DAMAGES IN ADVANCE.

THE LIMITATIONS SET FORTH ABOVE SHALL BE DEEMED TO APPLY TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW AND NOTWITHSTANDING THE FAILURE OF THE ESSENTIAL PURPOSE OF ANY LIMITED REMEDIES AVAILABLE TO THE SPECIFICATION RECIPIENT. THE SPECIFICATION RECIPIENT ACKNOWLEDGES AND AGREES THAT THE SPECIFICATION RECIPIENT HAS FULLY CONSIDERED THE FOREGOING ALLOCATION OF RISK AND FINDS IT REASONABLE, AND THAT THE FOREGOING LIMITATIONS ARE AN ESSENTIAL BASIS OF WIMEDIA AND THE WIMEDIA MEMBERS PERMITTING ACCESS TO THE SPECIFICATION. SPECIFICATION RECIPIENT FURTHER ACKNOWLEDGES AND AGREES THAT WIMEDIA AND THE WIMEDIA MEMBERS WOULD NOT HAVE PROVIDED THE SPECIFICATION RECIPIENT WITH ACCESS TO THE SPECIFICATION UNLESS THE SPECIFICATION RECIPIENT FULLY AGREED TO THE LIMITATIONS SET FORTH ABOVE. SPECIFICATION RECIPIENTS' SOLE AND EXCLUSIVE REMEDIES AND EXCLUSIVE LIABILITIES ARE SET FORTH IN THIS AGREEMENT.

8. Third Party Rights. Certain elements of the Specification may be subject to third party intellectual property rights, including without limitation, patent, trademark and copyright rights. WiMedia is not responsible and shall not be held responsible in any manner for identifying or failing to identify any or all such third party intellectual property rights.

9. Non-Applicability to Certain WiMedia Members. Notwithstanding anything to the contrary in this Agreement, no WiMedia promoter, contributor or adopter member shall be bound to the terms or conditions of this Agreement while it is a member of WiMedia. This Agreement does bind all WiMedia supporter members and non-members.

10. General. If any provision of this Agreement is found by a court of competent jurisdiction to be invalid or unenforceable, such invalidity or unenforceability shall not invalidate or render unenforceable any other part of this Agreement, but this Agreement shall be construed as not containing the particular provision or provisions held to be invalid or unenforceable. No delay or omission by either party to exercise any right occurring upon any noncompliance or default by the other party with respect to any of the terms of this Agreement shall impair any such right or power or be construed to be a waiver thereof. A waiver by either of the parties hereto of any of the covenants, conditions or agreements to be performed by the other shall not be construed to be a waiver of any succeeding breach thereof or of any covenant, condition or agreement herein contained. Nothing set forth in this Agreement shall be deemed or construed to render the parties as joint venturers, partners or employer and employee. This Agreement, together with any documents referenced herein, sets forth the entire, final and exclusive agreement between the parties as to the subject matter hereof and supersedes all prior and contemporaneous agreements, understandings, negotiations and discussions, whether oral or written, between the parties; provided, however, that a WiMedia promoter, contributor or adopter member shall not be bound to the terms of this Agreement while it is a promoter, contributor or adopter member of WiMedia. This Agreement may be modified only pursuant to a writing executed by authorized representatives of WiMedia and Specification Recipient. This Agreement, and all the rights and duties of the parties arising from or relating in any way to the subject matter of this Agreement or the transaction(s) contemplated by it, shall be governed by, construed and enforced in accordance with the laws of the State of California (excluding any conflict of laws provisions of the State of California that would refer to and apply the substantive laws of another jurisdiction). **SPECIFICATION RECIPIENT CONSENTS TO THE EXCLUSIVE PERSONAL JURISDICTION OF THE FEDERAL AND STATE COURTS AND VENUE LOCATED IN SAN FRANCISCO, CALIFORNIA.**

11. Trademarks. WiMedia is a registered trademark or service mark of the WiMedia Alliance, Inc. in the US and other countries. All other trademarks, registered trademarks, or service marks used in this document are the property of their respective owners and are hereby recognized. Specification Recipient shall not have any rights to reproduce WiMedia's trademarks or service marks except with WiMedia's prior written consent.

WLP TECHNICAL WORKING GROUP

Chair: Alan Berkema, Hewlett-Packard

Technical Editor: Ron Brown, Focus Enhancements

CONTRIBUTORS

Ankur Agiwal	Microsoft
Mohammad Alam	Microsoft
Simcha Aronson	Infineon Technologies
Geetha Arun	Synopsys
John Beattie	ITI Techmedia
Alan Berkema	Hewlett-Packard
Chuck Brabenac	Intel
Billy Brackenridge	Staccato Communications
Ron Brown	Focus Enhancements
Josh Cantrell	Focus Enhancements
Richard Chen	Philips
Joe Decuir	MCCI
Zhong Deng	Microsoft
Randy Erman	Stonestreet One
Mark Fidler	Hewlett-Packard
Kris Fleming	Intel
David Furodet	STMicroelectronics
Stephan Gehring	Tzero Technologies
Giriraj Goyal	Samsung Electronics
Marie Graham	ITI Techmedia
Peter Groset	Stonestreet One
Ghobad Heidari	Olympus
Manoj Hemrajani	Synopsys
Jin-Meng Ho	Texas Instruments
Preston Hunt	Intel
Sunil Jogi	Samsung Electronics
Peter Johansson	Staccato Communications
John Keys	Intel
Dongho Kim	Telecommunications Technology Association
Haim Kupershmidt	Wisair
Timothy Looney	Kodak
P.G. Madhavan	Microsoft
Janne Marin	Nokia
Sharad Mittal	Microsoft
Saleem Mohammad	Synopsys
Alaa Muqattash	Olympus
Arun Naniyat	Samsung Electronics
Kaisa Nyberg	Nokia
Jay O'Conor	Philips
Dave Patton	Hewlett-Packard
Alon Paycher	Texas Instruments
Vijay Kumar Peshkar	Wipro

Neeraj Poojary	Texas Instruments
Venkatesh Rajendran	Realtek Semiconductor
David Roberts	Microsoft
Sid Schrum	WiQuest Communications
Richard Solotke	Stonestreet One
J.P. Stewart	Microsoft
Fred Stivers	WiQuest Communications
Larry Taylor	Staccato Communications
Janne Tervonen	Nokia
Tim Thomas	Stonestreet One
Prashant Wason	Samsung Electronics
Patrick Worfolk	Tzero Technologies

TABLE OF CONTENTS

1. Scope

This specification defines a Logical Link Control layer networking protocol for the WiMedia radio platform (referred to as WLP) to model the behavior of an IEEE 802 [B1][1] environment, for example, IEEE 802.3 [B2]. This facilitates easy migration of applications compatible with an IEEE 802 environment to a WiMedia environment with few or no changes. For example, a TCP/IP protocol stack designed for an IEEE 802.3 environment will also work with a WiMedia environment, using this protocol. In addition to support for straightforward application migration, this protocol also preserves data structures to facilitate the design of bridges between a WiMedia network and other IEEE 802 or compatible wired or wireless networks.

[1] The numbers in brackets correspond to references listed in clause 2 or bibliography entries in Annex D.

2. References

This specification shall be used in conjunction with the following publications. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this specification are encouraged to investigate the possibility of applying the most recent editions of the standards listed below.

[B1] IEEE Std 802®-2001, IEEE Standard for Local and Metropolitan Area Networks: Overview and Architecture. 2001. New York: Institute of Electrical and Electronics Engineers, Inc.[2,3]

[B2] IEEE Std 802.3™-2005 IEEE Standard for Information technology— Telecommunications and information exchange between systems— Local and metropolitan area networks— Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications. 2005. New York: Institute of Electrical and Electronics Engineers, Inc.

[B3] IEEE Std 802.1D™-2004 IEEE Standard for Local and metropolitan area networks: Media Access Control Bridges. 2004. New York: Institute of Electrical and Electronics Engineers, Inc.

[B4] Distributed Medium Access Control (MAC) for Wireless Networks, Release 1.01. December 2006. San Ramon, California: WiMedia Alliance, Inc.[4]

[B5] RFC 4122, A Universally Unique IDentifier (UUID) URN Namespace, P. Leach, M. Mealling, and R. Salz. July 2005. Internet Engineering Task Force.[5]

[B6] NIST Special Publication 800-38A, Recommendation for Block Cipher Modes of Operation – *Methods and Techniques*, Morris Dworkin. December 2001. Gaithersburg, Maryland: National Institute of Standards and Technology (NIST), Technology Administration, U.S. Department of Commerce.[6]

---

[2] IEEE publications are available from the Institute of Electrical and Electronics Engineers, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331, USA (http://standards.ieee.org/).

[3] The IEEE standards referenced are trademarks belonging to the Institute of Electrical and Electronics Engineers, Inc.

[4] WiMedia publications are available from WiMedia Alliance, Incorporated, 2400 Camino Ramon, Suite 375, San Ramon, CA 94583, USA (http://www.wimedia.org).

[5] RFC documents are available from the RFC Editor, at http://www.rfc-editor.org.

[6] Referenced NIST publications are available from the National Institute of Standards and Technology Computer Security Resource Center, 100 Bureau Drive, Mail Stop 8930, Gaithersburg, MD 20899-8930, USA (http://csrc.nist.gov).

3. Definitions

For the purposes of this specification, the following terms and definitions apply. *The Authoritative Dictionary of IEEE Standards Terms*, Seventh Edition [B11], should be referenced for terms not defined in this clause.

3.1 abbreviated data frame: A frame format in which the header does not contain the original source and ultimate destination addresses, but instead relies on the MAC sublayer addresses of the transmitting and receiving devices. The frame format is suitable for exchange of frames between neighbors.

3.2 activate (a WLP service set): An action by a device to advertise its ability to generate or accept connections in order to communicate with other members of a WLP service set.

3.3 anchor cycle: A repeating period of superframes, at the start of which a device and all its neighbors will be in active mode and the device determines if it will act as a hibernation anchor based on a negotiation with its neighbors.

3.4 association: The overall process by which a device discovers neighbors, enrolls in WLP service sets (WSSs), activates WSSs, and connects to neighbors that have activated a WSS in common with the device.

3.5 bridge: A device that is capable of forwarding frames between segments.

3.6 bridge services: Services provided by a bridge to a device or remote bridge to forward frames from the device or remote bridge to a target or targets on another segment, and to forward frames from a source on another segment to the device or remote bridge.

3.7 client bridge: A bridge that is capable of forwarding frames to and from client devices.

3.8 client device: A device that is the original source or ultimate destination of data frames.

3.9 connect: An action by a device to identify a neighbor in a common WSS, acquire any required temporal security context, and prepare to exchange data frames.

3.10 device: An entity that implements the protocol defined in this specification.

3.11 enroll: A process by which a device obtains the necessary information to connect to other devices in a WLP service set.

3.12 enrollment session: A specific instance of the enrollment process carried out by two devices.

3.13 Ethernet type: A code used in frames to indicate the type of network traffic in the payload of the frame, as defined for use in IEEE 802 MAC frames in 10.4 of IEEE 802 [B1].

3.14 global cycle: A repeating period of superframes, to the start of which a device and all its neighbors synchronize for power management.

3.15 local cycle: A repeating period of superframes, at the start of which a device will be in active mode.

3.16 neighbor: An entity that is a neighbor as defined in the WiMedia MAC specification [B4] and implements the protocol defined in this specification.

3.17 node: Any addressable entity connected to a network, including devices as defined in this specification and entities that do not implement the protocol defined in this specification.

3.18 remote bridge: A device that, when paired with another remote bridge, is capable of forwarding frames between network segments attached to the remote bridges.

3.19 segment: A physical segment as defined in a wired network protocol such as IEEE 802.3, an equivalent to such a segment as defined in any compatible network protocol, or a logical segment that represents a link between two devices as defined in this specification.

3.20 service interval: The time between the start of two successive allocations of medium time to service a traffic stream.

3.21 standard data frame: A frame format in which the header contains the original source and ultimate destination addresses. The frame format is suitable for forwarding through a bridge.

3.22 WLP service set (WSS): A set of devices that share a common set of properties to permit communication between the members of the set.

4. Acronyms and abbreviations

ACW	anchor cycle weight
BPOIE	beacon period occupancy information element
BPST	beacon period start time
DRP	distributed reservation protocol
DS	differentiated services
EUI	extended unique identifier
GCSC	global cycle start countdown
GCST	global cycle start time
IE	information element
IP	internet protocol
IV	initialization vector
KDK	key derivation key
MAC	medium access control
MAS	medium access slot
MKID	master key identifier
MLME	MAC sublayer management entity
MSDU	MAC service data unit
MTU	maximum transmission unit
OUI	organizationally unique identifier
PCA	prioritized contention access
PHY	physical layer
PTK	pair-wise temporal key
PVR	personal video recorder
QoS	quality of service
SAP	service access point
SIMA	service interval-based MAS allocation
TIM	traffic indication map
TS	traffic stream
TSPEC	traffic specification
UUID	universally unique identifier
VLAN	virtual local area network
WLP	WiMedia logical link control protocol
WSS	WLP service set
WSSID	WLP service set identifier

5. General description

5.1 Architectural reference model

This specification defines a protocol, referred to as the WiMedia logical link control protocol (WLP), for data networking using the services of the WiMedia MAC. The protocol uses the MUX sublayer and service defined in the WiMedia MAC specification [B4]. The protocol corresponds to the logical link control sublayer of the standard ISO/OSI IEEE 802 reference model [B12], as shown in Figure 1.

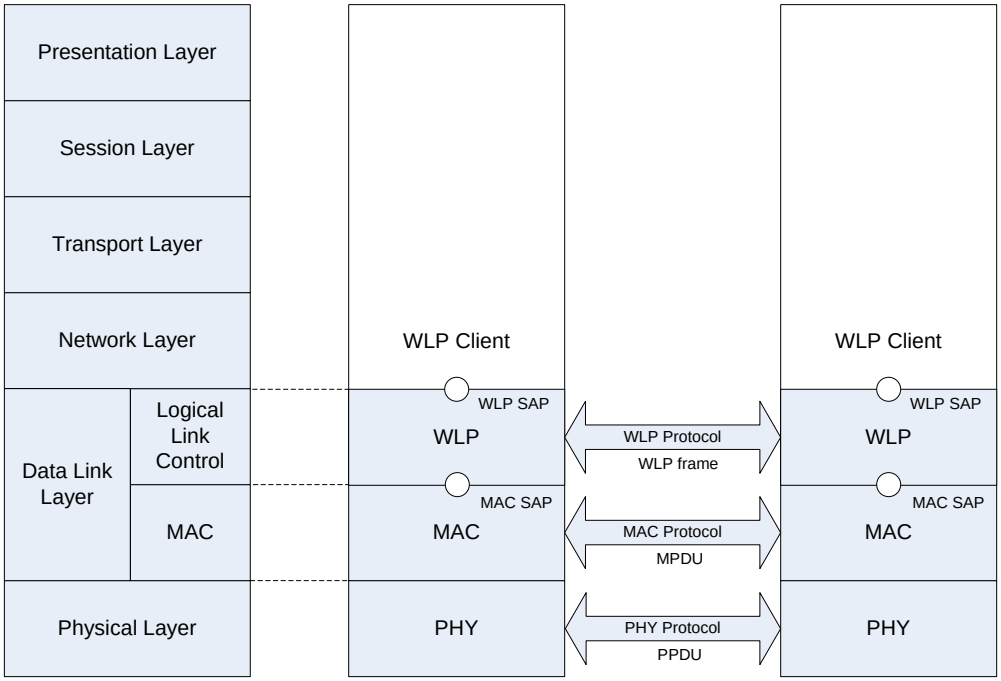


Figure 1 — Architectural reference model

5.2 Functional overview

This specification defines frame formats and requirements to support transfer of network layer packets over the WiMedia radio platform. It also defines support for bridging of frames between WLP and a wired Ethernet protocol or other similar networking protocols, as well as procedures for establishing secure relationships between devices.

The MUX service identifies frames with a Protocol ID value. WLP uses one WiMedia-defined Protocol ID value (0x0100) to identify WLP frames. WLP defines four types of frames: standard data frames, abbreviated data frames, control frames, and association frames. Clause 6 defines frame formats and IE formats used by devices.

5.3 Device overview

This specification defines the behavior of three functions that may be included in a device. Devices with a client function (client devices) are the original source or ultimate destination of data traffic conveyed in WLP frames.

Client devices may communicate directly with other client devices that belong to the same WSS, as described in 7.2. They may also communicate with other client devices and nodes, such as Ethernet stations, by using the services of client bridges.

Devices with a client bridge function (client bridges) forward frames to or from client devices that have requested bridge services. Client devices direct frames to a client bridge in order to communicate with destination nodes reachable through the client bridge. A client bridge and client device must belong to the same WSS for the client bridge to provide bridge services.

Devices with a remote bridge function (remote bridges) offer connectivity between network segments. Remote bridges forward frames to and from other remote bridges, such that each pair of remote bridges creates a new segment bridged to the segments attached to the remote bridges. Remote bridges implement IEEE 802.1D learning bridge mechanisms, and make forwarding decisions using a filter table. Remote bridges implement a spanning tree protocol according to [B3] in order to eliminate redundant paths (loops) throughout the network.

Figure 2 illustrates an example network that contains client devices, client bridges, and remote bridges. Some devices implement a single function, while others support multiple functions. In this example, assuming that all devices have registered with at least one bridge in radio range, any node can reach any other node.

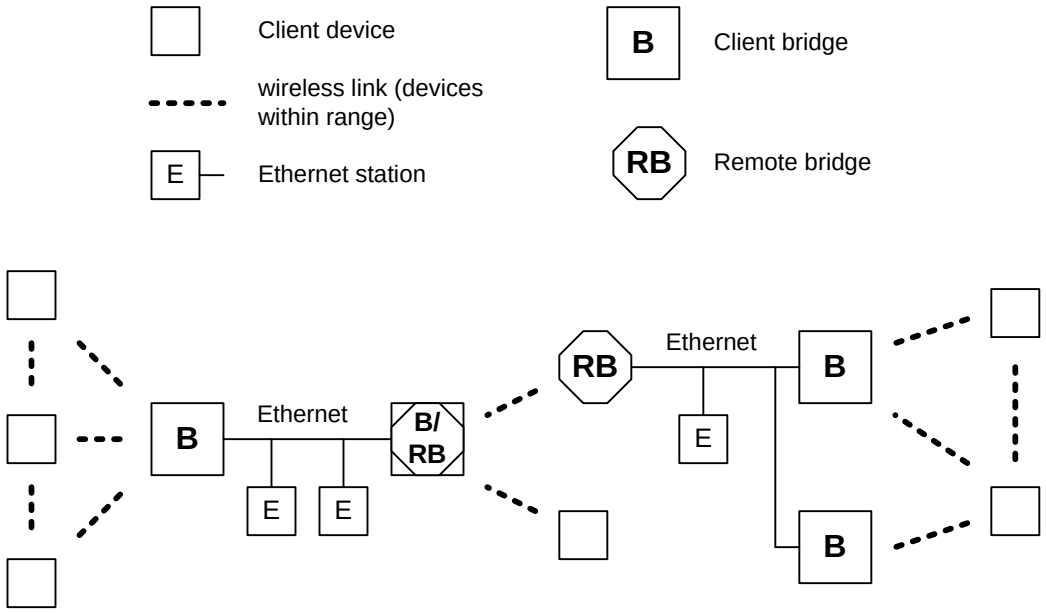


Figure 2 — Example network topology

5.4 Bridge services overview

A bridge provides IEEE 802.1D bridge functionality [B3] to allow forwarding of frames between client devices or remote bridges and other nodes on a network.

A bridge advertises bridge capabilities in a WLP IE. A neighbor requests bridge services from the bridge in order to initiate the forwarding of frames between the neighbor and other nodes reachable through the bridge. A bridge could offer connection to other client devices, to nodes reachable through wired or wireless ports using other protocols, or to both. A remote bridge may request bridge services from another remote bridge, creating a pair that can link networks that are based on other protocols. A device may request bridge services from more than one bridge. Such a device might receive the same frame from multiple bridges.

A bridge does not forward frames to or from neighbors that have not registered with it for bridge services. A bridge forwards data frames to and from neighbors that have registered with it for bridge services.

Traditionally, each physical interface in a bridge is assigned to a single port. For purposes of defining bridge functionality, this specification assumes that each neighbor that has registered for bridge services is assigned a bridge port in the bridge. Since these ports are all associated with a single WiMedia physical interface, this specification refers to these ports as virtual ports. Figure 3 illustrates an example bridge with virtual ports.

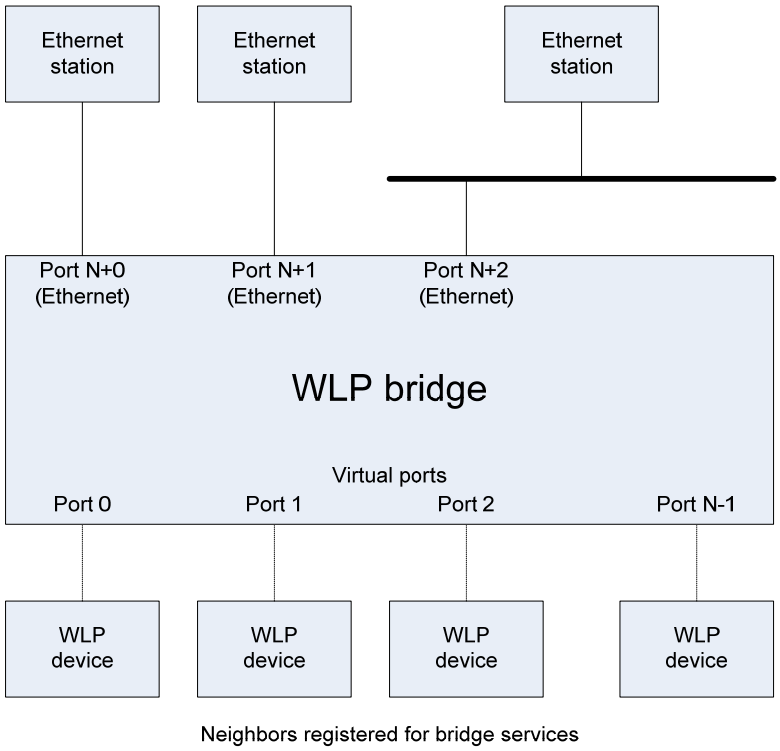


Figure 3 — Example WLP bridge with virtual ports

For each virtual port connected to a neighbor that is not a remote bridge, the bridge does not need to apply a spanning tree algorithm because such a device has explicitly declared that it is a client device and will not forward frames. For each virtual port connected to a neighbor that is a remote bridge, the bridge applies the spanning tree algorithm and may deactivate the port in order to remove loops. The bridge keeps a station cache for this virtual port as it does for LAN technology ports.

A bridge might have only virtual ports, in which case it would forward frames only between client devices and/or remote bridges.

Each neighbor that registers for bridge services enables specific multicast addresses and protocols for frame forwarding. The neighbor can also request forwarding based on the existence or value of a VLAN identifier in a frame. A bridge does not forward a frame onto the wireless medium if the frame's multicast address, protocol ID, and VLAN identifier do not pass the filter criteria for at least one neighbor.

A bridge may provide support for establishment of a DRP reservation upon request from a client device. A bridge that supports this feature advertises support in its WLP IE. The request from a client device or remote bridge includes TSPEC information and filtering parameters to identify the expected traffic.

5.5 WLP service sets

All WLP devices belong to one or more WLP service sets (WSSs) in order to segregate traffic on the medium based on user-controlled membership in the WSSs. WSSs are either secure, to permit traffic to be protected from various passive and active attacks, or non-secure, to permit traffic segregation with no protection from eavesdropping or identity spoofing.

Before two devices can exchange standard or abbreviated data frames, the devices must discover each other, enroll in and activate a common WLP service set (WSS), and establish a connection using the WSS properties, collectively referred to as the association process.

A device may, at any time, create a new WSS or enroll in an existing WSS. Enrollment results in learning the properties of the WSS, including, in the case of a secure WSS, the WSS master key. Once a device has created or enrolled in a WSS, it may activate the WSS, indicating a desire to communicate with other devices enrolled in the WSS. A device with an activated WSS can connect to any neighbor that has also activated the WSS. For a secure WSS, the connection process includes a WiMedia MAC 4-way handshake to establish a pair-wise temporal key (PTK). Once connected, devices can exchange WSS data frames.

Figure 4 provides a visual representation of the association process, including possible states and actions of a device.

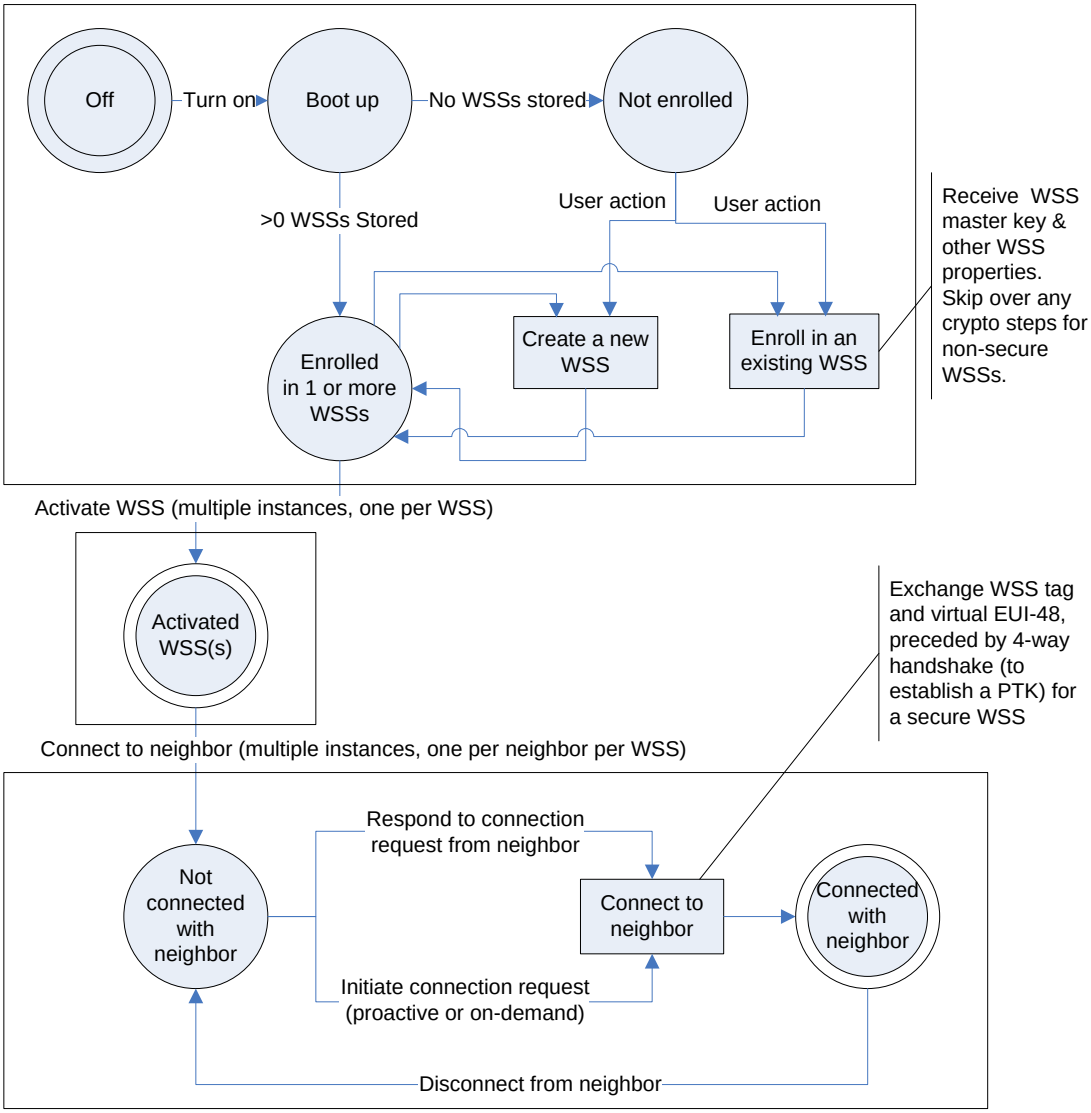


Figure 4 — Association process flow diagram

Subclause 7.2 describes how a device enrolls in a WSS, activates a WSS, and connects to other devices enrolled in an activated WSS.

A WLP device that participates in multiple WSSs effectively presents multiple network interfaces on a single physical interface. WLP allows such devices to assign a unique EUI-48 to each activated WSS. When a device requests bridge services in a WSS, the device must use a unique EUI-48 for that WSS in order to ensure that other devices are able to correctly forward frames to it.

5.6 Broadcast traffic announcement

A WLP device may address frames to broadcast or multicast targets. Transmission of broadcast or multicast data is under the control of the transmitting device which should attempt to minimize the time intended recipients are required to listen in order to receive the broadcast or multicast traffic.

For application-level control traffic, such as IP control packets and responses, the inter-arrival rate and traffic profile is highly dependent on the network topology and its rate of change. For small networks, the total offered load of control traffic from such protocols is small and irregular. This

specification provides a means for a source device to indicate to neighbor devices if and when broadcast or multicast traffic transfer will be attempted in the current superframe. This indication is made by inclusion of information in the device's WLP IE, as described in 7.3.5.

5.7 Power management

To minimize power consumption for power-sensitive devices, this specification defines a mechanism that enables devices to periodically go into active mode, to dynamically adjust their hibernation duty cycles, to inform their neighbors about their hibernation/active cycles, and to negotiate such cycles with their neighbors. This exchange of information enables a device to determine in which superframes each of its neighbors is active, so that it can potentially avoid the need to send broadcast or multicast traffic multiple times, and reduce the idle time it must wait for intended recipients to become active.

If all devices were to go into hibernation mode, the first device to return to active mode would need to scan for at least an entire superframe, which could result in consumption of more energy than required to remain in active mode. Furthermore, two or more devices could independently create different beacon period start times (BPSTs), which would result in significant protocol overhead to align the different BPSTs. To address these problems, this specification defines a hibernation anchor selection mechanism to enable devices to determine a connected set of anchors such that every device is either in the radio range of an anchor or is itself an anchor. The mechanism is designed to minimize the number of anchors and to rotate the anchor role so that devices share the burden of remaining in active mode.

5.8 Quality of service

Subclause 7.6 of this specification describes two types of quality of service (QoS) provisioning supported by this specification via two medium access mechanisms in the WiMedia MAC. For applications with traffic characteristics that are known and service quality requirements that are precisely specified, the reservation-based medium access mechanism, DRP, supports parameterized QoS. For applications with traffic characteristics and service quality requirements that are unknown or unspecified, the contention-based medium access mechanism, PCA, supports prioritized QoS. Legacy applications that do not have specified traffic characteristics or user priority use the AC_BE access category of PCA to obtain medium access with a corresponding user priority of *best effort*.

5.9 External requirements

The protocol defined in this specification requires the following optional features from the WiMedia MAC [B4].

- Prioritized contention access (PCA), including use of the PCA Availability IE, TIM IE, and PCA reservations.
- Distributed reservation protocol (DRP).
- Operation as a hibernation anchor.
- Transmission of frames with Imm-ACK policy
- MAC security mechanisms
- Use of a Generated DevAddr and non-NULL EUI-48 value

6. WLP frame formats

This clause defines the format of WLP frames and the WLP IE.

6.1 Data structure conventions

The following conventions and definitions apply throughout this clause.

6.1.1 Figures

Frames and IEs are described as a sequence of fields in a specific order. Figures in clause 6 depict fields in the order they are delivered to the MAC SAP, from left to right, where the left-most field is transmitted first in time. In field figures, bits within the field are numbered from the least-significant bit on the right to the most-significant bit on the left.

An example sequence of fields is illustrated in Figure 5.

octets: 2	1	…	4
First field transmitted (2 octets)	Second field transmitted (1 octet)	…	Last field transmitted (4 octets)

Figure 5 — Example sequence of fields

6.1.2 Octet order

Unless otherwise noted, fields larger than a single octet are encoded as a number and delivered to the MAC SAP in order from the octet containing the least-significant bits to the octet containing the most-significant bits.

An example of a bitmap specification for a two-octet field is illustrated in Figure 6.

bits: b15–b13	b12–b8	b7–b0
Most-significant bits of second octet transmitted	Least-significant bits of second octet transmitted	First octet transmitted

Figure 6 — Example bitmap specification for a field

When explicitly stated, fields encoded as an octet string are delivered to the MAC SAP in order from the first octet of the string to the last octet of the string. Fields that contain a UTF-8 string are treated as octet strings.

6.1.3 Concatenation

The concatenation operator (||) is used to indicate when two octet strings are joined and treated as a single octet string. The octet string on the left of the operator appears first in the combined string. If concatenation of a number larger than one octet is indicated, the number is converted to an octet string such that the first octet in the string contains the most-significant bits.

6.1.4 Encoding

Values specified in decimal are encoded in unsigned binary unless otherwise stated.

A bitmap is a sequence of bits, labeled as bit[0] through bit[N-1]. A bitmap is encoded in a field such that bit[0] corresponds to the least-significant bit of the field and subsequent bitmap elements correspond to subsequent significant bits of the field. Octets of the field are presented to the MAC SAP in ascending index value order.

Reserved fields are set to zero on transmission and ignored on reception. Fields are not set to reserved values on transmission. Unless otherwise noted, fields that are set to reserved values or are defined based on other fields that are set to reserved values are ignored on reception.

6.2 General WLP frame format

WLP frames are delivered to the WiMedia MUX service [B4] as MUX payloads. The MUX header for all WLP frames is the WLP Protocol ID (0x0100). The general WLP frame format is illustrated in Figure 7.

octets: 1	N
WLP Frame Type	Control/attribute fields or client data, optionally including header fields

Figure 7 — General WLP frame format

The WLP Frame Type field is set to a value from Table 1, which contains a list of valid WLP Frame Types, the names of the frame types, and the subclauses that describe the frame format for each of the frame types.

Table 1 — WLP Frame Type field encoding

Value	WLP Frame Type	Subclause
0	Standard Data	6.3
1	Abbreviated Data	6.4
2	Control	6.5
3	Association	6.6
4–255	Reserved	

6.3 Standard data frames

The format of a standard data frame is illustrated in Figure 8.

octets: 1	1	6	6	2	N
WLP Frame Type (=0)	WSS tag	Destination Address	Source Address	Type/Length	Client Data

Figure 8 — Standard data frame format

The WLP Frame Type field is set to zero, as shown in Table 1.

The WSS tag field is set to a value used by the transmitting device to identify the WSS for the data frame, as described in 7.2.5.

The Destination Address, Source Address, and Type/Length fields are formatted as described in IEEE 802.3 [B2] subclauses 3.2.3 through 3.2.7.

The Destination Address field is set to the EUI-48 [B10] of the ultimate destination of the frame. The EUI-48 is a sequence of 6 octets, labeled as eui[0] through eui[5]. Octets of the EUI-48 are passed to the MAC SAP in ascending index value order.

The Source Address field is set to the EUI-48 of the original source of the frame. The EUI-48 is a sequence of octets, labeled as eui[0] through eui[5]. Octets of the EUI-48 are passed to the MAC SAP in ascending index value order.

The Type/Length field is set to a type or length value as described in IEEE 802.3 subclause 3.2.6. The two octets of the field are encoded as an unsigned binary value, and are delivered to the MAC SAP in order from the octet containing the most-significant bits to the octet containing the least-significant bits.

The Client Data field contains the payload of the frame as received from the WLP client. The format is defined according to the value in the Type/Length field. The contents of the field are delivered to the MAC SAP in the same octet order as received from the WLP client.

If the Type/Length field is set to 802.1QTagType (0x8100), the Client Data field is formatted as described in IEEE 802.3 subclause 3.5.

The Client Data field in a frame generated by a WLP device does not include any pad, as described in IEEE 802.3 [B2] subclause 3.2.7. It is possible that the Client Data field in a received frame will contain a pad, as the frame could have been forwarded from a segment that required a minimum frame size.

6.4 Abbreviated data frames

An abbreviated data frame is a shorter version of the standard data frame that assumes that the original source and ultimate destination of the frame are the transmitting and recipient devices, respectively. The format of an abbreviated data frame is illustrated in Figure 9.

octets: 1	1	2	N
WLP Frame Type (=1)	WSS tag	Type/Length	Client Data

Figure 9 — Abbreviated data frame format

The WLP Frame Type field is set to one, as shown in Table 1.

The WSS tag field is set to a value used by the transmitting device to identify the WSS for the data frame.

The Type/Length and Client Data fields are set as defined in 6.3.

6.5 Control frames

The general format of a control frame is illustrated in Figure 10.

octets: 1	1	N
WLP Frame Type (=2)	Control Subtype	Control Subtype-specific data

Figure 10 — Control frame format

The WLP Frame Type field is set to two, as shown in Table 1.

The Control Subtype field is set to a value from Table 2, which contains a list of valid subtype values, descriptions, and the subclauses that describe the frame format for each of the control frame subtypes.

Table 2 — Control Subtype field encoding

Value	Control Subtype	Subclause
0	Bridge Services Request	6.5.1
1	Bridge Services Response	6.5.2
2	DRP Reservation Request	6.5.3
3	DRP Reservation Response	6.5.4
4	Local Cycle Change Request	6.5.5
5–255	Reserved	

6.5.1 Bridge Services Request

The format of a Bridge Services Request control frame is illustrated in Figure 11.

octets: 1	1	16	1	1	1	1	4×M	12×N	2×P
WLP Frame Type (=2)	Control Subtype (=0)	WSSID	Bridge Services Control	Protocol Count (=M)	Multicast Address Count (=N)	VLAN Identifier Count (=P)	Protocol Ranges	Multicast Address Ranges	VLAN Identifiers

Figure 11 — Bridge Services Request control frame format

The Control Subtype field is set to zero, as defined in Table 2.

The WSSID field is set to a value that identifies the WSS to enable for bridge services. The value is a UUID encoded as an octet string in the order the octets are shown in string representation in RFC 4122 [B5].

6.5.1.1 Bridge Services Control field

The Bridge Services Control field is illustrated in Figure 12.

bits: b7–b4	b3	b2	b1	b0
Reserved	Enable VLAN Forwarding	Enable Non-VLAN Forwarding	Enable Remote Bridge Services	Enable Client Bridge Services

Figure 12 — Bridge Services Control field format

The Enable VLAN Forwarding bit is set to one in a request to enable forwarding of VLAN frames, and is set to zero in a request to not forward any VLAN frames.

The Enable Non-VLAN Forwarding bit is set to one in a request to enable forwarding of non-VLAN frames, and is set to zero in a request to not forward any non-VLAN frames.

The Enable Remote Bridge Services bit is set to one in a request to enable forwarding of frames between the transmitting remote bridge and the recipient remote bridge or to update bridge services settings for a remote bridge, and is set to zero in a request to disable forwarding between the pair of remote bridges.

The Enable Client Bridge Services bit is set to one in a request to enable forwarding of frames between a client device and a bridge or to update bridge services settings for a client device, and is set to zero in a request to disable bridge services for a client device.

6.5.1.2 Count fields

The Protocol Count field is set to the number of Protocol Start and Protocol End pairs included in the frame. If the field is set to zero, it indicates a request to forward no frames.

The Multicast Address Count field is set to the number of Address Start and Address End pairs included in the frame. If the field is set to zero, it indicates a request to forward no multicast frames.

The VLAN Identifier Count field is set to the number of VLAN Identifiers included in the frame. If the field is zero and the Enable VLAN Forwarding bit is set to one, it indicates a request to forward frames regardless of the VLAN Identifier value.

6.5.1.3 Protocol Ranges field

The Protocol Ranges field consists of a sequence of Protocol Start and Protocol End values that identify ranges of protocols of frames to be forwarded to the client device, as illustrated in Figure 13. All desired protocol ranges are included in a request to enable or update bridge services settings. No protocol ranges are included in a request to disable bridge services.

octets: 2	**2**	**…**	**2**	**2**
Protocol Start 1	Protocol End 1	…	Protocol Start M	Protocol End M

Figure 13 — Protocol Ranges field format

The Protocol Start and Protocol End fields are set to Ethernet type values. Each pair of fields represents a request to the recipient to forward frames corresponding to any protocol identified by an Ethernet type in the range [Protocol Start, Protocol End], inclusive. The two octets of each field are encoded as an unsigned binary value, and are delivered to the MAC SAP in order from the octet containing the most-significant bits to the octet containing the least-significant bits.

6.5.1.4 Multicast Address Ranges field

The Multicast Address Ranges field consists of a sequence of Address Start and Address End values that identify multicast address ranges of frames to be forwarded to the client device, as illustrated in Figure 14. All desired multicast address ranges are included in a request to enable or update bridge services settings. No multicast address ranges are included in a request to disable bridge services.

octets: 6	**6**	**…**	**6**	**6**
Address Start 1	Address End 1	…	Address Start N	Address End N

Figure 14 — Multicast Address Ranges field format

The Address Start and Address End fields are set to EUI-48 values. Each pair of fields represents a request to the recipient to forward multicast frames with destination addresses in the range [Address Start, Address End], inclusive. If Address Start is set to the EUI-48 value 01-00-00-00-00-00 and Address End is set to FF-FF-FF-FF-FF-FF, it indicates a request to forward all multicast frames. Each field is a sequence of 6 octets, labeled as eui[0] through eui[5]. Octets of the EUI-48 are passed to the MAC SAP in ascending index value order.

6.5.1.5 VLAN Identifiers field

The VLAN Identifiers field is set to a list of IEEE 802.3 [B2] VLAN Identifier values. Each two-octet value is between 0 and 4095, and indicates a request to forward frames that contain a QTag prefix with VLAN Identifier set to that value. This field is reserved when the Enable VLAN Forwarding bit is set to zero.

6.5.2 Bridge Services Response

The format of a Bridge Services Response control frame is illustrated in Figure 15.

octets: 1	1	16	2
WLP Frame Type (=2)	Control Subtype (=1)	WSSID	Response

Figure 15 — Bridge Services Response control frame format

The Control Subtype field is set to one, as defined in Table 2.

The WSSID field is set to the same value as the WSSID field in the corresponding Bridge Services Request.

The Response field is a bit field set to zero to indicate a successful request, or to a binary value with one or more bits set to one as defined in Table 3 to indicate a failed request.

Table 3 — Response field encoding for a failed request

Bit	Meaning	Description
b0	Invalid address range	This bit is set to one if there is an incorrect address range specified. For example, Address Start is greater than Address End, or address ranges overlap.
b1	Invalid protocol range	This bit is set to one if there is an incorrect protocol range specified. For example, Protocol Start is greater than Protocol End, or protocol ranges overlap.
b2	Invalid VLAN identifier	This bit is set to one if a VLAN identifier is specified that is not in the range [0,0x0FFF], inclusive.
b3	Too many address ranges	This bit is set to one if the number of Address Start, Address End pairs is too high for the bridge to process.
b4	Too many protocol ranges	This bit is set to one if the number of Protocol Start, Protocol End pairs is too high for the bridge to process.
b5	Too many VLAN identifiers	This bit is set to one if the number of VLAN Identifiers is too high for the bridge to process.
b6	Unsupported protocol	This bit is set to one if a protocol is specified that is not in the range [0x0600–0xFFFF], inclusive.
b7	Invalid count field	This bit is set to one if the Count field values are not consistent with the length of the frame.
b8	Unsupported capability	This bit is set to one if a request is made for an unsupported capability, such as a request for bridge services from a non-bridge device.
b9	Resource limitation error	This bit is set to one if the bridge is unable to accept the request due to a lack of resources, such as space in forwarding tables.
b10	WSS not activated	This bit is set to one if the requesting device or the bridge has not activated the WSS indicated by the WSS tag field.
b11	Device not connected	This bit is set to one if the bridge requires a secure relationship to accept a bridge services request, and that relationship has not been established.
b12–b15	Reserved	

6.5.3 DRP Reservation Request

The format of a DRP Reservation Request control frame is illustrated in Figure 16.

octets: 1	1	1	25	variable
WLP Frame Type (=2)	Control Subtype (=2)	Request Parameters	TSPEC	Traffic Filtering Parameters

Figure 16 — DRP Reservation Request control frame format

The Control Subtype field is set to two, as defined in Table 2.

6.5.3.1 Request Parameters field

The Request Parameters field is illustrated in Figure 17.

bits: b7–b4	b3–b1	b0
Reservation Type	Stream Index	Establish

Figure 17 — Request Parameters field format

The Reservation Type field is set to the value for the bridge to use in the Reservation Type field of the DRP IE for the reservation.

The Stream Index field is set to the value for the bridge to use in the Stream Index field of the DRP IE for the reservation.

The DRP IE fields are defined in the WiMedia MAC specification [B4].

The Establish bit is set to one to indicate this is a request to establish or modify a reservation, or is set to zero to indicate this is a request to remove a reservation.

6.5.3.2 TSPEC field

The TSPEC field defines the traffic characteristics and service requirements of the traffic stream (TS) to service in the requested reservation. It is illustrated in Figure 18.

octets: 1	4	4	4	2	2	4	4
Service Type	Mean Data Rate (r)	Peak Data Rate (p)	Maximum Burst Size (b)	Maximum Packet Size (M)	Minimum Policed Unit (m)	Requested Service Rate (R)	Slack Term (S)

Figure 18 — TSPEC Parameters field format

The Service Type field is set to a value that specifies the primary service type requested in this TSPEC, encoded as shown in Table 4.

Table 4 — Service Type field encoding

Value	Service Type
0	Guaranteed [B19]
1	Controlled-load [B18]
2–255	Reserved

The Mean Data Rate (r) field is set to the average data rate, in octets per second, for transport of packets that belong to this traffic stream (TS).

The Peak Data Rate (p) field is set to the maximum data rate, in octets per second, for transfer of packets that belong to this TS. The Peak Data Rate value is the maximum data rate that will occur within any time interval greater than or equal to 256 microseconds.

The Maximum Burst Size (b) field is set to the maximum burst, in octets, of packets that belong to this TS. A value of zero indicates that there are no bursts. The Maximum Burst Size value is the maximum number of octets beyond that indicated by the Mean Data Rate field that will arrive over any time interval.

The Maximum Packet Size (M) field is set to the maximum size, in octets, of packets that belong to this TS.

The Minimum Policed Unit (m) field is set to a size value such that all packets with smaller size are counted as that size in terms of mean data rate, peak data rate, and maximum burst size specifications.

The Requested Service Rate (R) field is set to the data rate, in octets per second, for bandwidth reservation for transport of the packets that belong to this TS. The field value must be greater than or equal to the Mean Data Rate field value and less than or equal to the Peak Data Rate field value. This field is valid only when the Service Type field is set to Guaranteed, and is reserved otherwise.

The Slack Term (S) field is set to the difference, in microseconds, between the maximum allowed delay and the queuing delay resulting from using the Requested Service Rate R, in transfer of packets that belong to this TS. This field is valid only when the Service Type field is set to Guaranteed, and is reserved otherwise.

The values of the Mean Data Rate, Peak Data Rate, Maximum Burst Size, Maximum Packet Size, Minimum Policed Unit, and Requested Service Rate fields consider only the WLP Client Data portion of packets that belong to this TS.

6.5.3.3 Traffic Filtering Parameters field

The Traffic Filtering Parameters field is illustrated in Figure 19.

octets: 1	**$3 + 2\times M_1$**	**$3 + 2\times M_2$**	**…**	**$3 + 2\times M_N$**
Filter Set Count (=N)	Filter Set 1	Filter Set 2	…	Filter Set N

Figure 19 — Traffic Filtering Parameters field format

The Filter Set Count field is set to the number of filter sets included in the Traffic Filtering Parameters field.

The format of a Filter Set field is illustrated in Figure 20.

octets: 1	**2**	**M**	**M**
Filter Length (=M)	Offset	Mask	Value

Figure 20 — Filter Set field format

The Filter Length field is set to the number of octets in the WLP frame to compare with the contents of the Value field.

The Offset field is set to the index of the first octet in the WLP frame to compare, where zero indicates the filter comparison starts at the first octet in the WLP frame, that is, with the WLP Frame Type.

The Mask field is set to an octet string that indicates which bits in the WLP frame are compared. If a bit in the mask field is set to one, the corresponding bit in the WLP frame is compared. Otherwise, the corresponding bit is ignored.

The Value field is set to the sought values for the WLP frame. The field is encoded as an octet string. If all bits in the WLP frame that correspond to a one bit in the Mask field match the corresponding bits in this field, the WLP frame will be forwarded using the DRP reservation established based on this request.

6.5.4 DRP Reservation Response

The format of a DRP Reservation Response control frame is illustrated in Figure 21.

octets: 1	1	1	2
WLP Frame Type (=2)	Control Subtype (=3)	Response Parameters	Response

Figure 21 — DRP Reservation Response control frame format

The Control Subtype field is set to three, as defined in Table 2.

The Response Parameters field is set to the same value as the Request Parameters field in the corresponding DRP Reservation Request.

The Response field is a bit field set to zero to indicate a successful request, or to a binary value with one or more bits set to one as defined in Table 5 to indicate a failed request.

Table 5 — Response field encoding for a failed request

Bit	Meaning	Description
b0	Unsupported capability	This bit is set to one if a request is made of a bridge that cannot establish DRP reservation for a client device.
b1	Invalid traffic filtering parameters	This bit is set to one if the traffic filtering parameters specified are either inconsistent with parameters set by Bridge Services Request, or illegal. For example, the traffic filtering parameters involve a Protocol ID which is not covered by the Bridge Services Request.
b2	Unsupported traffic filtering parameters	This bit is set to one if the traffic filtering parameters specified are not supported by the bridge. For example, a filter length or the number of filter sets is too big.
b3	Not enough MASs available	This bit is set to one if the bridge is unable to accept the request due to MAS unavailability.
b4	Not enough resources	This bit is set to one if the bridge is unable to accept the request due to a lack of available resources in the bridge.
b5	Security violation	This bit is set to one if the traffic filtering parameters violate the security policy set for the bridge.
b6–b15	Reserved	

6.5.5 Local Cycle Change Request

The format of a Local Cycle Change Request control frame is shown in Figure 22.

octets: 1	1	1	2	1	…	2	1
WLP Frame Type (=2)	Control Subtype (=4)	Request Count (=N)	DevAddr 1	Local Cycle Index 1	…	DevAddr N	Local Cycle Index N

Figure 22 — Active Cycle Request control frame format

The Control Subtype field is set to four, as defined in Table 2.

The Request Count field is set to N, where N is the number of DevAddr/Local Cycle Index pairs that follow.

Each DevAddr field is set to the unicast or multicast DevAddr of a neighbor or group that is requested to change its local cycle.

Each Local Cycle Index field is set to the requested local cycle index value. The use of the local cycle index is specified in 7.5.1.

6.6 Association frames

Association frames are encoded as a list of attributes, each of which contains a specific piece of information relevant for the particular association subtype. The general format of an association frame is illustrated in Figure 23.

octets: 1	1	M_1	M_2	…	M_N
WLP Frame Type (=3)	Association Subtype	Attribute 1	Attribute 2	…	Attribute N

Figure 23 — Association frame format

The WLP Frame Type field is set to three, as shown in Table 1.

The Association Subtype field is set to the Attribute Value in the Message Type attribute included in the frame. This duplicate information is provided to permit identification without parsing the attribute fields, as well as when using only the attribute fields.

Certain attributes are required for each association subtype, as indicated below. Additional optional or vendor-specific attributes of types defined below are permitted in any attribute set as well. The required attributes appear in the order listed for each association subtype.

6.6.1 Attribute fields

The general format of an Attribute field is illustrated in Figure 24.

octets: 2	2	M
Attribute Type	Attribute Length (=M)	Attribute Value

Figure 24 — Attribute field format

The Attribute Type field is set to a value from Table 6, and determines the contents of the Attribute Value field.

The Attribute Length field is set to the length of the Attribute Value field, in octets.

The Attribute Value field is set as defined in Table 6, which lists valid Attributes for WLP devices. Attributes with Attribute Type values between 0x2000 and 0x20FF are specific to WLP. All values not listed in the table are reserved.

Table 6 — Attribute field encoding

Attribute Name	Attribute Type	Attribute Length (octets)	Attribute Value
Authenticator	0x1005	8	The message authentication code, encoded as an octet string, that protects the integrity of the attributes contained in the same frame, as further defined in 6.6.7–6.6.13
Device Name	0x1011	1–32	The friendly name of the sending device, encoded as an octet string in UTF-8 format
Device Password ID	0x1012	2	Identifies the source or type of Device Password used, as defined in Table 7
E-Hash1	0x1014	32	The enrollee's hash commitment for the first half of the Device Password, $E\text{-}H_1$, as described in 7.2.4.4. The value is encoded as an octet string.
E-Hash2	0x1015	32	The enrollee's hash commitment for the second half of the Device Password, $E\text{-}H_2$, as described in 7.2.4.4. The value is encoded as an octet string.
E-SNonce1	0x1016	16	A 128-bit secret nonce, $E\text{-}S_1$, generated by the enrollee to compute $E\text{-}H_1$, as described in 7.2.4.4
E-SNonce2	0x1017	16	A 128-bit secret nonce, $E\text{-}S_2$, generated by the enrollee to compute $E\text{-}H_2$, as described in 7.2.4.4
Encrypted Settings	0x1018	variable	A 128-bit initialization vector (IV), encoded as an octet string, followed by an encrypted field. The field, before being encrypted, contains a set of attributes and a trailing pad. The set of attributes required for various association frames is defined in 6.6.9–6.6.13. The trailing pad is 1–16 octets, such that the field length is an integer multiple of 16 octets. Each octet is set to the number of octets in the pad. The field is encrypted with 128-bit AES in CBC mode [B6] using IV as the 128-bit initialization vector and a key derived as described in 7.2.4.3. The encrypted field is encoded as an octet string.
Enrollee Nonce	0x101A	16	The enrollee's nonce, N_e, which is a 128-bit random number freshly generated by the enrollee for an enrollment session
Key Wrap Authenticator	0x101E	8	The first 64 bits of the HMAC-SHA-256 computed over the plaintext of an Encrypted Settings attribute. The value is encoded as an octet string.
Manufacturer	0x1021	0–64	The name of the manufacturer of the sending device, encoded as an octet string in UTF-8 format
Message Type	0x1022	1	Identifies the specific association message sent by the registrar or enrollee, as shown in Table 8
Model Name	0x1023	0–32	The model name of the sending device, encoded as an octet string in UTF-8 format
Model Number	0x1024	0–32	The model number of the sending device, encoded as an octet string in UTF-8 format
Public Key	0x1032	384	The sender's Diffie-Hellman public key, PK_e or PK_r, as defined in A.4 and A.5
Registrar Nonce	0x1039	16	The registrar's nonce, N_r, which is a 128-bit random number freshly generated by the registrar for an enrollment session
R-Hash1	0x103D	32	The registrar's hash commitment for the first half of the Device Password, $R\text{-}H_1$, as described in 7.2.4.4. The value is encoded as an octet string.

Attribute Name	Attribute Type	Attribute Length (octets)	Attribute Value
R-Hash2	0x103E	32	The registrar's hash commitment for the second half of the Device Password, $R\text{-}H_2$, as described in 7.2.4.4. The value is encoded as an octet string.
R-SNonce1	0x103F	16	A 128-bit secret nonce, $R\text{-}S_1$, generated by the registrar to compute $R\text{-}Hash_1$, as described in 7.2.4.4
R-SNonce2	0x1040	16	A 128-bit secret nonce, $R\text{-}S_2$, generated by the registrar to compute $R\text{-}Hash_2$, as described in 7.2.4.4
Serial Number	0x1042	0–32	The serial number of the sending device, encoded as an octet string in UTF-8 format
UUID-E	0x1047	16	The universally unique identifier (UUID) assigned to the enrollee. The value is encoded as an octet string in the order the octets are shown in string representation in RFC 4122 [B5]. The value is intended to uniquely identify an operational device and must remain unchanged for the lifetime of the device.
UUID-R	0x1048	16	The universally unique identifier (UUID) assigned to the registrar. The value is encoded as an octet string in the order the octets are shown in string representation in RFC 4122. The value is intended to uniquely identify an operational device and must remain unchanged for the lifetime of the device.
Primary Device Type	0x1054	8	The primary type or function of the sending device. The format of the Attribute Value field is defined in 6.6.1.8
Secondary Device Type	0x1055	8	A type or function of the sending device in addition to the Primary Device Type, as described in 6.6.1.8
Portable Device	0x1056	1	Set to one if the sending device is portable or zero if it is not portable
Application Extension	0x1058	17–1024	An application-defined field that permits inclusion of application-specific information in an association frame. The format of the Attribute Value field is defined in Figure 26.
WLP Version	0x2000	1	The protocol version of WLP implemented by the sending device
WSSID	0x2001	16	A WLP service set identifier. The value is a UUID encoded as an octet string in the order the octets are shown in string representation in RFC 4122.
WSS Name	0x2002	0–64	The friendly name of a WSS, encoded as an octet string in UTF-8 format
WSS Secure Status	0x2003	1	Set to one if the WSS is secure or zero if it is not secure
WSS Broadcast Address	0x2004	6	The multicast EUI-48, encoded as an octet string, used for broadcast traffic within a WSS
WSS Master Key	0x2005	16	A master key, encoded as an octet string, generated by a WSS founder and known by all WSS members, used for derivation of temporal keys
Accepting Enrollment	0x2006	1	Set to one if the registrar is accepting enrollment in this WSS, or zero otherwise
WSS Information	0x2007	variable	A set of attributes that define a WSS, including WSSID, WSS Name, WSS Secure Status, and WSS Broadcast Address, as defined in the D2 frame in 6.6.3
WSS Selection Method	0x2008	1	A value that indicates how a WSS is selected during enrollment, as described in Table 10

Attribute Name	Attribute Type	Attribute Length (octets)	Attribute Value
Association Methods List	0x2009	2, 4, 6, …, 30	A prioritized list of association methods, transmitted in order of the most-preferred first, each encoded as defined in Table 11
Selected Association Method	0x200A	2	A selected association method for an enrollment session, encoded as defined in Table 11
Enrollee Hash Commitment	0x200B	32	SHA-256(PK_e || N_e), encoded as an octet string, where PK_e is the enrollee's public key and N_e is the enrollee's nonce
WSS Tag	0x200C	1	A one-octet value that the sending device uses to identify the WSS in frames it transmits
WSS Virtual EUI-48	0x200D	6	An EUI-48, encoded as an octet string, that identifies the sending device within a WSS
WLP Association Error	0x200E	1	The result of an operation or step, as defined in Table 9
Vendor Extension	0x200F	4–1024	A vendor-defined field that permits inclusion of vendor-specific information in an association frame. The format of the Attribute Value field is defined in Figure 25.

6.6.1.1 Device Password ID attribute

Table 7 defines valid Attribute Values for the Device Password ID attribute. Attribute Values not listed in Table 7 are reserved.

Table 7 — Device Password ID attribute values

Attribute Value	Meaning
1	User-specified: The Device Password is a value provided by the user.
2	Machine-specified: The Device Password is a machine-generated random value.
6	Numeric-comparison: The Device Password is not used for the numeric comparison association method.

6.6.1.2 Message Type attribute

Table 8 defines the Attribute Value for the Message Type attribute that is included in each association frame. Attribute Values not listed in Table 8 are reserved.

Table 8 — Message Type attribute values

Attribute Value	Message type
2	D1
3	D2
4	M1
5	M2
7	M3

Attribute Value	Message type
8	M4
9	M5
10	M6
11	M7
12	M8
14	F0
32	E1
33	E2
34	C1
35	C2
36	C3
37	C4

6.6.1.3 Vendor Extension attribute

The format of the Attribute Value field for a Vendor Extension attribute is illustrated in Figure 25.

octets: 3	M
Vendor OUI	Vendor-defined Information

Figure 25 — Vendor Extension Attribute Value field format

The Vendor OUI field is set to the OUI value owned by the organization that defines the attribute, encoded as an octet string.

The Vendor-defined Information field is set as defined by the owner of the OUI. The field should contain an index or subtype field to permit multiple attributes to be defined by the vendor.

6.6.1.4 Application Extension attribute

The format of the Attribute Value field for an Application Extension attribute is illustrated in Figure 26.

octets: 16	M
Application UUID	Application-defined Information

Figure 26 — Application Extension Attribute Value field format

The Application UUID field is set to a globally unique UUID value that identifies this extension. The UUID is encoded as an octet string. The value must be a version 1 UUID based on a global IEEE 802 EUI-48, as defined in RFC 4122 [B5].

The Application-defined Information field is set as defined by the generator of the UUID.

6.6.1.5 WLP Association Error attribute

Table 9 defines valid values and associated error codes for the Attribute Value field of a WLP Association Error attribute. Attribute values not listed in Table 9 are reserved.

Table 9 — WLP Association Error attribute values

Attribute Value	Error	Description
0	No error	
1	Authenticator Failure	The device determined that an Authenticator or Key Wrap Authenticator attribute did not match that expected for the contents of a frame, and has abandoned the enrollment session.
2	Rogue Activity Suspected	The device received frames that potentially indicate an attempt to compromise the enrollment session, and has abandoned the enrollment session.
3	Device Busy	The device was unable to process the request, for example, because it received multiple requests but can only participate in one enrollment session at a time. The device has abandoned the enrollment session.
4	Setup Locked	The registrar was not accepting enrollment requests because of too many recent enrollment failures.
5	Registrar Not Ready	The registrar is not ready to continue enrollment, and has abandoned the enrollment session.
6	Invalid WSS Selection	The WSSID requested for enrollment or connection is not valid.
7	Message Timeout	The device reached a timeout waiting for the next association frame in the sequence, and has abandoned the enrollment session.
8	Enrollment Session Timeout	The device reached a timeout on completing the entire enrollment session, and has abandoned the enrollment session.
9	Device Password Invalid	The device determined that the Device Password was incorrect, and has abandoned the enrollment session.
10	Unsupported Version	The device received a frame that indicated a WLP version not supported by the recipient, and has abandoned the enrollment session.
11	Internal Error	An unexpected error specific to the device occurred. The device has abandoned any enrollment session in progress.
12	Undefined Error	An error not defined in this table occurred. The device has abandoned any enrollment session in progress.
13	Numeric Comparison Failure	The device received notice from the user that the numbers displayed for numeric comparison did not match, or the device reached a timeout waiting for the user to confirm that the numbers displayed matched. The device has abandoned the enrollment session.
14	Waiting For User Input	The device is not ready to send the next association frame in a sequence because it is waiting for user input. Any current enrollment session is still valid.

6.6.1.6 WSS Selection Method attribute

Table 10 defines the Attribute Value for the WSS Selection Method attribute, which indicates if the enrollee or registrar selects a WSS for enrollment. Attribute Values not listed in Table 10 are reserved.

Table 10 — WSS Selection Method attribute values

Attribute Value	WSS selection method	Description
1	Enrollee Selects	If this WSS selection method is used, the registrar is requested to provide information on all WSSs for which it is currently accepting enrollment. The enrollee will select the WSS in which it will seek to enroll.
2	Registrar Selects	If this WSS selection method is used, the registrar is requested to provide information about only a single WSS. The enrollee is not capable of selecting between multiple WSSs.

6.6.1.7 Association methods

Table 11 defines the encoding for the various association methods that may be supported by a device. Values not listed in Table 11 are reserved.

Table 11 — Association method encoding

Value	Association method
0x0008	Enrollee-display
0x0100	Registrar-display
0x0200	Numeric Comparison
0x0400	User-provided Password

6.6.1.8 Primary and Secondary Device Type attributes

The format of the Attribute Value field for a Primary or Secondary Device Type attribute is illustrated in Figure 27.

octets: 2	3	1	2
Category ID	OUI	OUI Subdivision	Subcategory ID

Figure 27 — Primary or Secondary Device Type Attribute Value field format

The Category ID field is set to a value from Table 12, and identifies the general category of the device primary or secondary function. Values not listed in Table 12 are reserved unless defined in another WiMedia specification.

Table 12 — Category ID field encoding

Value	Category
1	Computer
2	Input device
3	Printer, scanner, FAX, or copier
4	Camera
5	Storage Network
6	Infrastructure
7	Display
8	Multimedia device
9	Gaming device
10	Telephone
65535	Other

The OUI field is set to the OUI of the organization that defines valid values for the OUI Subdivision and Subcategory ID fields. The OUI is encoded as an octet string.

The OUI Subdivision field is defined by the OUI owner, and is used by the OUI owner to support multiple subcategory definition sets.

The Subcategory ID is defined by the OUI owner, and identifies the specific primary or secondary device function.

6.6.2 D1

The D1 association frame is sent by a device acting as an enrollee to discover certain information about a neighbor. The required attributes in a D1 association frame are listed in Table 13.

Table 13 — D1 association frame attributes

Attribute Name	Attribute Type	Notes
WLP Version	0x2000	The Attribute Value is set to 0x10.
Message Type	0x1022	The Attribute Value is set to 2.
UUID-E	0x1047	
WSS Selection Method	0x2008	
Device Name	0x1011	
Manufacturer	0x1021	This attribute is recommended, but not required.
Model Name	0x1023	This attribute is recommended, but not required.
Model Number	0x1024	This attribute is recommended, but not required.
Serial Number	0x1042	This attribute is recommended, but not required.

Attribute Name	Attribute Type	Notes
Primary Device Type	0x1054	This attribute is recommended, but not required.
WLP Association Error	0x200E	The Attribute Value is set to an error code (if any) from a previous enrollment session, to allow the registrar to take appropriate action, such as displaying an error message to the user.

6.6.3 D2

The D2 association frame is sent by a device acting as a registrar in response to a D1 frame. The required attributes in a D2 association frame are listed in Table 14.

Table 14 — D2 association frame attributes

Attribute Name	Attribute Type	Notes
WLP Version	0x2000	The Attribute Value is set to 0x10.
Message Type	0x1022	The Attribute Value is set to 3.
UUID-E	0x1047	
UUID-R	0x1048	
WSS Information	0x2007	This attribute can appear zero or more times in the frame in order to provide information on multiple WSSs. The required attributes within the WSS Information attribute are listed in Table 15.
Device Name	0x1011	
Manufacturer	0x1021	This attribute is recommended, but not required.
Model Name	0x1023	This attribute is recommended, but not required.
Model Number	0x1024	This attribute is recommended, but not required.
Serial Number	0x1042	This attribute is recommended, but not required.
Primary Device Type	0x1054	This attribute is recommended, but not required.
WLP Association Error	0x200E	

Table 15 — WSS Information attributes

Attribute Name	Attribute Type	Notes
WSSID	0x2001	
WSS Name	0x2002	
Accepting Enrollment	0x2006	
WSS Secure Status	0x2003	

Attribute Name	Attribute Type	Notes
WSS Broadcast Address	0x2004	

6.6.4 E1

The E1 association frame is sent by a device acting as an enrollee to begin an enrollment session. The required attributes in an E1 association frame are listed in Table 16.

Table 16 — E1 association frame attributes

Attribute Name	Attribute Type	Notes
WLP Version	0x2000	The Attribute Value is set to 0x10.
Message Type	0x1022	The Attribute Value is set to 32.
UUID-E	0x1047	
WSSID	0x2001	This attribute identifies the WSS selected by the enrollee.
Enrollee Hash Commitment	0x200B	The Attribute Value is set to SHA-256(PK_e || N_e), where PK_e and N_e are the enrollee's Diffie-Hellman public key and the enrollee's nonce, respectively.
Device Password ID	0x1012	
Association Methods List	0x2009	The Attribute Value is set to a list of association methods supported by the enrollee, in order of decreasing preference.
Device Name	0x1011	
Manufacturer	0x1021	This attribute is recommended, but not required.
Model Name	0x1023	This attribute is recommended, but not required.
Model Number	0x1024	This attribute is recommended, but not required.
Serial Number	0x1042	This attribute is recommended, but not required.
Primary Device Type	0x1054	This attribute is recommended, but not required.

6.6.5 E2

The E2 association frame is sent by a device acting as a registrar in response to an E1 frame. The required attributes in an E2 association frame are listed in Table 17.

Table 17 — E2 association frame attributes

Attribute Name	Attribute Type	Notes
WLP Version	0x2000	The Attribute Value is set to 0x10.
Message Type	0x1022	The Attribute Value is set to 33.
Registrar Nonce	0x1039	

Attribute Name	Attribute Type	Notes
UUID-R	0x1048	
Public Key	0x1032	The Attribute Value is set to PK_r, the registrar's Diffie-Hellman public key.
Device Password ID	0x1012	
Selected Association Method	0x200A	The Attribute Value is set to the association method selected by the registrar.
Device Name	0x1011	
Manufacturer	0x1021	This attribute is recommended, but not required.
Model Name	0x1023	This attribute is recommended, but not required.
Model Number	0x1024	This attribute is recommended, but not required.
Serial Number	0x1042	This attribute is recommended, but not required.
Primary Device Type	0x1054	This attribute is recommended, but not required.

6.6.6 M1

The M1 association frame is sent by a device acting as an enrollee, in response to an E2 frame, to continue an enrollment session. The required attributes in an M1 association frame are listed in Table 18.

Table 18 — M1 association frame attributes

Attribute Name	Attribute Type	Notes
WLP Version	0x2000	The Attribute Value is set to 0x10.
Message Type	0x1022	The Attribute Value is set to 4.
Enrollee Nonce	0x101A	The Attribute Value is set to N_e, the enrollee's nonce used in setting the Enrollee Hash Commitment attribute value sent in the preceding E1 association frame of this enrollment session.
Registrar Nonce	0x1039	The Attribute Value is set to the Registrar Nonce attribute value in the E2 association frame of this enrollment session.
UUID-E	0x1047	
Public Key	0x1032	The Attribute Value is set to PK_e, the enrollee's Diffie-Hellman public key used in setting the Enrollee Hash Commitment attribute value sent in the preceding E1 association frame of this enrollment session.
Device Password ID	0x1012	

6.6.7 M2

The M2 association frame is sent by a device acting as a registrar, in response to an M1 association frame. The required attributes in an M2 association frame are listed in Table 19.

Table 19 — M2 association frame attributes

Attribute Name	Attribute Type	Notes
WLP Version	0x2000	The Attribute Value is set to 0x10.
Message Type	0x1022	The Attribute Value is set to 5.
Enrollee Nonce	0x101A	The Attribute Value is set to the Enrollee Nonce attribute value in the M1 association frame of this enrollment session.
Registrar Nonce	0x1039	The Attribute Value is set to the Registrar Nonce attribute value in the E2 association frame of this enrollment session.
UUID-R	0x1048	
Authenticator	0x1005	The Attribute Value is set to the first 64 bits of HMAC-SHA-256$_{AuthKey}$(M1 || M2*)[7], where M1 is the attribute fields of the M1 association frame, M2* is the attribute fields of this frame excluding this attribute field, and AuthKey is calculated as specified in 7.2.4.3. This must be the last attribute in the M2 association frame.

6.6.8 M3

The M3 association frame is sent by a device acting as an enrollee, in response to an M2 association frame. The required attributes in an M3 association frame are listed in Table 20.

Table 20 — M3 association frame attributes

Attribute Name	Attribute Type	Notes
WLP Version	0x2000	The Attribute Value is set to 0x10.
Message Type	0x1022	The Attribute Value is set to 7.
Registrar Nonce	0x1039	The Attribute Value is set to the Registrar Nonce attribute value in the E2 association frame of this enrollment session.
E-Hash1	0x1014	
E-Hash2	0x1015	
Authenticator	0x1005	The Attribute Value is set to the first 64 bits of HMAC-SHA-256$_{AuthKey}$(M2 || M3*), where M2 is the attribute fields of the M2 association frame, M3* is the attribute fields of this frame excluding this attribute field, and AuthKey is calculated as in the M2 association frame. This must be the last attribute in the M3 association frame.

6.6.9 M4

The M4 association frame is sent by a device acting as a registrar, in response to an M3 association frame. The required attributes in an M4 association frame are listed in Table 21.

[7] HMAC-SHA-256$_{Key}$(Data) is defined in A.3.

Table 21 — M4 association frame attributes

Attribute Name	Attribute Type	Notes
WLP Version	0x2000	The Attribute Value is set to 0x10.
Message Type	0x1022	The Attribute Value field is set to 8.
Enrollee Nonce	0x101A	The Attribute Value field is set to the Enrollee Nonce attribute value in the M1 association frame of this enrollment session.
R-Hash1	0x103D	
R-Hash2	0x103E	
Encrypted Settings	0x1018	The Attribute Value field is set to IV || AES-Encrypt-CBC$_{KeyWrapKey,IV}$(S1r || pad)[8], where IV is a random 16-octet string freshly generated for this frame, S1$_r$ is a field containing a set of attributes, pad is set as defined in Table 6, and KeyWrapKey is calculated as specified in 7.2.4.3. The required attributes in the S1$_r$ field are listed in Table 22.
Authenticator	0x1005	The Attribute Value is set to the first 64 bits of HMAC-SHA-256$_{AuthKey}$(M3 || M4*), where M3 is the attribute fields of the M3 association frame, M4* is the attribute fields of this frame excluding this attribute field, and AuthKey is calculated as in the M2 association frame. This must be the last attribute in the M4 association frame.

Table 22 — S1$_r$ field attributes

Attribute Name	Attribute Type	Notes
R-SNonce1	0x103F	
Key Wrap Authenticator	0x101E	The Attribute Value is set to the first 64 bits of HMAC-SHA-256$_{AuthKey}$(S1$_r$*), where S1$_r$* is the attribute fields in the S1$_r$ field excluding this attribute field, and AuthKey is calculated as in the M2 association frame. This must be the last attribute in the S1$_r$ field.

6.6.10 M5

The M5 association frame is sent by a device acting as an enrollee, in response to an M4 association frame. The required attributes in an M5 association frame are listed in Table 23.

Table 23 — M5 association frame attributes

Attribute Name	Attribute Type	Notes
WLP Version	0x2000	The Attribute Value is set to 0x10.

[8] AES-Encrypt-CBC$_{Key,IV}$(Data) indicates 128-bit AES encryption in CBC mode [B6] with Key as the encryption key and IV as the 128-bit initialization vector.

Attribute Name	Attribute Type	Notes
Message Type	0x1022	The Attribute Value field is set to 9.
Registrar Nonce	0x1039	The Attribute Value field is set to the Registrar Nonce attribute value in the E2 association frame of this enrollment session.
Encrypted Settings	0x1018	The Attribute Value field is set to IV || AES-Encrypt-CBC$_{KeyWrapKey,IV}$($S1_e$), where IV is a random 16-octet string freshly generated for this frame, $S1_e$ is a field containing a set of attributes, pad is set as defined in Table 6, and KeyWrapKey is calculated as specified in 7.2.4.3. The required attributes in the $S1_e$ field are listed in Table 24.
Authenticator	0x1005	The Attribute Value is set to the first 64 bits of HMAC-SHA-256$_{AuthKey}$(M4 || M5*), where M4 is the attribute fields of the M4 association frame, M5* is the attribute fields of this frame excluding this attribute field, and AuthKey is calculated as in the M2 association frame. This must be the last attribute in the M5 association frame.

Table 24 — $S1_e$ field attributes

Attribute Name	Attribute Type	Notes
E-SNonce1	0x1016	
Key Wrap Authenticator	0x101E	The Attribute Value is set to the first 64 bits of HMAC-SHA-256$_{AuthKey}$($S1_e$*), where S1e* is the attribute fields in the S1e field excluding this attribute field, and AuthKey is calculated as in the M2 association frame. This must be the last attribute in the $S1_e$ field.

6.6.11 M6

The M6 association frame is sent by a device acting as a registrar, in response to an M5 association frame. The required attributes in an M6 association frame are listed in Table 25.

Table 25 — M6 association frame attributes

Attribute Name	Attribute Type	Notes
WLP Version	0x2000	The Attribute Value is set to 0x10.
Message Type	0x1022	The Attribute Value field is set to 10.
Enrollee Nonce	0x101A	The Attribute Value field is set to the Enrollee Nonce attribute value in the M1 association frame of this enrollment session.
Encrypted Settings	0x1018	The Attribute Value field is set to IV || AES-Encrypt-CBC$_{KeyWrapKey,IV}$($S2_r$), where IV is a random 16-octet string freshly generated for this frame, $S2_r$ is a field containing a set of attributes, pad is set as defined in Table 6, and KeyWrapKey is calculated as specified in 7.2.4.3. The required attributes in the $S2_r$ field are listed in Table 26.

Attribute Name	Attribute Type	Notes
Authenticator	0x1005	The Attribute Value is set to the first 64 bits of $HMAC\text{-}SHA\text{-}256_{AuthKey}$(M5 || M6*), where M5 is the attribute fields of the M5 association frame, M6* is the attribute fields of this frame excluding this attribute field, and AuthKey is calculated as in the M2 association frame. This must be the last attribute in the M6 association frame.

Table 26 — $S2_r$ field attributes

Attribute Name	Attribute Type	Notes
R-SNonce2	0x1040	
Key Wrap Authenticator	0x101E	The Attribute Value is set to the first 64 bits of $HMAC\text{-}SHA\text{-}256_{AuthKey}$($S2_r$*), where $S2_r$* is the attribute fields in the $S2_r$ field excluding this attribute field, and AuthKey is calculated as in the M2 association frame. This must be the last attribute in the $S2_r$ field.

6.6.12 M7

The M7 association frame is sent by a device acting as an enrollee, in response to an M6 association frame, or an M2 association frame if the association method is numeric comparison. The required attributes in an M7 association frame are listed in Table 27.

Table 27 — M7 association frame attributes

Attribute Name	Attribute Type	Notes
WLP Version	0x2000	The Attribute Value is set to 0x10.
Message Type	0x1022	The Attribute Value field is set to 11.
Registrar Nonce	0x1039	The Attribute Value field is set to the Registrar Nonce attribute value in the E2 association frame for this enrollment session.
Encrypted Settings	0x1018	This attribute is not included when the Numeric Comparison association method is used for the enrollment session. If required, the Attribute Value field is set to IV || $AES\text{-}Encrypt\text{-}CBC_{KeyWrapKey,IV}$($S2_e$), where IV is a random 16-octet string freshly generated for this frame, $S2_e$ is a field containing a set of attributes, pad is set as defined in Table 6, and KeyWrapKey is calculated as specified in 7.2.4.3. The required attributes in the $S2_e$ field are listed in Table 28.
Authenticator	0x1005	If the association method is not Numeric Comparison, the Attribute Value is set to the first 64 bits of $HMAC\text{-}SHA\text{-}256_{AuthKey}$(M6 || M7*), where M6 is the attribute fields of the M6 association frame, M7* is the attribute fields of this frame excluding this attribute field, and AuthKey is calculated as in the M2 association frame. If the association method is Numeric Comparison, the Attribute Value is set to the first 64 bits of $HMAC\text{-}SHA\text{-}256_{AuthKey}$(M2 || M7*), where M2 is the attribute fields of the M2 association frame, and M7* and AuthKey are the same as the previous case. This must be the last attribute in the M7 association frame.

Table 28 — $S2_e$ field attributes

Attribute Name	Attribute Type	Notes
E-SNonce2	0x1017	
Key Wrap Authenticator	0x101E	The Attribute Value is set to the first 64 bits of $HMAC\text{-}SHA\text{-}256_{AuthKey}(S2_e^*)$, where $S2_e^*$ is the attribute fields in the $S2_e$ field excluding this attribute field, and AuthKey is calculated as in the M2 association frame. This must be the last attribute in the $S2_e$ field.

6.6.13 M8

The M8 association frame is sent by a device acting as a registrar, in response to an M7 association frame. The required attributes in an M8 association frame are listed in Table 29.

Table 29 — M8 association frame attributes

Attribute Name	Attribute Type	Notes
WLP Version	0x2000	The Attribute Value is set to 0x10.
Message Type	0x1022	The Attribute Value field is set to 12.
Enrollee Nonce	0x101A	The Attribute Value field is set to the Enrollee Nonce attribute value in the M1 association frame of this enrollment session.
Encrypted Settings	0x1018	The Attribute Value field is set to IV || $AES\text{-}Encrypt\text{-}CBC_{KeyWrapKey,IV}$(WSS Properties), where IV is a random 16-octet string freshly generated for this frame, WSS Properties is a field containing a set of attributes, pad is set as defined in Table 6, and KeyWrapKey is calculated as specified in 7.2.4.3. The required attributes in the WSS Properties field are listed in Table 30.
Authenticator	0x1005	The Attribute Value is set to the first 64 bits of $HMAC\text{-}SHA\text{-}256_{AuthKey}$(M7 || M8*), where M7 is the attribute fields of the M7 association frame, M8* is the attribute fields of this frame excluding this attribute field, and AuthKey is calculated as in the M2 association frame. This must be the last attribute in the M8 association frame.

Table 30 — WSS Properties field attributes

Attribute Name	Attribute Type	Notes
WSSID	0x2001	
WSS Name	0x2002	
WSS Broadcast Address	0x2004	
WSS Master Key	0x2005	
Key Wrap Authenticator	0x101E	The Attribute Value is set to the first 64 bits of $HMAC\text{-}SHA\text{-}256_{AuthKey}$(WSS Properties*), where WSS Properties* is the contents of the WSS Properties field excluding this attribute, and AuthKey is calculated as in the M2 association frame. This must be the last attribute in the WSS Properties field.

6.6.14 C1

The C1 association frame is sent by a device to check the WSS of a neighbor. The required attributes in a C1 association frame are listed in Table 31.

Table 31 — C1 association frame attributes

Attribute Name	Attribute Type	Notes
WLP Version	0x2000	The Attribute Value is set to 0x10.
Message Type	0x1022	The Attribute Value field is set to 34.
WSSID	0x2001	

6.6.15 C2

The C2 association frame is sent by a device in response to a C1 association frame. The required attributes in a C2 association frame are listed in Table 32.

Table 32 — C2 association frame attributes

Attribute Name	Attribute Type	Notes
WLP Version	0x2000	The Attribute Value is set to 0x10.
Message Type	0x1022	The Attribute Value field is set to 35.
WSSID	0x2001	

6.6.16 C3

The C3 association frame is sent by a device to establish a connection. The required attributes in a C3 association frame are listed in Table 33.

Table 33 — C3 association frame attributes

Attribute Name	Attribute Type	Notes
WLP Version	0x2000	The Attribute Value is set to 0x10.
Message Type	0x1022	The Attribute Value field is set to 36.
WSSID	0x2001	
WSS Tag	0x200C	
WSS Virtual EUI-48	0x200D	

6.6.17 C4

The C4 association frame is sent by a device in response to a C3 association frame. The required attributes in a C4 association frame are listed in Table 34.

Table 34 — C4 association frame attributes

Attribute Name	Attribute Type	Notes
WLP Version	0x2000	The Attribute Value is set to 0x10.
Message Type	0x1022	The Attribute Value field is set to 37.
WSSID	0x2001	
WSS Tag	0x200C	
WSS Virtual EUI-48	0x200D	

6.6.18 F0

The F0 association frame is sent by a device in response to an association frame to indicate a failure and stop the enrollment session or connection before completion. The required attributes in an F0 association frame are listed in Table 35.

Table 35 — F0 association frame attributes

Attribute Name	Attribute Type	Notes
WLP Version	0x2000	The Attribute Value field is set to 0x10.
Message Type	0x1022	The Attribute Value field is set to 14.
Enrollee Nonce	0x101A	The Attribute Value field is set to the Enrollee Nonce attribute value in the M1 association frame, or zero if the M1 association frame was not received.
Registrar Nonce	0x1039	The Attribute Value field is set to the Registrar Nonce attribute value in the E2 association frame, or zero if the E2 association frame was not received.
WLP Association Error	0x200E	

6.7 WLP IE

The WLP IE is included in beacons by all devices. It provides information about the device and its participation in WSSs. The format of the IE is illustrated in Figure 28.

octets: 1	1	2	2	2	0 or 10	M	N
Element ID (=250)	Length (=6+(0 or 10)+M+N)	Capabilities	Cycle Parameters	ACW/ AnchorAddr	Bridge Information	WSSID Hash List	Broadcast Traffic Indications

Figure 28 —WLP IE format

The Element ID field for the WLP IE is set to 250.

The Length field is set to the total number of octets of the fields that follow the Length field in this IE.

6.7.1 Capabilities field

The Capabilities field format is illustrated in Figure 29.

bits: b15–b12	b11–b8	b7–b5	b4	b3	b2	b1	b0
WSSID Hash List Length	Broadcast Traffic Indications Count	Reserved	Discoverable	DRP Establishment	Remote Bridge	Client Bridge	Client Device

Figure 29 — Capabilities field format

The WSSID Hash List Length field is set to the number of WSSID Hash values included in the WSSID Hash List field.

The Broadcast Traffic Indications Count field is set to the number of Traffic Indication fields in the Broadcast Traffic Indications field.

The Discoverable bit is set to one if the device has been enabled by the user to accept a new enrollment, or is set to zero otherwise.

The DRP Establishment bit is set to one if the device is a bridge and is capable of accepting a DRP reservation request from a client device, or is set to zero otherwise.

The Remote Bridge bit is set to one if the device can act as one side of a remote bridge pair, connecting the segments attached to the remote bridges. It is set to zero otherwise.

The Client Bridge bit is set to one if the device can provide bridge services to client devices, as defined in 7.4, or is set to zero otherwise.

The Client Device bit is set to one if the device can act as a client device. It is set to zero otherwise.

6.7.2 Cycle Parameters and ACW/AnchorAddr fields

The Cycle Parameters field format is illustrated in Figure 30.

bits: b15	b14–b11	b10–b0
Selecting Anchor	Local Cycle Index	Global Cycle Start Countdown

Figure 30 — Cycle Parameters field format

The Selecting Anchor bit is set to one if the device is negotiating the hibernation anchor role in the current cycle, or is set to zero otherwise.

The Local Cycle Index field indicates the current local cycle length used by the device. The local cycle is $2^{\text{Local Cycle Index}}$ superframes.

The Global Cycle Start Countdown (GCSC) field is a countdown that is set to the number of superframes remaining before the device starts a new global cycle. If the GCSC field is zero, the device will start a new global cycle in the next superframe.

If the Selecting Anchor bit is set to one, the ACW/AnchorAddr field is set to the anchor cycle weight (ACW) of the device, which is the number of anchor cycles (ACs) the device has not been an anchor, capped at 255.

If the Selecting Anchor bit is set to zero, the ACW/AnchorAddr field is set to the AnchorAddr, which is the DevAddr of the anchor selected by the device.

6.7.3 Bridge Information field

If either the Remote Bridge bit or the Client Bridge bit is set to one, the Bridge Information field is formatted as illustrated in Figure 31. If neither bit is set to one, the Bridge Information field is zero length.

octets: 1	1	8
Load Metric	Remaining Capacity	Local Segment ID (LSID)

Figure 31 — Bridge Information field format

The Load Metric field is set to a value that indicates the number of MASs potentially available for additional WLP traffic per superframe, averaged over the last wBridgeActivityWindow superframes. A value of zero indicates that no additional MASs are available.

The Remaining Capacity field is illustrated in Figure 32.

bits: b7	b6	b5–b0
Reserved	Remote Bridge	Additional Clients

Figure 32 — Remaining Capacity field format

The Remote Bridge bit is set to one if the bridge can accept a bridge services request from another remote bridge. It is set to zero otherwise.

The Additional Clients field is set to the number of additional client devices that the bridge can serve.

The Local Segment ID field is set to the Root Identifier of the 802.1D [B3] MAC bridge spanning tree protocol running at the bridge, encoded as an octet string.

6.7.4 WSSID Hash List fields

The WSSID Hash List field contains zero or more WSSID hash values that indicate which WSSs a device has activated. A WSSID hash value is one octet. If the device has not activated any WSSs, the field is zero length. WSSID hash values are transmitted in ascending order. If the device has activated multiple WSSs with the same WSSID hash, the WSSID hash value is included only once in the list.

6.7.5 Broadcast Traffic Indications field

The Broadcast Traffic Indications field contains zero or more Traffic Indication fields, an example of which is illustrated in Figure 33. Each Traffic Indication field informs neighbors of the device of its intent to transmit frames carrying WLP broadcast or multicast traffic. A WLP device includes one Traffic Indication field for each WSS to which it intends to transmit broadcast or multicast frames in the current superframe.

The format of a Traffic Indication field is illustrated in Figure 33.

octets: 1	1	L
WSS Tag	MAS List Length (=L)	MAS List

Figure 33 — Traffic Indication field format

The WSS Tag field identifies the WSS to which the device intends to transmit multicast or broadcast traffic.

The MAS List Length field is set to the number of MAS numbers included in the MAS List field.

The MAS List field contains one or more MAS numbers in the current superframe in which the device intends to send broadcast or multicast traffic. A MAS number is one octet. The MAS numbers are transmitted in increasing value order.

7. Functional description

This clause specifies functionality of the WiMedia logical link control protocol.

The association process, including device discovery, WSS enrollment and activation, and connection to a neighbor, is described in 7.2. General rules for frame transfer that apply to client devices and bridges are described in 7.3. Rules specific to bridges are described in 7.4. Power management features and rules for all devices that permit efficient use of power for mobile devices are described in 7.5. Quality of service requirements are described in 7.6.

7.1 General requirements

A device shall construct all frames and IEs as defined in clause 6. In association frames, a device shall include all the attributes listed in the attributes tables. It shall include the attributes in the order listed and shall place the attributes in the list before any other attributes, except as noted. If a device receives an attribute with an Attribute Type not defined in this specification, it may ignore that attribute.

A device that generates or receives data frames is referred to as a client device. A client device shall always include a WLP IE, as shown in Figure 28, in its beacon. It shall set the Client Device bit in the Capabilities field in the IE to one.

A device that is capable of and ready to forward data to and from its neighbors is referred to as a bridge. It shall always include a WLP IE in its beacon. It shall set the Client Bridge bit, the Remote Bridge bit, or both bits in the Capabilities field of the IE to one.

A device shall update its WLP IE field values each time it transmits a beacon.

7.2 Association

This subclause specifies rules to permit a device to discover neighbors, enroll in WSSs, activate WSSs, and connect to neighbors that have activated a WSS in common with the device.

A device shall send association frames to unicast addresses only.

A device shall use the Imm-ACK acknowledgment policy at the MAC sublayer to send association frames. Subject to timeout rules in this subclause, a device shall retransmit any association frame for which an Imm-ACK frame was not received.

If a response is required, a registrar or enrollee shall respond to an association frame within wResponseTimeout seconds of receiving the association frame. If the response depends on a user action, the device may respond with an F0 frame with the WLP Association Error attribute value set to Waiting For User Input (14) to restart the timeout interval. A device may repeat such a response multiple times, but shall send either the F0 frame or the response frame within wResponseTimeout seconds of the preceding frame in each case. On receipt of an F0 frame with the WLP Association Error attribute value set to Waiting for User Input, a device shall restart its wPerMessageTimeout interval as if it just sent the preceding association frame.

A device shall set the WLP Version attribute value in all association frames to 0x10. If a device receives an association frame with a WLP Version attribute value other than 0x10, it shall respond with an F0 frame with the Association Error attribute value set to Unsupported Version (10).

7.2.1 WLP service set (WSS)

A WSS provides a context for exchange of frames between devices. The properties that define a WSS are described in Table 36. Any device may become the founder of a new WSS by establishing a new set of WSS properties.

Table 36 — WSS properties

Property	Size	Description
WLP service set identifier (WSSID)	16 octets	A globally unique value that identifies the set
WSSID hash	1 octet	A hash of the WSSID used in the WLP IE to identify activated WSSs
WSS name	variable, 0–64 octets	A text string that identifies the WSS, for display to a device user
WSS broadcast address	6 octets	A multicast EUI-48 used to address broadcast traffic within the WSS
WSS secure status	1 octet	One if the WSS is secure or zero if the WSS is non-secure
WSS master key	16 octets	A master key known by all WSS members, used in the 4-way handshake described in the WiMedia MAC specification [B4]. Only present for a secure WSS.

The WSS founder shall select a UUID to use for the WSSID. It shall select a version 1 UUID based on a global IEEE 802 EUI-48, as defined in RFC 4122 [B5]. The WSS founder should select a WSSID such that the WSSID hash is not currently in use within the device's neighborhood.

The WSSID hash for a WSSID is the result of an octet-wise exclusive-OR of all octets in the WSSID, and can be computed by any device aware of the WSSID.

The WSS founder shall set the WSS name to a text string suitable for display to a user.

The WSS founder shall select a multicast EUI-48 for the WSS broadcast address. The WSS founder shall select a multicast EUI-48 [B10], based on the WiMedia Alliance OUI, 00-13-88, within the WLP range, [01-13-88-00-01-00, 01-13-88-00-01-FF], inclusive.

The WSS founder shall set the WSS secure status to one unless the device user has taken a specific action to allow the device to create a non-secure WSS.

The WSS founder shall select a fresh WSS master key according to the criteria for selecting a cryptographic grade random number in A.6.

7.2.2 WSS local properties

In addition to the WSS properties, a device maintains a set of device-specific properties related to an activated WSS. These WSS local properties are defined in Table 37.

Table 37 — WSS local properties

Property	Size	Description
WSS tag	1 octet	A value selected by the device to identify the WSS context in transmitted frames
WSS virtual EUI-48	6 octets	A unicast EUI-48 used as by the device as its virtual local address for communication within the WSS

The WSS tag is selected by the device when it activates the WSS, as defined in 7.2.5. It uniquely identifies the WSS for frames the device transmits.

The WSS virtual EUI-48 is selected by the device when it activates the WSS. The device may select its WiMedia MAC EUI-48 for use as a WSS virtual EUI-48. A device shall not register for bridge services for a WSS if the WSS virtual EUI-48 is used for any other WSS.

7.2.3 WSS discovery

A device can become enrolled in a WSS in one of two ways. It can create a new WSS with new properties, in which case it is automatically enrolled in the WSS, but can only communicate with other devices that subsequently enroll in the WSS. It can also enroll in an existing WSS established by another device, in which case it can communicate with that device and any other devices also enrolled in that WSS. In order to enroll in an existing WSS, a device must first discover the existence of another device accepting enrollment for that WSS.

Actual discovery mechanisms are outside the scope of this standard, but could include a scan of various PHY channels for available and activated WSSs or out of band knowledge gained through an alternative communication mechanism. A device may use the D1 and D2 frame exchange to discover information about a WSS advertised in a neighbor's WLP IE.

During discovery and a subsequent enrollment session, a device that is already enrolled in an existing WSS is referred to as a registrar, and a device seeking to enroll in the WSS is referred to as an enrollee. These roles are temporary and last only for the duration of the enrollment session.

A device shall be capable of acting as a registrar. A device shall be capable of acting as an enrollee.

To check the WSS properties of a WSS activated by a neighbor, a device shall send a D1 association frame to the neighbor. A device shall not send a D1 frame to a neighbor unless the Discoverable bit is set to one in the latest WLP IE received from the neighbor.

A device that receives a D1 association frame shall respond with a D2 association frame that contains device information and WSS information, or an F0 association frame that indicates why the discovery request is not accepted or WSS information is not available. A registrar may respond with a D2 frame that includes partial information and a non-zero error code. A registrar should not send a D2 frame with a non-zero error code if the frame contains WSS information with the Accepting Enrollment attribute value set to one. If an enrollee receives a D2 or F0 frame with the Association Error attribute value set to a non-zero value, it shall not respond with an E1 frame, and shall not send another D1 frame to the same registrar for at least wMinDiscoveryRepeatTime.

In a D2 frame, a registrar may respond with zero or more WSS Information fields. If the WSS Selection Method attribute value in the preceding D1 frame was set to Registrar Selects, the registrar shall not set the Accepting Enrollment attribute value to one in more than one WSS Information field in the D2 frame.

If a D2 association frame is received, an enrollee has enough information to determine if the WSS is secure or non-secure and if it is already enrolled in the WSS or not. If the WSS is non-secure, the enrollee may retain the WSS properties, at which point it is enrolled in the WSS. An enrollee shall not retain the WSS properties to enroll in a non-secure WSS unless the device user has taken a specific action to allow the device to enroll in a non-secure WSS. If the WSS is secure, enrollment requires additional frame exchanges as defined in 7.2.4.

7.2.4 Secure WSS enrollment

After receiving a D2 association frame from a registrar that indicates a secure WSS and does not indicate an error, an enrollee may initiate an enrollment session by sending an E1 association frame, defined in 6.6.4, to the registrar. Except as indicated below, the registrar and enrollee shall each alternately respond with the next association frame in sequence, E2, then the M1 through the M8 association frames, or shall respond with an F0 association frame that indicates the reason for stopping the enrollment session in the WLP Association Error attribute. On receipt of the M8 frame, the enrollee may retain the WSS master key, along with other public WSS properties, which constitutes enrollment in the WSS.

Certain association frames are identified by nonces and authenticator attributes. If a frame is received with either a non-matching nonce or an invalid authenticator attribute, the recipient shall not respond to the frame (except as required by the ACK policy indicated in the frame).

If the sender of an association frame does not receive an expected response within wPerMessageTimeout seconds of successfully sending the association frame, the sender shall stop the enrollment session. If the sender of an association frame does not receive a response within wEnrollmentTimeout seconds of sending or receiving the E1 association frame, the sender shall also stop the enrollment session. If an enrollment session for a secure WSS is stopped prior to the transmission or reception of M8, a device shall discard all state information corresponding to the enrollment session, except any error logs that may be kept. It also shall send an F0 association frame to the other device. If the most-recent association frame received was ignored because of a non-matching nonce or invalid authenticator attribute, the device shall set the WLP Association Error attribute value to Rogue Activity Suspected (2). Otherwise, it shall indicate the nature of the timeout reached.

7.2.4.1 Association method

In an E1 association frame, an enrollee shall include an Association Methods List attribute that contains an ordered list of association methods that it supports. Each association method in the list is encoded as a value from Table 11. The Association Methods List attribute value field is encoded as a sequence of 16-bit fields, where the first field represents the most preferred association method, and the last represents the least preferred.

This list is either set by the manufacturer or managed by the user. The maximum length of the list is 16 entries. An enrollee shall support the Numeric Comparison association method and shall include Numeric Comparison (0x0200) in this list.

In an E2 association frame, a registrar shall include a Selected Association Method attribute that contains a single association method to be used for the enrollment session. A registrar should select the first association method that it supports from the list provided by the enrollee. A registrar shall support the Numeric Comparison association method.

The following subclauses define specific requirements for each association method.

7.2.4.1.1 Display association methods

The following specific requirements apply if the registrar and enrollee agree to use the Registrar-display or Enrollee-display association method. These association methods use a numeric value generated by one device and entered into the other to confirm enrollment. For the Registrar-display association method, the registrar shall generate and display a numeric value that is entered into the enrollee. For the Enrollee-display association method, the enrollee shall generate and display a numeric value that is entered into the registrar. The numeric value is referred to as the Display Value.

The M1 through M8 association frames are exchanged using the Display Value as the Device Password for authentication of DHKey. If a device will display the Device Password, it shall do so at any time prior to sending the M2 or M3 frame. If a device will query the user for the Device Password, it shall do so at any time prior to sending the M3 or M4 frame.

The Display Value is either 4 or 8 numeric digits (0–9). The Display Value is treated as an octet string, where each octet contains the UTF-8 representation of the digit (0x30–0x39). The first octet contains the left-most digit on the display, and successive octets contain successive digits.

A device should generate a new Display Value for each enrollment session, although it may re-use the same value from an enrollment session that completed successfully. If an enrollment session does not complete successfully, a device shall generate a new Display Value for any future sessions.

If the Display Value is 8 digits, a device shall generate the Display Value such that the following condition is met, where d_1 is the left-most digit:

$$(3\times d_1 + d_2 + 3\times d_3 + d_4 + 3\times d_5 + d_6 + 3\times d_7 + d_8) \bmod 10 = 0$$

When a Display Value is entered by a user, a device shall check the number to see if it is 4 or 8 digits, and if the previous condition is met for an 8-digit number. If not, the device shall not transmit an association frame that uses that number, and should indicate to the user that the number is invalid and allow re-entry.

7.2.4.1.2 User-provided Password association method

The following specific requirements apply if the registrar and enrollee agree to use the User-provided Password association method. This association method uses a password entered by the user into both the registrar and enrollee to authenticate enrollment.

The registrar and enrollee shall exchange the M1 through M8 association frames using the password entered by the user, encoded in UTF-8 format, as the Device Password for authentication of DHKey. A device shall query the user for the password to use at any time prior to sending the M3 or M4 frame.

A device should not repeat use of a password if it was used in a recent enrollment session that did not complete successfully. Such a password is not secure as it could be discovered by an attacker through a man-in-the-middle attack.

7.2.4.1.3 Numeric Comparison association method

The following specific requirements apply if the registrar and enrollee agree to use the Numeric Comparison association method:

After sending M1, the enrollee shall compute and display the numeric comparison value as described in A.7.

After receiving M1, the registrar shall compute

$$\text{HashCommitCheck} = \text{SHA-256}(PK_e \,||\, N_e)$$

using the attribute values provided by the enrollee in M1. If HashCommitCheck does not equal the value in the Enrollee Hash Commitment attribute value in the previous E1 association frame, the registrar shall send an F0 association frame with the WLP Association Error attribute value set to Rogue Activity Suspected (2) and shall abort the enrollment session.

If the hash commitment values match, the registrar shall display the numeric comparison value as described in A.7. The registrar shall not send the M2 association frame unless the user indicates to the registrar that the numbers match. If the user does not indicate that the numbers match, the registrar shall send an F0 association frame with the WLP Association Error attribute value set to Numeric Comparison Failure (13).

On receipt of an M2 association frame, if the user indicates to the enrollee that the numbers match, it shall send an M7 association frame to the registrar. The enrollee shall not respond with an M3 frame. The enrollee shall not include an Encrypted Settings attribute in the M7 frame. If the user does not indicate that the numbers match, the enrollee shall send an F0 association frame with the WLP Association Error attribute value set to Numeric Comparison Failure (13).

7.2.4.2 DHKey generation

A registrar shall generate DHKey as follows, where PK_e is the enrollee's public key contained in M1, R is the registrar's private key, and p is a prime number defined in A.6.

$$DHKey_{registrar} = \text{SHA-256}(PK_e^{\,R} \bmod p)$$

An enrollee shall generate DHKey as follows, where PK_r is the registrar's public key contained in M2, E is the enrollee's private key, and p is a constant defined in A.6.

$$DHKey_{enrollee} = \text{SHA-256}(PK_r^{\,E} \bmod p)$$

Both $PK_e^{\,R} \bmod p$ and $PK_r^{\,E} \bmod p$ have the same number of bits as p, with as many leading zero bits included as needed.

If the public and private keys are constructed and exchanged properly, $DHKey_{registrar}$ and $DHKey_{enrollee}$ will be the same value. The value is referred to as DHKey in this specification.

7.2.4.3 AuthKey and KeyWrapKey generation

The keys used in authentication and encryption, AuthKey and KeyWrapKey, respectively, are derived from a key deriving key (KDK).

A device shall determine KDK for an enrollment session as:

$$KDK = HMAC\text{-}SHA\text{-}256_{DHKey}(N_e \,||\, N_r)$$

DHKey is defined in 7.2.4.2; N_e and N_r are the enrollee's nonce and registrar's nonce contained in M1 and E2, respectively.

A device shall determine AuthKey for an enrollment session as:

$$AuthKey = HMAC\text{-}SHA\text{-}256_{KDK}(0x00000001 \,||\, \text{“WLP 1.0”} \,||\, 0x00000180)$$

Similarly, a device shall determine KeyWrapKey for an enrollment session as:

$$KeyWrapKey = \text{first 128 bits of } HMAC\text{-}SHA\text{-}256_{KDK}(0x00000002 \,||\, \text{“WLP 1.0”} \,||\, 0x00000180)$$

7.2.4.4 Hash derivation

The enrollee and registrar hash values are derived from the Device Password hashed using AuthKey, which is generated as specified in 7.2.4.3.

A device shall derive two 128-bit PSK values from the Device Password as follows:

$$PSK_1 = \text{first 128 bits of } HMAC\text{-}SHA\text{-}256_{AuthKey}(1^{st} \text{ half of Device Password})$$
$$PSK_2 = \text{first 128 bits of } HMAC\text{-}SHA\text{-}256_{AuthKey}(2^{nd} \text{ half of Device Password})$$

If the Device Password is an odd length, the 1^{st} half of Device Password will have one more octet than the 2^{nd} half of Device Password.

The Enrollee shall create two 128-bit random numbers for its secret nonces, $E\text{-}S_1$ and $E\text{-}S_2$, and shall compute $E\text{-}H_1$ and $E\text{-}H_2$ as follows:

$$E\text{-}H_1 = HMAC\text{-}SHA\text{-}256_{AuthKey}(E\text{-}S_1 \,||\, PSK_1 \,||\, PK_e \,||\, PK_r)$$
$$E\text{-}H_2 = HMAC\text{-}SHA\text{-}256_{AuthKey}(E\text{-}S_2 \,||\, PSK_2 \,||\, PK_e \,||\, PK_r)$$

The Registrar shall create two 128-bit random numbers for its secret nonces, $R\text{-}S_1$ and $R\text{-}S_2$, and shall compute $R\text{-}H_1$ and $R\text{-}H_2$ as follows:

$$R\text{-}H_1 = HMAC\text{-}SHA\text{-}256_{AuthKey}(R\text{-}S_1 \,||\, PSK_1 \,||\, PK_e \,||\, PK_r)$$
$$R\text{-}H_2 = HMAC\text{-}SHA\text{-}256_{AuthKey}(R\text{-}S_2 \,||\, PSK_2 \,||\, PK_e \,||\, PK_r)$$

The enrollee and registrar shall gradually exchange and verify the hash commitment values and the secret nonces as defined in the M3–M7 association frames. To verify a secret nonce, a recipient device shall calculate the corresponding hash commitment based on its own Device Password value and check if it matches the value received in a previous association frame in this enrollment session. If a verification fails, the receiving device shall respond to the last received association frame with an F0 association frame with the WLP Association Error attribute value set to Rogue Activity Suspected (2). The enrollee and registrar shall stop the enrollment session and discard all keys and nonces generated during the session.

7.2.5 WSS activation

In order to enable connection to other devices in a WSS, a device must activate the WSS. Prior to activating a WSS, a device must be enrolled in the WSS. To activate a WSS, a device shall include the WSS hash in the WLP IE in its beacon in each superframe. A device may deactivate a WSS by removing the WSS hash from its beacon. A device may activate multiple WSSs simultaneously.

When a device activates a WSS, it shall select a WSS tag to identify the WSS in frames that it transmits. The device shall not use the same WSS tag value for more than one activated WSS. The device should use the WSSID hash value for the WSS tag. In the event of conflict, the device should use the first unused value higher than the WSSID hash, modulo 256.

A device seeking to determine if a neighbor has activated a particular WSS, based on a WSSID hash included in the neighbor's beacon, may send a C1 association frame to that neighbor. The device shall include a WSSID in the C1 frame that identifies a WSS in which it is enrolled. If a device receives a C1 association frame that identifies a WSS it has activated, it shall respond with a C2 association frame that identifies the same WSS. If a device receives a C1 association frame that identifies a WSS it has not activated, it shall respond with an F0 association frame with WLP Association Error attribute value set to Invalid WSS Selection (6).

7.2.6 Connection

Prior to exchanging data frames with a neighbor within a WSS, a device shall connect to the neighbor. Prior to connecting, both devices must have activated the WSS to be used for the communication context.

A device seeking to establish a connection shall transmit a C3 association frame to a target device. If a device receives a C3 frame that identifies a WSS it has activated, it shall transmit a corresponding C4 association frame. Otherwise, it shall transmit an F0 association frame to indicate the connection failure. In a C3 or C4 frame, a device shall include the WSS tag value selected when the device activated the WSS. It is not necessary for each device to initiate a C3/C4 frame exchange – one exchange provides both devices with the information necessary to form a connection.

If the WSS is secure, a device shall always send a C3 or C4 association frame as a MAC secure frame. This requires that a device use the 4-way handshake as defined in the WiMedia MAC specification [B4] to establish a secure relationship and generate a PTK for use with the target device for this WSS. For this process, the device shall use the WSSID as the MKID and the WSS master key as the master key. A device shall also distribute a GTK for protecting WSS broadcast traffic, as defined in the WiMedia MAC specification. Because the WSS master key is known to all devices enrolled in the WSS, it is possible for any device enrolled in the WSS that receives the frames exchanged in the 4-way handshake to obtain the PTK. Frames protected with the PTK should only be considered secure within the WSS.

A device shall attempt to connect to a neighbor if the following conditions are all met:

- a frame from the WLP client addresses that neighbor, including multicast and broadcast frames; and
- the WSSID hash for the frame's WSS is included in the neighbor's WSSID hash list; and
- the device has not attempted to connect to that neighbor in the frame's WSS since the device last activated that WSS; and
- the device has not attempted to connect to that neighbor in the frame's WSS since the neighbor last added the WSSID hash for that WSS to its WSSID hash list.

A device that has established a connection to a target device in a secure WSS shall transmit MAC secure frames to the target device as defined in Table 72 in the WiMedia MAC specification [B4]. The device shall assume the connection to the target device exists until:

- it receives a WLP IE from the target device that does not contain the WSSID hash of the WSS; or
- it does not receive a beacon from the target device for wConnectionTimeout+1 consecutive superframes.

If one of these events occurs, the device shall no longer consider the target device connected, and shall discard the PTK associated with the target device in order to end the secure relationship.

A device shall use the MAC secure frame format for all control frames and data frames sent with the WSS tag of a secure WSS.

7.3 Frame transfer

A device shall transmit data frames only to neighbors that have activated the same WSS and have an established connection with the device in the WSS as described in 7.3.1 and 7.3.2.

7.3.1 Standard data frames

To transmit a standard data frame in a WSS, a client device shall:

- set the Source Address field to its WSS virtual EUI-48;
- set the Destination Address field to the EUI-48 of the ultimate destination of the frame, which is a WSS virtual EUI-48 if the destination is a WLP device; and
- address the frame at the MAC SAP to:
  - the physical unicast EUI-48 of the destination device if the destination is a neighbor; or
  - the physical unicast EUI-48 of a neighbor bridge with which the device is currently registered for bridge services; or
  - the multicast EUI-48 of the ultimate destination; or
  - the WSS broadcast address if the ultimate destination is the broadcast EUI-48.

To forward a standard data frame within a WSS, a bridge shall:

- set the Source Address field to the EUI-48 of the original source of the frame, which is a WSS virtual EUI-48 if the source is a WLP device;
- set the Destination Address field to the EUI-48 of the ultimate destination of the frame, which is a WSS virtual EUI-48 if the destination is a WLP device; and
- address the frame at the MAC SAP to:
  - the physical unicast EUI-48 of the destination device if the destination is a neighbor; or
  - the physical unicast EUI-48 of a paired remote bridge; or
  - the multicast EUI-48 of the ultimate destination; or
  - the WSS broadcast address if the ultimate destination is the broadcast EUI-48.

On receipt of a standard data frame, a client device shall:

- determine the WSS of the received frame using the WSS tag and the source EUI-48 indicated at the MAC SAP, and discard the frame if it has not activated that WSS or has not connected to the device identified by the source EUI-48 indicated at the MAC SAP; otherwise
- deliver the frame to the WLP client, indicating the values in the Source Address and Destination Address fields as the original source and ultimate destination of the frame, respectively.

On receipt of a standard data frame, a bridge shall:

- determine the WSS of the received frame using the WSS tag and the source EUI-48 indicated at the MAC SAP, and discard the frame if it has not activated that WSS or has not connected to the device identified by the source EUI-48 indicated at the MAC SAP; otherwise
- forward the frame onto other bridge ports, indicating the values in the Source Address and Destination Address fields as the original source and ultimate destination of the frame, respectively.

A device shall discard a frame received from a bridge that is currently not providing bridge services to it if the Destination Address of the frame is a multicast or broadcast EUI-48 and the Source Address is not the bridge's WSS virtual EUI-48.

7.3.2 Abbreviated data frames

A client device shall not transmit an abbreviated data frame unless the ultimate destination is its neighbor. A bridge shall not transmit an abbreviated data frame, except when acting as a client device.

To transmit an abbreviated data frame in a WSS, a client device shall address the frame at the MAC SAP to:

- the physical unicast EUI-48 or multicast EUI-48 of the ultimate destination; or
- the WSS broadcast address if the ultimate destination is the broadcast EUI-48.

On receipt of an abbreviated data frame, a client device shall:

- determine the WSS of the received frame using the WSS tag and the source EUI-48 indicated at the MAC SAP, and discard the frame if it has not activated that WSS or has not connected to the device identified by the source EUI-48 indicated at the MAC SAP; otherwise
- deliver the frame to the WLP client, indicating the WSS virtual EUI-48 of the source and the WSS virtual EUI-48 of the destination as the original source and ultimate destination of the frame, respectively. The device shall determine these values from information received when connecting to the source device, and from the WSS tag, source EUI-48, and destination EUI-48 indicated at the MAC SAP.

7.3.3 Use of PCA

A device shall support the ability to transmit and receive frames using PCA as defined in the WiMedia MAC specification [B4]. A device shall announce availability to receive and acknowledge PCA traffic using a PCA Availability IE. If the device announces availability in any MASs, it shall announce availability in at least one MAS within the first MAS zone, unless all MASs in the zone are in use for a beacon period or non-PCA reservation. A device shall announce availability in all MASs claimed in its PCA reservations, if any.

If there are no available MASs in the first MAS zone, a device should create a PCA reservation with sufficient MASs to send its own multicast and broadcast traffic.

7.3.4 Use of DRP

A device shall support the ability to accept reservations from neighbors as defined in the WiMedia MAC specification [B4]. Unless there is a reservation conflict or other resource constraint, a device shall accept any unicast reservation request it receives from a neighbor with which it is connected.

7.3.5 WLP broadcast and multicast transmission

A device should include a Traffic Indication field in the Broadcast Traffic Indications field of its WLP IE for each WSS to which it intends to transmit broadcast or multicast frames in the current superframe.

If a device is the owner of a reservation that targets the multicast recipients, the Traffic Indication field may identify MASs included in the reservation.

When selecting MASs to indicate in a Traffic Indication field, a device should take into account the availability of its neighbors as declared in their beacons. If there are no MASs in which all potential targets are available, a device shall transmit a frame multiple times such that each potential target is available during at least one of the transmission times.

7.3.6 Maximum transmission unit (MTU)

A device shall not transmit any frame for forwarding with Client Data larger than an MTU of 4048 octets, as derived in Table 38. A bridge shall be able to forward frames with Client Data sizes up to 4048 octets. If a QTag Prefix [B2] is present in the frame, it is considered part of the Client Data.

Table 38 — MTU calculation

mMaxFramePayloadSize	4095 octets
MAC security header and MIC	20 octets
MUX Header	2 octets
WLP standard data frame header	16 octets
Margin for future expansion	9 octets
MTU for Client Data	4048 octets

7.4 Bridge operation

A client device and a bridge shall obtain and provide bridge services for frame forwarding in the context of a specific WSS. A pair of remote bridges shall provide bridge services to each other after one has requested bridge services and the other has responded with an indication of a successful request, also in the context of a specific WSS.

7.4.1 Enabling bridge services

A device may request a bridge to forward frames to or from other nodes by sending a Bridge Services Request control frame to the bridge. Prior to requesting bridge services from a bridge, the device shall establish a connection to the bridge.

If the device requesting bridge services will also forward frames to and from other segments, the device shall indicate this in the Bridge Services Request control frame by setting the Enable Remote Bridge Services bit to one.

A device may also transmit a Bridge Services Request control frame to update protocol or multicast forwarding filters, or to terminate the bridge services requested. Each time a bridge receives a Bridge Services Request control frame from a device, it shall discard any information retained from previous requests from that device, and use only the information contained in the received request.

A device may send a Bridge Services Request control frame to enable bridge serviceswith the Protocol Count field set to zero. If the request is accepted, the recipient bridge shall forward frames from the device to other nodes, but shall not forward frames from other nodes to the device.

A device may request bridge services from multiple bridges.

When a bridge receives a Bridge Services Request control frame, it shall respond with a Bridge Services Response control frame. If the bridge responds to a new request with a Response field that indicates a failure, it shall not provide bridge services for the device. If the bridge responds to an update request with a Response field that indicates a failure, it shall continue to provide bridge services to the device as if no request were received.

A bridge shall only provide bridge services to a device that has established a connection to it in a WSS and shall terminate bridge services to that device if either the bridge or the device deactivates the WSS.

To provide bridge services to a client device, a bridge should forward any frame that addresses the device, including multicast and broadcast frames, unless a filtering rule in the following subclauses indicates otherwise. To provide bridge services to a peer remote bridge, a bridge should forward all frames to the peer unless a filtering rule in the following subclauses indicates otherwise. A remote bridge that receives a bridge services request may also send a bridge services request to the same peer, in order to establish filters in each direction between the two bridges. Each remote bridge shall filter frames sent to its peer as requested by its peer.

7.4.1.1 VLAN and non-VLAN frame forwarding

A device informs a bridge whether to forward VLAN and non-VLAN frames to it from other nodes. VLAN frames are identified by a value in the Type/Length field of 802.1QTagType (0x8100) [B2]. If the device did not enable the VLAN forwarding in the Bridge Services Request control frame, then the bridge shall not forward any VLAN frame to the device. If the device did not enable non-VLAN forwarding in the Bridge Services Request control frame, then the bridge shall not forward any non-VLAN frame to the device.

For VLAN frames, the bridge shall assume the Client Data field format is consistent with that described in 802.3 subclause 3.5 [B2]. If a device enabled VLAN forwarding and included a non-zero length list of VLAN identifiers in the Bridge Services Request control frame, the bridge shall not forward any VLAN frame with a VLAN Identifier not equal to a value in the list.

The bridge shall apply the protocol-specific forwarding rules in 7.4.1.2 and the multicast forwarding rules in 7.4.1.3 to both VLAN and non-VLAN frames.

7.4.1.2 Protocol-specific forwarding

A device informs a bridge which protocol-specific frames to forward to it from other nodes. A device shall indicate the protocols of frames to be forwarded to it in the Protocol Ranges field in a Bridge Services Request control frame.

A bridge shall not forward a frame to a device unless the frame contains a protocol ID that the device included in the Protocol Ranges field when it most-recently registered for bridge services.

If the value in the Type/Length field of a frame is 802.1QTagType, the bridge shall assume the Client Data field format is consistent with that described in 802.3 subclause 3.5 and use the value in the MAC Client Length/Type field as the potential protocol ID. If the value in the Type/Length field of a frame is not 802.1QTagType, the bridge shall use the value in the Type/Length field as the potential protocol ID. If the value of the potential protocol ID is less than 0x5DD, then the bridge shall determine the protocol ID to check as described in IEEE 802.3 [B2] subclause 3.2.6. Otherwise, the bridge shall consider the potential protocol ID to be the protocol ID to check.

7.4.1.3 Multicast forwarding

A device may request a bridge to forward destination address-specific multicast frames to it from other nodes. The device shall indicate the multicast address of the multicast traffic to be forwarded to it in the Multicast Address Ranges field in a Bridge Services Request control frame.

In order to receive broadcast traffic from a bridge, a device should include the WSS broadcast address (a multicast EUI-48) for each WSS that it has activated.

A bridge shall not forward a frame with a multicast destination address unless the destination address was included in the Multicast Address Ranges field by a device for which it is providing bridge services.

7.4.2 Disabling bridge services

A device may request a bridge to disable bridge services by sending a Bridge Services Request control frame to the bridge with the Enable Remote Bridge Services and Enable Client Bridge Services bits set to zero. A bridge that receives such a request shall cease forwarding frames to or from that device. If a remote bridge makes such a request, both the sender and recipient bridges shall cease forwarding frames to or from the other.

A bridge shall maintain bridge services provided for a neighbor even if the bridge does not receive a WLP IE from the neighbor for up to wBridgeServiceTimeout superframes. If a bridge does not receive a WLP IE from a registered neighbor for wBridgeServiceTimeout+1 superframes, it shall terminate bridge services for that neighbor.

7.4.3 DRP reservations

A client device may request a client bridge with which it has enabled bridge services to establish a DRP reservation for traffic addressed to the client device if the bridge indicates support for DRP establishment in its WLP IE. A remote bridge may also make this request of a peer remote bridge, if supported.

The client device or remote bridge shall set the Traffic Filtering Parameters field in a DRP Reservation Request control frame to be consistent with the parameters for bridge services enabled for the device, in terms of requested protocols, multicast addresses, and WSS properties.

A bridge that supports DRP establishment and receives a DRP Reservation Request control frame shall establish a reservation according to the TSPEC field included in the frame, if possible, and report the result in a DRP Reservation Response control frame. If the reservation was successfully established, the bridge shall forward frames in the reservation that meet the filter parameters in the DRP Reservation Request control frame, and no others.

A bridge that supports DRP establishment shall at a minimum support a request with a Filter Set Count equal to one and Filter Length set to 16 octets.

7.4.4 Forwarding rules for bridges

7.4.4.1 Forwarding frames to a device

A bridge shall forward unicast, multicast and broadcast frames to neighbors that have successfully registered with it for bridge services, subject to the following rules:

- The bridge shall not use the abbreviated data frame format.
- The bridge shall set the WSS Tag field in the frame to correspond to the WSS that corresponds to the destination address of the frame.
- The bridge shall forward a unicast frame using WiMedia MAC [B4] unicast services.
- The bridge shall forward a broadcast frame to each neighbor using either the WiMedia MAC unicast services or the WiMedia MAC multicast services with a WSS broadcast address as the destination EUI-48 at the MAC SAP.
- The bridge should use the MAC multicast services to forward a multicast or broadcast frame that must be transmitted to multiple neighbors.
- The bridge shall not use MAC unicast services to forward a frame to a neighbor if the neighbor has not enabled the protocol that is associated with the frame for forwarding, as described in 7.4.1.2.
- The bridge shall not use MAC unicast services to forward a multicast frame to a neighbor if the neighbor has not enabled the multicast address of the frame for forwarding.
- The bridge shall not forward a multicast or broadcast frame unless at least one neighbor has enabled the protocol associated with the frame for forwarding.
- The bridge shall not forward a multicast frame unless at least one neighbor has enabled the multicast address of the frame for forwarding.
- The bridge shall not forward a broadcast frame unless at least one neighbor has enabled the WSS broadcast address (a multicast EUI-48) of the frame for forwarding.
- The bridge shall not forward unicast frames to neighbors that have not registered with it for bridge services.
- The bridge shall store frames prior to forwarding to support power-management mechanisms as specified in 7.5.
- If the bridge supports DRP establishment, it shall use established DRP reservations to forward standard data frames that meet the following criterion: The frame contents at the offset defined in any filter set of the DRP Reservation Request, filtered by the corresponding mask, match the corresponding values.
- The bridge shall pass the DeliveryID received from the MAC SAP in connection with a received frame back to the MAC SAP for forwarding the frame to a neighbor if the DeliveryID value is smaller than eight.

Any neighbor of a bridge might receive a multicast or broadcast frame transmitted by the bridge using MAC multicast services, even though the device has not enabled the protocol or multicast address of the frame for forwarding, or even registered for bridge services.

7.4.4.2 Forwarding frames from a device

A bridge shall not forward any frame that was received from a neighbor for which it is not providing bridge services.

A bridge shall follow these rules when receiving frames from neighbors for which it is providing bridge services:

— The bridge shall not forward frames other than standard data frames.
— The bridge shall not filter frames it receives based on the VLAN, protocol, or address filtering parameters of the bridge services request.

7.4.4.3 Remote bridge requirements

A remote bridge that accepts a request to forward frames from another remote bridge, as indicated by the Enable Remote Bridge Services bit, shall initiate an analysis of the network for loops using the IEEE 802.1D MAC bridge rapid spanning tree protocol [B3]. Each remote bridge shall implement 802.1 bridging, including disabling segments that create loops and associating EUI-48s with segments. A remote bridge shall forward frames to another remote bridge using the MAC unicast service only. A remote bridge shall not forward a frame received from another remote bridge if the frame was sent using the MAC multicast service.

7.5 Power management

A power-sensitive device should use WiMedia MAC [B4] facilities such as the PCA Availability IE, TIM IE, and Hibernation Mode IE as well as WLP facilities such as the Broadcast Traffic Indications field to conserve power. This subclause includes specific requirements to permit devices to coordinate hibernation periods for power conservation.

7.5.1 Local cycle

A device shall be in active mode in the first superframe of its local cycle. A device shall start its local cycle at a BPST, and shall determine its local cycle as follows:

Local cycle = 2^n superframes, where n is its local cycle index; 0 <= n <= wMaxLocalCycleIndex.

A device shall set the Local Cycle Index in the Cycle Parameters field in its WLP IE to its current local cycle index. A device may indicate that it never hibernates by setting its local cycle index to zero. A device shall align its local cycle with the global cycle as described in 7.5.2.

After the first superframe in a local cycle, a device may stay in active mode for a variable duration. A device may exit active mode if it has no traffic buffered for any neighbors in active mode, and no traffic is pending for it from active neighbors as indicated by a TIM IE. To exit active mode, a device should enter hibernation mode as described in the WiMedia MAC specification.

7.5.2 Global cycle

In order to synchronize with neighbors' local cycles, a device shall maintain a global cycle coordinated with its neighbors. A global cycle starts every wMaxGCSC+1 superframes. A device shall set the Global Cycle Start Countdown (GCSC) field in its WLP IE to the number of superframes before the start of the next global cycle, not including the current superframe. The device shall set the GCSC value to wMaxGCSC in the first superframe of every global cycle and decrement it by one in each subsequent superframe, except as noted below. A value of zero indicates that the next global cycle start time (GCST) occurs at the start of the next superframe.

Before a device includes a WLP IE in its beacon, it shall establish a GCST. To establish a GCST, the device shall set its GCSC field in the current superframe to one less than the value of the

GCSC fields contained in beacons it received in the previous superframe. If the device received beacons containing two or more different GCSC values in the previous superframe, it shall set its GCSC field to one less than the smallest received GCSC value. If the device did not receive any beacon with a GCSC field, and it has not previously established a GCST, it may choose any GCST.

Devices using two or more unaligned GCSTs may come into range. To address this situation, a device shall follow these rules:

- If the GCSC fields in one or more beacons the device received in the previous superframe indicate a single GCST that is not aligned with the device's own GCST, the device shall check the difference between its GCST and the unaligned GCST. If that difference is equal to (wMaxGCSC+1)/2 superframes, the device shall set its GCSC field for the current superframe to a random integer drawn from a uniform distribution over the interval [0, wMaxGCSC]. Otherwise, if the unaligned GCST falls within the second half of the device's global cycle, the device shall set its GCSC for the current superframe to one less than the unaligned GCSC value in the previous superframe, to align GCSTs.
- If the GCSC fields in beacons the device received in the previous superframe indicate two or more distinct GCSTs that are not aligned with the device's own GCST, the device shall set its GCSC field for the current superframe to one less than the smallest received GCSC value, to align with the next GCST.
- In all other cases, the device shall decrement its GCSC by one in each superframe.

If any calculated GCSC value is less than zero, a device shall set its GCSC field for the current superframe to wMaxGCSC.

A device that changes its GCST shall include the new GCSC value in the GCSC field of subsequently transmitted beacons. The device shall be in active mode at the start of its local cycles based on its new GCST. The device shall also be in active mode at the start of its local cycles based on its previous GCST for at least $2^{wMaxLocalCycleIndex}$ superframes.

A device that receives one or more GCSC values that are different from its own shall stay in active mode for at least mMaxLostBeacons[9] additional superframes.

7.5.3 Cycle scheduling

If M is the maximum value a device receives in a Local Cycle Index field, then all neighbors of that device will be in active mode in superframes that are integer multiples of 2^M superframes relative to the GCST of the device and its neighbors. This is the time when a power-sensitive device should send its broadcast/multicast traffic to avoid the need to send such traffic multiple times.

A device may reduce its local cycle index at any time. A device shall not increase its local cycle index except when all its neighbors are in active mode, since this is the time when all neighbors can receive the updated local cycle index.

Figure 34 illustrates the scheduling concepts specified in 7.5.

[9] mMaxLostBeacons is defined in the WiMedia MAC specification [B4].

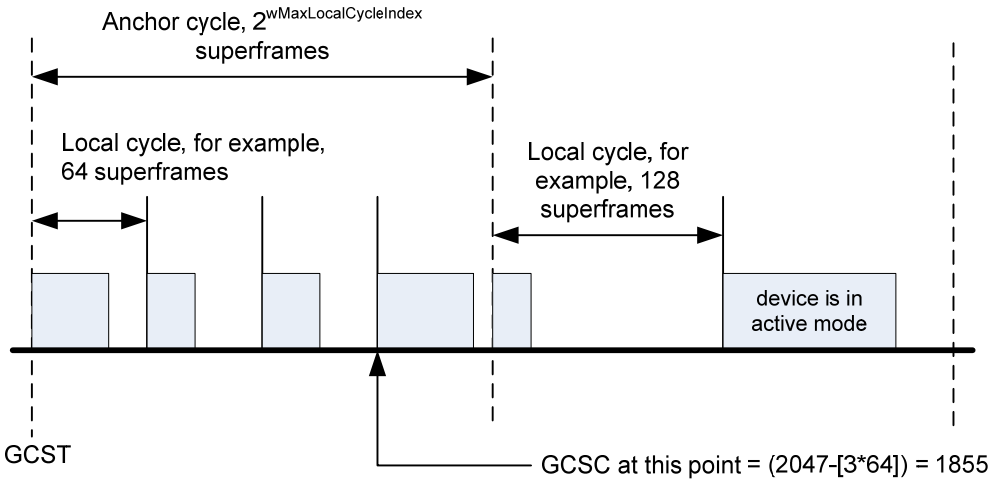


Figure 34 — Example local and anchor cycles

7.5.4 Negotiation of local cycle

In some cases, the local cycle index of a recipient device is too large, resulting in buffer overflow and violation of QoS requirements at the source device. To address this problem, a device may send a Local Cycle Change Request control frame to one or more of its neighbors to request a change of their local cycle indexes. An addressed recipient should select a new local cycle index that is equal to or less than the indicated values in the request. The Local Cycle Change Request control frame may be sent to a specific neighbor, a multicast address, or a WSS broadcast address.

7.5.5 Hibernation anchor selection

The hibernation anchor selection mechanism enables devices to determine a connected set of hibernation anchors such that every device is either in the radio range of a hibernation anchor or is itself a hibernation anchor. Devices select their hibernation anchors once every anchor cycle at the beginning (within the first few superframes) of that anchor cycle according to the following rules.

A device shall start a new anchor cycle every $2^{wMaxLocalCycleIndex}$ superframes relative to its GCST.

A device shall determine its anchor cycle weight (ACW) as the smaller of wMaxACW or the number of anchor cycles when the device has not been a hibernation anchor. A value of zero indicates that the device served as a hibernation anchor in the last anchor cycle. A device shall set the Anchor Cycle Weight (ACW) field in its WLP IE to its anchor cycle weight in the first superframe of an anchor cycle and successive superframes until it selects a hibernation anchor. Once a device selects a hibernation anchor, it shall announce its selected hibernation anchor by replacing its ACW with its AnchorAddr in all subsequent beacons transmitted in the anchor cycle.

A device yet to select a hibernation anchor shall select its hibernation anchor for an anchor cycle according to the following rules:

A. The device may select itself as a hibernation anchor at any time.

B. If the device is mains-powered it shall select itself as a hibernation anchor and shall announce that selection in every superframe, including the first superframe of every anchor cycle.

C. Let X be the set of the device's neighbors that have selected themselves as hibernation anchors, and let Y be the set of the device's neighbors that have not selected a hibernation anchor.

The device shall select any device from set X as its hibernation anchor in the next superframe if:

— the neighbor graph of set X is a connected graph (according to graph theory); and
— each device in set Y has a neighbor in set X.

The neighbor graph of set X is a graph with vertices corresponding to devices in set X, and edges that connect a pair of vertices if the corresponding devices are neighbors. For purposes of this test, devices are considered neighbors if each reports the other's DevAddr in the appropriate beacon slot in its BPOIE in any of the latest mMaxLostBeacons+1 superframes.

D. The device shall select itself as a hibernation anchor in the next superframe if in the current superframe all of the following conditions are satisfied:

— It has received an ACW or AnchorAddr from all of its neighbors.
— All neighbors that have an ACW value higher than its own have indicated selection of a hibernation anchor.
— All neighbors that have the same ACW value as its own and have higher beacon slot numbers than its own have indicated selection of a hibernation anchor.
— It does not select a neighbor to be its hibernation anchor as described in rule C.

E. The device shall select itself as a hibernation anchor if it has not selected a neighbor as its hibernation anchor as described in rule C within wMaxCycleWait superframes after the start of an anchor cycle.

If a device selects itself as a hibernation anchor, it shall remain in active mode for the rest of the anchor cycle, and shall include a Hibernation Anchor IE in its beacon in each superframe. In the Hibernation Anchor IE it shall include hibernation information about each of its hibernating neighbors, as defined in the WiMedia MAC specification.

Once a device selects a hibernation anchor in an anchor cycle, it shall not change that selection for the rest of the anchor cycle.

A device shall stay in active mode until all neighbors of the device have selected a hibernation anchor.

7.6 Quality of service

7.6.1 DS field mapping for IP packets

If an IPv4 [B13] or IPv6 [B21] packet is to be transmitted using PCA, the transmitting device shall assign a user priority based on the packet's DS field [B23] as shown in Table 39.

Table 39 — DS field to user priority mapping

DS Field (in hex)	User Priority
0x38–0x3F	7
0x30–0x37	6
0x28–x2F	5
0x20–0x27	4
0x18–0x1F	3
0x10–0x17	2
0x08–0x0F	1

DS Field (in hex)	User Priority
0x00–0x07	0

7.6.2 Parameterized QoS

If a WLP device supports higher-layer end-to-end QoS signaling protocols as a part of a larger network providing end-to-end QoS delivery, it should provide Controlled-load [B18] and Guaranteed [B19] services. If a device supports Controlled-load or Guaranteed services, it shall use the distributed reservation protocol (DRP) [B4] to reserve medium time. Traffic characteristics and service quality of an application are encoded in the form of a Traffic Specification (TSPEC), based on which network resources, such as medium time, will be reserved to provide the QoS requested. Annex B provides an example implementation.

7.7 WLP parameters

Table 40 contains the values for WLP parameters.

Table 40 — WLP parameters

Parameter	Value
wBridgeActivityWindow	16 superframes
wBridgeServiceTimeout	160 superframes
wEnrollmentTimeout	2 minutes
wMaxACW	255
wMaxCycleWait	6 superframes
wMaxGCSC	2047 superframes
wMaxLocalCycleIndex[10]	8
wMinDiscoveryRepeatTime	1 second
wPerMessageTimeout	15 seconds
wResponseTimeout	5 seconds

[10] wMaxLocalCycleIndex is determined from the WiMedia MAC [B3] maximum hibernation time. wMaxLocalCycleIndex is set to 8, which dictates a maximum local cycle of 256. This is compatible with the maximum hibernation period of 255 superframes allowed in the WiMedia MAC, since devices must be in active mode for at least one superframe in every local cycle.

Annex A (normative) Mathematical functions used for association

This annex describes details of mathematical functions used in association.

A.1 Representation of numbers

In this clause, the following octet order convention is used: Text representing integers is presented most-significant octet first, with each octet consisting of two hexadecimal characters. Octets might or might not be separated with spaces for clarity.

A.2 Secure hash algorithm (SHA-256)

A cryptographic hash function takes a message of any length and produces a fixed-length bit string as an output. This specification uses SHA-256 as specified in FIPS 180-2 [B8]. The SHA-256 algorithm produces a 256-bit long hash.

The nomenclature used in this specification to denote the use of a SHA-256 hash is:

MessageDigest = SHA-256(MessageText)

A.3 Keyed-hash message authentication code (HMAC-SHA-256)

A keyed-hash message authentication code, or HMAC, is used to simultaneously verify both the data integrity and the authenticity of a message. This specification uses HMAC as specified in RFC 2104 [B16], with the SHA-256 hash function. See RFC 2104 or FIPS PUB 198 [B9] for more information.

The nomenclature used in this specification to indicate the use of HMAC is:

AuthenticationCode = $\text{HMAC-SHA-256}_{\text{Key}}$(MessageText)

Key must be exactly 256 bits long (including any possible leading zero bits).

A.4 3072-bit MODP group for Diffie-Hellman exchange

This specification uses the 3072-bit MODP group (group id 15) defined in RFC 3526 [B26].

The hexadecimal value of the prime number p is:

```
FFFFFFFF FFFFFFFF C90FDAA2 2168C234 C4C6628B 80DC1CD1
29024E08 8A67CC74 020BBEA6 3B139B22 514A0879 8E3404DD
EF9519B3 CD3A431B 302B0A6D F25F1437 4FE1356D 6D51C245
E485B576 625E7EC6 F44C42E9 A637ED6B 0BFF5CB6 F406B7ED
EE386BFB 5A899FA5 AE9F2411 7C4B1FE6 49286651 ECE45B3D
C2007CB8 A163BF05 98DA4836 1C55D39A 69163FA8 FD24CF5F
83655D23 DCA3AD96 1C62F356 208552BB 9ED52907 7096966D
670C354E 4ABC9804 F1746C08 CA18217C 32905E46 2E36CE3B
E39E772C 180E8603 9B2783A2 EC07A28F B5C55DF0 6F4C52C9
DE2BCBF6 95581718 3995497C EA956AE5 15D22618 98FA0510
15728E5A 8AAAC42D AD33170D 04507A33 A85521AB DF1CBA64
ECFB8504 58DBEF0A 8AEA7157 5D060C7D B3970F85 A6E1E4C7
ABF5AE8C DB0933D7 1E8C94E0 4A25619D CEE3D226 1AD2EE6B
F12FFA06 D98A0864 D8760273 3EC86A64 521F2B18 177B200C
BBE11757 7A615D6C 770988C0 BAD946E2 08E24FA0 74E5AB31
43DB5BFC E0FD108E 4B82D120 A93AD2CA FFFFFFFF FFFFFFFF
```

The generator g is: 2.

A.5 Public key generation

Registrars and enrollees shall generate public keys by the following method:

1. Generate a private key X, a 256-bit random number. Requirements for random number generation are specified in A.6.
2. Calculate the corresponding public key PK = g^X mod p, where g and p are constants defined in A.4. PK is a 3072-bit (384 octet) number, with as many leading zero bits as needed.

For security reasons, a device should generate fresh public keys for each new enrollment session. A device may re-use a public key provided that its private key has been kept secret.

A.6 Cryptographic grade random number generation

A device shall select a cryptographic grade random number according to the following criteria:

— It shall derive a random number from a physical entropy source, such as RF noise, thermal noise, or another unpredictable physical phenomenon. RFC 4086 [B29] provides detailed information on the generation of cryptographic grade random numbers and provides guidance for achieving suitable randomness.
— It shall not derive a random number from information that was used in any previous operation, and shall not use information that was used in random number derivation in any subsequent operation.
— It shall not use a random number with a value of zero or one.
— It shall select a random number from the available number space with equal probability of selecting each possible number.

A.7 Numeric Comparison

Numeric comparison is a user verification process that authenticates the identity of the registrar and the enrollee by asking the user to visually compare and accept a short number that is displayed on both the registrar and the enrollee. The number is calculated as:

ComparisonNumber = first 32 bits of SHA-256(PK_e || PK_r || N_e || N_r || “displayed digest”)

The enrollee shall compute $ComparisonNumber_e$ based on information received in the E2 association frame and sent in the M1 association frame. It shall display $ComparisonNumber_e$ mod 10^N to the user, where N is the display size. In this specification, N is always 2.

The registrar shall compute $ComparisonNumber_r$ based on the information sent in the E2 association frame and received in the M1 association frame. It shall display $ComparisonNumber_r$ mod 10^N.

A device shall display numbers to the user in decimal form, using only the numbers “0” through “9”. The device shall display exactly 2 digits padded with “0” at the beginning if the number to display is less than 2 digits long.

Annex B (informative) Guidelines for use of a TSPEC in SIMA

This annex provides guidelines for using a TSPEC in service interval-based MAS allocation (SIMA) to satisfy a delay requirement.

B.1 Traffic characterization using token bucket model

A token bucket TSPEC is an aggregate TSPEC [B18][B19][B20] that provides a standard set of parameters to characterize a traffic source, based on which networking resources can be reserved for parameterized QoS provisioning. The token bucket TSPEC is a quintuple of mean data rate (r), peak data rate (p), maximum burst size (b), maximum packet size (M) and minimum policed unit (m). The behavior of a traffic stream with a token bucket TSPEC is confined by the theoretical model of a fluid twin token bucket. In this subclause, the fluid twin token bucket model is briefly described to serve as the basis for the guideline of SIMA.

The fluid twin token bucket model provides standard terminology for describing the behavior of a network traffic source. As shown in Figure 35, the model has three parameters, mean rate *r*, peak rate *p* and maximum burst size *b*. A token bucket injects data into the network only if there is an equivalent amount of tokens available and when a packet is transmitted, it consumes and removes exactly the same number of tokens. The arrival curve represents the cumulative maximum number of bits the traffic source may possibly inject during any time interval *t*. Figure 36 shows the arrival curve of a token bucket model. The arrival curve is the basis for traffic characterization using the token bucket model and for the guidelines of SIMA described in this annex.

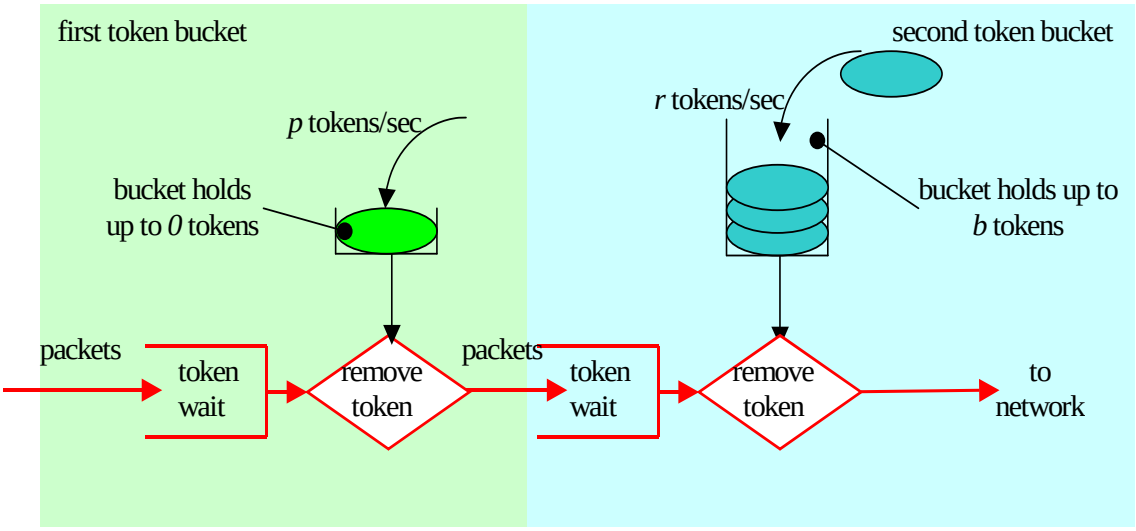


Figure 35 — Fluid twin token bucket model

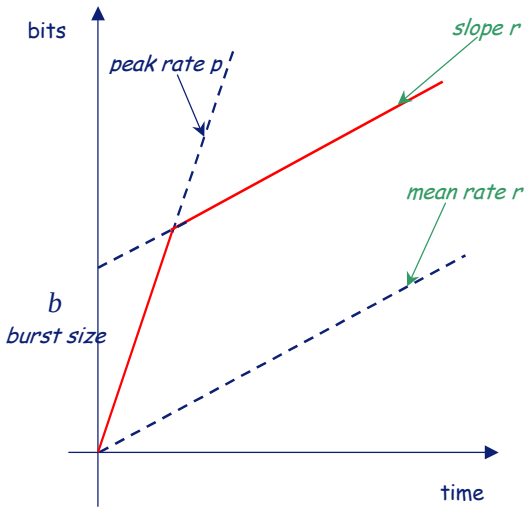
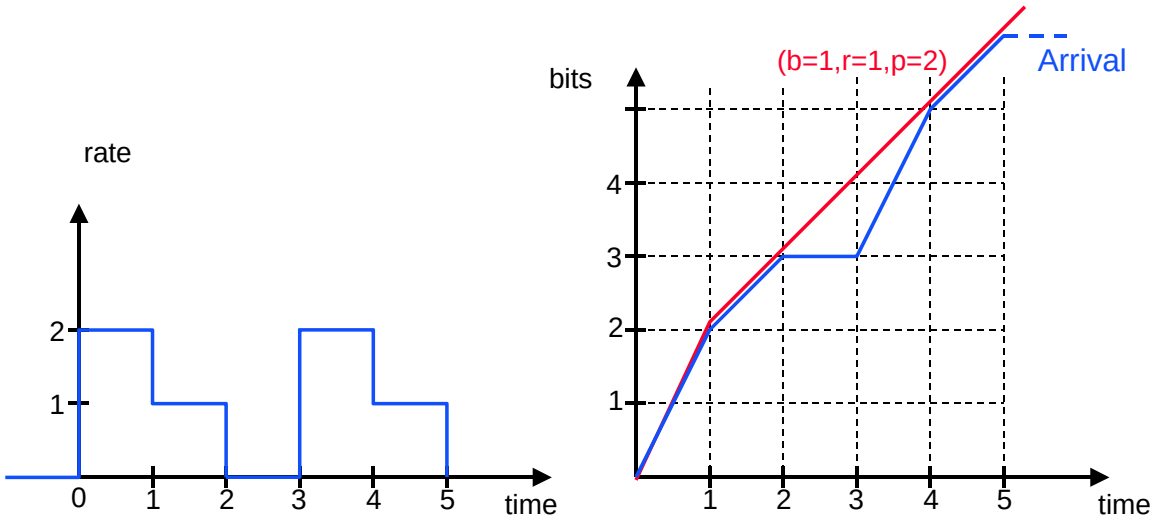


Figure 36 — Arrival curve of twin token bucket policer

Figure 37 illustrates how arbitrary traffic can be bound by the token bucket model and hence characterized with parameters of {*r, p, b*} using the model. Figure 37a shows the instant rate of the example traffic source; while Figure 37b shows the twin token bucket model that characterizes the same traffic source.



a) example traffic rate b) the token bucket model that characterizes the traffic source

Figure 37 —Example of token bucket model for an example traffic source

B.2 Queuing delay and service rate

In provisioning of Guaranteed QoS service, the requirement of maximum allowed delay is specified using the service TSPEC parameters of Requested Service Rate (R) and Slack Term (S). For a traffic stream (TS) with traffic characteristics of {*r, b, p*}, there exists a theoretical minimum service rate for a certain delay bound constraint. This subclause first derives the theoretical minimum service rate based on the fluid twin token bucket model, after which the maximum allowed delay is calculated from the service TSPEC parameters.

Figure 38(a) depicts the general relationship among token bucket TSPEC, minimum service rate and the resulting delay bound values. In the figure, the red line represents the arrival curve of a TS with traffic characteristics of {*r, b, p*}. The slope of the service rate line (in solid blue) is the effective rate at which the TS is serviced. The vertical grey line represents minimum buffer space necessary to avoid overflow for the TS. Using the distance formula, the length of the vertical grey line, hence the minimum buffer space needed is given by:

$$buf = \frac{p-g}{p-r} * b$$

Equation 1 — Minimum buffer space

To calculate delay bound, consider the fact that the last bit of the fully loaded buffer is going to experience the maximum delay as that bit needs to wait for the entire buffer to be serviced. As the TS is serviced with rate *g*, the delay bound *d* can be expressed as:

$$d = \frac{buf}{g}$$

d is shown with the red horizontal line that is also the maximum horizontal distance between the arrival curve and the service rate line.

Considering Equation 1, it can be seen that the queuing delay is mathematically bounded by:

$$d = \frac{p-g}{p-r} * \frac{b}{g}$$

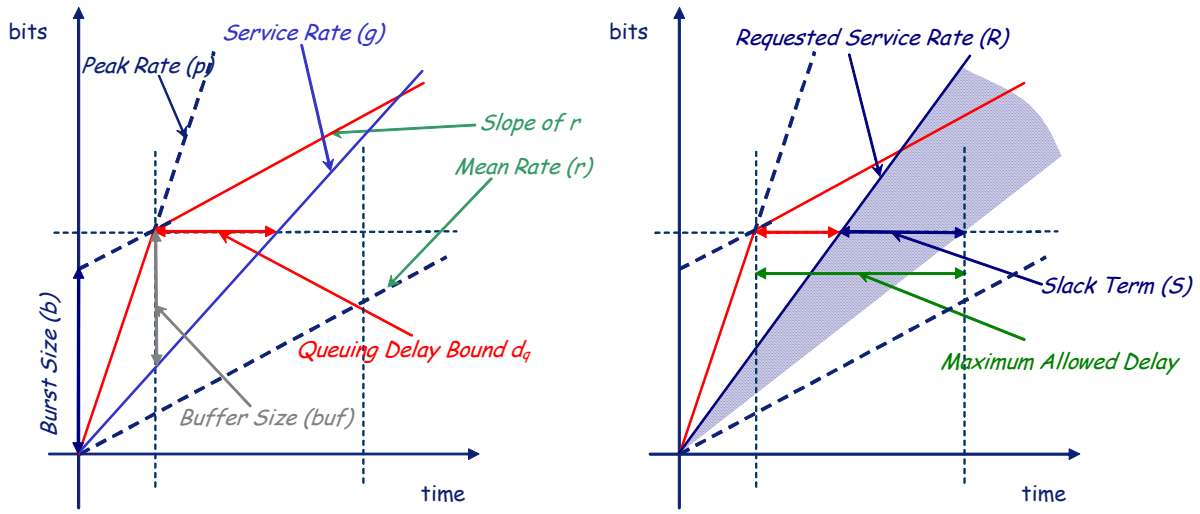
Equation 2 — Maximum queuing delay

Hence, the minimum service rate *g* necessary to guarantee delay bound *d* for a TS with characteristics of {*r, b, p*} is given by:

$$g = \frac{p}{1 + d * \frac{p-r}{b}}$$

Equation 3 — Minimum service rate

As seen in Figure 38(a) and also in Equation 3, it requires a larger service rate *g*, or bandwidth, to satisfy a smaller queuing delay requirement *d*.



(a) Minimum service rate to satisfy a queuing delay bound

(b) Requested service rate, slack term and maximum allowed queueing delay

Figure 38 — Minimum service rate to transport the TS according to the TSPEC

Figure 38(b) depicts the relationship between service TSPEC and token bucket TSPEC. As seen in the figure, slack term *S* signifies the difference between maximum allowed delay (shown with the green horizontal line) and the delay resulting from the requested service rate *R* (shown with the red horizontal line). From Equation 2, the delay resulting from *R* is:

$$d_R = \frac{p-R}{p-r} * \frac{b}{g}$$

Hence the requirement of maximum allowed delay can be expressed in terms of service TSPEC as below:

$$d_s = d_R + S = \frac{p-R}{p-r} * \frac{b}{g} + S$$

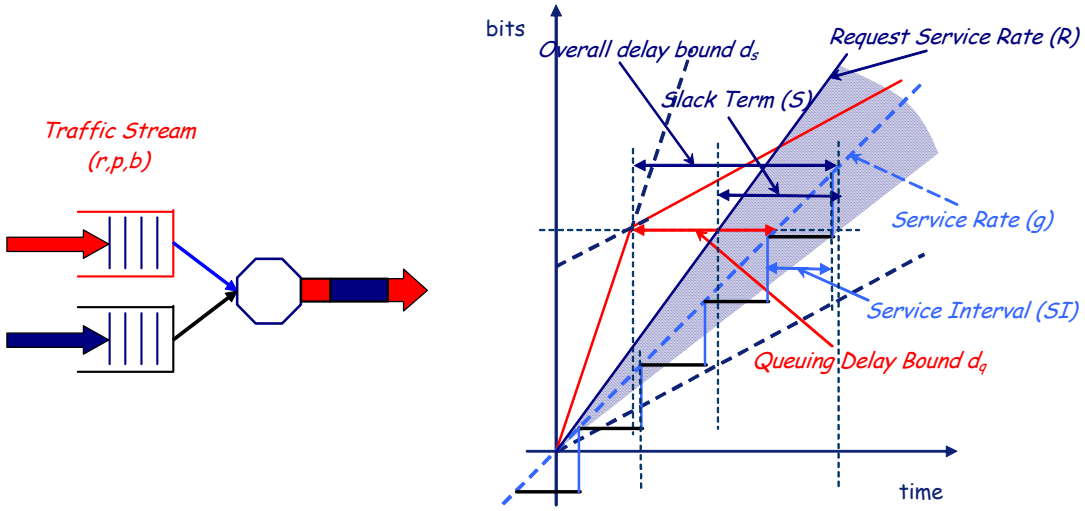
Equation 4 — Maximum allowed delay

As seen in Equation 4, service TSPEC of *R, S* together specifies the maximum allowed delay for the TS with the token bucket TSPEC of {*r, b, p*}, And as represented in the shaded area in the figure, the specification of slack term allows using a range of service rates that are lower than *R* to satisfy the TS's maximum allowed delay.

B.3 Service interval-based MAS allocation

In Time-Division-Multiple-Access (TDMA) based systems such as devices using DRP, medium time is divided into service periods in which a TS is serviced at a higher rate (than the service rate), for example, link rate. Service periods are separated by other periods in which other TSs are serviced. The needed service rate is provided by arranging service periods such that the average rate over both service periods and other periods equals the service rate. Figure 39(a) illustrates such a TDMA system. In order to satisfy the delay bound requirement, the service periods are arranged in periodic manner such that the latency performance can be managed easily and bounded. In such a scheme, MASs are allocated periodically across superframes with a service interval determined based on the delay bound requirement (specified in terms of service TSPEC) as well as token bucket TSPEC. This periodic MAS allocation scheme is referred to as the service

interval based MAS allocation scheme (SIMA). In order to derive the delay bound of SIMA, the actual service line of SIMA with a service rate *g* and strict periodic service interval *SI* is depicted in Figure 39(b). Without loss of generality, SIMA service line is approximated with a solid staircase to simplify the analysis of the worst case.



(a) TDMA system

(b) SIMA service line and delay bound

Figure 39 — SIMA delay bound

As seen in the figure, the service delay caused by the periodic service periods is bounded by the service interval *SI*. Taking into account the additional queuing delay d_q caused by the burstiness of the TS itself, as derived in Equation 2, the overall delay of SIMA is bounded by:

$$d_s \leq SI + d_q$$

Equation 5 — Overall delay bound

SIMA consists of two steps. The first step is the calculation of *SI* and service rate *g* based on a TSPEC (including both the token bucket and service TSPECs), local resources such as buffer space, and local conditions such as link quality. The shaded area in Figure 39(b) also shows the relationship between the service TSPEC and SIMA and is used in B.4 to determine *SI* and *g*. The second step is to make a MAS reservation according to MAC policies and the existing reservation bitmap based on the values of *SI* and *g* calculated in the first step.

B.4 General considerations for choosing a service interval

As seen in B.2 and B.3, the service rate needed to satisfy the specified delay requirement varies with the choice of service interval. The delay requirement d_s, specifies a range of service intervals to choose from. The choice of service interval determines the upper bound of d_q, which in turn determines the service rate. As indicated in Equation 5 and Equation 3, a larger service interval results in a smaller upper bound on d_q, which in turn requires a larger service rate. This is the basis for the trade-off between service interval and bandwidth.

As described in B.2, a delay requirement is usually specified using a service TSPEC {*R*,*S*}. This has the advantage of simplifying implementation of such a trade-off. The shaded area in Figure 39(b) indicates any service rates lower than *R* that are feasible to satisfy the overall delay bound. As seen in the figure, the maximum service interval for requested rate *R* is the slack term *S*.

Therefore a low-complexity implementation could be as simple as choosing *R* as the service rate and selection of a service interval smaller than or equal to *S*.

In addition, the choice of service interval is constrained by MAC medium allocation policies.

Usage of SIMA to derive a DRP reservation based on a TSPEC can be summarized as follows:

— Make proper trade-off between service interval *SI* and service rate *g* according to Equation 3 and Equation 5 based on available bandwidth.
— Take into account the MAC policies when choosing *SI*.

B.5 Example of SIMA

Making a reservation of network resources using the WiMedia MAC distributed reservation protocol (DRP) [B4] requires determining the number of Medium Access Slots (MASs) and the locations of the MASs within the WiMedia superframe. The number of MASs per superframe depends on many factors such as traffic source bandwidth characteristics, PHY transmission data rate, link condition and/or transmission distance, MSDU sizes, and acknowledgement type. The locations of the MASs within the superframe depends on the total number of MASs, the service interval or latency requirement, traffic source burstiness, etc. In addition, both the total number and the locations of MASs are constrained by MAC reservation policies.

In this subclause, the derivation of a DRP reservation for an example multimedia application using an internet protocol television (IPTV) is used to demonstrate relevant MAC reservation policies and to illustrate the trade-off between service interval and reservation bandwidth that is also referred to as service rate.

Consider a multimedia application with a wireless IPTV and a personal video recorder (PVR) recording the same program. At some point, the user picks up a remote control and tunes the set-top box (STB) to start the IPTV program. The PVR in the STB simultaneously starts to record the same program to a wirelessly connected external hard disk drive that is located in a closet next to the living room. Assume the video source of the service provider generates an MPEG-4 elementary stream using Real-Time Transport Protocol (RTP) [B26] as transport, and the token bucket TSPEC of the stream is as the follows:

Mean Data Rate r = 4.13 Mbps
Peak Data Rate p = 14.8 Mbps
Maximum Burst Size b = 131350 octets
Maximum packet size M = 1490 octets[11]
Minimum policed unit m = 49 octets
Nominal packet size = 1427 octets
Delay d_s = 64 ms

In this example, there are two identical streams transmitted on the medium. Therefore there are two DRP reservations, referred to as STB and PVR.

B.5.1 Reservation bandwidth and service interval

According to Equation 5, the service interval needs to be selected in the range of [0, 64 ms] to satisfy a delay requirement of 64 ms, which consequently determines the permissible queuing latency bound d_q. As seen in Figure 38 as well as Equation 3, a larger service rate *g* is required in

[11] The maximum packet size (M) and minimum packet size (m) are used to determine bandwidth efficiency and MAC/PHY layer overhead, consequently to derive total number of MASs from specified service rate.

order to meet smaller queuing delay bound d_q. That is, it requires more medium time, or bandwidth, to satisfy a smaller d_q. This is because serving a traffic stream at a higher rate helps to reduce accumulation of frames in the transmission queue, which are arriving in a bursty manner and therefore reduces queuing latency. In conclusion, the reservation bandwidth required to satisfy the latency requirement varies with the choice of service interval.

Using Equation 3, we can calculate the lower bound and upper bound of reservation bandwidth, g_{min} and g_{max}, to elaborate on the effect of choice of service interval on reservation bandwidth.

The lower bound of reservation bandwidth g_{min} can be achieved when choosing service interval *SI* close to 4 ms, which leaves d_q = 60 ms. Therefore, the minimum reservation bandwidth required is:

$$g_{\min} = \frac{p}{1 + d_q * \frac{p - r}{b}} = \frac{14.8}{1 + 0.060 * \frac{14.8 - 4.13}{0.13135 * 8}} = 9.2\ Mbps$$

Equation 6 — Minimum reservation bandwidth

Apparently, the upper bound of reservation bandwidth is peak rate p, resulting in d_q = 0 ms. From Equation 5, we can determine that *SI* = 64 ms in this case. Therefore, the maximum reservation bandwidth is given by:

$$g_{\max} = \frac{p}{1 + 0 * \frac{p - r}{b}} = p = 14.8\ Mbps$$

Equation 7 — Maximum reservation bandwidth

From Equation 6 and Equation 7 for this specific stream, it can be seen that by choosing different service intervals, the reservation bandwidth required may increase as much as:

$$\frac{g_{\max} - g_{\min}}{g_{\min}} = \frac{14.8 - 9.2}{9.2} = 61\%$$

Equation 8 — Maximum reservation bandwidth overhead

If the same traffic stream is serviced more frequently, e.g. with half of the service interval, i.e. $SI = 32ms$ which leaves $d_q = 32ms$, then the reservation bandwidth required is:

$$g = \frac{p}{1 + d_q * \frac{p - r}{b}} = \frac{14.8}{1 + 0.032 * \frac{14.8 - 4.13}{0.13135 * 8}} = 11.17\ Mbps$$

Equation 9 — Reservation bandwidth with 32 ms service interval

As seen from Equation 9, by reducing the service interval by a factor of 2, the additional reservation bandwidth (with respect to the lower bound) required to meet the same delay requirement is reduced to:

$$\frac{g - g_{\min}}{g_{\min}} = \frac{11.17 - 9.2}{9.2} = 21\%$$

Equation 10 — Reservation bandwidth overhead with 32 ms service interval

B.5.2 Overview of relevant MAC reservation policies

MAC reservation policies concern both reservation limits and reservation locations. MAC policies regarding reservation limits include restrictions on the total medium time in units of number of MASs, and on reservation block sizes. Block size limits depend on the location of the block within an allocation zone. The permissible maximum block size of any safe reservation is 8 MASs. Blocks with size of 4 MASs or less are safe anywhere independent of their locations with the exception of the last 3 MASs within an allocation zone. Safe blocks larger than 4 MASs are restricted to the first 8 MASs of each allocation zone.

Before we describe the MAC policies on reservation locations, the isozone structure of the superframe is first introduced to facilitate the elaboration on the MAC polices. Allocation zones of a WiMedia superframe excluding allocation zone zero are further grouped into 4 subsets of allocation zones, called isozones, as depicted in Figure 40. Each isozone is identified by its index, called iso-index, ranged from 0 through 4 inclusive. The MASs within an isozone are distributed evenly across the superframe. More specifically, the MASs located in the same row and “adjacent” allocation zones within an isozone are separated from each other by a uniform interval that depends on the isozone in which the MASs are located. Such an interval is referred to as the native service interval of the isozone. Table 41 lists the native service interval, and comprising allocation zones of each isozone. Notice that higher-indexed isozones are capable of supporting smaller service interval, hence tighter delay bound. Also it’s worth noting that these properties apply to the specific case listed in the last row in Table 41, in which a reservation has blocks in every allocation zone, optionally excluding allocation zone zero. This type of reservation is referred to as a row component in MAC specification. In this type of reservation, its “native service interval” is close to the duration of an allocation zone (4.096ms). MAC policies on reservation locations require a row component to be located at as high-indexed MAS locations as possible within all the allocation zones. Therefore the bottom part of the 2-D representation of WiMedia superframe can be considered as the “virtual” isozone 4 with a dynamic boundary, as far as supported service interval is concerned.

In order to make room for subsequent reservations that may request smaller service interval or tighter delay bound, MAC polices on reservation locations require the selection of reservation blocks in the isozones with as low iso-indices as possible, provided the locations meet the application’s latency requirement.

For the same purpose of making room for subsequent reservation requests of row components, that is, with smaller service interval, MAC reservation policies on locations also require non-row components to be located within the first 8 MASs of their zones if possible and as close to the beginning of their allocation zones as possible.

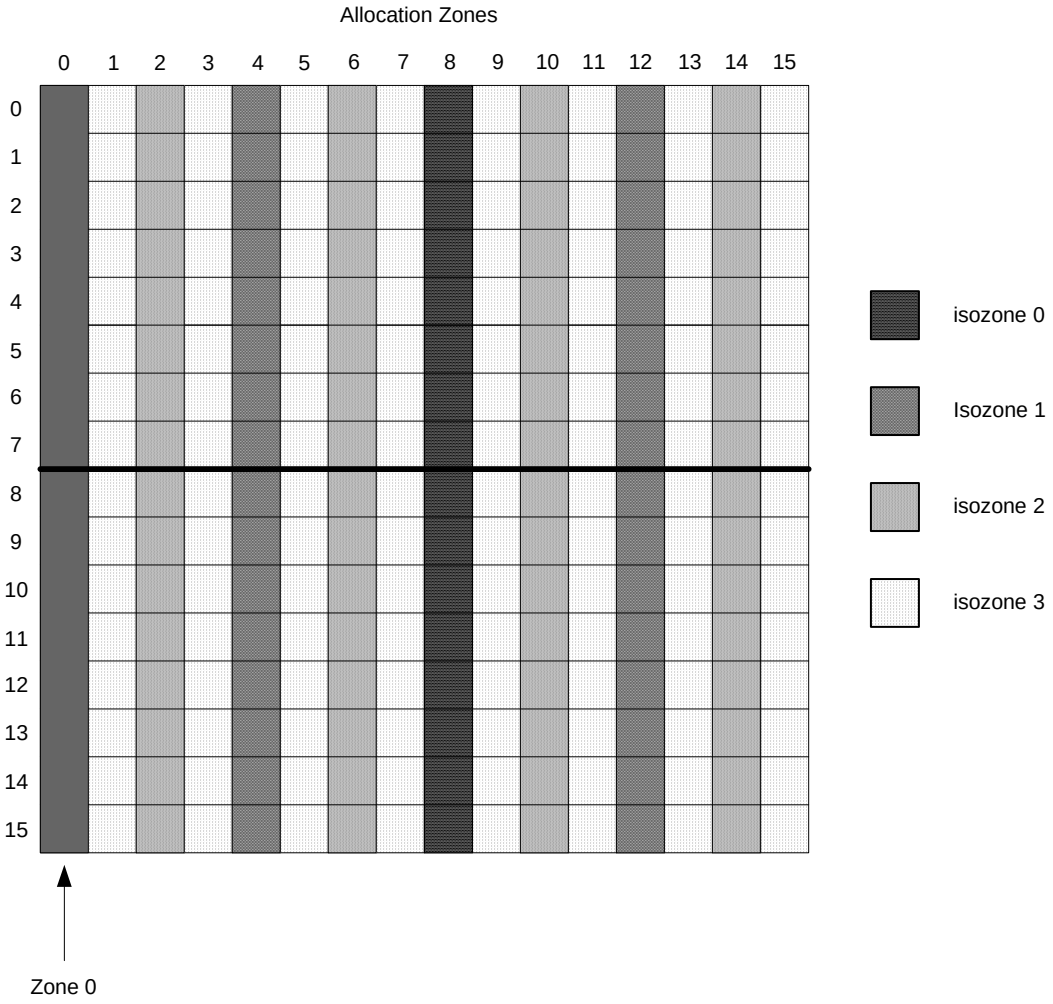


Figure 40 — Isozone structure in two-dimensional view of a WiMedia superframe

Isozone Index	Number of allocation zones (k)	Comprising allocation zones	Native service interval (milliseconds)
0	1	8	16×4.096
1	2	4,12	8×4.096
2	4	2,6,10,14	4×4.096
3	8	1,3,5,7,9,11,13,15	2×4.096
4	15	All	~4.096

Table 41 — Native service intervals of isozones

B.5.3 Determine service interval according to MAC reservation policies

Service interval of a reservation is tightly dependent on the number of allocation zones per superframe, which a reservation occupies. The number is referred as value k as listed in Table 41. Therefore, the k value is first determined based on the related MAC allocation polices and

consequently *SI* is determined based on the value of k. The goal of this procedure is to search for a k value that strikes a good balance among service rate, block size, latency requirement and bandwidth efficiency with the constraints of MAC reservation polices.

If both the set-top box and PVR transmit the video stream at a PHY data rate of 106.7 Mb/s, we assume that the service rate of approximately 10Mbps to transmit such a stream would require roughly 40 MASs per superframe – if an allowance is made for transmission errors and subsequent retries[12]. Therefore the total number of MASs for each of these streams would be well below the mTotalMASLimit (112 MASs).

As described in B.5.2, the maximum block size for a safe reservation is 8 MASs. Hence the lower bound on the number of allocation zones needed for each reservation is:

$$k_{\min} = \frac{40}{8} = 5$$

Upper bound of k is when the reservation is created using row components such that it occupies every allocation zones excluding allocation zone zero. Therefore the maximum k is:

$$k_{\max} = 15$$

Therefore, the k value should be chosen in the range [5, 15], if feasible.

As seen in Table 41, the higher k, the higher iso-index. In addition, the higher the k value, the smaller the block size, given a fixed total number of MASs.

If k is chosen as the upper bound of 15, the resulting reservation would be located in the “virtual” isozone 4 with the highest iso-index. MAC policies on reservation locations require to allocate in as low-indexed isozone as possible.

At the other end of the spectrum, if k is chosen to be the lower bound 5, STB reservation would already occupy isozone 2, as indicated in Table 41. One conforming reservation with k=5 is depicted in Figure 41. As seen in the figure, the PVR reservation would be forced to locate its reservation blocks in isozone 3 in order to confine its reservation blocks in the first 8 MASs of their allocation zones.

As seen from the figure, the corresponding maximum service interval is:

$SI_{\max} = 16.384ms$ (e.g. between allocation zone 2 and 6);

This consequently determines the upper bound of queuing delay as:

$$d_q = 64 - 16.384 = 47.616ms$$

The corresponding required reservation bandwidth is:

[12] Data Link Layer (L2) and Physical Layer (L1) overhead are also taken into account using the TSPEC parameters Maximum Packet Size and Minimum Policed Unit.

$$g_5 = \frac{p}{1 + d_q * \frac{p - r}{b}} = \frac{14.8}{1 + 0.047616 * \frac{14.8 - 4.13}{0.13135 * 8}} = 9.98 Mbps$$

In this case, the additional reservation bandwidth with regard to the lower bound is:

$$\frac{g_5 - g_{\min}}{g_{\min}} = \frac{9.98 - 9.2}{9.2} = 8.5\%$$

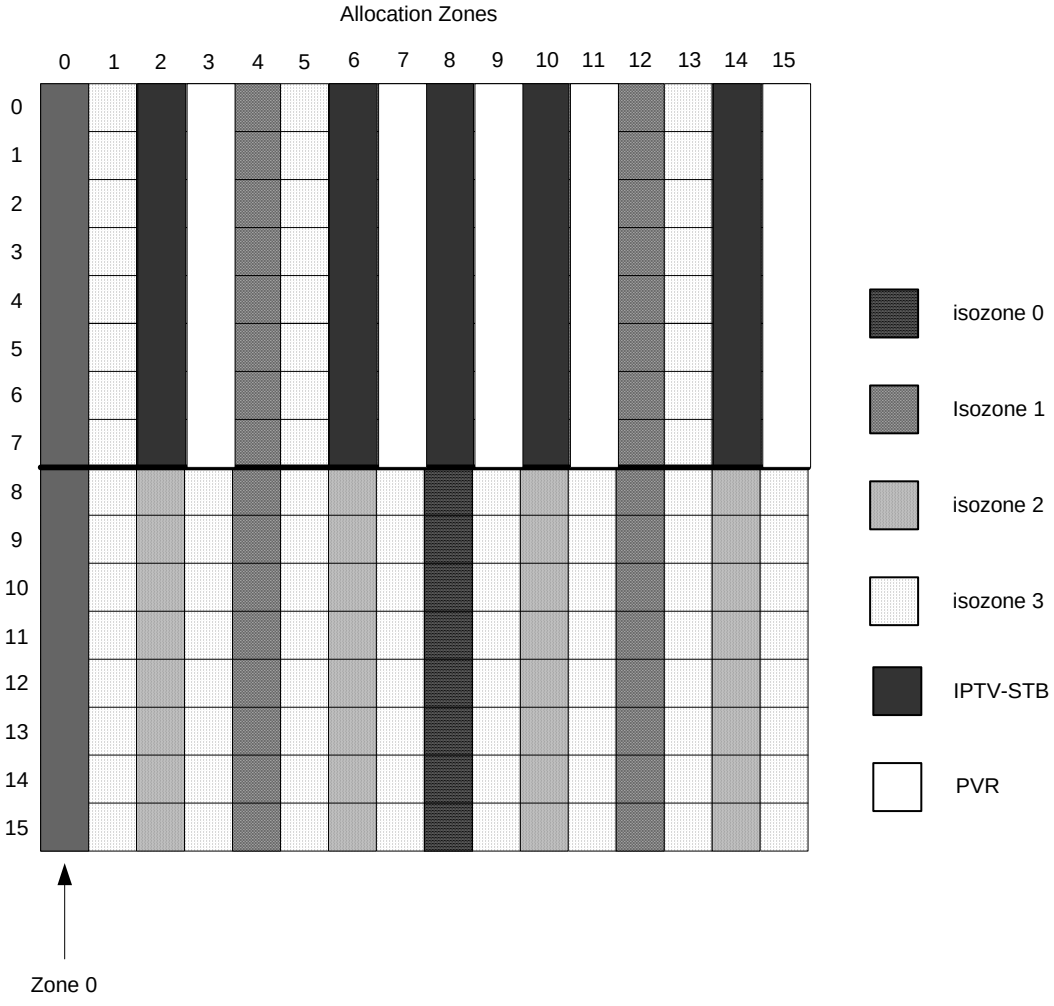


Figure 41 — STB and PVR reservations with k=5

As seen from Table 41, any reservation that contains reservation blocks in isozone 3 allows k to be at least 8. Moreover, choosing a larger value of k, or smaller value of *SI*, allows for larger queuing delay d_q, which results in less reservation bandwidth *g* as indicated in Equation 3. Therefore k of larger value than 5, e.g. 8 should also be considered. In addition, k=8 also allows for the reservation blocks to be evenly distributed over the superframe. This property reduces delay jitter, which is a welcome property for real-time A/V applications such as IPTV.

On the other hand, a larger value of k, or smaller value of *SI* causes fragmentation of medium access time (or smaller block size), which may result in poor bandwidth efficiency.

The corresponding block size with k=8 is $\frac{40}{8} = 5$

Such reservations are shown in Figure 42. Unfortunately, the MAC policy on reservation limits requires that reservation blocks with size larger than 4 MAS be located above the midline in the two-dimensional view of the superframe in order to qualify as a safe reservation. In this case the PVR has to declare its reservation as unsafe. Consequently, the PVR will have to move or change its reservation location or block size, if a late-coming application stream issues a relinquish request. A move or change of an existing reservation may cause disruption in service, which may result in an unpleasant user experience.

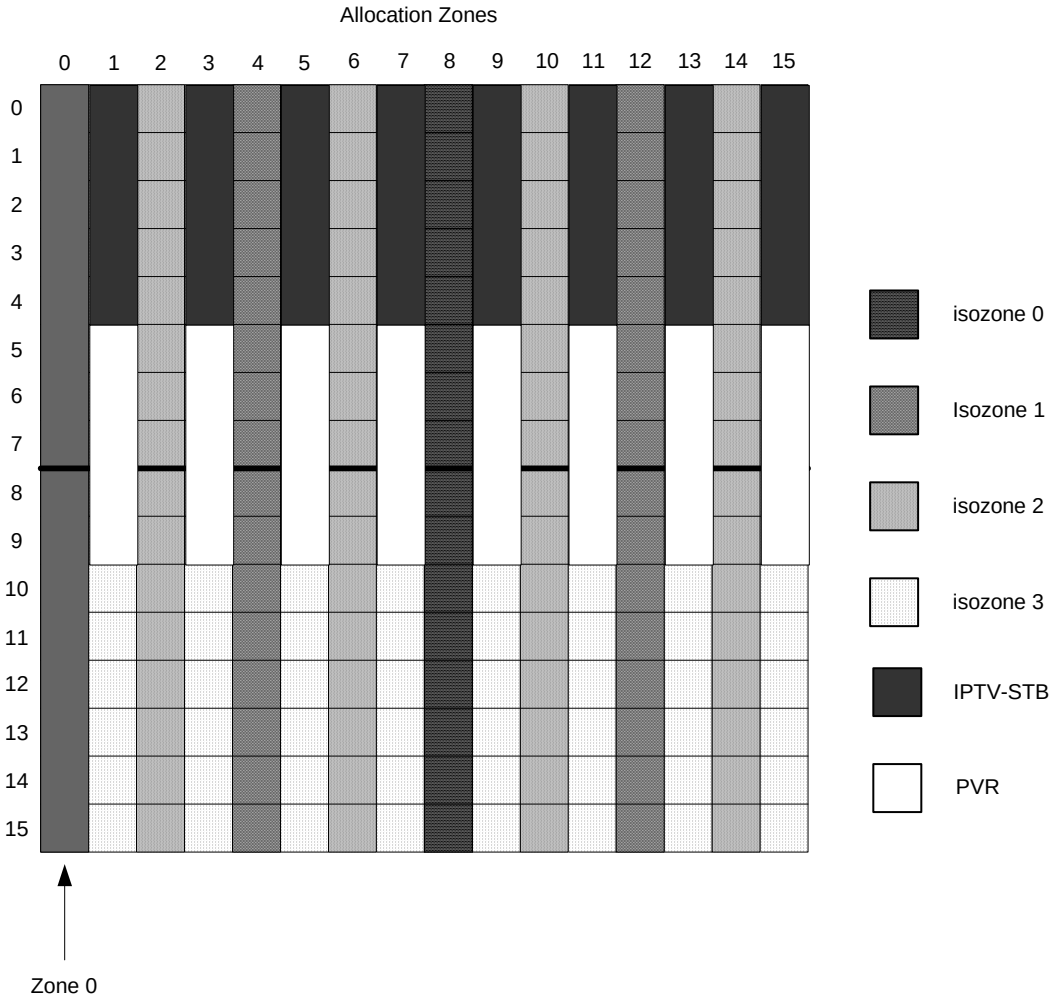


Figure 42 — STB and PVR reservations with k=8

As a result, the searching range of k is narrowed down to [8,15] striving for both safe STB and PVR reservations to further reduce reservation bandwidth and strike a balance in regard to block size. As described in the WiMedia MAC specification, an observation of MAC policies on reservation limit indicates that reservation blocks with size of 4 MAS or less are safe independent of their locations in the superframe, which allows for flexibility in placement of the reservation blocks. Moreover, choosing a block size less than 4 may cause significant loss in bandwidth efficiency, depending on nominal frame size. Therefore, in this case choosing block size of 4 MAS seems to be a good trade-off between service interval and bandwidth efficiency as well as easy placement of

reservation blocks in the superframe. With block size of 4 MAS in mind, the number of allocation zones is determined by:

$$k = \frac{40}{4} = 10$$

The average service interval is:

$$SI_{ave} = \frac{SP}{k} = \frac{65.536}{10} = 6.5536ms$$

Figure 43 shows such safe reservations for both STB and PVR with k=10.

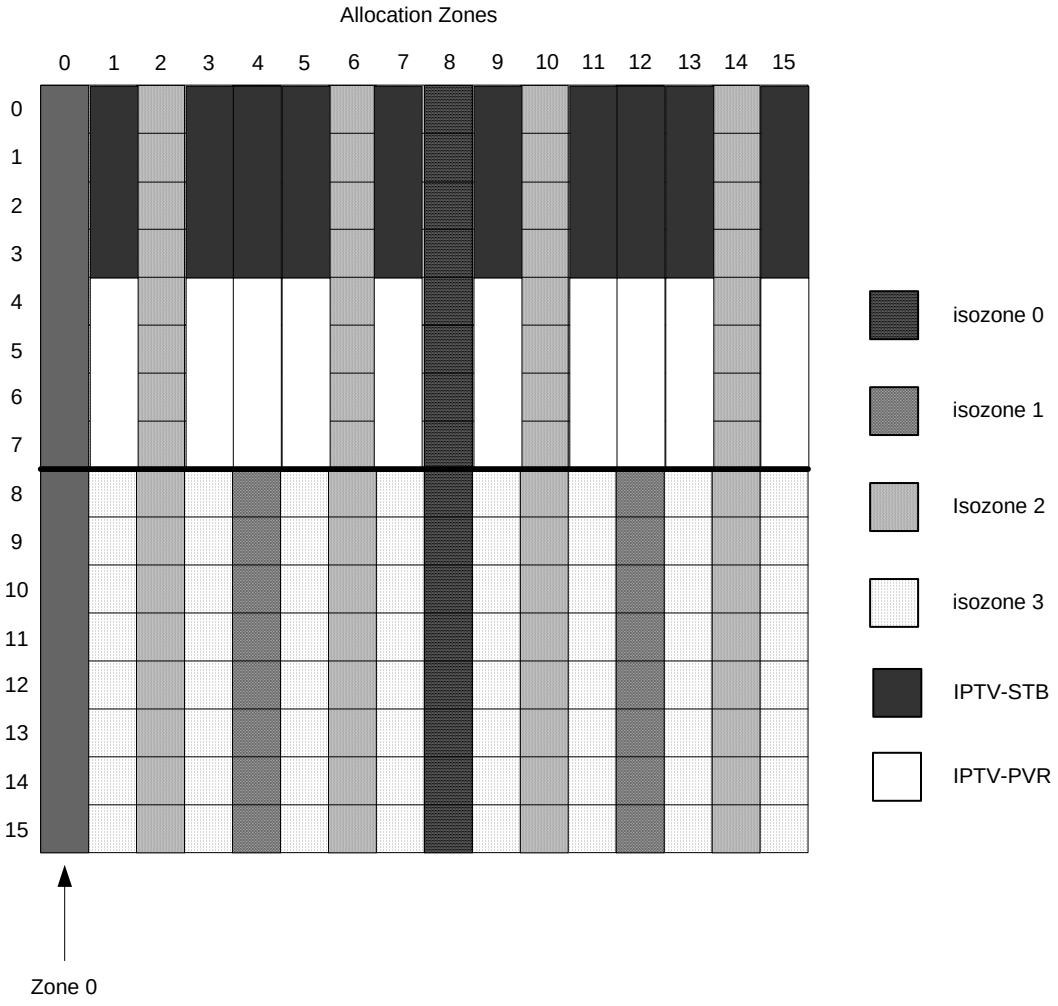


Figure 43 — STB and PVR reservations with k=10

As seen from the figure, the corresponding maximum service interval is:

$SI_{\max} = 8.192ms$ (e.g. between allocation zone 1 and 3);

This consequently determines the upper bound of queuing delay as:

$$d_q = 64 - 8.192 = 55.808ms$$

The corresponding required reservation bandwidth is:

$$g_{10} = \frac{p}{1 + d_q * \frac{p - r}{b}} = \frac{14.8}{1 + 0.055808 * \frac{14.8 - 4.13}{0.13135 * 8}} = 9.45Mbps$$

In this case, the additional reservation bandwidth with regard to the lower bound is only:

$$\frac{g_{10} - g_{\min}}{g_{\min}} = \frac{9.45 - 9.2}{9.2} = 2.7\%$$

B.5.4 Evaluation of the conforming reservations

In the case of combination of STB and PVR applications, choosing either k=5 or k=10 results in safe and conforming reservations. In this subclause, the schemes are evaluated in the aspects of latency, reservation bandwidth and making room for subsequent reservation requests with service interval requirements. Since STB and PVR are mains-powered applications, power consumption is not a major concern. Therefore, the schemes are not evaluated in the aspect of power-saving performance.

In terms of latency, the average and maximum service interval of the k=10 scheme is half of the k=5 scheme, which lowers the scheduling latency by the factor of 2. This allows for larger queuing delay d_q, in turn, reducing the reservation bandwidth. In this specific case when total number of MASs is fixed, the k=10 scheme leaves more bandwidth allowance for transmission errors and subsequent retransmissions.

In terms of reservation bandwidth, with a smaller value of *SI*, the k=10 scheme allows for larger queuing latency, which, in turn, requires less reservation bandwidth. On the other hand, the k=5 scheme allows for longer contiguous medium time in its reservation blocks, which results in a bit better bandwidth efficiency. Taking into account both aspects, roughly speaking, the overall required reservation bandwidth is similar for both schemes.

In terms of making room for subsequent reservation requests with a service interval requirement, as seen in Figure 41, the minimum service interval that the remaining top half of the superframe can support is 8×4.096=32.768 ms (e.g. between allocation zone 4 and 12, or 5 and 13). On the other hand, as seen in Figure 43, the k=10 scheme leaves the entire isozone 2 open, which can accommodate reservation requests that require a service interval of 4×4.096=16.384 ms. Therefore, the k=10 scheme is better in making room for subsequent reservations with service interval requirements.

In conclusion, without consuming more reservation bandwidth, the k=10 scheme provides smaller service interval for the STB and PVR applications and leaves room for potential reservation requests with service interval requirements, and, therefore, is a better choice of reservation scheme.

Annex C (informative) Test vectors

The following examples illustrate frame encoding and transmit order for various options in this protocol.

The frame examples in this annex include the MUX Header as well as the WLP frame in order to clarify octet order over the air.

C.1 WLP IE

C.1.1 Client device

Table 42 — Field values for a client device WLP IE

Field		Value	
Element ID		250 (WLP IE)	
Length		18	
Capabilities	WSSID Hash List Length (b15–b12)	2	0x2101
	Broadcast Traffic Indications Count (b11–b8)	1	
	Reserved (b7–b5)	0	
	Discoverable (b4)	0	
	DRP Establishment (b3)	0	
	Remote Bridge (b2)	0	
	Client Bridge (b1)	0	
	Client Device (b0)	1	
Cycle Parameters	Selecting Anchor (b15)	0	0x223C
	Local Cycle Index (b14–b11)	4	
	Global Cycle Start Countdown (b10–b0)	572	
AnchorAddr		0xEEEC	
WSSID Hash List		0x23, 0x54	
Broadcast Traffic Indications	WSS Tag	0x23	
	MAS List Length	8	
	MAS List	8,9,10,11,12,13,14,15	

The octets comprising the WLP IE are passed to the MLME SAP, to become part of the beacon, in the following order:

```
FA                                  (Element ID)
12                                  (Length)
01 21                               (Capabilities)
3C 22                               (Cycle Parameters)
EC EE                               (AnchorAddr)
23 54                               (WSSID Hash List)
23 08 08 09   0A 0B 0C 0D   0E 0F   (Broadcast Traffic Indications)
```

C.1.2 Bridge

Table 43 — Field values for a bridge WLP IE

Field			Value	
Element ID			250 (WLP IE)	
Length			22	
Capabilities	WSSID Hash List Length (b15–b12)		2	0x211E
	Broadcast Traffic Indications Count (b11–b8)		1	
	Reserved (b7–b5)		0	
	Discoverable (b4)		1	
	DRP Establishment (b3)		1	
	Remote Bridge (b2)		1	
	Client Bridge (b1)		1	
	Client Device (b0)		0	
Cycle Parameters	Selecting Anchor (b15)		1	0xA7FE
	Local Cycle Index (b14–b11)		4	
	Global Cycle Start Countdown (b10–b0)		2046	
ACW			6	
Bridge Information	Load Metric		205	
	Remaining Capacity	Reserved (b7)	0	0x0C
		Remote Bridge (b6)	0	
		Additional Clients (b5–b0)	12	
	Local Segment ID		0x10, 0x00, 0x00, 0x02, 0xEA, 0x46,0x43, 0x53	
WSSID Hash List			0x23, 0x54	
Broadcast Traffic Indications		WSS Tag	0x55	
		MAS List Length	2	
		MAS List	128, 129	

The octets comprising the WLP IE are passed to the MLME SAP, to become part of the beacon, in the following order:

```
FA                                  (Element ID)
16                                  (Length)
1E 21                               (Capabilities)
FE A7                               (Cycle Parameters)
06 00                               (ACW)
CD 0C 10 00   00 02 EA 46   43 53   (Bridge Information)
23 54                               (WSSID Hash List)
55 02 80 81                         (Broadcast Traffic Indications)
```

C.2 Standard data frames

C.2.1 Type/Length field contains a protocol ID

Table 44 — Field values for a standard data frame with a protocol ID field

Field	Value
Protocol ID (MUX Header)	0x0100 (WLP)
WLP Frame Type	0 (Standard Data)
WSS Tag	0xF3
Destination Address	01:13:88:00:01:02
Source Address	00:08:09:45:21:01
Type	0x0806
Client Data	00 01 08 00 06 04 00 01 00 08 09 45 21 01 C0 A8 00 01 FF FF FF FF FF FF C0 A8 00 02

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                   (MUX Header)
00                                      (WLP Frame Type)
F3                                      (WSS Tag)
01 13 88 00   01 02                     (Destination Address)
00 08 09 45   21 01                     (Source Address)
08 06                                   (Type)
00 01 08 00   06 04 00 01   00 08 09 45 (Client Data)
21 01 C0 A8   00 01 FF FF   FF FF FF FF
C0 A8 00 02
```

C.2.2 Type/length field contains a length

Table 45 — Field values for a standard data frame with a length field

Field	Value
Protocol ID (MUX Header)	0x0100 (WLP)
WLP Frame Type	0 (Standard Data)
WSS Tag	0xF3
Destination Address	01:13:88:00:01:02
Source Address	00:08:09:45:21:01
Length	0x001C
802.2 LLC	0xAA AA 03
OUI	0x00 00 00
Protocol Type	0x08 06
Client Data	00 01 08 00 06 04 00 01 00 08 09 45 21 01 C0 A8 00 01 FF FF FF FF FF FF C0 A8 00 02

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                       (MUX Header)
00                                          (WLP Frame Type)
F3                                          (WSS Tag)
01 13 88 00   01 02                         (Destination Address)
00 08 09 45   21 01                         (Source Address)
00 1C                                       (Length)
AA AA 03                                    (802.2 LLC)
00 00 00                                    (OUI)
08 06                                       (Protocol Type)
00 01 08 00   06 04 00 01   00 08 09 45     (Client Data)
21 01 C0 A8   00 01 FF FF   FF FF FF FF
C0 A8 00 02
```

C.3 Abbreviated data frames

Table 46 — Field values for an abbreviated data frame

Field	Value
Protocol ID (MUX Header)	0x0100 (WLP)
WLP Frame Type	1 (Abbreviated Data)
WSS Tag	0xF3
Type/Length	0x0800
Client Data	45 00 00 2E 00 02 00 00 FF 01 3A 7A C0 A8 00 01 C0 A8 00 02 08 00 E6 A0 00 02 12 34 45 63 68 6F 20 52 65 71 75 65 73 74 20 44 61 74 61 00

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                       (MUX Header)
01                                          (WLP Frame Type)
F3                                          (WSS Tag)
08 00                                       (Type/Length)
45 00 00 2E   00 02 00 00   FF 01 3A 7A     (Client Data)
C0 A8 00 01   C0 A8 00 02   08 00 E6 A0
00 02 12 34   45 63 68 6F   20 52 65 71
75 65 73 74   20 44 61 74   61 00
```

C.4 Control frames

C.4.1 Bridge Services Request

Table 47 — Field values for a Bridge Services Request control frame

Field	Value
Protocol ID (MUX Header)	0x0100 (WLP)
WLP Frame Type	2 (Control)
Control Subtype	0 (Bridge Services Request)
WSSID	1D015E9C-2930-4C3B-9B50-987121F32E9D

Field		Value	
Bridge Services Control	Reserved (b7–b4)	0	0x0E
	Enable VLAN Forwarding (b3)	1	
	Enable Non-VLAN Forwarding (b2)	1	
	Enable Remote Bridge Services (b1)	1	
	Enable Client Bridge Services (b0)	0	
Protocol Count		1	
Multicast Address Count		1	
VLAN Identifier Count		0	
Protocol Ranges	Protocol Start 1	0x0600	
	Protocol End 1	0xFFFF	
Multicast Address Ranges	Address Start 1	01-13-88-00-01-CC	
	Address End 1	01-13-88-00-01-CF	
VLAN Identifiers		-	

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                    (MUX Header)
02 00                                    (WLP Frame Type and Control Subtype)
1D 01 5E 9C   29 30 4C 3B   9B 50 98 71   (WSSID)
21 F3 2E 9D
0E                                       (Bridge Services Control)
01 01 00                                 (Count fields)
06 00 FF FF                              (Protocol Ranges)
01 13 88 00   01 CC 01 13   88 00 01 CF   (Multicast Address Ranges)
```

C.4.2 Bridge Services Response

Table 48 — Field values for a Bridge Services Response control frame

Field	Value
Protocol ID (MUX Header)	0x0100 (WLP)
WLP Frame Type	2 (Control)
Control Subtype	1 (Bridge Services Response)
WSSID	1D015E9C-2930-4C3B-9B50-987121F32E9D

Field		Value	
Response	Reserved (b15–b12)	0	0x0200
	Device not connected (b11)	0	
	WSS not activated (b10)	0	
	Resource limitation error (b9)	1	
	Unsupported capability (b8)	0	
	Invalid count field (b7)	0	
	Unsupported protocol (b6)	0	
	Too many VLAN identifiers (b5)	0	
	Too many protocol ranges (b4)	0	
	Too many address ranges (b3)	0	
	Invalid VLAN identifier (b2)	0	
	Invalid protocol range (b1)	0	
	Invalid address range (b0)	0	

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                    (MUX Header)
02 01                                    (WLP Frame Type and Control Subtype)
1D 01 5E 9C   29 30 4C 3B   9B 50 98 71   (WSSID)
21 F3 2E 9D
00 02                                    (Response)
```

C.4.3 DRP Reservation Request

Table 49 — Field values for a DRP Reservation Request control frame

Fields		Value	
Protocol ID (MUX Header)		0x0100 (WLP)	
WLP Frame Type		2 (Control)	
Control Subtype		2 (DRP Reservation Request)	
Request Parameters	Reservation Type (b7–b4)	1 (Hard)	0x1B
	Stream Index (b3–b1)	5	
	Establish (b0)	1	

Fields			Value
TSPEC	Service Type		1 (Controlled Load)
	Mean Data Rate		3,250,000 (0x00319750)
	Peak Data Rate		12,500,000 (0x00BEBC20)
	Maximum Burst Size		4,500 (0x00001194)
	Maximum Packet Size		1500 (0x05DC)
	Minimum Policed Unit		512 (0x0200)
	Requested Service Rate		0 (0x00000000)
	Slack Term		0 (0x00000000)
Traffic Filtering Parameters	Filter Set Count		1
	Filter Set 1	Filter Length	15
		Offset	1 (0x0001)
		Mask	FF FF FF FF FF FF FF FF 00 00 00 00 00 00 FF FF
		Value	F3 01 13 88 00 01 02 00 00 00 00 00 00 08 00

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                   (MUX Header)
02 02                                   (WLP Frame Type and Control Subtype)
1B                                      (Request Parameters)
                                        (TSPEC:
01                                              Service Type
50 97 31 00                                     Mean Data Rate (r)
20 BC BE 00                                     Peak Data Rate (p)
94 11 00 00                                     Maximum Burst Size (b)
DC 05                                           Maximum Packet Size (M)
00 02                                           Minimum Policed Unit (m)
00 00 00 00                                     Requested Service Rate (R)
00 00 00 00                                     Slack Term (s) )
                              (         (Traffic Filtering Parameters:
01                                              Filter Set Count
0F                                              Filter Length
01 00                                           Offset
FF FF FF FF   FF FF FF 00   00 00 00 00         Mask
00 FF FF
F3 01 13 88   00 01 02 00   00 00 00 00         Value)
00 08 00
```

C.4.4 DRP Reservation Response

Table 50 — Field values for a DRP Reservation Response control frame

Field	Value
Protocol ID (MUX Header)	0x0100 (WLP)
WLP Frame Type	2 (Control)
Control Subtype	3 (DRP Reservation Response)

Field		Value	
Response Parameters	Reservation Type (b7–b4)	1 (Hard)	0x1B
	Stream Index (b3–b1)	5	
	Establish (b0)	1	
Response	Reserved (b15–b6)	0	0x0008
	Security violation (b5)	0	
	Not enough resources (b4)	0	
	Not enough MASs available (b3)	1	
	Unsupported traffic filtering parameters (b2)	0	
	Invalid traffic filtering parameters (b1)	0	
	Unsupported capability (b0)	0	

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                   (MUX Header)
02 03                   (WLP Frame Type and Control Subtype)
1B 08 00                (Control Subtype-specific data)
```

C.4.5 Local Cycle Change Request

Table 51 — Field values for a Local Cycle Change Request control frame

Field	Value
Protocol ID (MUX Header)	0x0100 (WLP)
WLP Frame Type	2 (Control)
Control Subtype	4 (Local Cycle Change Request)
Request Count	2
DevAddr 1	0xFF03
Local Cycle Index 1	4
DevAddr 2	0x5A01
Local Cycle Index 2	3

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                   (MUX Header)
02 04                   (WLP Frame Type and Control Subtype)
02 03 FF 04   01 5A 03  (Control Subtype-specific data)
```

C.5 Association frames

This subclause specifies example sequences of association frames. The derivation of the cryptographic numbers used is specified in C.6.

C.5.1 Numeric Comparison

The following example frames are exchanged during a successful enrollment session using the Numeric Comparison association method.

C.5.1.1 D1

Table 52 — Field values for the D1 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		2 (D1)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	2 (D1)
Attribute 3	Attribute Type	0x1047 (UUID-E)
	Attribute Length	16 (0x0010)
	Attribute Value	BFB4E4B0-F179-4E0F-AEC6-94ABB1421477
Attribute 4	Attribute Type	0x2008 (WSS Selection Method)
	Attribute Length	1 (0x0001)
	Attribute Value	1 (Enrollee Selects)
Attribute 5	Attribute Type	0x1011 (Device Name)
	Attribute Length	20 (0x0014)
	Attribute Value	“WiMedia Sample Phone”
Attribute 6	Attribute Type	0x1021 (Manufacturer)
	Attribute Length	7 (0x0007)
	Attribute Value	“WiMedia”
Attribute 7	Attribute Type	0x1023 (Model Name)
	Attribute Length	13 (0x000D)
	Attribute Value	“WiMedia Phone”
Attribute 8	Attribute Type	0x1024 (Model Number)
	Attribute Length	1 (0x0001)
	Attribute Value	“1”

Field			Value
Attribute 9	Attribute Type		0x1042 (Serial Number)
	Attribute Length		6 (0x0006)
	Attribute Value		“123456”
Attribute 10	Attribute Type		0x1054 (Primary Device Type)
	Attribute Length		8 (0x0008)
	Attribute Value	Category ID	10 (0x000A) (Telephone)
		OUI	00-13-88
		OUI Subdivision	0
		Subcategory ID	3 (0x0003)
Attribute 11	Attribute Type		0x200E (WLP Association Error)
	Attribute Length		1 (0x0001)
	Attribute Value		0 (No Error)

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                   (MUX Header)
03                                                      (WLP Frame Type)
02                                                      (Association Subtype)
00 20 01 00   10                                        (WLP Version)
22 10 01 00   02                                        (Message Type)
47 10 10 00                                             (UUID-E)
BF B4 E4 B0   F1 79 4E 0F   AE C6 94 AB   B1 42 14 77
08 20 01 00   01                                        (WSS Selection Method)
11 10 14 00                                             (Device Name)
57 69 4D 65   64 69 61 20   53 61 6D 70   6C 65 20 50
68 6F 6E 65
21 10 07 00   57 69 4D 65   64 69 61                    (Manufacturer)
23 10 0D 00                                             (Model Name)
57 69 4D 65   64 69 61 20   50 68 6F 6E   65
24 10 01 00   31                                        (Model Number)
42 10 06 00   31 32 33 34   35 36                       (Serial Number)
54 10 08 00   0A 00 00 13   88 00 03 00                 (Primary Device Type)
0E 20 01 00   00                                        (WLP Association Error)
```

C.5.1.2 D2

Table 53 — Field values for the D2 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		3 (D2)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10

Field				Value
Attribute 2	Attribute Type			0x1022 (Message Type)
	Attribute Length			1 (0x0001)
	Attribute Value			3 (D2)
Attribute 3	Attribute Type			0x1047 (UUID-E)
	Attribute Length			16 (0x0010)
	Attribute Value			BFB4E4B0-F179-4E0F-AEC6-94ABB1421477
Attribute 4	Attribute Type			0x1048 (UUID-R)
	Attribute Length			16 (0x0010)
	Attribute Value			BFB4E4D2-F179-4E0F-AEC6-94ABB1421477
Attribute 5	Attribute Type			0x2007 (WSS Information)
	Attribute Length			48 (0x0030)
	Attribute Value	Attribute 5a	Attribute Type	0x2001 (WSSID)
			Attribute Length	16 (0x0010)
			Attribute Value	1D015E9D-2930-4C3B-9B50-987121F32E9D
		Attribute 5b	Attribute Type	0x2002 (WSS Name)
			Attribute Length	4 (0x0004)
			Attribute Value	“”Open”
		Attribute 5c	Attribute Type	0x2006 (Accepting Enrollment)
			Attribute Length	1 (0x0001)
			Attribute Value	1
		Attribute 5d	Attribute Type	0x2003 (WSS Secure Status)
			Attribute Length	1 (0x0001)
			Attribute Value	0
		Attribute 5e	Attribute Type	0x2004 (WSS Broadcast Address)
			Attribute Length	6 (0x0006)
			Attribute Value	01-13-88-00-01-CC

Field				Value
Attribute 6	Attribute Type			0x2007 (WSS Information)
	Attribute Length			51 (0x0033)
	Attribute Value	Attribute 6a	Attribute Type	0x2001 (WSSID)
			Attribute Length	16 (0x0010)
			Attribute Value	1D015E9C-2930-4C3B-9B50-987121F32E9D
		Attribute 6b	Attribute Type	0x2002 (WSS Name)
			Attribute Length	7 (0x0007)
			Attribute Value	“Private”
		Attribute 6c	Attribute Type	0x2006 (Accepting Enrollment)
			Attribute Length	1 (0x0001)
			Attribute Value	1
		Attribute 6d	Attribute Type	0x2003 (WSS Secure Status)
			Attribute Length	1 (0x0001)
			Attribute Value	1
		Attribute 6e	Attribute Type	0x2004 (WSS Broadcast Address)
			Attribute Length	6 (0x0006)
			Attribute Value	01-13-88-00-01-4E
Attribute 7	Attribute Type			0x1011 (Device Name)
	Attribute Length			20 (0x0014)
	Attribute Value			“WiMedia SamplePhone”
Attribute 8	Attribute Type			0x1021 (Manufacturer)
	Attribute Length			7 (0x0007)
	Attribute Value			“WiMedia”
Attribute 9	Attribute Type			0x1023 (Model Name)
	Attribute Length			13 (0x000D)
	Attribute Value			“WiMedia Phone”
Attribute 10	Attribute Type			0x1024 (Model Number)
	Attribute Length			1 (0x0001)
	Attribute Value			“1”
Attribute 11	Attribute Type			0x1042 (Serial Number)
	Attribute Length			6 (0x0006)
	Attribute Value			“123478”

Field			Value
Attribute 12	Attribute Type		0x1054 (Primary Device Type)
	Attribute Length		8 (0x0008)
	Attribute Value	Category ID	10 (0x000A) (Telephone)
		OUI	00-13-88
		OUI Subdivision	0
		Subcategory ID	3 (0x0003)
Attribute 13	Attribute Type		0x200E (WLP Association Error)
	Attribute Length		1 (0x0001)
	Attribute Value		0 (No Error)

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                   (MUX Header)
03                                                      (WLP Frame Type)
03                                                      (Association Subtype)
00 20 01 00   10                                        (WLP Version)
22 10 01 00   03                                        (Message Type)
47 10 10 00                                             (UUID-E)
BF B4 E4 B0   F1 79 4E 0F   AE C6 94 AB   B1 42 14 77
48 10 10 00                                             (UUID-R)
BF B4 E4 D2   F1 79 4E 0F   AE C6 94 AB   B1 42 14 77
07 20 30 00                                             (WSS Information attribute contains:
01 20 10 00                                                 WSSID
1D 01 5E 9D   29 30 4C 3B   9B 50 98 71   21 F3 2E 9D
02 20 04 00   4F 70 65 6E                                   WSS Name
06 20 01 00   01                                            Accepting Enrollment
03 20 01 00   00                                            WSS Secure Status
04 20 06 00   01 13 88 00   01 CC                          WSS Broadcast Address)
07 20 33 00                                             (WSS Information attribute contains:
01 20 10 00                                                 WSSID
1D 01 5E 9C   29 30 4C 3B   9B 50 98 71   21 F3 2E 9D
02 20 07 00   50 72 69 76   61 74 65                       WSS Name
06 20 01 00   01                                            Accepting Enrollment
03 20 01 00   01                                            WSS Secure Status
04 20 06 00   01 13 88 00   01 4E                          WSS Broadcast Address)
11 10 14 00                                             (Device Name)
57 69 4D 65   64 69 61 20   53 61 6D 70   6C 65 20 50
68 6F 6E 65
21 10 07 00   57 69 4D 65   64 69 61                     (Manufacturer)
23 10 0D 00                                             (Model Name)
57 69 4D 65   64 69 61 20   50 68 6F 6E   65
24 10 01 00   31                                        (Model Number)
42 10 06 00   31 32 33 34   37 38                       (Serial Number)
54 10 08 00   0A 00 00 13   88 00 03 00                 (Primary Device Type)
0E 20 01 00   00                                        (WLP Association Error)
```

C.5.1.3 E1

Table 54 — Field values for the E1 association frame

Field	Value
Protocol ID (MUX Header)	0x0100 (WLP)
WLP Frame Type	3 (Association)

Field		Value
Association Subtype		32 (E1)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	32 (E1)
Attribute 3	Attribute Type	0x1047 (UUID-E)
	Attribute Length	16 (0x0010)
	Attribute Value	BFB4E4B0-F179-4E0F-AEC6-94ABB1421477
Attribute 4	Attribute Type	0x2001 (WSSID)
	Attribute Length	16 (0x0010)
	Attribute Value	1D015E9C-2930-4C3B-9B50-987121F32E9D
Attribute 5	Attribute Type	0x200B (Enrollee Hash Commitment)
	Attribute Length	32 (0x0020)
	Attribute Value	de77 cd25 d982 d499 f96c ac2f bf9a 5ef3 c6ce f1cf 20e2 2513 db81 a474 0cd1 52f0
Attribute 6	Attribute Type	0x1012 (Device Password ID)
	Attribute Length	2 (0x0002)
	Attribute Value	6 (0x0006) (Numeric Comparison)
Attribute 7	Attribute Type	0x2009 (Association Methods List)
	Attribute Length	2 (0x0002)
	Attribute Value	0x0200 (Numeric Comparison)
Attribute 8	Attribute Type	0x1011 (Device Name)
	Attribute Length	20 (0x0014)
	Attribute Value	“WiMedia Sample Phone”
Attribute 9	Attribute Type	0x1021 (Manufacturer)
	Attribute Length	7 (0x0007)
	Attribute Value	“WiMedia”
Attribute 10	Attribute Type	0x1023 (Model Name)
	Attribute Length	13 (0x000D)
	Attribute Value	“WiMedia Phone”

Field			Value
Attribute 11	Attribute Type		0x1024 (Model Number)
	Attribute Length		1 (0x0001)
	Attribute Value		“1”
Attribute 12	Attribute Type		0x1042 (Serial Number)
	Attribute Length		6 (0x0006)
	Attribute Value		“123456”
Attribute 13	Attribute Type		0x1054 (Primary Device Type)
	Attribute Length		8 (0x0008)
	Attribute Value	Category ID	10 (0x000A) (Telephone)
		OUI	00-13-88
		OUI Subdivision	0
		Subcategory ID	3 (0x0003)

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                   (MUX Header)
03                                                      (WLP Frame Type)
20                                                      (Association Subtype)
00 20 01 00   10                                        (WLP Version)
22 10 01 00   20                                        (Message Type)
47 10 10 00                                             (UUID-E)
BF B4 E4 B0   F1 79 4E 0F   AE C6 94 AB   B1 42 14 77
01 20 10 00                                             (WSSID)
1D 01 5E 9C   29 30 4C 3B   9B 50 98 71   21 F3 2E 9D
0B 20 20 00                                             (Enrollee Hash Commitment)
DE 77 CD 25   D9 82 D4 99   F9 6C AC 2F   BF 9A 5E F3
C6 CE F1 CF   20 E2 25 13   DB 81 A4 74   0C D1 52 F0
12 10 02 00   06 00                                     (Device Password ID)
09 20 02 00   00 02                                     (Association Methods List)
11 10 14 00                                             (Device Name)
57 69 4D 65   64 69 61 20   53 61 6D 70   6C 65 20 50
68 6F 6E 65
21 10 07 00   57 69 4D 65   64 69 61                   (Manufacturer)
23 10 0D 00                                             (Model Name)
57 69 4D 65   64 69 61 20   50 68 6F 6E   65
24 10 01 00   31                                        (Model Number)
42 10 06 00   31 32 33 34   35 36                      (Serial Number)
54 10 08 00   0A 00 00 13   88 00 03 00                (Primary Device Type)
```

C.5.1.4 E2

Table 55 — Field values for the E2 association frame

Field	Value
Protocol ID (MUX Header)	0x0100 (WLP)
WLP Frame Type	3 (Association)
Association Subtype	33 (E2)

Field		Value
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	33 (E2)
Attribute 3	Attribute Type	0x1039 (Registrar Nonce)
	Attribute Length	16 (0x0010)
	Attribute Value	0xADEE 5C75 F058 24FE 4D51 1B62 39B2 96D0
Attribute 4	Attribute Type	0x1048 (UUID-R)
	Attribute Length	16 (0x0010)
	Attribute Value	BFB4E4D2-F179-4E0F-AEC6-94ABB1421477
Attribute 5	Attribute Type	0x1032 (Public Key)
	Attribute Length	384 (0x0180)
	Attribute Value	0xDC14 C6F6 D85B 3D58 B54A BB30 6D55 6829 2ED7 85D3 9ED7 3643 666A 1B4A 4684 654F 88BB EDF0 414C 59C7 0DD9 90B4 47B3 C325 0A4A 2367 3EA9 361A 79BE 3376 0906 EF12 7627 FA9E 7F91 07E7 3675 9CFF 990C 44FC E240 7E7C E1C7 D61A 83B8 5C82 85A9 BF94 7CC1 E582 642A 8A86 3E4E 0D57 F258 4B25 5229 C4D3 5355 1E86 AC2B BCE4 13C7 E554 1CC2 E68D 7101 D578 30CD E1C9 1BD4 8C03 D190 1472 01F3 9697 F65C C2F4 45E8 5162 3BEA 585C 8205 D8E8 CA91 B54D AEFB 6FE5 AC46 E942 B5EA 6E04 495B D2F6 CB11 88C1 B44A 342E 5DAB 2917 165E 0935 D743 69B7 6698 68C9 D4D5 B148 33F3 1E56 9499 1E73 353A 33F5 F4DC 61FF 5752 517B 7180 6DA2 E47E FC78 D22D D8DA C4F1 1501 9D57 5D60 B787 6140 4413 BFF6 E314 329B F1E5 2B92 38F8 7964 A5A3 00C7 26C0 950F AC94 6459 3C30 6ECE 4D92 813F D714 2E16 18B3 EFBB 3FEA 25F9 E177 0859 2507 D8BE 73EF D569 761E 7FF4 B016 EDD0 C5C3 85A8 EC16 1A44 F2D6 7C1C 6B39 7D8F 6C3F A797 BCD9 5E3F B8F4 ECBA 7EBF 6620 570E F491 4E75 EAF9 752B A471 FAF7 CCC5 5373 069C 2153 1194
Attribute 6	Attribute Type	0x1012 (Device Password ID)
	Attribute Length	2 (0x0002)
	Attribute Value	6 (0x0006) (Numeric Comparison)
Attribute 7	Attribute Type	0x200A (Selected Association Method)
	Attribute Length	2 (0x0002)
	Attribute Value	0x0200 (Numeric Comparison)
Attribute 8	Attribute Type	0x1011 (Device Name)
	Attribute Length	20 (0x0014)
	Attribute Value	“WiMedia Sample Phone”

Field			Value
Attribute 9	Attribute Type		0x1021 (Manufacturer)
	Attribute Length		7 (0x0007)
	Attribute Value		“WiMedia”
Attribute 10	Attribute Type		0x1023 (Model Name)
	Attribute Length		13 (0x000D)
	Attribute Value		“WiMedia Phone”
Attribute 11	Attribute Type		0x1024 (Model Number)
	Attribute Length		1 (0x0001)
	Attribute Value		“1”
Attribute 12	Attribute Type		0x1042 (Serial Number)
	Attribute Length		6 (0x0006)
	Attribute Value		“123478”
Attribute 13	Attribute Type		0x1054 (Primary Device Type)
	Attribute Length		8 (0x0008)
	Attribute Value	Category ID	10 (0x000A) (Telephone)
		OUI	00-13-88
		OUI Subdivision	0
		Subcategory ID	3 (0x0003)

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                   (MUX Header)
03                                                      (WLP Frame Type)
21                                                      (Association Subtype)
00 20 01 00   10                                        (WLP Version)
22 10 01 00   21                                        (Message Type)
39 10 10 00                                             (Registrar Nonce)
D0 96 B2 39   62 1B 51 4D   FE 24 58 F0   75 5C EE AD
48 10 10 00                                             (UUID-R)
BF B4 E4 D2   F1 79 4E 0F   AE C6 94 AB   B1 42 14 77
32 10 80 01                                             (Public Key)
94 11 53 21   9C 06 73 53   C5 CC F7 FA   71 A4 2B 75
F9 EA 75 4E   91 F4 0E 57   20 66 BF 7E   BA EC F4 B8
3F 5E D9 BC   97 A7 3F 6C   8F 7D 39 6B   1C 7C D6 F2
44 1A 16 EC   A8 85 C3 C5   D0 ED 16 B0   F4 7F 1E 76
69 D5 EF 73   BE D8 07 25   59 08 77 E1   F9 25 EA 3F
BB EF B3 18   16 2E 14 D7   3F 81 92 4D   CE 6E 30 3C
59 64 94 AC   0F 95 C0 26   C7 00 A3 A5   64 79 F8 38
92 2B E5 F1   9B 32 14 E3   F6 BF 13 44   40 61 87 B7
60 5D 57 9D   01 15 F1 C4   DA D8 2D D2   78 FC 7E E4
A2 6D 80 71   7B 51 52 57   FF 61 DC F4   F5 33 3A 35
73 1E 99 94   56 1E F3 33   48 B1 D5 D4   C9 68 98 66
B7 69 43 D7   35 09 5E 16   17 29 AB 5D   2E 34 4A B4
C1 88 11 CB   F6 D2 5B 49   04 6E EA B5   42 E9 46 AC
E5 6F FB AE   4D B5 91 CA   E8 D8 05 82   5C 58 EA 3B
62 51 E8 45   F4 C2 5C F6   97 96 F3 01   72 14 90 D1
03 8C D4 1B   C9 E1 CD 30   78 D5 01 71   8D E6 C2 1C
54 E5 C7 13   E4 BC 2B AC   86 1E 55 53   D3 C4 29 52
25 4B 58 F2   57 0D 4E 3E   86 8A 2A 64   82 E5 C1 7C
```

```
94 BF A9 85   82 5C B8 83   1A D6 C7 E1   7C 7E 40 E2
FC 44 0C 99   FF 9C 75 36   E7 07 91 7F   9E FA 27 76
12 EF 06 09   76 33 BE 79   1A 36 A9 3E   67 23 4A 0A
25 C3 B3 47   B4 90 D9 0D   C7 59 4C 41   F0 ED BB 88
4F 65 84 46   4A 1B 6A 66   43 36 D7 9E   D3 85 D7 2E
29 68 55 6D   30 BB 4A B5   58 3D 5B D8   F6 C6 14 DC
12 10 02 00   06 00                                    (Device Password ID)
0A 20 02 00   00 02                                    (Selected Association Method)
11 10 14 00                                            (Device Name)
57 69 4D 65   64 69 61 20   53 61 6D 70   6C 65 20 50
68 6F 6E 65
21 10 07 00   57 69 4D 65   64 69 61                   (Manufacturer)
23 10 0D 00                                            (Model Name)
57 69 4D 65   64 69 61 20   50 68 6F 6E   65
24 10 01 00   31                                       (Model Number)
42 10 06 00   31 32 33 34   37 38                      (Serial Number)
54 10 08 00   0A 00 00 13   88 00 03 00                (Primary Device Type)
```

C.5.1.5 M1

Table 56 — Field values for the M1 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		4 (M1)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	4 (M1)
Attribute 3	Attribute Type	0x101A (Enrollee Nonce)
	Attribute Length	16 (0x0010)
	Attribute Value	0x2696 1EAD 7E25 C69D 3C2F 7DB6 2444 A5B7
Attribute 4	Attribute Type	0x1039 (Registrar Nonce)
	Attribute Length	16 (0x0010)
	Attribute Value	0xADEE 5C75 F058 24FE 4D51 1B62 39B2 96D0
Attribute 5	Attribute Type	0x1047 (UUID-E)
	Attribute Length	16 (0x0010)
	Attribute Value	BFB4E4B0-F179-4E0F-AEC6-94ABB1421477

Field		Value
Attribute 6	Attribute Type	0x1032 (Public Key)
	Attribute Length	384 (0x0180)
	Attribute Value	0x5A0D 3D4E 049F AA93 9FFA 6A37 5B9C 3C16 A4C3 9753 D19F F7DA 36BC 391E A72F C0F6 8C92 9BDB 4005 52ED 84E0 900C 7A44 C322 2FD5 4D71 4825 6862 886B FB40 16BD 2D03 C4C4 CF47 6567 C291 770E 47BD 59D0 AA53 23CF DDFC 5596 E0D6 558C 480E E8B0 C625 9983 4D45 81A7 96A0 1981 4687 8916 4504 AFBD 29CE 9936 E86A 290C 5F00 F8BA 986B 4801 0F3E 5C07 9C7F 351D DCA2 EE1F D508 46B3 7BF7 463C 2B0F 3D00 1B13 17AC 3069 CD89 E2E4 927E D3D4 0875 A604 9AF6 49D2 DC34 9DB5 995A 7525 D70A 3A1C 9B67 3F54 82F8 3343 BD90 D45E 9C39 62DC 4A4B F2B4 ADB3 7E91 66B2 DDB3 1CCF 11C5 B9E6 C98E 0A9A 3377 ABBA 56B0 F428 3B2E AA69 F536 8BC1 07E1 C225 99F8 8DD1 924D 0899 C5F1 5346 2C91 1A82 9307 8AEF EE9F B238 9A78 5483 3FCE A61C FECB B49F 828C 361A 981A 5FED ECF1 3796 AE36 E36C 15A1 6670 AF96 996C 3C45 A30E 900E 18C8 58F6 232B 5F70 72BD D9E4 7D7F C612 46EF 5D19 7657 39F3 8509 2843 79BC 319D 9409 E8FE 236B D29B 0335 A5BC 5BB0 424E E44D E8A1 9F86 4A15 9FDA 907D 6F5A 30EB C0A1 7E36 28E4 90E5
Attribute 7	Attribute Type	0x1012 (Device Password ID)
	Attribute Length	2 (0x0002)
	Attribute Value	6 (0x0006) (Numeric Comparison)

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                   (MUX Header)
03                                                      (WLP Frame Type)
04                                                      (Association Subtype)
00 20 01 00   10                                        (WLP Version)
22 10 01 00   04                                        (Message Type)
1A 10 10 00                                             (Enrollee Nonce)
B7 A5 44 24   B6 7D 2F 3C   9D C6 25 7E   AD 1E 96 26
39 10 10 00                                             (Registrar Nonce)
D0 96 B2 39   62 1B 51 4D   FE 24 58 F0   75 5C EE AD
47 10 10 00                                             (UUID-E)
BF B4 E4 B0   F1 79 4E 0F   AE C6 94 AB   B1 42 14 77
32 10 80 01                                             (Public Key)
E5 90 E4 28   36 7E A1 C0   EB 30 5A 6F   7D 90 DA 9F
15 4A 86 9F   A1 E8 4D E4   4E 42 B0 5B   BC A5 35 03
9B D2 6B 23   FE E8 09 94   9D 31 BC 79   43 28 09 85
F3 39 57 76   19 5D EF 46   12 C6 7F 7D   E4 D9 BD 72
70 5F 2B 23   F6 58 C8 18   0E 90 0E A3   45 3C 6C 99
96 AF 70 66   A1 15 6C E3   36 AE 96 37   F1 EC ED 5F
1A 98 1A 36   8C 82 9F B4   CB FE 1C A6   CE 3F 83 54
78 9A 38 B2   9F EE EF 8A   07 93 82 1A   91 2C 46 53
F1 C5 99 08   4D 92 D1 8D   F8 99 25 C2   E1 07 C1 8B
36 F5 69 AA   2E 3B 28 F4   B0 56 BA AB   77 33 9A 0A
8E C9 E6 B9   C5 11 CF 1C   B3 DD B2 66   91 7E B3 AD
B4 F2 4B 4A   DC 62 39 9C   5E D4 90 BD   43 33 F8 82
54 3F 67 9B   1C 3A 0A D7   25 75 5A 99   B5 9D 34 DC
D2 49 F6 9A   04 A6 75 08   D4 D3 7E 92   E4 E2 89 CD
69 30 AC 17   13 1B 00 3D   0F 2B 3C 46   F7 7B B3 46
08 D5 1F EE   A2 DC 1D 35   7F 9C 07 5C   3E 0F 01 48
6B 98 BA F8   00 5F 0C 29   6A E8 36 99   CE 29 BD AF
04 45 16 89   87 46 81 19   A0 96 A7 81   45 4D 83 99
25 C6 B0 E8   0E 48 8C 55   D6 E0 96 55   FC DD CF 23
53 AA D0 59   BD 47 0E 77   91 C2 67 65   47 CF C4 C4
03 2D BD 16   40 FB 6B 88   62 68 25 48   71 4D D5 2F
22 C3 44 7A   0C 90 E0 84   ED 52 05 40   DB 9B 92 8C
F6 C0 2F A7   1E 39 BC 36   DA F7 9F D1   53 97 C3 A4
16 3C 9C 5B   37 6A FA 9F   93 AA 9F 04   4E 3D 0D 5A
12 10 02 00   06 00                                    (Device Password ID)
```

C.5.1.6 M2

Table 57 — Field values for the M2 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		5 (M2)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	5 (M2)
Attribute 3	Attribute Type	0x101A (Enrollee Nonce)
	Attribute Length	16 (0x0010)
	Attribute Value	0x2696 1EAD 7E25 C69D 3C2F 7DB6 2444 A5B7
Attribute 4	Attribute Type	0x1039 (Registrar Nonce)
	Attribute Length	16 (0x0010)
	Attribute Value	0xADEE 5C75 F058 24FE 4D51 1B62 39B2 96D0
Attribute 5	Attribute Type	0x1048 (UUID-R)
	Attribute Length	16 (0x0010)
	Attribute Value	BFB4E4D2-F179-4E0F-AEC6-94ABB1421477
Attribute 6	Attribute Type	0x1005 (Authenticator)
	Attribute Length	8 (0x0008)
	Attribute Value	fac7 1f9c 0529 3ff7

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                   (MUX Header)
03                                                      (WLP Frame Type)
05                                                      (Association Subtype)
00 20 01 00   10                                        (WLP Version)
22 10 01 00   05                                        (Message Type)
1A 10 10 00                                             (Enrollee Nonce)
B7 A5 44 24   B6 7D 2F 3C   9D C6 25 7E   AD 1E 96 26
39 10 10 00                                             (Registrar Nonce)
D0 96 B2 39   62 1B 51 4D   FE 24 58 F0   75 5C EE AD
48 10 10 00                                             (UUID-R)
BF B4 E4 D2   F1 79 4E 0F   AE C6 94 AB   B1 42 14 77
05 10 08 00   FA C7 1F 9C   05 29 3F F7                 (Authenticator)
```

C.5.1.7 M7

Table 58 — Field values for the M7 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		11 (M7)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	11 (M7)
Attribute 3	Attribute Type	0x1039 (Registrar Nonce)
	Attribute Length	16 (0x0010)
	Attribute Value	0xADEE 5C75 F058 24FE 4D51 1B62 39B2 96D0
Attribute 4	Attribute Type	0x1005 (Authenticator)
	Attribute Length	8 (0x0008)
	Attribute Value	8721 22c6 2097 42fc

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                   (MUX Header)
03                                                      (WLP Frame Type)
0B                                                      (Association Subtype)
00 20 01 00   10                                        (WLP Version)
22 10 01 00   0B                                        (Message Type)
39 10 10 00                                             (Registrar Nonce)
D0 96 B2 39   62 1B 51 4D   FE 24 58 F0   75 5C EE AD
05 10 08 00   87 21 22 C6   20 97 42 FC                 (Authenticator)
```

C.5.1.8 M8

Table 59 — Field values for the M8 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		12 (M8)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10

Field					Value	
Attribute 2	Attribute Type				0x1022 (Message Type)	
	Attribute Length				1 (0x0001)	
	Attribute Value				12 (M8)	
Attribute 3	Attribute Type				0x101A (Enrollee Nonce)	
	Attribute Length				16 (0x0010)	
	Attribute Value				0x2696 1EAD 7E25 C69D 3C2F 7DB6 2444 A5B7	
Attribute 4	Attribute Type				0x1018 (Encrypted Settings)	
	Attribute Length				96 (0x0060)	
	Attribute Value	IV			16e3 affd c1cd 1ddb f923 841f 3a69 8805	
		Attribute 4a	Attribute Type		0x2001 (WSSID)	Encrypted: 0eed 3a42 b4b9 ac56 59b0 a34e cae4 07c7 2b02 8ba2 d867 4b44 6992 a844 5832 0e46 cfc2 f820 eb49 1f03 45b5 3b87 ba92 4caf 2387 071d aff2 00ad 7dbb 9e0c 9f6d da31 b15e 4cba 47a8 138b 945f c008 1248 1399
			Attribute Length		16 (0x0010)	
			Attribute Value		1D015E9C-2930-4C3B-9B50-987121F32E9D	
		Attribute 4b	Attribute Type		0x2002 (WSS Name)	
			Attribute Length		7 (0x0007)	
			Attribute Value		“Private”	
		Attribute 4c	Attribute Type		0x2004 (WSS Broadcast Address)	
			Attribute Length		6 (0x0006)	
			Attribute Value		01-13-88-00-01-4E	
		Attribute 4d	Attribute Type		0x2005 (WSS Master Key)	
			Attribute Length		16 (0x0010)	
			Attribute Value		3f87 1019 c4ee 73f5 811b bb45 7103 35ce	
		Attribute 4e	Attribute Type		0x101E (Key Wrap Authenticator)	
			Attribute Length		8 (0x0008)	
			Attribute Value		1122 4d84 acea 8105	
		pad			0707 0707 0707 07	
Attribute 5	Attribute Type				0x1005 (Authenticator)	
	Attribute Length				8 (0x0008)	
	Attribute Value				9114 eca4 99ca 4c7e	

The octets that make up the Encrypted Settings attribute cleartext are:

```
01 20 10 00                                          (WSSID)
1D 01 5E 9C   29 30 4C 3B   9B 50 98 71   21 F3 2E 9D
```

```
02 20 07 00   50 72 69 76   61 74 65                  (WSS Name)
04 20 06 00   01 13 88 00   01 4E                     (WSS Broadcast Address)
05 20 10 00                                           (WSS Master Key)
3F 87 10 19   C4 EE 73 F5   81 1B BB 45   71 03 35 CE
1E 10 08 00   11 22 4D 84   AC EA 81 05               (Key Wrap Authenticator)
07 07 07 07   07 07 07                                (pad)
```

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                 (MUX Header)
03                                                    (WLP Frame Type)
0C                                                    (Association Subtype)
00 20 01 00   10                                      (WLP Version)
22 10 01 00   0C                                      (Message Type)
1A 10 10 00                                           (Enrollee Nonce)
B7 A5 44 24   B6 7D 2F 3C   9D C6 25 7E   AD 1E 96 26
18 10 60 00                                           (Encrypted Settings)
16 E3 AF FD   C1 CD 1D DB   F9 23 84 1F   3A 69 88 05
0E ED 3A 42   B4 B9 AC 56   59 B0 A3 4E   CA E4 07 C7
2B 02 8B A2   D8 67 4B 44   69 92 A8 44   58 32 0E 46
CF C2 F8 20   EB 49 1F 03   45 B5 3B 87   BA 92 4C AF
23 87 07 1D   AF F2 00 AD   7D BB 9E 0C   9F 6D DA 31
B1 5E 4C BA   47 A8 13 8B   94 5F C0 08   12 48 13 99
05 10 08 00   91 14 EC A4   99 CA 4C 7E               (Authenticator)
```

C.5.2 Registrar-display

The following example frames are exchanged during a successful Registrar-display enrollment session.

C.5.2.1 D1

Table 60 — Field values for the D1 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		2 (D1)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	2 (D1)
Attribute 3	Attribute Type	0x1047 (UUID-E)
	Attribute Length	16 (0x0010)
	Attribute Value	BFB4F9C1-F179-4E0F-AEC6-94ABB1421477
Attribute 4	Attribute Type	0x2008 (WSS Selection Method)
	Attribute Length	1 (0x0001)
	Attribute Value	2 (Registrar Selects)

Field			Value
Attribute 5	Attribute Type		0x1011 (Device Name)
	Attribute Length		13 (0x000D)
	Attribute Value		“Sample Remote”
Attribute 6	Attribute Type		0x1021 (Manufacturer)
	Attribute Length		7 (0x0007)
	Attribute Value		“WiMedia”
Attribute 7	Attribute Type		0x1023 (Model Name)
	Attribute Length		21 (0x0015)
	Attribute Value		“WiMedia Sample Remote”
Attribute 8	Attribute Type		0x1024 (Model Number)
	Attribute Length		1 (0x0001)
	Attribute Value		“4”
Attribute 9	Attribute Type		0x1042 (Serial Number)
	Attribute Length		6 (0x0006)
	Attribute Value		“1238E1”
Attribute 10	Attribute Type		0x1054 (Primary Device Type)
	Attribute Length		8 (0x0008)
	Attribute Value	Category ID	2 (0x0002) (Input Device)
		OUI	00-13-88
		OUI Subdivision	0
		Subcategory ID	6 (0x0006)
Attribute 11	Attribute Type		0x200E (WLP Association Error)
	Attribute Length		1 (0x0001)
	Attribute Value		0 (No Error)

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                   (MUX Header)
03                                                      (WLP Frame Type)
02                                                      (Association Subtype)
00 20 01 00   10                                        (WLP Version)
22 10 01 00   02                                        (Message Type)
47 10 10 00                                             (UUID-E)
BF B4 F9 C1   F1 79 4E 0F   AE C6 94 AB   B1 42 14 77
08 20 01 00   02                                        (WSS Selection Method)
11 10 0D 00                                             (Device Name)
53 61 6D 70   6C 65 20 52   65 6D 6F 74   65
21 10 07 00   57 69 4D 65   64 69 61                    (Manufacturer)
23 10 15 00                                             (Model Name)
57 69 4D 65   64 69 61 20   53 61 6D 70   6C 65 20 52
65 6D 6F 74   65
24 10 01 00   34                                        (Model Number)
42 10 06 00   31 32 33 38   45 31                       (Serial Number)
```

```
54 10 08 00   02 00 00 13   88 00 06 00              (Primary Device Type)
0E 20 01 00   00                                     (WLP Association Error)
```

C.5.2.2 D2

Table 61 — Field values for the D2 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		3 (D2)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	3 (D2)
Attribute 3	Attribute Type	0x1047 (UUID-E)
	Attribute Length	16 (0x0010)
	Attribute Value	BFB4F9C1-F179-4E0F-AEC6-94ABB1421477
Attribute 4	Attribute Type	0x1048 (UUID-R)
	Attribute Length	16 (0x0010)
	Attribute Value	BFB4F9B6-F179-4E0F-AEC6-94ABB1421477

Field				Value
Attribute 5	Attribute Type			0x2007 (WSS Information)
	Attribute Length			50 (0x0032)
	Attribute Value	Attribute 5a	Attribute Type	0x2001 (WSSID)
			Attribute Length	16 (0x0010)
			Attribute Value	A8DE9103-4047-41B0-8BA7-0286EBCBA8F1
		Attribute 5b	Attribute Type	0x2002 (WSS Name)
			Attribute Length	6 (0x0006)
			Attribute Value	“My WSS”
		Attribute 5c	Attribute Type	0x2006 (Accepting Enrollment)
			Attribute Length	1 (0x0001)
			Attribute Value	1
		Attribute 5d	Attribute Type	0x2003 (WSS Secure Status)
			Attribute Length	1 (0x0001)
			Attribute Value	1
		Attribute 5e	Attribute Type	0x2004 (WSS Broadcast Address)
			Attribute Length	6 (0x0006)
			Attribute Value	01-13-88-00-01-CC
Attribute 6	Attribute Type			0x1011 (Device Name)
	Attribute Length			14 (0x000E)
	Attribute Value			“Sample Display”
Attribute 7	Attribute Type			0x1021 (Manufacturer)
	Attribute Length			7 (0x0007)
	Attribute Value			“WiMedia”
Attribute 8	Attribute Type			0x1023 (Model Name)
	Attribute Length			22 (0x0016)
	Attribute Value			“WiMedia Sample Display”
Attribute 9	Attribute Type			0x1024 (Model Number)
	Attribute Length			1 (0x0001)
	Attribute Value			“2”
Attribute 10	Attribute Type			0x1042 (Serial Number)
	Attribute Length			6 (0x0006)
	Attribute Value			1238D6

Field			Value
Attribute 11	Attribute Type		0x1054 (Primary Device Type)
	Attribute Length		8 (0x0008)
	Attribute Value	Category ID	7 (0x0007) (Display)
		OUI	00-13-88
		OUI Subdivision	0
		Subcategory ID	4 (0x0004)
Attribute 12	Attribute Type		0x200E (WLP Association Error)
	Attribute Length		1 (0x0001)
	Attribute Value		0 (No Error)

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                   (MUX Header)
03                                                      (WLP Frame Type)
03                                                      (Association Subtype)
00 20 01 00   10                                        (WLP Version)
22 10 01 00   03                                        (Message Type)
47 10 10 00                                             (UUID-E)
BF B4 F9 C1   F1 79 4E 0F   AE C6 94 AB   B1 42 14 77
48 10 10 00                                             (UUID-R)
BF B4 F9 B6   F1 79 4E 0F   AE C6 94 AB   B1 42 14 77
07 20 32 00                                             (WSS Information attribute contains:
01 20 10 00                                                 WSSID
A8 DE 91 03   40 47 41 B0   8B A7 02 86   EB CB A8 F1
02 20 06 00   4D 79 20 57   53 53                           WSS Name
06 20 01 00   01                                            Accepting Enrollment
03 20 01 00   01                                            WSS Secure Status
04 20 06 00   01 13 88 00   01 CC                         WSS Broadcast Address)
11 10 0E 00                                             (Device Name)
53 61 6D 70   6C 65 20 44   69 73 70 6C   61 79
21 10 07 00   57 69 4D 65   64 69 61                     (Manufacturer)
23 10 16 00                                             (Model Name)
57 69 4D 65   64 69 61 20   53 61 6D 70   6C 65 20 44
69 73 70 6C   61 79
24 10 01 00   32                                        (Model Number)
42 10 06 00   31 32 33 38   44 36                       (Serial Number)
54 10 08 00   07 00 00 13   88 00 04 00                 (Primary Device Type)
0E 20 01 00   00                                        (WLP Association Error)
```

C.5.2.3 E1

Table 62 — Field values for the E1 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		32 (E1)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10

Field		Value
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	32 (E1)
Attribute 3	Attribute Type	0x1047 (UUID-E)
	Attribute Length	16 (0x0010)
	Attribute Value	BFB4F9C1-F179-4E0F-AEC6-94ABB1421477
Attribute 4	Attribute Type	0x2001 (WSSID)
	Attribute Length	16 (0x0010)
	Attribute Value	A8DE9103-4047-41B0-8BA7-0286EBCBA8F1
Attribute 5	Attribute Type	0x200B (Enrollee Hash Commitment)
	Attribute Length	32 (0x0020)
	Attribute Value	de77 cd25 d982 d499 f96c ac2f bf9a 5ef3 c6ce f1cf 20e2 2513 db81 a474 0cd1 52f0
Attribute 6	Attribute Type	0x1012 (Device Password ID)
	Attribute Length	2 (0x0002)
	Attribute Value	2 (0x0002) (Machine-specified)
Attribute 7	Attribute Type	0x2009 (Association Methods List)
	Attribute Length	4 (0x0004)
	Attribute Value	0x0100 (Registrar-display) 0x0200 (Numeric Comparison)
Attribute 5	Attribute Type	0x1011 (Device Name)
	Attribute Length	13 (0x000D)
	Attribute Value	“Sample Remote”
Attribute 6	Attribute Type	0x1021 (Manufacturer)
	Attribute Length	7 (0x0007)
	Attribute Value	“WiMedia”
Attribute 7	Attribute Type	0x1023 (Model Name)
	Attribute Length	21 (0x0015)
	Attribute Value	“WiMedia Sample Remote”
Attribute 8	Attribute Type	0x1024 (Model Number)
	Attribute Length	1 (0x0001)
	Attribute Value	“4”

Field			Value
Attribute 9	Attribute Type		0x1042 (Serial Number)
	Attribute Length		6 (0x0006)
	Attribute Value		“1238E1”
Attribute 10	Attribute Type		0x1054 (Primary Device Type)
	Attribute Length		8 (0x0008)
	Attribute Value	Category ID	2 (0x0002) (Input Device)
		OUI	00-13-88
		OUI Subdivision	0
		Subcategory ID	6 (0x0006)

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                   (MUX Header)
03                                                      (WLP Frame Type)
20                                                      (Association Subtype)
00 20 01 00   10                                        (WLP Version)
22 10 01 00   20                                        (Message Type)
47 10 10 00                                             (UUID-E)
BF B4 F9 C1   F1 79 4E 0F   AE C6 94 AB   B1 42 14 77
01 20 10 00                                             (WSSID)
A8 DE 91 03   40 47 41 B0   8B A7 02 86   EB CB A8 F1
0B 20 20 00                                             (Enrollee Hash Commitment)
DE 77 CD 25   D9 82 D4 99   F9 6C AC 2F   BF 9A 5E F3
C6 CE F1 CF   20 E2 25 13   DB 81 A4 74   0C D1 52 F0
12 10 02 00   02 00                                     (Device Password ID)
09 20 04 00   00 01 00 02                              (Association Methods List)
11 10 0D 00                                             (Device Name)
53 61 6D 70   6C 65 20 52   65 6D 6F 74   65
21 10 07 00   57 69 4D 65   64 69 61                    (Manufacturer)
23 10 15 00                                             (Model Name)
57 69 4D 65   64 69 61 20   53 61 6D 70   6C 65 20 52
65 6D 6F 74   65
24 10 01 00   34                                        (Model Number)
42 10 06 00   31 32 33 38   45 31                       (Serial Number)
54 10 08 00   02 00 00 13   88 00 06 00                 (Primary Device Type)
```

C.5.2.4 E2

Table 63 — Field values for the E2 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		33 (E2)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10

Field		Value
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	33 (E2)
Attribute 3	Attribute Type	0x1039 (Registrar Nonce)
	Attribute Length	16 (0x0010)
	Attribute Value	0xADEE 5C75 F058 24FE 4D51 1B62 39B2 96D0
Attribute 4	Attribute Type	0x1048 (UUID-R)
	Attribute Length	16 (0x0010)
	Attribute Value	BFB4F9B6-F179-4E0F-AEC6-94ABB1421477
Attribute 5	Attribute Type	0x1032 (Public Key)
	Attribute Length	384 (0x0180)
	Attribute Value	0xDC14 C6F6 D85B 3D58 B54A BB30 6D55 6829 2ED7 85D3 9ED7 3643 666A 1B4A 4684 654F 88BB EDF0 414C 59C7 0DD9 90B4 47B3 C325 0A4A 2367 3EA9 361A 79BE 3376 0906 EF12 7627 FA9E 7F91 07E7 3675 9CFF 990C 44FC E240 7E7C E1C7 D61A 83B8 5C82 85A9 BF94 7CC1 E582 642A 8A86 3E4E 0D57 F258 4B25 5229 C4D3 5355 1E86 AC2B BCE4 13C7 E554 1CC2 E68D 7101 D578 30CD E1C9 1BD4 8C03 D190 1472 01F3 9697 F65C C2F4 45E8 5162 3BEA 585C 8205 D8E8 CA91 B54D AEFB 6FE5 AC46 E942 B5EA 6E04 495B D2F6 CB11 88C1 B44A 342E 5DAB 2917 165E 0935 D743 69B7 6698 68C9 D4D5 B148 33F3 1E56 9499 1E73 353A 33F5 F4DC 61FF 5752 517B 7180 6DA2 E47E FC78 D22D D8DA C4F1 1501 9D57 5D60 B787 6140 4413 BFF6 E314 329B F1E5 2B92 38F8 7964 A5A3 00C7 26C0 950F AC94 6459 3C30 6ECE 4D92 813F D714 2E16 18B3 EFBB 3FEA 25F9 E177 0859 2507 D8BE 73EF D569 761E 7FF4 B016 EDD0 C5C3 85A8 EC16 1A44 F2D6 7C1C 6B39 7D8F 6C3F A797 BCD9 5E3F B8F4 ECBA 7EBF 6620 570E F491 4E75 EAF9 752B A471 FAF7 CCC5 5373 069C 2153 1194
Attribute 6	Attribute Type	0x1012 (Device Password ID)
	Attribute Length	2 (0x0002)
	Attribute Value	2 (0x0002) (Machine-generated)
Attribute 7	Attribute Type	0x200A (Selected Association Method)
	Attribute Length	2 (0x0002)
	Attribute Value	0x0100 (Registrar-display)
Attribute 8	Attribute Type	0x1011 (Device Name)
	Attribute Length	14 (0x000E)
	Attribute Value	“Sample Display”
Attribute 9	Attribute Type	0x1021 (Manufacturer)
	Attribute Length	7 (0x0007)
	Attribute Value	“WiMedia”

Field			Value
Attribute 10	Attribute Type		0x1023 (Model Name)
	Attribute Length		22 (0x0016)
	Attribute Value		“WiMedia Sample Display”
Attribute 11	Attribute Type		0x1024 (Model Number)
	Attribute Length		1 (0x0001)
	Attribute Value		“2”
Attribute 12	Attribute Type		0x1042 (Serial Number)
	Attribute Length		6 (0x0006)
	Attribute Value		1238D6
Attribute 13	Attribute Type		0x1054 (Primary Device Type)
	Attribute Length		8 (0x0008)
	Attribute Value	Category ID	7 (0x0007) (Display)
		OUI	00-13-88
		OUI Subdivision	0
		Subcategory ID	4 (0x0004)

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                   (MUX Header)
03                                                      (WLP Frame Type)
21                                                      (Association Subtype)
00 20 01 00   10                                        (WLP Version)
22 10 01 00   21                                        (Message Type)
39 10 10 00                                             (Registrar Nonce)
D0 96 B2 39   62 1B 51 4D   FE 24 58 F0   75 5C EE AD
48 10 10 00                                             (UUID-R)
BF B4 F9 B6   F1 79 4E 0F   AE C6 94 AB   B1 42 14 77
32 10 80 01                                             (Public Key)
94 11 53 21   9C 06 73 53   C5 CC F7 FA   71 A4 2B 75
F9 EA 75 4E   91 F4 0E 57   20 66 BF 7E   BA EC F4 B8
3F 5E D9 BC   97 A7 3F 6C   8F 7D 39 6B   1C 7C D6 F2
44 1A 16 EC   A8 85 C3 C5   D0 ED 16 B0   F4 7F 1E 76
69 D5 EF 73   BE D8 07 25   59 08 77 E1   F9 25 EA 3F
BB EF B3 18   16 2E 14 D7   3F 81 92 4D   CE 6E 30 3C
59 64 94 AC   0F 95 C0 26   C7 00 A3 A5   64 79 F8 38
92 2B E5 F1   9B 32 14 E3   F6 BF 13 44   40 61 87 B7
60 5D 57 9D   01 15 F1 C4   DA D8 2D D2   78 FC 7E E4
A2 6D 80 71   7B 51 52 57   FF 61 DC F4   F5 33 3A 35
73 1E 99 94   56 1E F3 33   48 B1 D5 D4   C9 68 98 66
B7 69 43 D7   35 09 5E 16   17 29 AB 5D   2E 34 4A B4
C1 88 11 CB   F6 D2 5B 49   04 6E EA B5   42 E9 46 AC
E5 6F FB AE   4D B5 91 CA   E8 D8 05 82   5C 58 EA 3B
62 51 E8 45   F4 C2 5C F6   97 96 F3 01   72 14 90 D1
03 8C D4 1B   C9 E1 CD 30   78 D5 01 71   8D E6 C2 1C
54 E5 C7 13   E4 BC 2B AC   86 1E 55 53   D3 C4 29 52
25 4B 58 F2   57 0D 4E 3E   86 8A 2A 64   82 E5 C1 7C
94 BF A9 85   82 5C B8 83   1A D6 C7 E1   7C 7E 40 E2
FC 44 0C 99   FF 9C 75 36   E7 07 91 7F   9E FA 27 76
12 EF 06 09   76 33 BE 79   1A 36 A9 3E   67 23 4A 0A
25 C3 B3 47   B4 90 D9 0D   C7 59 4C 41   F0 ED BB 88
4F 65 84 46   4A 1B 6A 66   43 36 D7 9E   D3 85 D7 2E
29 68 55 6D   30 BB 4A B5   58 3D 5B D8   F6 C6 14 DC
```

```
12 10 02 00   02 00                                     (Device Password ID)
0A 20 02 00   00 01                                     (Selected Association Method)
11 10 0E 00                                             (Device Name)
53 61 6D 70   6C 65 20 44   69 73 70 6C   61 79
21 10 07 00   57 69 4D 65   64 69 61                    (Manufacturer)
23 10 16 00                                             (Model Name)
57 69 4D 65   64 69 61 20   53 61 6D 70   6C 65 20 44
69 73 70 6C   61 79
24 10 01 00   32                                        (Model Number)
42 10 06 00   31 32 33 38   44 36                       (Serial Number)
54 10 08 00   07 00 00 13   88 00 04 00                 (Primary Device Type)
```

C.5.2.5 M1

Table 64 — Field values for the M1 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		4 (M1)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	4 (M1)
Attribute 3	Attribute Type	0x101A (Enrollee Nonce)
	Attribute Length	16 (0x0010)
	Attribute Value	0x2696 1EAD 7E25 C69D 3C2F 7DB6 2444 A5B7
Attribute 4	Attribute Type	0x1039 (Registrar Nonce)
	Attribute Length	16 (0x0010)
	Attribute Value	0xADEE 5C75 F058 24FE 4D51 1B62 39B2 96D0
Attribute 5	Attribute Type	0x1047 (UUID-E)
	Attribute Length	16 (0x0010)
	Attribute Value	BFB4F9C1-F179-4E0F-AEC6-94ABB1421477

Field		Value
Attribute 6	Attribute Type	0x1032 (Public Key)
	Attribute Length	384 (0x0180)
	Attribute Value	0x5A0D 3D4E 049F AA93 9FFA 6A37 5B9C 3C16 A4C3 9753 D19F F7DA 36BC 391E A72F C0F6 8C92 9BDB 4005 52ED 84E0 900C 7A44 C322 2FD5 4D71 4825 6862 886B FB40 16BD 2D03 C4C4 CF47 6567 C291 770E 47BD 59D0 AA53 23CF DDFC 5596 E0D6 558C 480E E8B0 C625 9983 4D45 81A7 96A0 1981 4687 8916 4504 AFBD 29CE 9936 E86A 290C 5F00 F8BA 986B 4801 0F3E 5C07 9C7F 351D DCA2 EE1F D508 46B3 7BF7 463C 2B0F 3D00 1B13 17AC 3069 CD89 E2E4 927E D3D4 0875 A604 9AF6 49D2 DC34 9DB5 995A 7525 D70A 3A1C 9B67 3F54 82F8 3343 BD90 D45E 9C39 62DC 4A4B F2B4 ADB3 7E91 66B2 DDB3 1CCF 11C5 B9E6 C98E 0A9A 3377 ABBA 56B0 F428 3B2E AA69 F536 8BC1 07E1 C225 99F8 8DD1 924D 0899 C5F1 5346 2C91 1A82 9307 8AEF EE9F B238 9A78 5483 3FCE A61C FECB B49F 828C 361A 981A 5FED ECF1 3796 AE36 E36C 15A1 6670 AF96 996C 3C45 A30E 900E 18C8 58F6 232B 5F70 72BD D9E4 7D7F C612 46EF 5D19 7657 39F3 8509 2843 79BC 319D 9409 E8FE 236B D29B 0335 A5BC 5BB0 424E E44D E8A1 9F86 4A15 9FDA 907D 6F5A 30EB C0A1 7E36 28E4 90E5
Attribute 7	Attribute Type	0x1012 (Device Password ID)
	Attribute Length	2 (0x0002)
	Attribute Value	2 (0x0002) (Machine-generated)

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                   (MUX Header)
03                                                      (WLP Frame Type)
04                                                      (Association Subtype)
00 20 01 00   10                                        (WLP Version)
22 10 01 00   04                                        (Message Type)
1A 10 10 00                                             (Enrollee Nonce)
B7 A5 44 24   B6 7D 2F 3C   9D C6 25 7E   AD 1E 96 26
39 10 10 00                                             (Registrar Nonce)
D0 96 B2 39   62 1B 51 4D   FE 24 58 F0   75 5C EE AD
47 10 10 00                                             (UUID-E)
BF B4 F9 C1   F1 79 4E 0F   AE C6 94 AB   B1 42 14 77
32 10 80 01                                             (Public Key)
E5 90 E4 28   36 7E A1 C0   EB 30 5A 6F   7D 90 DA 9F
15 4A 86 9F   A1 E8 4D E4   4E 42 B0 5B   BC A5 35 03
9B D2 6B 23   FE E8 09 94   9D 31 BC 79   43 28 09 85
F3 39 57 76   19 5D EF 46   12 C6 7F 7D   E4 D9 BD 72
70 5F 2B 23   F6 58 C8 18   0E 90 0E A3   45 3C 6C 99
96 AF 70 66   A1 15 6C E3   36 AE 96 37   F1 EC ED 5F
1A 98 1A 36   8C 82 9F B4   CB FE 1C A6   CE 3F 83 54
78 9A 38 B2   9F EE EF 8A   07 93 82 1A   91 2C 46 53
F1 C5 99 08   4D 92 D1 8D   F8 99 25 C2   E1 07 C1 8B
36 F5 69 AA   2E 3B 28 F4   B0 56 BA AB   77 33 9A 0A
8E C9 E6 B9   C5 11 CF 1C   B3 DD B2 66   91 7E B3 AD
B4 F2 4B 4A   DC 62 39 9C   5E D4 90 BD   43 33 F8 82
54 3F 67 9B   1C 3A 0A D7   25 75 5A 99   B5 9D 34 DC
D2 49 F6 9A   04 A6 75 08   D4 D3 7E 92   E4 E2 89 CD
69 30 AC 17   13 1B 00 3D   0F 2B 3C 46   F7 7B B3 46
08 D5 1F EE   A2 DC 1D 35   7F 9C 07 5C   3E 0F 01 48
6B 98 BA F8   00 5F 0C 29   6A E8 36 99   CE 29 BD AF
04 45 16 89   87 46 81 19   A0 96 A7 81   45 4D 83 99
25 C6 B0 E8   0E 48 8C 55   D6 E0 96 55   FC DD CF 23
53 AA D0 59   BD 47 0E 77   91 C2 67 65   47 CF C4 C4
03 2D BD 16   40 FB 6B 88   62 68 25 48   71 4D D5 2F
22 C3 44 7A   0C 90 E0 84   ED 52 05 40   DB 9B 92 8C
F6 C0 2F A7   1E 39 BC 36   DA F7 9F D1   53 97 C3 A4
16 3C 9C 5B   37 6A FA 9F   93 AA 9F 04   4E 3D 0D 5A
12 10 02 00   02 00                                    (Device Password ID)
```

C.5.2.6 M2

Table 65 — Field values for the M2 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		5 (M2)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	5 (M2)
Attribute 3	Attribute Type	0x101A (Enrollee Nonce)
	Attribute Length	16 (0x0010)
	Attribute Value	0x2696 1EAD 7E25 C69D 3C2F 7DB6 2444 A5B7
Attribute 4	Attribute Type	0x1039 (Registrar Nonce)
	Attribute Length	16 (0x0010)
	Attribute Value	0xADEE 5C75 F058 24FE 4D51 1B62 39B2 96D0
Attribute 5	Attribute Type	0x1048 (UUID-R)
	Attribute Length	16 (0x0010)
	Attribute Value	BFB4F9B6-F179-4E0F-AEC6-94ABB1421477
Attribute 6	Attribute Type	0x1005 (Authenticator)
	Attribute Length	8 (0x0008)
	Attribute Value	f3b3 e923 06b9 9161

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                   (MUX Header)
03                                                      (WLP Frame Type)
05                                                      (Association Subtype)
00 20 01 00   10                                        (WLP Version)
22 10 01 00   05                                        (Message Type)
1A 10 10 00                                             (Enrollee Nonce)
B7 A5 44 24   B6 7D 2F 3C   9D C6 25 7E   AD 1E 96 26
39 10 10 00                                             (Registrar Nonce)
D0 96 B2 39   62 1B 51 4D   FE 24 58 F0   75 5C EE AD
48 10 10 00                                             (UUID-R)
BF B4 F9 B6   F1 79 4E 0F   AE C6 94 AB   B1 42 14 77
05 10 08 00   F3 B3 E9 23   06 B9 91 61                 (Authenticator)
```

C.5.2.7 M3

Table 66 — Field values for the M3 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		7 (M3)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	7 (M3)
Attribute 3	Attribute Type	0x1039 (Registrar Nonce)
	Attribute Length	16 (0x0010)
	Attribute Value	0xADEE 5C75 F058 24FE 4D51 1B62 39B2 96D0
Attribute 4	Attribute Type	0x1014 (E-Hash1)
	Attribute Length	32 (0x0020)
	Attribute Value	e68e b195 96a6 fb01 22d6 81ff 29c2 9ec7 b4a7 c6b2 9700 ee19 4717 3b62 ea14 4a4a
Attribute 5	Attribute Type	0x1015 (E-Hash2)
	Attribute Length	32 (0x0020)
	Attribute Value	0d93 41fb 5ee5 3b3e f0a2 b62d c870 693e dc8e 3886 d6b4 606d b664 7776 8b16 cf86
Attribute 6	Attribute Type	0x1005 (Authenticator)
	Attribute Length	8 (0x0008)
	Attribute Value	d71a ccb9 1532 b085

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                   (MUX Header)
03                                                      (WLP Frame Type)
07                                                      (Association Subtype)
00 20 01 00  10                                         (WLP Version)
22 10 01 00  07                                         (Message Type)
39 10 10 00                                             (Registrar Nonce)
D0 96 B2 39  62 1B 51 4D  FE 24 58 F0  75 5C EE AD
14 10 20 00                                             (E-Hash1)
E6 8E B1 95  96 A6 FB 01  22 D6 81 FF  29 C2 9E C7
B4 A7 C6 B2  97 00 EE 19  47 17 3B 62  EA 14 4A 4A
15 10 20 00                                             (E-Hash2)
0D 93 41 FB  5E E5 3B 3E  F0 A2 B6 2D  C8 70 69 3E
DC 8E 38 86  D6 B4 60 6D  B6 64 77 76  8B 16 CF 86
05 10 08 00  D7 1A CC B9  15 32 B0 85                   (Authenticator)
```

C.5.2.8 M4

Table 67 — Field values for the M4 association frame

Field					Value	
Protocol ID (MUX Header)					0x0100 (WLP)	
WLP Frame Type					3 (Association)	
Association Subtype					8 (M4)	
Attribute 1	Attribute Type				0x2000 (WLP Version)	
	Attribute Length				1 (0x0001)	
	Attribute Value				0x10	
Attribute 2	Attribute Type				0x1022 (Message Type)	
	Attribute Length				1 (0x0001)	
	Attribute Value				8 (M4)	
Attribute 3	Attribute Type				0x101A (Enrollee Nonce)	
	Attribute Length				16 (0x0010)	
	Attribute Value				0x2696 1EAD 7E25 C69D 3C2F 7DB6 2444 A5B7	
Attribute 4	Attribute Type				0x103D (R-Hash1)	
	Attribute Length				32 (0x0020)	
	Attribute Value				7e6b 1daa a28c 9e0b 4303 ca88 a5e6 faae cb2d 8262 bb7e 8c96 53f3 6546 e54b f0ae	
Attribute 5	Attribute Type				0x103E (R-Hash2)	
	Attribute Length				32 (0x0020)	
	Attribute Value				9378 1060 45b9 068e fd70 7388 c0f1 c6b5 6dce 4714 7ba9 aa86 f706 ccf5 83d0 a671	
Attribute 6	Attribute Type				0x1018 (Encrypted Settings)	
	Attribute Length				64 (0x0040)	
	Attribute Value	IV			eb38 9539 a05d 06ce d49c f12c 8e10 a236	
		Attribute 6a	Attribute Type		0x103F (R-SNonce1)	Encrypted: 1b84 7206 c9fa 3f01 62f7 8533 e239 b6ca 6a54 3e68 c11a 4a25 2962 5649 1f14 d446 02e1 5902 0449 fe32 7648 3ef9 223f 47c3
			Attribute Length		16 (0x0010)	
			Attribute Value		0x39EF B757 E156 3CE4 F4D3 BB9F CF33 1197	
		Attribute 6b	Attribute Type		0x101E (Key Wrap Authenticator)	
			Attribute Length		8 (0x0008)	
			Attribute Value		a83e 5c30 61b0 0f04	
		pad			1010 1010 1010 1010 1010 1010 1010 1010	

Field		Value
Attribute 7	Attribute Type	0x1005 (Authenticator)
	Attribute Length	8 (0x0008)
	Attribute Value	82fc cb38 7e8c 164a

The octets that make up the Encrypted Settings attribute cleartext are:

```
3F 10 10 00                                             (R-SNonce1)
97 11 33 CF   9F BB D3 F4   E4 3C 56 E1   57 B7 EF 39
1E 10 08 00   A8 3E 5C 30   61 B0 0F 04                 (Key Wrap Authenticator)
10 10 10 10   10 10 10 10   10 10 10 10   10 10 10 10   (pad)
```

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                   (MUX Header)
03                                                      (WLP Frame Type)
08                                                      (Association Subtype)
00 20 01 00   10                                        (WLP Version)
22 10 01 00   08                                        (Message Type)
1A 10 10 00                                             (Enrollee Nonce)
B7 A5 44 24   B6 7D 2F 3C   9D C6 25 7E   AD 1E 96 26
3D 10 20 00                                             (R-Hash1)
7E 6B 1D AA   A2 8C 9E 0B   43 03 CA 88   A5 E6 FA AE
CB 2D 82 62   BB 7E 8C 96   53 F3 65 46   E5 4B F0 AE
3E 10 20 00                                             (R-Hash2)
93 78 10 60   45 B9 06 8E   FD 70 73 88   C0 F1 C6 B5
6D CE 47 14   7B A9 AA 86   F7 06 CC F5   83 D0 A6 71
18 10 40 00                                             (Encrypted Settings)
EB 38 95 39   A0 5D 06 CE   D4 9C F1 2C   8E 10 A2 36
1B 84 72 06   C9 FA 3F 01   62 F7 85 33   E2 39 B6 CA
6A 54 3E 68   C1 1A 4A 25   29 62 56 49   1F 14 D4 46
02 E1 59 02   04 49 FE 32   76 48 3E F9   22 3F 47 C3
05 10 08 00   82 FC CB 38   7E 8C 16 4A                 (Authenticator)
```

C.5.2.9 M5

Table 68 — Field values for the M5 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		9 (M5)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	9 (M5)
Attribute 3	Attribute Type	0x1039 (Registrar Nonce)
	Attribute Length	16 (0x0010)
	Attribute Value	0xADEE 5C75 F058 24FE 4D51 1B62 39B2 96D0

Field				Value	
Attribute 4	Attribute Type			0x1018 (Encrypted Settings)	
	Attribute Length			64 (0x0040)	
	Attribute Value	IV		d319 22c4 ea44 34e1 e242 f9f9 d14f 5e53	
		Attribute 4a	Attribute Type	0x1016 (E-SNonce1)	Encrypted: 4a64 c0e0 09a8 8ab4 5293 5aa3 343e b05f c332 bc1a d469 4d0a 0d3d 1801 f6f4 4497 93c8 d603 b9c8 4e22 13bb eb3e e9ec 0629
			Attribute Length	16(0x0010)	
			Attribute Value	0xFD1E 858E 9974 4690 3CC6 4535 33A9 F347	
		Attribute 4b	Attribute Type	0x101E (Key Wrap Authenticator)	
			Attribute Length	8 (0x0008)	
			Attribute Value	aa50 b214 ac9f ecdd	
		pad		1010 1010 1010 1010 1010 1010 1010 1010	
Attribute 5	Attribute Type			0x1005 (Authenticator)	
	Attribute Length			8 (0x0008)	
	Attribute Value			40ae 0abc dd68 c124	

The octets that make up the Encrypted Settings attribute cleartext are:

```
16 10 10 00                                             (E-SNonce1)
47 F3 A9 33   35 45 C6 3C   90 46 74 99   8E 85 1E FD
1E 10 08 00   AA 50 B2 14   AC 9F EC DD                 (Key Wrap Authenticator)
10 10 10 10   10 10 10 10   10 10 10 10   10 10 10 10   (pad)
```

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                   (MUX Header)
03                                                      (WLP Frame Type)
09                                                      (Association Subtype)
00 20 01 00   10                                        (WLP Version)
22 10 01 00   09                                        (Message Type)
39 10 10 00                                             (Registrar Nonce)
D0 96 B2 39   62 1B 51 4D   FE 24 58 F0   75 5C EE AD
18 10 40 00                                             (Encrypted Settings)
D3 19 22 C4   EA 44 34 E1   E2 42 F9 F9   D1 4F 5E 53
4A 64 C0 E0   09 A8 8A B4   52 93 5A A3   34 3E B0 5F
C3 32 BC 1A   D4 69 4D 0A   0D 3D 18 01   F6 F4 44 97
93 C8 D6 03   B9 C8 4E 22   13 BB EB 3E   E9 EC 06 29
05 10 08 00   40 AE 0A BC   DD 68 C1 24                 (Authenticator)
```

C.5.2.10 M6

Table 69 — Field values for the M6 association frame

Field	Value
Protocol ID (MUX Header)	0x0100 (WLP)
WLP Frame Type	3 (Association)
Association Subtype	10 (M6)

Field					Value	
Attribute 1	Attribute Type				0x2000 (WLP Version)	
	Attribute Length				1 (0x0001)	
	Attribute Value				0x10	
Attribute 2	Attribute Type				0x1022 (Message Type)	
	Attribute Length				1 (0x0001)	
	Attribute Value				10 (M6)	
Attribute 3	Attribute Type				0x101A (Enrollee Nonce)	
	Attribute Length				16 (0x0010)	
	Attribute Value				0x2696 1EAD 7E25 C69D 3C2F 7DB6 2444 A5B7	
Attribute 4	Attribute Type				0x1018 (Encrypted Settings)	
	Attribute Length				64 (0x0040)	
	Attribute Value	IV			341a 5283 cd11 4f7a 364b 61dd 689e 28f7	
		Attribute 4a	Attribute Type		0x1040 (R-SNonce2)	Encrypted: 0249 da63 4d1a 94e6 65b9 d1b3 f032 753c 2d1f b169 8567 336b 193b 96ce 45b8 d2ef e380 907c 00bf b73d 3271 b295 bb32 526b
			Attribute Length		16 (0x0010)	
			Attribute Value		0xEF2B 8978 6D43 149B FCDB 1B31 CB0A E6E3	
		Attribute 4b	Attribute Type		0x101E (Key Wrap Authenticator)	
			Attribute Length		8 (0x0008)	
			Attribute Value		976c ecf0 310f 9e2c	
		pad			1010 1010 1010 1010 1010 1010 1010 1010	
Attribute 5	Attribute Type				0x1005 (Authenticator)	
	Attribute Length				8 (0x0008)	
	Attribute Value				562c 2203 7c53 2ea2	

The octets that make up the Encrypted Settings attribute cleartext are:

```
40 10 10 00                                          (R-SNonce2)
E3 E6 0A CB   31 1B DB FC   9B 14 43 6D   78 89 2B EF
1E 10 08 00   97 6C EC F0   31 0F 9E 2C              (Key Wrap Authenticator)
10 10 10 10   10 10 10 10   10 10 10 10   10 10 10 10  (pad)
```

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                (MUX Header)
03                                                   (WLP Frame Type)
0A                                                   (Association Subtype)
00 20 01 00   10                                     (WLP Version)
22 10 01 00   0A                                     (Message Type)
1A 10 10 00                                          (Enrollee Nonce)
B7 A5 44 24   B6 7D 2F 3C   9D C6 25 7E   AD 1E 96 26
18 10 40 00                                          (Encrypted Settings)
34 1A 52 83   CD 11 4F 7A   36 4B 61 DD   68 9E 28 F7
02 49 DA 63   4D 1A 94 E6   65 B9 D1 B3   F0 32 75 3C
```

```
2D 1F B1 69   85 67 33 6B   19 3B 96 CE   45 B8 D2 EF
E3 80 90 7C   00 BF B7 3D   32 71 B2 95   BB 32 52 6B
05 10 08 00   56 2C 22 03   7C 53 2E A2              (Authenticator)
```

C.5.2.11 M7

Table 70 — Field values for the M7 association frame

Field					Value	
Protocol ID (MUX Header)					0x0100 (WLP)	
WLP Frame Type					3 (Association)	
Association Subtype					11 (M7)	
Attribute 1	Attribute Type				0x2000 (WLP Version)	
	Attribute Length				1 (0x0001)	
	Attribute Value				0x10	
Attribute 2	Attribute Type				0x1022 (Message Type)	
	Attribute Length				1 (0x0001)	
	Attribute Value				11 (M7)	
Attribute 3	Attribute Type				0x1039 (Registrar Nonce)	
	Attribute Length				16 (0x0010)	
	Attribute Value				0xADEE 5C75 F058 24FE 4D51 1B62 39B2 96D0	
Attribute 4	Attribute Type				0x1018 (Encrypted Settings)	
	Attribute Length				64 (0x0040)	
	Attribute Value	IV			81fb 6b9d c2a8 b1a5 abcc ba5a deb7 d061	
		Attribute 4a	Attribute Type		0x1017 (E-SNonce2)	Encrypted: d8ba 9df5 71b7 7278 e8f9 f68c ab68 826e b0d1 8a57 7dd1 7cac adec 0ce8 6032 0d30 9763 9b40 c224 e95c 26d1 3a6f 94f3 59b8
			Attribute Length		16 (0x0010)	
			Attribute Value		0x97D1 FF6A 2D6C 65C4 29C7 057F 927A 657B	
		Attribute 4b	Attribute Type		0x101E (Key Wrap Authenticator)	
			Attribute Length		8 (0x0008)	
			Attribute Value		402f 8ae6 6a05 7ab4	
		pad			1010 1010 1010 1010 1010 1010 1010 1010	
Attribute 5	Attribute Type				0x1005 (Authenticator)	
	Attribute Length				8 (0x0008)	
	Attribute Value				230c 4afc fc43 9329	

The octets that make up the Encrypted Settings attribute cleartext are:

```
17 10 10 00                                          (E-SNonce2)
7B 65 7A 92   7F 05 C7 29   C4 65 6C 2D   6A FF D1 97
```

```
1E 10 08 00   40 2F 8A E6   6A 05 7A B4               (Key Wrap Authenticator)
10 10 10 10   10 10 10 10   10 10 10 10   10 10 10 10  (pad)
```

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                                 (MUX Header)
03                                                    (WLP Frame Type)
0B                                                    (Association Subtype)
00 20 01 00   10                                      (WLP Version)
22 10 01 00   0B                                      (Message Type)
39 10 10 00                                           (Registrar Nonce)
D0 96 B2 39   62 1B 51 4D   FE 24 58 F0   75 5C EE AD
18 10 40 00                                           (Encrypted Settings)
81 FB 6B 9D   C2 A8 B1 A5   AB CC BA 5A   DE B7 D0 61
D8 BA 9D F5   71 B7 72 78   E8 F9 F6 8C   AB 68 82 6E
B0 D1 8A 57   7D D1 7C AC   AD EC 0C E8   60 32 0D 30
97 63 9B 40   C2 24 E9 5C   26 D1 3A 6F   94 F3 59 B8
05 10 08 00   23 0C 4A FC   FC 43 93 29               (Authenticator)
```

C.5.2.12 M8

Table 71 — Field values for the M8 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		12 (M8)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	12 (M8)
Attribute 3	Attribute Type	0x101A (Enrollee Nonce)
	Attribute Length	16 (0x0010)
	Attribute Value	0x2696 1EAD 7E25 C69D 3C2F 7DB6 2444 A5B7

Field					Value	
Attribute 4	Attribute Type				0x1018 (Encrypted Settings)	
	Attribute Length				96 (0x0060)	
	Attribute Value	IV			cea4 c2a6 44f8 a67b aca4 8de3 d40b f39e	
		Attribute 4a	Attribute Type		0x2001 (WSSID)	Encrypted: 04da 8477 1fc0 dea4 98a1 ac48 7b05 7155 69d0 3a5b 1c9e a959 7927 f459 3391 2c70 364d ccd2 af12 6df8 a600 3194 abcd eb39 3743 290f 9107 434b a147 5b27 0245 6268 af92 42e9 0559 9cb6 d9b3 1d5e 9e89 34d2
			Attribute Length		16 (0x0010)	
			Attribute Value		A8DE9103-4047-41B0-8BA7-0286EBCBA8F1	
		Attribute 4b	Attribute Type		0x2002 (WSS Name)	
			Attribute Length		6 (0x0006)	
			Attribute Value		“My WSS”	
		Attribute 4c	Attribute Type		0x2004 (WSS Broadcast Address)	
			Attribute Length		6 (0x0006)	
			Attribute Value		01-13-88-00-01-CC	
		Attribute 4d	Attribute Type		0x2005 (WSS Master Key)	
			Attribute Length		16 (0x0010)	
			Attribute Value		f69f c951 0350 84c6 333a d198 2c25 91f2	
		Attribute 4e	Attribute Type		0x101E (Key Wrap Authenticator)	
			Attribute Length		8 (0x0008)	
			Attribute Value		306f 368b b882 9070	
		pad			0808 0808 0808 0808	
Attribute 5	Attribute Type				0x1005 (Authenticator)	
	Attribute Length				8 (0x0008)	
	Attribute Value				edec f22d 40ac 393c	

The octets that make up the Encrypted Settings attribute cleartext are:

```
01 20 10 00                                     (WSSID)
A8 DE 91 03   40 47 41 B0   8B A7 02 86   EB CB A8 F1
02 20 06 00   4D 79 20 57   53 53                   (WSS Name)
04 20 06 00   01 13 88 00   01 CC                   (WSS Broadcast Address)
05 20 10 00                                     (WSS Master Key)
F6 9F C9 51   03 50 84 C6   33 3A D1 98   2C 25 91 F2
1E 10 08 00   30 6F 36 8B   B8 82 90 70          (Key Wrap Authenticator)
08 08 08 08   08 08 08 08                       (pad)
```

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                           (MUX Header)
03                                              (WLP Frame Type)
```

```
0C                                                  (Association Subtype)
00 20 01 00   10                                    (WLP Version)
22 10 01 00   0C                                    (Message Type)
1A 10 10 00                                         (Enrollee Nonce)
B7 A5 44 24   B6 7D 2F 3C   9D C6 25 7E   AD 1E 96 26
18 10 60 00                                         (Encrypted Settings)
CE A4 C2 A6   44 F8 A6 7B   AC A4 8D E3   D4 0B F3 9E
04 DA 84 77   1F C0 DE A4   98 A1 AC 48   7B 05 71 55
69 D0 3A 5B   1C 9E A9 59   79 27 F4 59   33 91 2C 70
36 4D CC D2   AF 12 6D F8   A6 00 31 94   AB CD EB 39
37 43 29 0F   91 07 43 4B   A1 47 5B 27   02 45 62 68
AF 92 42 E9   05 59 9C B6   D9 B3 1D 5E   9E 89 34 D2
05 10 08 00   ED EC F2 2D   40 AC 39 3C               (Authenticator)
```

C.5.3 Activation

In the previous examples, three WSSID values were used. The values and corresponding WSSID hash values as would be included in a beacon to activate the WSS are shown in Table 72.

Table 72 — Example WSSID and WSSID hash values

WSSID	WSSID Hash
1D015E9C-2930-4C3B-9B50-987121F32E9D	F3
1D015E9D-2930-4C3B-9B50-987121F32E9D	F2
A8DE9103-4047-41B0-8BA7-0286EBCBA8F1	C3

C.5.4 Connection

The following example frames are exchanged during connection between the two devices in the Numeric Comparison example.

C.5.4.1 C1

Table 73 — Field values for the C1 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		34 (C1)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	34 (C1)
Attribute 3	Attribute Type	0x2001 (WSSID)
	Attribute Length	16 (0x0010)
	Attribute Value	1D015E9C-2930-4C3B-9B50-987121F32E9D

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                    (MUX Header)
03 22                                    (WLP Frame Type and Association Subtype)
00 20 01 00   10                         (WLP Version attribute)
22 10 01 00   22                         (Message Type attribute)
01 20 10 00   1D 01 5E 9C   29 30 4C 3B  (WSSID attribute)
9B 50 98 71   21 F3 2E 9D
```

C.5.4.2 C2

Table 74 — Field values for the C2 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		35 (C2)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	35 (C2)
Attribute 3	Attribute Type	0x2001 (WSSID)
	Attribute Length	16 (0x0010)
	Attribute Value	1D015E9C-2930-4C3B-9B50-987121F32E9D

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                    (MUX Header)
03 23                                    (WLP Frame Type and Association Subtype)
00 20 01 00   10                         (WLP Version attribute)
22 10 01 00   23                         (Message Type attribute)
01 20 10 00   1D 01 5E 9C   29 30 4C 3B  (WSSID attribute)
9B 50 98 71   21 F3 2E 9D
```

C.5.4.3 C3

Table 75 — Field values for the C3 association frame

Field	Value
Protocol ID (MUX Header)	0x0100 (WLP)
WLP Frame Type	3 (Association)
Association Subtype	36 (C3)

Field		Value
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	36 (C3)
Attribute 3	Attribute Type	0x2001 (WSSID)
	Attribute Length	16 (0x0010)
	Attribute Value	1D015E9C-2930-4C3B-9B50-987121F32E9D
Attribute 4	Attribute Type	0x200C (WSS tag)
	Attribute Length	1 (0x0001)
	Attribute Value	F3
Attribute 5	Attribute Type	0x200D (WSS Virtual EUI-48)
	Attribute Length	6 (0x0006)
	Attribute Value	00-13-88-12-38-E1

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                   (MUX Header)
03 24                                   (WLP Frame Type and Association Subtype)
00 20 01 00   10                        (WLP Version attribute)
22 10 01 00   24                        (Message Type attribute)
01 20 10 00   1D 01 5E 9C   29 30 4C 3B   (WSSID attribute)
9B 50 98 71   21 F3 2E 9D
0C 20 01 00   F3                        (WSS Tag attribute)
0D 20 06 00   00 13 88 12   38 E1       (WSS Virtual EUI-48 attribute)
```

C.5.4.4 C4

Table 76 — Field values for the C4 association frame

Field		Value
Protocol ID (MUX Header)		0x0100 (WLP)
WLP Frame Type		3 (Association)
Association Subtype		37 (C4)
Attribute 1	Attribute Type	0x2000 (WLP Version)
	Attribute Length	1 (0x0001)
	Attribute Value	0x10
Attribute 2	Attribute Type	0x1022 (Message Type)
	Attribute Length	1 (0x0001)
	Attribute Value	37 (C4)

Field		Value
Attribute 3	Attribute Type	0x2001 (WSSID)
	Attribute Length	16 (0x0010)
	Attribute Value	1D015E9C-2930-4C3B-9B50-987121F32E9D
Attribute 4	Attribute Type	0x200C (WSS tag)
	Attribute Length	1 (0x0001)
	Attribute Value	F3
Attribute 5	Attribute Type	0x200D (WSS Virtual EUI-48)
	Attribute Length	6 (0x0006)
	Attribute Value	00-13-88-12-38-D6

The octets comprising the MSDU arrive at the MAC SAP in the following order:

```
01 00                                   (MUX Header)
03 25                                   (WLP Frame Type and Association Subtype)
00 20 01 00   10                        (WLP Version attribute)
22 10 01 00   25                        (Message Type attribute)
01 20 10 00   1D 01 5E 9C   29 30 4C 3B  (WSSID attribute)
9B 50 98 71   21 F3 2E 9D
0C 20 01 00   F3                        (WSS Tag attribute)
0D 20 06 00   00 13 88 12   38 D6       (WSS Virtual EUI-48 attribute)
```

C.6 Derivation of association frame cryptographic numbers

This subclause specifies in detail each value calculated during an example enrollment session.

C.6.1 Random numbers used in examples

The following random numbers are used in examples in this clause:

E (enrollee's private key) =

```
4400 51d6 f0b5 5ea9 67ab 31c6 8a8b 5e37 d910 dae0 e2d4 59a4 8645 9caa df36 7516
```

R (registrar's private key) =

```
5dae c786 7980 a324 8ce3 578f c75f 1b0f 2df8 9d30 6fa4 52cd e07a 048a ded9 2656
```

N_e (enrollee's nonce) =

```
2696 1ead 7e25 c69d 3c2f 7db6 2444 a5b7
```

N_r (registrar's nonce) =

```
adee 5c75 f058 24fe 4d51 1b62 39b2 96d0
```

E-S_1 (enrollee's first secret nonce) =

```
fd1e 858e 9974 4690 3cc6 4535 33a9 f347
```

E-S_2 (enrollee's second secret nonce) =

```
97d1 ff6a 2d6c 65c4 29c7 057f 927a 657b
```

R-S_1 (registrar's first secret nonce) =

```
39ef b757 e156 3ce4 f4d3 bb9f cf33 1197
```

R-S_2 (registrar's second secret nonce) =

```
ef2b 8978 6d43 149b fcdb 1b31 cb0a e6e3
```

C.6.2 Public key calculation

$PK_e = g^E \bmod p =$

```
5a0d 3d4e 049f aa93 9ffa 6a37 5b9c 3c16 a4c3 9753 d19f f7da 36bc 391e a72f c0f6
8c92 9bdb 4005 52ed 84e0 900c 7a44 c322 2fd5 4d71 4825 6862 886b fb40 16bd 2d03
c4c4 cf47 6567 c291 770e 47bd 59d0 aa53 23cf ddfc 5596 e0d6 558c 480e e8b0 c625
9983 4d45 81a7 96a0 1981 4687 8916 4504 afbd 29ce 9936 e86a 290c 5f00 f8ba 986b
4801 0f3e 5c07 9c7f 351d dca2 ee1f d508 46b3 7bf7 463c 2b0f 3d00 1b13 17ac 3069
cd89 e2e4 927e d3d4 0875 a604 9af6 49d2 dc34 9db5 995a 7525 d70a 3a1c 9b67 3f54
82f8 3343 bd90 d45e 9c39 62dc 4a4b f2b4 adb3 7e91 66b2 ddb3 1ccf 11c5 b9e6 c98e
0a9a 3377 abba 56b0 f428 3b2e aa69 f536 8bc1 07e1 c225 99f8 8dd1 924d 0899 c5f1
5346 2c91 1a82 9307 8aef ee9f b238 9a78 5483 3fce a61c fecb b49f 828c 361a 981a
5fed ecf1 3796 ae36 e36c 15a1 6670 af96 996c 3c45 a30e 900e 18c8 58f6 232b 5f70
72bd d9e4 7d7f c612 46ef 5d19 7657 39f3 8509 2843 79bc 319d 9409 e8fe 236b d29b
0335 a5bc 5bb0 424e e44d e8a1 9f86 4a15 9fda 907d 6f5a 30eb c0a1 7e36 28e4 90e5
```

$PK_r = g^R \bmod p =$

```
dc14 c6f6 d85b 3d58 b54a bb30 6d55 6829 2ed7 85d3 9ed7 3643 666a 1b4a 4684 654f
88bb edf0 414c 59c7 0dd9 90b4 47b3 c325 0a4a 2367 3ea9 361a 79be 3376 0906 ef12
7627 fa9e 7f91 07e7 3675 9cff 990c 44fc e240 7e7c e1c7 d61a 83b8 5c82 85a9 bf94
7cc1 e582 642a 8a86 3e4e 0d57 f258 4b25 5229 c4d3 5355 1e86 ac2b bce4 13c7 e554
1cc2 e68d 7101 d578 30cd e1c9 1bd4 8c03 d190 1472 01f3 9697 f65c c2f4 45e8 5162
3bea 585c 8205 d8e8 ca91 b54d aefb 6fe5 ac46 e942 b5ea 6e04 495b d2f6 cb11 88c1
b44a 342e 5dab 2917 165e 0935 d743 69b7 6698 68c9 d4d5 b148 33f3 1e56 9499 1e73
353a 33f5 f4dc 61ff 5752 517b 7180 6da2 e47e fc78 d22d d8da c4f1 1501 9d57 5d60
b787 6140 4413 bff6 e314 329b f1e5 2b92 38f8 7964 a5a3 00c7 26c0 950f ac94 6459
3c30 6ece 4d92 813f d714 2e16 18b3 efbb 3fea 25f9 e177 0859 2507 d8be 73ef d569
761e 7ff4 b016 edd0 c5c3 85a8 ec16 1a44 f2d6 7c1c 6b39 7d8f 6c3f a797 bcd9 5e3f
b8f4 ecba 7ebf 6620 570e f491 4e75 eaf9 752b a471 faf7 ccc5 5373 069c 2153 1194
```

C.6.3 DHKey calculation

SharedSecret = $PK_e^R \bmod p = PK_r^E \bmod p =$

```
bcec d344 c6f4 2f35 aced 542b 7ceb 684a 623b f9ad 3ebf 2a64 9afc be7c 9fd2 127e
1d2b 08ba b247 3cdd bf44 fa3f 98a5 6ad7 5ee7 5a66 e0dc 0bfb c246 fb57 9a6d 5275
3222 ea82 e4fc ee51 fef5 3d24 af4c 5f00 fdba f7b3 c55a 0e4f 8b5f 2e27 51b5 ca3f
9898 8ca3 08b5 11bd 2e35 7767 84dc 852f 8519 9eb0 52aa 12a3 b4f5 e9ca be79 8610
11a6 c34e 9b11 6f06 fcb3 b59e e739 75cf 6529 118f 63b0 68f2 2422 cbac 11e1 18f1
fc3a 06c7 9787 f8c0 ee90 f878 64b9 fac6 5f75 6725 6abd 1da2 1122 d83e 4026 e9d4
835e 5e77 10cd 5ab4 7e88 7d10 dd75 56bf 5f27 679d 634a a1c2 f8a8 cfc3 1859 cb72
d0e0 8efa 9b01 a88b 213f b604 63fa eb63 2449 7b77 4420 76cf 81b9 9556 34dc eeeb
bcc1 9b17 1857 d823 d190 798f 391e 1910 b7ce eccc baa5 0856 32cf 7660 bb06 9b82
721f 7c33 61a4 512b 8a25 ac32 f16e a332 2e87 2f54 d2db 8ea7 b815 e125 cd47 b0c6
2a51 ae42 5f6c 6956 8ec4 3bb8 810f 62e8 447c cb19 0f59 ad1c 212a 50aa 20f0 66c5
732c a60e 6728 ea2b c91a 82fe cc80 6f81 3330 a694 4aff c69a 562f 3501 514c c70f
```

DHKey = SHA-256(SharedSecret) =

```
2d42 85c2 3196 26f2 c2c7 2c5a 2855 3f54 41d2 c521 8c0c fbb6 60cc 57a1 dfa1 a68f
```

C.6.4 Enrollee hash commitment

HashCommit = SHA-256(PK_e || N_e) =

```
de77 cd25 d982 d499 f96c ac2f bf9a 5ef3 c6ce f1cf 20e2 2513 db81 a474 0cd1 52f0
```

C.6.5 Numeric comparison

ComparisonNumber = first 32 bits of SHA-256(PK_e || PK_r || N_e || N_r || “displayed digest”) =

```
ee21 efad
```

DisplayedDigits = ComparisonNumber mod 10^2 = 53 (decimal)

C.6.6 Key derivation

KDK = HMAC-SHA-256$_{DHKey}$(N_e || N_r) =

```
3788 c2b5 ca51 d5ee 7430 6d02 60c5 aed1 adb8 cd53 cbe1 a387 1ce2 968b e34e 5a68
```

Input to hash function for AuthKey = 0x00000001 || “WLP 1.0” || 0x00000180 =

```
0000 0001 574c 5020 312e 3000 0001 80
```

AuthKey = HMAC-SHA-256$_{KDK}$(0x00000001 || “WLP 1.0” || 0x00000180) =

```
617f 2b2a dac5 aa1b f320 3bd1 7bfa 570d c11d 1b3b bc28 f816 b781 d7c2 5e20 7795
```

Input to hash function for KeyWrapKey = 0x00000002 || “WLP 1.0” || 0x00000180 =

```
0000 0002 574c 5020 312e 3000 0001 80
```

KeyWrapKey = first 128 bits of HMAC-SHA-256$_{KDK}$(0x00000002 || “WLP 1.0” || 0x00000180) =

```
ffd4 0b4e 0a38 42a6 334f dd5b 95d8 ffcb
```

C.6.7 Proof-of-device-password for Registrar-display association method

Display Value = 80874911

DevicePassword = ASCII “80874911” =

```
3830 3837 3439 3131
```

PSK_1 = first 128 bits of HMAC-SHA-256$_{AuthKey}$(“8087”) =

```
8237 359b 6a7b 299e 790a 93c2 2364 761e
```

PSK_2 = first 128 bits of HMAC-SHA-256$_{AuthKey}$(“4911”) =

```
f563 380f 9f6c 06af ad17 a2b7 ec7b f61f
```

E-H_1 = HMAC-SHA-256$_{AuthKey}$(E-S_1 || PSK_1 || PK_e || PK_r) =

```
e68e b195 96a6 fb01 22d6 81ff 29c2 9ec7 b4a7 c6b2 9700 ee19 4717 3b62 ea14 4a4a
```

E-H_2 = HMAC-SHA-256$_{AuthKey}$(E-S_2 || PSK_2 || PK_e || PK_r) =

```
0d93 41fb 5ee5 3b3e f0a2 b62d c870 693e dc8e 3886 d6b4 606d b664 7776 8b16 cf86
```

R-H_1 = HMAC-SHA-256$_{AuthKey}$(R-S_1 || PSK_1 || PK_e || PK_r) =

```
7e6b 1daa a28c 9e0b 4303 ca88 a5e6 faae cb2d 8262 bb7e 8c96 53f3 6546 e54b f0ae
```

R-H_2 = HMAC-SHA-256$_{AuthKey}$(R-S_2 || PSK_2 || PK_e || PK_r) =

```
9378 1060 45b9 068e fd70 7388 c0f1 c6b5 6dce 4714 7ba9 aa86 f706 ccf5 83d0 a671
```

C.6.8 Proof-of-device-password for User-provided Password association method

DevicePassword = ASCII “jinmeng” =

```
6a69 6e6d 656e 67
```

PSK_1 = first 128 bits of HMAC-SHA-256$_{AuthKey}$(“jinm”) =

```
3164 b4a1 1d9d 48f4 20e9 e4c3 dfa8 c30e
```

PSK_2 = first 128 bits of HMAC-SHA-256$_{AuthKey}$(“eng”) =

```
c872 718b 8b17 39ad fa0b 985f 6b74 3f68
```

E-H_1 = HMAC-SHA-256$_{AuthKey}$(E-S_1 || PSK_1 || PK_e || PK_r) =

```
f5c2 e4c6 6280 ed4f afb8 c5bc f1ae 4057 9bc9 c8b9 6183 e589 cb0f 558c 836b 315a
```

$E\text{-}H_2 = HMAC\text{-}SHA\text{-}256_{AuthKey}(E\text{-}S_2 \,||\, PSK_2 \,||\, PK_e \,||\, PK_r) =$

`3fcb 7a0c 954a af00 f3e1 0680 5684 b43a 0cc8 c1e6 ee4b 2dc8 3b37 0598 14c0 b4e6`

$R\text{-}H_1 = HMAC\text{-}SHA\text{-}256_{AuthKey}(R\text{-}S_1 \,||\, PSK_1 \,||\, PK_e \,||\, PK_r) =$

`ed86 b673 a426 6f90 a3f9 43c0 3adf 1a8b 0dfa 1737 7358 bce5 2ab2 307c 093d 9382`

$R\text{-}H_2 = HMAC\text{-}SHA\text{-}256_{AuthKey}(R\text{-}S_2 \,||\, PSK_2 \,||\, PK_e \,||\, PK_r) =$

`bee5 a9f8 6db6 ce15 2128 47dd 3820 2988 65b4 d848 58be 381c 51b4 e87f 58fa e5d7`

Annex D (informative) Bibliography

[B7] ECMA-368 High Rate Ultra Wideband PHY and MAC Standard. 2005. Geneva: Ecma International.[13]

[B8] FIPS PUB 180-2, Secure Hash Standard. August 2002. National Institute of Standards and Technology.[14]

[B9] FIPS PUB 198, The Keyed-Hash Message Authentication Code (HMAC). March 2002. National Institute of Standards and Technology.

[B10] "Guidelines for use of a 48-bit Extended Unique Identifier (EUI-48™)", http://standards.ieee.org/regauth/oui/tutorials/EUI48.html. 2005. New York: Institute of Electrical and Electronics Engineers, Inc.

[B11] IEEE 100, The Authoritative Dictionary of IEEE Standards Terms, Seventh Edition. 200x. New York: Institute of Electrical and Electronics Engineers, Inc.

[B12] ISO/IEC 7498-1:1994, Information Technology—Open Systems Interconnection—Basic Reference Model: The Basic Model. 1994. City of Publication: Publisher.[15]

[B13] RFC 791, Internet Protocol. September 1981. Internet Engineering Task Force.

[B14] RFC 1191, Path MTU Discovery, J. Mogul, S. Deering. November 1990. Internet Engineering Task Force.

[B15] RFC 1981, Path MTU Discovery for IP version 6, J. McCann, S. Deering, J. Mogul. August 1996. Internet Engineering Task Force.

[B16] RFC 2104, HMAC: Keyed-Hashing for Message Authentication, H. Krawczyk, M. Bellare, R. Canetti. February 1997. Internet Engineering Task Force.

[B17] RFC 2205, Resource ReSerVation Protocol (RSVP), R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin. September 1997. Internet Engineering Task Force.

[B18] RFC 2211, Specification of the Controlled-Load Network Element Service, J. Wroclawski. September 1997. Internet Engineering Task Force.

---

[13] Ecma publications are available from Ecma International, Rue du Rhône 114, CH-1204 Geneva (http://www.ecma-international.org).

[14] NIST computer security publications are available from National Technical Information Service (NTIS), 5285 Port Royal Road, Springfield, VA 22161 (http://csrc.nist.gov/publications/).

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

[B19] RFC 2212, Specification of the Guaranteed Quality of Service, S. Shenker, C. Partridege R. Guerin. September 1997. Internet Engineering Task Force.

[B20] RFC 2215, General Characterization Parameters for Integrated Service Network Elements, S. Shenker, J. Wroclawski. September 1997. Internet Engineering Task Force.

[B21] RFC2460, Internet Protocol, Version 6 (IPv6) Specification, S. Deering, R. Hinden. December 1998. Internet Engineering Task Force.

[B22] RFC 2462, IPv6 Stateless Address Autoconfiguration, S. Thomson, T. Narten. December 1998. Internet Engineering Task Force.

[B23] RFC 2474, Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, K. Nichols, S. Blake, F. Baker, D. Black. December 1998. Internet Engineering Task Force.

[B24] RFC 2631, Diffie-Hellman Key Agreement Method, E. Rescorla. June 1999. Internet Engineering Task Force.

[B25] RFC 3290, An Informal Management Model for Diffserv Routers, Y. Bernet, S. Blake, D. Grossman, A. Smith. May 2002. Internet Engineering Task Force.

[B26] RFC 3526, More Modular Exponential (MODP) Diffie-Hellman groups for Internet Key Exchange (IKE), T. Kivinen, M. Kojo. May 2003. The Internet Engineering Task Force.

[B27] RFC 3550, RTP: A Transport Protocol for Real-Time Applications, H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson. July 2003. Internet Engineering Task Force.

[B28] RFC 3927, Dynamic Configuration of IPv4 Link-Local Addresses (proposed standard), S. Cheshire, B. Aboba, E. Guttman. May 2005. Internet Engineering Task Force.

[B29] RFC 4086, Randomness Requirements for Security, D. Eastlake, 3rd, J. Schiller, S. Crocker. June 2005. Internet Engineering Task Force.

[B30] RFC draft, NSLP for Quality-of-Service signaling, J. Manner, G. Karagiannis, A. McDonald. June 2006. Internet Engineering Task Force.

[B31] UPnP™ QoS Architecture:1.0, D. Hlasny, J. Manbeck, N. Gadiraju, S. Palm, R. Bopardikar, R.Y. Chen, R. Bardini, B. Fairman, A. Bhagwat. Dec. 2004. UPnP Forum.[16]

[16] UPnP publications are available from the UPnP Forum (http://www.upnp.org).