Network Working Group
Internet-Draft
Expires: August 28, 2006

P. Calhoun, Editor
Cisco Systems, Inc.
M. Montemurro, Editor
Chantry Networks
D. Stanley, Editor
Aruba Networks

February 24, 2006

CAPWAP Protocol Specification
draft-ietf-capwap-protocol-specification-00

Status of this Memo

Copyright Notice

Abstract

Wireless LAN product architectures have evolved from single autonomous access points to systems consisting of a centralized controller and Wireless Termination Points (WTPs). The general goal of centralized control architectures is to move access control, including user authentication and authorization, mobility management

and radio management from the single access point to a centralized controller.

This specification defines the Control And Provisioning of Wireless Access Points (CAPWAP) Protocol. The CAPWAP protocol meets the IETF CAPWAP working group protocol requirements. The CAPWAP protocol is designed to be flexible, allowing it to be used for a variety of wireless technologies. This document describes the base CAPWAP protocol, including an extension which supports the IEEE 802.11 wireless LAN protocol. Future extensions will enable support of additional wireless technologies.

Table of Contents

1. Introduction

The emergence of centralized architectures, in which simple IEEE 802.11 WTPs are managed by an Access Controller (AC) suggests that a standards based, interoperable protocol could radically simplify the deployment and management of wireless networks. WTPs require a set of dynamic management and control functions related to their primary task of connecting the wireless and wired mediums. Traditional protocols for managing WTPs are either manual static configuration via HTTP, proprietary Layer 2 specific or non-existent (if the WTPs are self-contained). This document describes the CAPWAP Protocol, a standard, interoperable protocol which enables an AC to manage a collection of WTPs. The protocol is defined to be independent of layer 2 technology. An IEEE 802.11 binding is provided to support IEEE 802.11 wireless LAN networks.

CAPWAP assumes a network configuration consisting of multiple WTPs communicating via the Internet Protocol (IP) to an AC. WTPs are viewed as remote RF interfaces controlled by the AC. The AC forwards all L2 frames to be transmitted by a WTP to that WTP via the CAPWAP protocol. L2 frames from mobile nodes (STAs) are forwarded by the WTP to the AC using the CAPWAP protocol. Both Split-MAC and Local MAC arhcitectures are supported. Figure 1 illustrates this arrangement as applied to an IEEE 802.11 binding.

```
+-+         802.11 frames         +-+
| |--------------------------------| |
| |              +-+               | |
| |--------------| |---------------| |
| |  802.11 PHY/ | |     CAPWAP     | |
| | MAC sublayer | |               | |
+-+              +-+               +-+
STA              WTP                AC
```

Figure 1: Representative CAPWAP Architecture for Split MAC

Provisioning WTPs with security credentials, and managing which WTPs are authorized to provide service are traditionally handled by proprietary solutions. Allowing these functions to be performed from a centralized AC in an interoperable fashion increases manageability and allows network operators to more tightly control their wireless network infrastructure.

Goals

Goals for the CAPWAP protocol are listed below:

1. To centralize the bridging, forwarding, authentication and policy enforcement functions for a wireless network. Optionally, the AC may also provide centralized encryption of user traffic. Centralization of these functions will enable reduced cost and higher efficiency by applying the capabilities of network processing silicon to the wireless network, as in wired LANs.

2. To enable shifting of the higher level protocol processing from the WTP. This leaves the time critical applications of wireless control and access in the WTP, making efficient use of the computing power available in WTPs which are the subject to severe cost pressure.

3. To provide a generic encapsulation and transport mechanism, enabling the CAPWAP protocol to be applied to other access point types in the future, via a specific wireless binding.

The CAPWAP protocol concerns itself solely with the interface between the WTP and the AC. Inter-AC, or mobile node (STA) to AC communication is strictly outside the scope of this document.

1.1. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [1].

1.2. Contributing Authors

This section lists and acknowledges the authors of significant text and concepts included in this specification. [Note: This section needs work to accurately reflect the contribution of each author and this work will be done in revision 01 of this document.]

The CAPWAP Working Group selected the Lightweight Access Point Protocol (LWAPP) [add reference, when available]to be used as the basis of the CAPWAP protocol specification. The following people are authors of the LWAPP document:

Bob O'Hara, Cisco Systems, Inc.,170 West Tasman Drive, San Jose, CA 95134
Phone: +1 408-853-5513, Email: bob.ohara@cisco.com

Pat Calhoun, Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 9513
4
Phone: +1 408-853-5269, Email: pcalhoun@cisco.com

Rohit Suri, Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134
Phone: +1 408-853-5548, Email: rsuri@cisco.com

Nancy Cam Winget, Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA
95134
Phone: +1 408-853-0532, Email: ncamwing@cisco.com

Scott Kelly, Facetime Communications, 1159 Triton Dr, Foster City, CA 9440
4
Phone: +1 650 572-5846, Email: scott@hyperthought.com

Michael Glenn Williams, Nokia, Inc., 313 Fairchild Drive, Mountain View, CA
94043
Phone: +1 650-714-7758, Email: Michael.G.Williams@Nokia.com

Sue Hares, Nexthop Technologies, Inc., 825 Victors Way, Suite 100, Ann Arbo
r, MI 48108
Phone: +1 734 222 1610, Email: shares@nexthop.com

DTLS is used as the security solution for the CAPWAP protocol. The following people are authors of significant DTLS-related text included in this document:

Scott Kelly, Facetime Communications, 1159 Triton Dr, Foster City, CA 9440
4
Phone: +1 650 572-5846, Email: scott@hyperthought.com

Eric Rescorla, Network Resonance, 2483 El Camino Real, #212,Palo Alto CA, 9
4303
Email: ekr@networkresonance.com

The concept of using DTLS to secure the CAPWAP protocol was part of the Secure Light Access Point Protocol (SLAPP) proposal [add reference when available]. The following people are authors of the SLAPP proposal:

Partha Narasimhan, Aruba Networks, 1322 Crossman Ave, Sunnyvale, CA 94089
Phone: +1 408-480-4716, Email: partha@arubanetworks.com

Dan Harkins, Tropos Networks, 555 Del Rey Avenue, Sunnyvale, CA, 95085
Phone: +1 408 470 7372, Email: dharkins@tropos.com

Subbu Ponnuswammy, Aruba Networks, 1322 Crossman Ave, Sunnyvale, CA 94089
Phone: +1 408-754-1213, Email: subbu@arubanetworks.com

[Ed note: Additional authors to be added as required.]

1.3. Acknowledgements

The authors thank Michael Vakulenko for contributing text that describes how CAPWAP can be used over a layer 3 (IP/UDP) network.

The authors thank Russ Housley and Charles Clancy for their assistance in provide a security review of the LWAPP specification. Charles' review can be found at [14].

[Ed note: Additional acknowledgements to be added as required.]

2. Protocol Overview

The CAPWAP protocol is a generic protocol defining AC and WTP control and data plane communication via a CAPWAP protocol transport mechanism. CAPWAP control messages, and optionally CAPWAP data messages are secured using Datagram Transport Layer Security (DTLS). DTLS is a standards-track IETF protocol based upon TLS. The underlying security-related protocol mechanisms of TLS have been successfully deployed for many years.

The CAPWAP protocol Transport layer carries two types of payload, CAPWAP Data messages and CAPWAP Control messages. CAPWAP Data messages are forwarded wireless frames. CAPWAP protocol Control messages are management messages exchanged between a WTP and an AC. The CAPWAP Data and Control packets are sent over separate UDP ports. Since both data and control frames can exceed the PMTU, the payload of a CAPWAP data or control message can be fragmented. The fragmentation behavior is highly dependent upon the lower layer transport and is defined in Section 3.

The CAPWAP Protocol begins with a discovery phase. The WTPs send a Discovery Request message, causing any Access Controller (AC) receiving the message to respond with a Discovery Response message. From the Discovery Response messages received, a WTP will select an AC with which to establish a secure DTLS session, using the DTLS initialization request message. [MTU discovery mechanism? to determine the MTU supported by the network between the WTP and AC.] CAPWAP protocol messages will be fragmented to the maximum length discovered to be supported by the network.

Once the WTP and the AC have completed DTLS session establishment, a configuration exchange occurs in which both devices to agree on version information. During this exchange the WTP may receive provisioning settings. For the IEEE 802.11 binding, this information typically includes a name (IEEE 802.11 Service Set Identifier, SSID) security parameters, the data rates to be advertised and the associated radio channel(s) to be used. The WTP is then enabled for operation.

When the WTP and AC have completed the version and provision exchange and the WTP is enabled, the CAPWAP protocol is used to encapsulate the wireless data frames sent between the WTP and AC. The CAPWAP protocol will fragment the L2 frames if the size of the encapsulated wireless user data (Data) or protocol control (Management) frames causes the resultant CAPWAP protocol packet to exceed the MTU supported between the WTP and AC. Fragmented CAPWAP packets are reassembled to reconstitute the original encapsulated payload.

The CAPWAP protocol provides for the delivery of commands from the AC to the WTP for the management of mobile units (STAs) that are communicating with the WTP. This may include the creation of local data structures in the WTP for the mobile units and the collection of statistical information about the communication between the WTP and the mobile units. The CAPWAP protocol provides a mechanism for the AC to obtain statistical information collected by the WTP.

The CAPWAP protocol provides for a keep alive feature that preserves the communication channel between the WTP and AC. If the AC fails to appear alive, the WTP will try to discover a new AC.

This Document uses terminology defined in [5].

2.1. Wireless Binding Definition

The CAPWAP protocol is independent of a specific WTP radio technology. Elements of the CAPWAP protocol are designed to accommodate the specific needs of each wireless technology in a standard way. Implementation of the CAPWAP protocol for a particular wireless technology must follow the binding requirements defined for that technology. This specification includes a binding for the IEEE 802.11 standard(see Section 11).

When defining a binding for other wireless technologies, the authors MUST include any necessary definitions for technology-specific messages and all technology-specific message elements for those messages. At a minimum, a binding MUST provide the definition for a binding-specific Statistics message element, carried in the WTP Event Request message, and a Mobile message element, carried in the Mobile Configure Request. If technology specific message elements are required for any of the existing CAPWAP messages defined in this specification, they MUST also be defined in the technology binding document.

The naming of binding-specific message elements MUST begin with the name of the technology type, e.g., the binding for IEEE 802.11, provided in this specification, begins with "IEEE 802.11"."

2.2. CAPWAP State Machine Definition

The following state diagram represents the lifecycle of a WTP-AC session:

```
              /-------------\
            |             v
            |       +------------+
            |      C|    Idle    |<---------------------------------------+
            |       +------------+                                       |
            |        ^    |a    ^                                        |
            |        |    |     \----\                 y                 |
            |        |    |          |   +-------------+------------+    |
            |        |    |          |   |             | DTLS-rekey |    |
            |        |    |          |   |  +--------->+------------+    |
            |        |    |          |   |  |                  |6        ^
            |        |    |          |t  V  | x                V         |
            |        |    |        +--------+--+       +------------+    |
            |       /     |       C|    Run    |------>| DTLS-Reset |<---|----\
            |     /       |       r+-----------+     u +------------+    |     |
            |    /        |              ^                ^     v|       |     |
            |   |         v              |                |      |       |     |
            |   |   +--------------+     |          /----/       V       |     |
            |   |  C|  Discovery   |    q|        k|           +-------+ |     |
            |   |  b+--------------+    +-------------+        | Reset |-+ w   |
            |   |     |d     f|  ^      |  Configure  |        +-------+       |
            |   |     |       |  |      +-------------+                        |
            |   |e    v       |  |              ^                              |
            |  +---------+    v  |i            2|                              |
            | C| Sulking |   +------------+    +--------------+                |
            |  +---------+  C| DTLS-Init  |--->| DTLS-Complete|                |
            |                +------------+ z  +--------------+                |
            |                   |h                         |4                  |
            |                   |                          v                 o /
             \                  |                        +------------+-------/
              \-----------------/                        | Image Data |C
                                                         +------------+n
```

Figure 2: CAPWAP State Machine

The CAPWAP protocol state machine, depicted above, is used by both the AC and the WTP. For every state defined, only certain messages are permitted to be sent and received. In all of the CAPWAP control messages defined in this document, the state for which each command is valid is specified.

Note that in the state diagram figure above, the 'C' character is used to represent a condition that causes the state to remain the same.

The following text discusses the various state transitions, and the events that cause them.

Idle to Discovery (a): This is the initialization state.

WTP: The WTP enters the Discovery state prior to transmitting the first Discovery Request message (see Section 5.1). Upon entering this state, the WTP sets the DiscoveryInterval timer (see Section 12). The WTP resets the DiscoveryCount counter to zero (0) (see Section 13). The WTP also clears all information from ACs (e.g., AC Addresses) it may have received during a previous Discovery phase.

AC: The AC does not need to maintain state information for the WTP upon reception of the Discovery Request message, but it MUST respond with a Discovery Response message (see Section 5.2).

Discovery to Discovery (b): This is the state in which the WTP determines which AC to connect to.

WTP: This event occurs when the DiscoveryInterval timer expires. The WTP transmits a Discovery Request message to every AC from which the WTP has not received a Discovery Response message. For every transition to this event, the WTP increments the DiscoveryCount counter. See Section 5.1 for more information on how the WTP knows the ACs to which ACs it should transmit the Discovery Request messages. The WTP restarts the DiscoveryInterval timer.

AC: This is a no-op.

Discovery to Sulking (d): This state occurs on a WTP when Discovery or connectivity to the AC fails.

WTP: The WTP enters this state when the DiscoveryInterval timer expires and the DiscoveryCount variable is equal to the MaxDiscoveries variable (see Section 13). Upon entering this state, the WTP shall start the SilentInterval timer. While in the Sulking state, all received CAPWAP protocol messages received shall be ignored.

AC: This is a no-op.

Sulking to Idle (e): This state occurs on a WTP when it must restart the discovery phase.

WTP: The WTP enters this state when the SilentInterval timer (see Section 12) expires.

AC: This is a no-op.

Discovery to DTLS-Init (f): This state is used by the WTP to confirm its commitment to an AC that it wishes to be provided service and to simultaneously establish a secure channel with that AC.

WTP: The WTP selects the best AC based on the information it gathered during the Discovery Phase. It then sends a ClientHello to its preferred AC, sets the WaitJoin timer, and awaits the outcome of the DTLS handshake.

AC: The AC enters this state for the given WTP upon reception of a ClientHello. The AC responds by sending either the ServerHello or the HelloVerifyRequest to the WTP. For the AC, this is a meta-state; in actuality, it remains in the Discovery state. To do otherwise resuls in loss of the stateless nature of the cookie exchange.

DTLS-Init to Idle (h): This state transition is used when the DTLS Initialization process failed.

WTP: This state transition occurs if the WTP is unable to successfully establish a DTLS session.

AC: This state transition occurs if the AC is unable to successfully establish a DTLS session.

DTLS-Init to Discovery (i): This state transition is used to return the WTP to discovery mode when an unresponsive AC is encountered.

WTP: The WTP enters the Discovery state when the DTLS handshake fails.

AC: This state transition is invalid.

DTLS-Init to DTLS-Complete (z): This state transition is used to indicate DTLS session establishment.

WTP: The DTLS-Complete state is entered when the WTP receives the Finished message from the AC.

AC: The DTLS-Complete state is entered when the AC receives the Finished mesage from the WTP.

DTLS-Complete to Configure (2): This state transition is used by the WTP and the AC to exchange configuration information.

WTP: The WTP enters the Configure state when it successfully completes DTLS session establishment and determines that its version number and the version number advertised by the AC are the same. The WTP transmits the Configure Request message(see Section 7.2) message to the AC with a snapshot of its current configuration. The WTP also starts the ResponseTimeout timer (see Section 12).

AC: This state transition occurs when the AC receives the Configure Request message from the WTP. The AC must transmit a Configure Response message(see Section 7.3) to the WTP, and may include specific message elements to override the WTP's configuration.

DTLS Complete to Image Data (4): This state transition is used by the WTP and the AC to download executable firmware.

WTP: The WTP enters the Image Data state when it successfully comletes DTLS session establishment, and determines that its version number and the version number advertised by the AC are different. The WTP transmits the Image Data Request (see Section 8.1) message requesting that the AC's latest firmware be initiated.

AC: This state transition occurs when the AC receives the Image Data Request message from the WTP. The AC must transmit an Image Data Response message(see Section 8.2) to the WTP, which includes a portion of the firmware.

Image Data to Image Data (n): The Image Data state is used by WTP and the AC during the firmware download phase.

WTP: The WTP enters the Image Data state when it receives a Image Data Response message indicating that the AC has more data to send.

AC: This state transition occurs when the AC receives the Image Data Request message from the WTP while already in the Image Data state, and it detects that the firmware download has not completed.

Configure to DTLS-Reset (k): This state is used to reset the DTLS connection prior to restarting the WTP with a new configuration.

WTP: The WTP enters the DTLS-Reset state when it determines that a new configuration is required.

AC: The AC transitions to the DTLS-Reset state when the DTLS connection tear-down is complete.

Image Data to DTLS-Reset (o): This state transition is used to reset the DTLS connection prior to restarting the WTP after an image download.

WTP: The WTP enters the DTLS-Reset state when image download completes.

AC: The AC enters the DTLS-Reset state upon receipt of TLS Finished message from the WTP.

Configure to Run (q): This state transition occurs when the WTP and AC enter their normal state of operation.

WTP: The WTP enters this state when it receives a successful Configure Response message from the AC. The WTP initializes the HeartBeat Timer (see Section 12), and transmits the Change State Event Request message (see Section 7.6).

AC: This state transition occurs when the AC receives the Change State Event Request message (see Section 7.6) from the WTP. The AC responds with a Change State Event Response (see Section 7.7) message. The AC must start the NeighborDeadInterval timer (see Section 12).

Run to Run (r): This is the normal state of operation.

WTP: This is the WTP’s normal state of operation. There are many events that result this state transition:

Configuration Update: The WTP receives a Configuration Update Request message(see Section 7.4). The WTP MUST respond with a Configuration Update Response message (see Section 7.5).

Change State Event: The WTP receives a Change State Event Response message, or determines that it must initiate a Change State Event Request message, as a result of a failure or change in the state of a radio.

Echo Request: The WTP receives an Echo Request message (see Section 6.1), to which it MUST respond with an Echo Response message(see Section 6.2).

Clear Config Indication: The WTP receives a Clear Config Indication message (see Section 7.8). The WTP MUST reset its configuration back to manufacturer defaults.

WTP Event: The WTP generates a WTP Event Request message to send information to the AC (see Section 8.5). The WTP receives a WTP Event Response message from the AC (see Section 8.6).

Data Transfer: The WTP generates a Data Transfer Request message to the AC (see Section 8.7). The WTP receives a Data Transfer Response message from the AC (see Section 8.8).

WLAN Config Request: The WTP receives a WLAN Config Request message (see Section 11.8.1), to which it MUST respond with a WLAN Config Response message (see Section 11.8.2).

Mobile Config Request: The WTP receives a Mobile Config Request message (see Section 9.1), to which it MUST respond with a Mobile Config Response message (see Section 9.2).

AC: This is the AC's normal state of operation:

Configuration Update: The AC sends a Configuration Update Request message (see Section 7.4) to the WTP to update its configuration. The AC receives a Configuration Update Response message (see Section 7.5) from the WTP.

Change State Event: The AC receives a Change State Event Request message (see Section 7.6), to which it MUST respond with the Change State Event Response message (see Section 7.7).

Echo: The AC sends an Echo Request message Section 6.1) or receives the corresponding Echo Response message (see Section 6.2) from the WTP.

Clear Config Indication: The AC sends a Clear Config Indication message (see Section 7.8).

WLAN Config: The AC sends a WLAN Config Request message (see Section 11.8.1) or receives the corresponding WLAN Config Response message (see Section 11.8.2) from the WTP.

Mobile Config: The AC sends a Mobile Config Request message (see Section 9.1) or receives the corresponding Mobile Config Response message (see Section 9.2) from the WTP.

Data Transfer: The AC receives a Data Transfer Request message from the AC (see Section 8.7) and MUST generate a corresponding Data Transfer Response message (see Section 8.8).

WTP Event: The AC receives a WTP Event Request message from the AC (see Section 8.5) and MUST generate a corresponding WTP Event Response message (see Section 8.6).

Run to Idle (t): This event occurs when an error occurs in the communication between the WTP and the AC.

WTP: The WTP enters the Idle state when the underlying reliable transport in unable to transmit a message within the RetransmitInterval timer (see Section 12), and the maximum number of RetransmitCount counter has reached the MaxRetransmit variable (see Section 13).

AC: The AC enters the Idle state when the underlying reliable transport in unable to transmit a message within the RetransmitInterval timer (see Section 12), and the maximum number of RetransmitCount counter has reached the MaxRetransmit variable (see Section 13).

Run to DTLS-Reset(u): This state transition is used to when the AC or WTP wish to tear down the connection.

WTP: The WTP enters the DTLS-Reset state when it initiates orderly termination of the DTLS connection; The WTP sends a TLS Finished message to the AC.

AC: The AC enters the DTLS-Reset state upon receipt of a TLS Finished message from the WTP.

Run to DTLS-Rekey (x): This state is used to initiate a new DTLS handshake. Either the WTP or AC may initiate the state transition. DTLS protected CAPWAP packets may continue to flow while a new handshake is being performed. Because packets may be reordered, records encrypted under the new cipher suite may be received before one side receives the ChangeCipherSpec from the other side.

The epoch value in the DTLS record header allows the data from the two associations/cryptographic states to be distinguished. Implementations SHOULD retain the state for the old association until it is likely that all old records have been received or dropped, e.g., for the maximum packet lifetime. If the state is dropped too early, the only effect will be that some data is lost, which is a condition that systems running over unreliable protocols need to consider in any case.

Because the new handshake is performed over the existing DTLS association, both sides can be confident that the handshake was properly initiated and was not tampered with. All data is protected under either the old or new keys--and these can be distinguished by both the epoch and the authentication (MAC) verification. Thus, there is no period during which data is unprotected.

WTP: The WTP enters the DTLS-Rekey state when either (1) a rekey is required, or (2) the AC initiates a DTLS handshake.

AC: The AC enters the DTLS-Rekey state when either (1) a rekey is required, or (2) the WTP initiates a DTLS handshake.

DTLS-rekey to Run (y): This event occurs when the DTLS rehandshake is completed.

WTP: This state transition occurs when the WTP completes the DTLS rehandshake.

AC: This state transition occurrs when the AC completed the DTLS rehandshake.

DTLS-rekey to Reset (6): This event occurs when the DTLS rehandshake exchange phase times out.

WTP: This state transition occurs when the WTP does not successfully complete the DTLS rehandshake phase.

AC: This state transition occurs when the AC does not successfully complete the DTLS rehandshake phase.

DTLS-Reset to Reset (v): This state transition is used to complete DTLS session tear-down.

WTP: The WTP enters the Reset state when it has completed DTLS session clean-up, and it is ready to complete the CAPWAP protocol session clean-up.

AC: The AC enters the Reset state when it has completed DTLS session clean-up, and it is ready to complete the CAPWAP protocol session clean-up.

Reset to Idle (w): This event occurs when the state machine is restarted.

WTP: The WTP reboots. After reboot the WTP will start its CAPWAP state machine in the Idle state.

AC: The AC clears any state associated with the WTP. The AC generally does this as a result of the reliable link layer timing out.

2.3. Use of DTLS in the CAPWAP Protocol

DTLS is used as a tightly-integrated secure wrapper for the CAPWAP protocol. Certain errors may occur during the DTLS negotiation and/or the resulting session; the following section describes those, along with handling requirements. It is important to note that the CAPWAP protocol, being the controlling entity for the DTLS session, must establish its own timers outside of DTLS (e.g. WaitJoin), and is responsible for terminating sessions which timeout. DTLS implements a retransmission backoff timer, but will not terminate a session unless instructed to do so.

2.3.1. DTLS Error Handling Requirements

DTLS uses all of the same handshake messages and flows as TLS, with three principal changes:

1. A stateless cookie exchange has been added to prevent denial of service attacks.

2. Modifications to the handshake header have been made to handle message loss, reordering, and fragmentation

3. Retransmission timers to handle message loss have been added.

Each of these features can cause the DTLS session to fail, as discussed below. For reference, an illustration of a normal DTLS session establishment (in this particular case, using certificates for authentication) is as follows:

```
            Client (WTP)                         Server (AC)
            ------------                         ------------
            ClientHello           ------>

                                  <-----       HelloVerifyRequest
                                               (contains cookie)

            ClientHello           ------>
            (with cookie)
                                  <------      ServerHello (seq=1)
                                  <------      Certificate (seq=2)
                                  <------      ServerHelloDone (seq=3)
            Certificate*
            ClientKeyExchange
            CertificateVerify*
            [ChangeCipherSpec]
            Finished               ------>

                                               [ChangeCipherSpec]
                                   <------     Finished
```

2.3.2. DTLS Cookie Exchange Failure

The cookie exchange is optional in DTLS. For use with the CAPWAP protocol, it may not be required if the network on which the AC and WTP reside is entirely within the same administrative domain. However, if AC-WTP communications traverse multiple administrative domains, the cookie exchange SHOULD be supported. There are three potential points of failure in Hello exchange, assuming cookies are used:

- o The AC does not respond to the ClientHello (this may occur independently of cookie usage)

- o The WTP does not respond to the HelloVerifyRequest

- o The ClientHello contains an invalid cookie

In determining appropriate error handling behavior for any of these cases, it is important to remember that the stateless cookie implements a defense mechanism from the point of view of the AC. That is, it is explictly designed to minimize AC-side processing prior to verifying that the WTP can receive and respond to packets at the specified address. Hence, any processing associated with this mechanism SHOULD be minimized.

In the case of AC non-responsiveness to the ClientHello, the WaitJoin timer will eventually expire. When this occurs, the WTP SHOULD log an error message and choose an alternative AC if one exists, or return to the CAPWAP protocol Discovery state.

In the case of WTP non-responsiveness to the HelloVerifyRequest, the DTLS implementation purposely does not set a timer (the HelloVerifyRequest is stateless by design). This means that DTLS itself will provide no indication of WTP non-responsiveness. To mitigate this, the AC MAY log a message when sending a HelloVerifyRequest, and SHOULD log a message upon receipt of a valid corresponding ClientHello. In this way, optional external detection of non-responsive WTP’s can be used to troubleshoot such problems using data from the AC alone. In reality, administrators will typically have access to WTP logs as well, making detection of such problems straightforward.

In case of an invalid cookie in the ClientHello, the AC MUST terminate the DTLS handshake, returing to Discovery state. A DTLS alert MAY be sent to the WTP indicating the failure.

2.3.3. DTLS Re-Assembly Failure

Since DTLS handshake messages are potentially larger than the maximum record size, DTLS supports fragmenting of handshake messages across multiple records. There are several potential causes of re-assembly errors, including overlapping and/or lost fragments. The DTLS implementation should return an error to the CAPWAP protocol implementation when such errors occur. The precise error value is an API issue, and hence is beyond the scope of this document. Upon receipt of such an error, the CAPWAP protocol implementation SHOULD log an appropriate error message. Whether processing continues or the DTLS session is terminated is implementation dependent.

3. CAPWAP Transport

The CAPWAP protocol uses UDP as a transport, and can be used with IPv4 or IPv6. This section details the specifics of how the CAPWAP protocol works in conjunction with IP.

3.1. UDP Transport

Communication between a WTP and an AC is established according to the standard UDP client/server model. One of the CAPWAP requirements is to allow a WTP to reside behind a firewall and/or Network Address Translation (NAT) device. Since the connection is initiated by the WTP (client) to the well-known UDP port of the AC (server), the use of UDP is a logical choice.

CAPWAP protocol control packets sent between the WTP and the AC use well known UDP port 12222. CAPWAP protocol data packets sent between the WTP and the AC use UDP port [to be IANA assigned].

3.2. AC Discovery

A WTP and an AC will frequently not reside in the same IP subnet (broadcast domain). When this occurs, the WTP must be capable of discovering the AC, without requiring that multicast services are enabled in the network. This section describes how AC discovery is performed by WTPs.

As the WTP attempts to establish communication with an AC, it sends the Discovery Request message and receives the corresponding response message from the AC(s). The WTP must send the Discovery Request message to either the limited broadcast IP address (255.255.255.255), a well known multicast address or to the unicast IP address of the AC. Upon receipt of the Discovery Request message, the AC issues a Discovery Response message to the unicast IP address of the WTP, regardless of whether the Discovery Request message was sent as a broadcast, multicast or unicast message.

WTP use of a limited IP broadcast, multicast or unicast IP address is implementation dependent.

When a WTP transmits a Discovery Request message to a unicast address, the WTP must first obtain the IP address of the AC. Any static configuration of an AC's IP address on the WTP non-volatile storage is implementation dependent. However, additional dynamic schemes are possible, for example:

DHCP: A comma delimited ASCII encoded list of AC IP addresses is embedded in the DHCP vendor specific option 43 extension. An example of the actual format of the vendor specific payload for IPv4 is of the form "10.1.1.1, 10.1.1.2".

DNS: The DNS name "CAPWAP-AC-Address" MAY be resolvable to one or more AC addresses.

3.3. Fragmentation/Reassembly

While fragmentation and reassembly services are provided by IP, the CAPWAP protocol also provides such services. Environments where the CAPWAP protocol is used involve firewall, Network Address Translation (NAT) and "middle box" devices, which tend to drop IP fragments in order to minimize possible Denial of Service attacks. By providing fragmentation and reassembly at the application layer, any fragmentation required due to the tunneling component of the CAPWAP protocol becomes transparent to these intermediate devices. Consequently, the CAPWAP protocol is not impacted by any network configurations.

4. CAPWAP Packet Formats

This section contains the CAPWAP protocol packet formats. A CAPWAP protocol packet consists of a CAPWAP Transport Layer packet header followed by a CAPWAP message. The CAPWAP message can be either of type Control or Data, where Control packets carry signaling, and Data packets carry user payloads. The CAPWAP frame formats for CAPWAP Data packets, and for DTLS encapsulated CAPWAP Data and Control packets. are as shown below:

```
   CAPWAP Data Packet :
    +---------------------------------+
    | IP  |UDP  | CAPWAP | Wireless  |
    | Hdr |Hdr  | Header | Payload   |
    +---------------------------------+



    CAPWAP + Optional DTLS Data Packet Security:
    +-------------------------------------------------+
    | IP  |UDP | DTLS  | CAPWAP  | Wireless | DTLS   |
    | Hdr |Hdr | Hdr   | Hdr     | Payload  | Trailer|
    +-------------------------------------------------+
                \--authenticated-----------/
                        \---     encrypted-----------/



    CAPWAP Control Packet (DTLS Security Required):
    +------------------------------------------------------------+
    | IP  |UDP | DTLS | CAPWAP | Control | Message    | DTLS    |
    | Hdr |Hdr | Hdr  | Header | Header  | Element(s) | Trailer |
    +------------------------------------------------------------+
                \-------authenticated----------------/
                       \------------encrypted-------------------/
```

UDP: All CAPWAP packets are encapsulated within UDP. Section Section 3.1 defines the specific UDP usage.

CAPWAP Header: All CAPWAP protocol packets use a common header that immediately follows the UDP header. This header, is defined in Section 4.1.

Wireless Payload: A CAPWAP protocol packet that contains a wireless payload is known as a data frame. The CAPWAP protocol does not dictate the format of the wireless payload, which is defined by the appropriate wireless standard. Additional information is in Section 4.2.

Control Header: The CAPWAP protocol includes a signalling component, known as the CAPWAP control protocol. All CAPWAP control packets include a Control Header, which is defined in Section 4.3.1.

Message Elements: A CAPWAP Control packet includes one or more message elements, which are found immediately following the control header. These message elements are in a Type/Length/value style header, defined in Section 4.3.2.

4.1. CAPWAP Transport Header

All CAPWAP protocol messages are encapsulated using a common header format, regardless of the CAPWAP control or CAPWAP Data transport used to carry the messages. However, certain flags are not applicable for a given transport. Refer to the specific transport section in order to determine which flags are valid.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|VER| RID |F|L|R|    Frag ID    |            Length             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Status/WLANs         |  Payload...  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

4.1.1. VER Field

A 2 bit field which contains the version of CAPWAP used in this packet. The value for this draft is 0.

4.1.2. RID Field

A 3 bit field which contains the Radio ID number for this packet. WTPs with multiple radios but a single MAC Address use this field to indicate which radio is associated with the packet.

4.1.3. F Bit

The Fragment 'F' bit indicates whether this packet is a fragment. When this bit is one (1), the packet is a fragment and MUST be combined with the other corresponding fragments to reassemble the complete information exchanged between the WTP and AC.

4.1.4. L Bit

The Not Last 'L' bit is valid only if the 'F' bit is set and indicates whether the packet contains the last fragment of a fragmented exchange between WTP and AC. When this bit is 1, the packet is not the last fragment. When this bit is 0, the packet is the last fragment.

4.1.5. R Bit

The R bit is reserved and set to 0 in this version of the CAPWAP protocol.

4.1.6. Fragment ID

An 8 bit field whose value is assigned to each group of fragments making up a complete set. The fragment ID space is managed individually for every WTP/AC pair. The value of Fragment ID is incremented with each new set of fragments. The Fragment ID wraps to zero after the maximum value has been used to identify a set of fragments. The CAPWAP protocol only supports up to 2 fragments per frame.

4.1.7. Length

The 16 bit length field contains the number of bytes in the Payload. The field is encoded as an unsigned number.

4.1.8. Status and WLANS

The interpretation of this 16 bit field is binding specific. Refer to the transport portion of the binding for a specific wireless technology for the definition of this field.

4.1.9. Payload

This field contains the header for a CAPWAP Data Message or CAPWAP Control Message, followed by the data associated with that message.

4.2. CAPWAP Data Messages

A CAPWAP protocol data message is a forwarded wireless frame. The CAPWAP protocol defines two different modes of encapsulations; IEEE 802.3 and native wireless. IEEE 802.3 encapsulation requires that the bridging function be performed in the WTP. An IEEE 802.3 encapsulated user payload frame has the following format:

```
+-------------------------------------------------------+
| IP Header | UDP Header | CAPWAP Header | 802.3 Frame |
+-------------------------------------------------------+
```

The CAPWAP protocol also defines the native wireless encapsulation mode. The actual format of the encapsulated CAPWAP data frame is subject to the rules defined under the specific wireless technology binding. As a consequence, each wireless technology binding MUST define a section entitled "Payload encapsulation", which defines the format of the wireless payload that is encapsulated within the CAPWAP Data messages.

In the event that the encapsulated frame would exceed the transport layer's MTU, the sender is responsible for the fragmentation of the frame, as specified in Section 3.3.

4.3. CAPWAP Control Messages Overview

The CAPWAP Control protocol provides a control channel between the WTP and the AC. Control messages are divided into the following distinct message types:

Discovery: CAPWAP Discovery messages are used to identify potential ACs, their load and capabilities.

WTP Configuration: The WTP Configuration messages are used by the AC to push a specific configuration to the WTP it has a control channel with. Messages that deal with the retrieval of statistics from the WTP also fall in this category.

Mobile Session Management: Mobile session management messages are used by the AC to push specific mobile policies to the WTP.

Firmware Management: Messages in this category are used by the AC to push a new firmware image to the WTP.

Discovery, WTP Configuration and Mobile Session Management messages MUST be implemented. Firmware Management MAY be implemented.

In addition, technology specific bindings may introduce new control channel commands.

4.3.1. Control Message Format

All CAPWAP control messages are sent encapsulated within the CAPWAP header (see Section 4.1). Immediately following the CAPWAP header, is the control header, which has the following format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Message Type |    Seq Num    |      Msg Element Length       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Msg Element [0..N]       |
```

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

4.3.1.1. Message Type

The Message Type field identifies the function of the CAPWAP control message. The valid values for Message Type are the following:

Description	Value
Discovery Request	1
Discovery Response	2
Configure Request	3
Configure Response	4
Configuration Update Request	5
Configuration Update Response	6
WTP Event Request	7
WTP Event Response	8
Change State Event Request	9
Change State Event Response	10
Echo Request	11
Echo Response	12
Unused	13
Image Data Request	14
Image Data Response	15
Reset Request	16
Reset Response	17
Primary Discovery Request	18
Primary Discovery Response	19
Data Transfer Request	20
Data Transfer Response	21
Clear Config Indication	22
WLAN Config Request	23
WLAN Config Response	24
Mobile Config Request	25
Mobile Config Response	26

4.3.1.2. Sequence Number

The Sequence Number Field is an identifier value to match request/ response packet exchanges. When a CAPWAP packet with a request message type is received, the value of the sequence number field is copied into the corresponding response packet.

When a CAPWAP control message is sent, its internal sequence number counter is monotonically incremented, ensuring that no two requests pending have the same sequence number. This field will wrap back to zero.

4.3.1.3. Message Element Length

The Length field indicates the number of bytes following the Sequence Num field.

4.3.1.4. Message Element[0..N]

The message element(s) carry the information pertinent to each of the control message types. Every control message in this specification specifies which message elements are permitted.

4.3.2. Message Element Format

The message element is used to carry information pertinent to a control message. Every message element is identified by the Type field, whose numbering space is managed via IANA (see Section 16). The total length of the message elements is indicated in the Message Element Length field.

All of the message element definitions in this document use a diagram similar to the one below in order to depict its format. Note that in order to simplify this specification, these diagrams do not include the header fields (Type and Length). The header field values are defined in the Message element descriptions.

Additional message elements may be defined in separate IETF documents.

The format of a message element uses the TLV format shown here:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Type             |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Value ...   |
+-+-+-+-+-+-+-+-+
```

Where Type (16 bit) identifies the character of the information carried in the Value field and Length (16 bits) indicates the number of bytes in the Value field.

4.3.2.1. Generic Message Elements

This section includes message elements that are not bound to a specific control message.

4.3.2.1.1. Vendor Specific

The Vendor Specific Payload is used to communicate vendor specific information between the WTP and the AC. The value contains the following format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Vendor Identifier                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Element ID           |   Value...    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 104 for Vendor Specific

Length: >= 7

Vendor Identifier: A 32-bit value containing the IANA assigned "SMI Network Management Private Enterprise Codes" [17]

Element ID: A 16-bit Element Identifier which is managed by the vendor.

Value: The value associated with the vendor specific element.

4.3.3. Quality of Service

It is recommended that CAPWAP control messages be sent by both the AC and the WTP with an appropriate Quality of Service precedence value, ensuring that congestion in the network minimizes occurrences of CAPWAP control channel disconnects. Therefore, a Quality of Service enabled CAPWAP device should use:

802.1P: The precedence value of 7 SHOULD be used.

DSCP: The DSCP tag value of 46 SHOULD be used.

5. CAPWAP Discovery Operations

The Discovery messages are used by a WTP to determine which ACs are available to provide service, and the capabilities and load of the ACs.

5.1. Discovery Request

The Discovery Request message is used by the WTP to automatically discover potential ACs available in the network. The Discovery Request message provides ACs with the primary capabilities of the WTP. A WTP must exchange this information to ensure subsequent exchanges with the ACs are consistent with the WTP’s functional characteristics. A WTP must transmit this command even if it has a statically configured AC.

Discovery Request messages MUST be sent by a WTP in the Discover state after waiting for a random delay less than MaxDiscoveryInterval, after a WTP first comes up or is (re)initialized. A WTP MUST send no more than the maximum of MaxDiscoveries Discovery Request messages, waiting for a random delay less than MaxDiscoveryInterval between each successive message.

This is to prevent an explosion of WTP Discovery Request messages. An example of this occurring is when many WTPs are powered on at the same time.

Discovery Request messages MUST be sent by a WTP when no Echo Response messages are received for NeighborDeadInterval and the WTP returns to the Idle state. Discovery Request messages are sent after NeighborDeadInterval. They MUST be sent after waiting for a random delay less than MaxDiscoveryInterval. A WTP MAY send up to a maximum of MaxDiscoveries Discovery Request messages, waiting for a random delay less than MaxDiscoveryInterval between each successive message.

If a Discovery Response message is not received after sending the maximum number of Discovery Request messages, the WTP enters the Sulking state and MUST wait for an interval equal to SilentInterval before sending further Discovery Request messages.

The Discovery Request message may be sent as a unicast, broadcast or multicast message.

Upon receiving a Discovery Request message, the AC will respond with a Discovery Response message sent to the address in the source address of the received discovery request message.

The following subsections define the message elements that MUST be included in the Discovery Request message.

5.1.1. Discovery Type

The Discovery message element is used to configure a WTP to operate in a specific mode.

```
 0
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
| Discovery Type|
+-+-+-+-+-+-+-+-+
```

Type: 58 for Discovery Type

Length: 1

Discovery Type: An 8-bit value indicating how the AC was discovered. The following values are supported:

0 - Broadcast

1 - Configured

5.1.2. WTP Descriptor

The WTP descriptor message element is used by the WTP to communicate it’s current hardware/firmware configuration. The value contains the following fields.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Hardware   Version                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Software   Version                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Boot   Version                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Max Radios  | Radios in use |    Encryption Capabilities    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 3 for WTP Descriptor

Length: 16

Hardware Version: A 32-bit integer representing the WTP's hardware version number

Software Version: A 32-bit integer representing the WTP's Firmware version number

Boot Version: A 32-bit integer representing the WTP's boot loader's version number

Max Radios: An 8-bit value representing the number of radios (where each radio is identified via the RID field) supported by the WTP

Radios in use: An 8-bit value representing the number of radios present in the WTP

Encryption Capabilities: This 16-bit field is used by the WTP to communicate it's capabilities to the AC. Since most WTP's support link layer encryption, the AC may make use of these services. There are binding dependent encryption capabilities. A WTP that does not have any encryption capabilities would set this field to zero (0). Refer to the specific binding for further specification of the Encryption Capabilities field.

5.1.3. WTP Radio Information

The WTP radios information message element is used to communicate the radio information in a specific slot. The Discovery Request MUST include one such message element per radio in the WTP. The Radio-Type field is used by the AC in order to determine which technology specific binding is to be used with the WTP.

The value contains two fields, as shown.

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Radio ID    |           Radio Type          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 4 for WTP Radio Information

Length: 3

Radio ID: The Radio Identifier, which typically refers to an interface index on the WTP

Radio Type: The type of radio present. Note this bitfield can be used to specify support for more than a single type of PHY/MAC. The following values are supported:

1 - 802.11b: An IEEE 802.11b radio.

2 - 802.11a: An IEEE 802.11a radio.

4 - 802.11g: An IEEE 802.11g radio.

8 - 802.11n: An IEEE 802.11n radio.

65535 - all: Used to specify all radios in the WTP.

5.1.4. WTP MAC Type

The WTP MAC-Type message element allows the WTP to communicate its mode of operation to the AC. A WTP that advertises support for both modes allows the AC to select the mode to use, based on local policy.

```
 0
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|   MAC Type    |
+-+-+-+-+-+-+-+-+
```

Type: TBD for WTP MAC Type

Length: 1

MAC Type: The MAC mode of operation supported by the WTP. The following values are supported

0 - Local-MAC: Local-MAC is the default mode that MUST be supported by all WTPs.

1 - Split-MAC: Split-MAC support is optional, and allows the AC to receive and process native wireless frames.

2 - Both: WTP is capable of supporting both Local-MAC and Split-MAC.

5.1.5. WTP Frame Type

The WTP Frame-Type message element allows the WTP to communicate the tunneling modes of operation which it supports to the AC. A WTP that advertises support for all modes allows the AC to select which mode will be used, based on its local policy.

```
  0
  0 1 2 3 4 5 6 7
 +-+-+-+-+-+-+-+-+
 |   Frame Type  |
 +-+-+-+-+-+-+-+-+
```

Type: TBD for WTP Frame Type

Length: 1

Frame Type: The Frame type specifies the encapsulation modes supported by the WTP. The following values are supported

1 - Local Bridging: Local Bridging allows the WTP to perform the bridging function. This value MUST NOT be used when the MAC Type is set to Split-MAC.

2 - 802.3 Bridging: 802.3 Bridging requires the WTP and AC to encapsulate all user payload as native IEEE 802.3 frames (see Section 4.2). This value MUST NOT be used when the MAC Type is set to Split-MAC.

4 - Native Bridging: Native Bridging requires the WTP and AC to encapsulate all user payloads as native wireless frames, as defined by the wireless binding (see Section 4.2).

7 - All: The WTP is capable of supporting all frame types.

5.2. Discovery Response

The Discovery Response message provides a mechanism for an AC to advertise its services to requesting WTPs.

Discovery Response messages are sent by an AC after receiving a Discovery Request message from a WTP.

When a WTP receives a Discovery Response message, it MUST wait for an interval not less than DiscoveryInterval for receipt of additional Discovery Response messages. After the DiscoveryInterval elapses, the WTP enters the DTLS-Init state and selects one of the ACs that sent a Discovery Response message and send a DTLS Handshake to that AC.

The following subsections define the message elements that MUST be included in the Discovery Response Message.

5.2.1. AC Address

The AC address message element is used to communicate the identity of the AC. The value contains two fields, as shown.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Reserved    |                  MAC Address                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 MAC Address                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 2 for AC Address

Length: 7

Reserved: MUST be set to zero

Mac Address: The MAC Address of the AC

5.2.2. AC Descriptor

The AC payload message element is used by the AC to communicate it’s current state. The value contains the following fields.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Reserved    |                Hardware  Version ...          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     HW Ver    |                Software  Version ...          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     SW Ver    |           Stations            |      Limit    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Limit     |           Radios              |   Max Radio   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Max Radio   |    Security   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 6 for AC Descriptor

Length: 18

Reserved: MUST be set to zero

Hardware Version: The AC’s hardware version number

Software Version: The AC’s Firmware version number

Stations: The number of mobile stations currently associated with the AC

Limit: The maximum number of stations supported by the AC

Radios: The number of WTPs currently attached to the AC

Max Radio: The maximum number of WTPs supported by the AC

Security: A 8 bit bit mask specifying the authentication credential type supported by the AC. The following values are supported (see Section 10):

1 - X.509 Certificate Based

2 - Pre-Shared Secret

5.2.3. AC Name

The AC name message element contains an ASCII representation of the AC’s identity. The value is a variable length byte string. The string is NOT zero terminated.

```
  0
  0 1 2 3 4 5 6 7
 +-+-+-+-+-+-+-+-+
 | Name ...
 +-+-+-+-+-+-+-+-+
```

Type: 31 for AC Name

Length: > 0

Name: A variable length ASCII string containing the AC’s name

5.2.4. WTP Manager Control IPv4 Address

The WTP Manager Control IPv4 Address message element is sent by the AC to the WTP during the discovery process and is used by the AC to provide the interfaces available on the AC, and the current number of WTPs connected. In the event that multiple WTP Manager Control IPV4 Address message elements are returned, the WTP is expected to perform load balancing across the multiple interfaces.

```
      0                   1                   2                   3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                           IP Address                          |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |          WTP Count            |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 99 for WTP Manager Control IPv4 Address

Length: 6

IP Address: The IP Address of an interface.

WTP Count: The number of WTPs currently connected to the interface.

5.2.5. WTP Manager Control IPv6 Address

The WTP Manager Control IPv6 Address message element is sent by the AC to the WTP during the discovery process and is used by the AC to provide the interfaces available on the AC, and the current number of WTPs connected. This message element is useful for the WTP to perform load balancing across multiple interfaces.

```
      0                   1                   2                   3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                           IP Address                          |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                           IP Address                          |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                           IP Address                          |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                           IP Address                          |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |          WTP Count            |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 142 for WTP Manager Control IPv6 Address

Length: 18

IP Address: The IP Address of an interface.

WTP Count: The number of WTPs currently connected to the interface.

5.3. Primary Discovery Request

The Primary Discovery Request message is sent by the WTP to determine whether its preferred (or primary) AC is available.

A Primary Discovery Request message is sent by a WTP when it has a primary AC configured, and is connected to another AC. This generally occurs as a result of a failover, and is used by the WTP as a means to discover when its primary AC becomes available. As a consequence, this message is only sent by a WTP when it is in the Run state.

The frequency of the Primary Discovery Request messages should be no more often than the sending of the Echo Request message.

Upon receipt of a Discovery Request message, the AC responds with a Primary Discovery Response message sent to the address in the source address of the received Primary Discovery Request message.

The following subsections define the message elements that MUST be included in the Primary Discovery message.

5.3.1. Discovery Type

The Discovery Type message element is defined in Section 5.1.1.

5.3.2. WTP Descriptor

The WTP Descriptor message element is defined in Section 5.1.2.

5.3.3. WTP MAC Type

The Discovery Type message element is defined in Section 5.1.4.

5.3.4. WTP Frame Type

The WTP Frame Type message element is defined in Section 5.1.5.

5.3.5. WTP Radio Information

A WTP Radio Information message element must be present for every radio in the WTP. This message element is defined in Section 5.1.3.

5.4. Primary Discovery Response

The Primary Discovery Response message enables an AC to advertise its availability and services to requesting WTPs that are configured to have the AC as its primary AC.

Primary Discovery Response messages are sent by an AC after receiving a Primary Discovery Request message.

When a WTP receives a Primary Discovery Response message, it may establish a CAPWAP protocol connection to its primary AC, based on the configuration of the WTP Fallback Status message element on the WTP.

The following subsections define the message elements that MUST be included in the Primary Discovery Request message.

5.4.1. AC Descriptor

The Discovery Type message element is defined in Section 5.2.2.

5.4.2. AC Name

The AC Name message element is defined in Section 5.2.3.

5.4.3. WTP Manager Control IPv4 Address

A WTP Radio Information message element MAY be present for every radio in the WTP which are reachable via IPv4. This message element is defined in Section 5.2.4.

5.4.4. WTP Manager Control IPv6 Address

A WTP Radio Information message element must be present for every radio in the WTP which are reachable via IPv6. This message element is defined in Section 5.2.5.

6. Control Channel Management

The Control Channel Management messages are used by the WTP and AC to maintain a control communication channel.

6.1. Echo Request

The Echo Request message is a keep alive mechanism for CAPWAP control messages.

Echo Request messages are sent periodically by a WTP in the Run state (see Section 2.2) to determine the state of the connection between the WTP and the AC. The Echo Request message is sent by the WTP when the Heartbeat timer expires. The WTP MUST start its NeighborDeadInterval timer when the Heartbeat timer expires.

The Echo Request message carries no message elements.

When an AC receives an Echo Request message it responds with an Echo Response message.

6.2. Echo Response

The Echo Response message acknowledges the Echo Request message, and is only processed while in the Run state (see Section 2.2).

An Echo Response message is sent by an AC after receiving an Echo Request message. After transmitting the Echo Response message, the AC SHOULD reset its Heartbeat timer to expire in the value configured for EchoInterval. If another Echo Request message is not received by the AC when the timer expires, the AC SHOULD consider the WTP to be no longer be reachable.

The Echo Response message carries no message elements.

When a WTP receives an Echo Response message it stops the NeighborDeadInterval timer, and initializes the Heartbeat timer to the EchoInterval.

If the NeighborDeadInterval timer expires prior to receiving an Echo Response message, the WTP enters the Idle state.

7. WTP Configuration Management

Wireless Termination Point Configuration messages are used to exchange configuration information between the AC and the WTP.

7.1. Configuration Consistency

The CAPWAP protocol provides flexibility in how WTP configuration is managed. A WTP has two options:

1. The WTP retains no configuration and accepts the configuration provided by the AC.

2. The WTP retains the configuration of parameters provided by the AC that are non-default values.

If the WTP opts to save configuration locally, the CAPWAP protocol state machine defines the Configure state, which allows for configuration exchange. In the Configure state, the WTP sends its current configuration overrides to the AC via the Configure Request message. A configuration override is a parameter that is non-default. One example is that in the CAPWAP protocol, the default antenna configuration is internal omni antenna. A WTP that either has no internal antennas, or has been explicitly configured by the AC to use external antennas, sends its antenna configuration during the configure phase, allowing the AC to become aware of the WTP's current configuration.

Once the WTP has provided its configuration to the AC, the AC sends its own configuration. This allows the WTP to inherit the configuration and policies from the AC.

An AC maintains a copy of each active WTP's configuration. There is no need for versioning or other means to identify configuration changes. If a WTP becomes inactive, the AC MAY delete the configuration associated with it. If a WTP fails, and connects to a new AC, it provides its overridden configuration parameters, allowing the new AC to be aware of the WTP's configuration.

This model allows for resiliency in case of an AC failure, that another AC can provide service to the WTP. In this scenario, the new AC would be automatically updated with WTP configuration changes, eliminating the need for inter-AC communication or the need for all ACs to be aware of the configuration of all WTPs in the network.

Once the CAPWAP protocol enters the Run state, the WTPs begin to provide service. It is quite common for administrators to require that configuration changes be made while the network is operational.

Therefore, the Configuration Update Request is sent by the AC to the WTP to make these changes at run-time.

7.1.1. Configuration Flexibility

The CAPWAP protocol provides the flexibility to configure and manage WTPs of varying design and functional characteristics. When a WTP first discovers an AC, it provides primary functional information relating to its type of MAC and to the nature of frames to be exchanged. The AC configures the WTP appropriately. The AC also establishes corresponding internal operations to deal with the WTP according to its functionalities.

7.2. Configure Request

The Configure Request message is sent by a WTP to deliver its current configuration to its AC.

Configure Request messages are sent by a WTP while in the Configure state.

The Configure Request message carries binding specific message elements. Refer to the appropriate binding for the definition of this structure.

When an AC receives a Configure Request message it will act upon the content of the packet and respond to the WTP with a Configure Response message.

The Configure Request message includes multiple Administrative State message Elements. There is one such message element for the WTP, and one message element per radio in the WTP.

The following subsections define the message elements that MUST be included in the Configure Request message.

7.2.1. Administrative State

The administrative event message element is used to communicate the state of a particular radio. The value contains the following fields.

```
  0                   1
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |    Radio ID   |  Admin State  |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 27 for Administrative State

Length: 2

Radio ID: An 8-bit value representing the radio to configure. The Radio ID field may also include the value of 0xff, which is used to identify the WTP itself. Therefore, if an AC wishes to change the administrative state of a WTP, it would include 0xff in the Radio ID field.

Admin State: An 8-bit value representing the administrative state of the radio. The following values are supported:

1 - Enabled

2 - Disabled

7.2.2. AC Name

The AC Name message element is defined in Section Section 5.2.3.

7.2.3. AC Name with Index

The AC Name with Index message element is sent by the AC to the WTP to configure preferred ACs. The number of instances where this message element would be present is equal to the number of ACs configured on the WTP.

```
 0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Index     |   AC Name...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 90 for AC Name with Index

Length: > 2

Index: The index of the preferred server (e.g., 1=primary, 2=secondary).

AC Name: A variable length ASCII string containing the AC’s name.

7.2.4. WTP Board Data

The WTP Board Data message element is sent by the WTP to the AC and contains information about the hardware present.

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |            Card ID            |         Card Revision         |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                           WTP Model                           |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                           WTP Model                           |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                       WTP Serial Number                       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                       WTP Serial Number                       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                       WTP Serial Number                       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                       WTP Serial Number                       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                       WTP Serial Number                       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                       WTP Serial Number                       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                      Ethernet MAC Address                     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |      Ethernet MAC Address     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 50 for WTP Board Data

Length: 26

Card ID: A 2 byte hardware identifier.

Card Revision: A 2 byte Revision of the card.

WTP Model: 8 byte WTP Model Number.

WTP Serial Number: 24 byte WTP Serial Number.

Ethernet MAC Address: MAC Address of the WTP’s Ethernet interface.

7.2.5. Statistics Timer

The statistics timer message element value is used by the AC to inform the WTP of the frequency which it expects to receive updated statistics.

```
    0                   1
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |        Statistics Timer       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 37 for Statistics Timer

Length: 2

Statistics Timer: A 16-bit unsigned integer indicating the time, in seconds

7.2.6. WTP Static IP Address Information

The WTP Static IP Address Information message element is used by an AC to configure or clear a previously configured static IP address on a WTP.

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                          IP Address                           |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                            Netmask                            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                            Gateway                            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |    Static     |
   +-+-+-+-+-+-+-+-+
```

Type: 82 for WTP Static IP Address Information

Length: 13

IP Address: The IP Address to assign to the WTP. This field is only valid if the static field is set to one.

Netmask: The IP Netmask. This field is only valid if the static field is set to one.

Gateway: The IP address of the gateway. This field is only valid if the static field is set to one.

Netmask: The IP Netmask. This field is only valid if the static field is set to one.

Static: An 8-bit boolean stating whether the WTP should use a static IP address or not. A value of zero disables the static IP address, while a value of one enables it.

7.2.7. WTP Reboot Statistics

The WTP Reboot Statistics message element is sent by the WTP to the AC to communicate reasons why reboots have occurred.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Crash Count          |    CAPWAP Initiated Count     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Link Failure Count       | Failure Type  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 67 for WTP Reboot Statistics

Length: 7

Crash Count: The number of reboots that have occurred due to a WTP crash. A value of 65535 implies that this information is not available on the WTP.

CAPWAP Initiated Count: The number of reboots that have occurred at the request of a CAPWAP protocol message, such as a change in configuration that required a reboot or an explicit CAPWAP reset request. A value of 65535 implies that this information is not available on the WTP.

Link Failure Count: The number of times that a CAPWAP protocol connection with an AC has failed.

Failure Type: The last WTP failure. The following values are supported:

0 - Link Failure

1 - CAPWAP Initiated (see Section 8.3)

2 - WTP Crash

255 - Unknown (e.g., WTP doesn’t keep track of info)

7.3. Configure Response

The Configure Response message is sent by an AC and provides a mechanism for the AC to override a WTP's requested configuration.

Configure Response messages are sent by an AC after receiving a Configure Request message.

The Configure Response message carries binding specific message elements. Refer to the appropriate binding for the definition of this structure.

When a WTP receives a Configure Response message it acts upon the content of the message, as appropriate. If the Configure Response message includes a Change State Event message element that causes a change in the operational state of one of the Radio, the WTP will transmit a Change State Event to the AC, as an acknowledgement of the change in state.

The following subsections define the message elements that MUST be included in the Configure Response message.

7.3.1. Decryption Error Report Period

The Decryption Error Report Period message element value is used by the AC to inform the WTP how frequently it should send decryption error report messages.

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Radio ID    |        Report Interval        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 38 for Decryption Error Report Period

Length: 3

Radio ID: The Radio Identifier, typically refers to some interface index on the WTP

Report Interval: A 16-bit unsigned integer indicating the time, in seconds

7.3.2. Change State Event

The Change State message element is used to communicate a change in the operational state of a radio. The value contains two fields, as shown.

```
     0                   1
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |   Radio ID    |     State     |     Cause     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 26 for Change State Event

Length: 3

Radio ID: The Radio Identifier, typically refers to some interface index on the WTP.

State: An 8-bit boolean value representing the state of the radio. A value of one disables the radio, while a value of two enables it.

Cause: In the event of a radio being inoperable, the cause field would contain the reason the radio is out of service.

Cause: In the event of a radio being inoperable, the cause field would contain the reason the radio is out of service. The following values are supported:

0 - Normal

1 - Radio Failure

2 - Software Failure

7.3.3. CAPWAP Timers

The CAPWAP Timers message element is used by an AC to configure CAPWAP timers on a WTP.

```
     0                   1
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |   Discovery   | Echo Request  |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 68 for CAPWAP Timers

Length: 2

Discovery: The number of seconds between CAPWAP Discovery packets, when the WTP is in the discovery mode.

Echo Request: The number of seconds between WTP Echo Request CAPWAP messages.

7.3.4. AC IPv4 List

The AC List message element is used to configure a WTP with the latest list of ACs in a cluster.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       AC IP Address[]                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 59 for AC List

Length: 4

The AC IP Address: An array of 32-bit integers containing an AC’s IPv4 Address.

7.3.5. AC IPv6 List

The AC List message element is used to configure a WTP with the latest list of ACs in a cluster.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       AC IP Address[]                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       AC IP Address[]                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       AC IP Address[]                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       AC IP Address[]                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 141 for AC IPV6 List

Length: 16

The AC IP Address: An array of 32-bit integers containing an AC's IPv6 Address.

7.3.6. WTP Fallback

The WTP Fallback message element is sent by the AC to the WTP to enable or disable automatic CAPWAP fallback in the event that a WTP detects its preferred AC, and is not currently connected to it.

```
 0
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|     Mode      |
+-+-+-+-+-+-+-+-+
```

Type: 91 for WTP Fallback

Length: 1

Mode: The 8-bit value indicates the status of automatic CAPWAP fallback on the WTP. A value of zero disables fallback, while a value of one enables it. When enabled, if the WTP detects that its primary AC is available, and it is not connected to it, it SHOULD automatically disconnect from its current AC and reconnect to its primary. If disabled, the WTP will only reconnect to its primary through manual intervention (e.g., through the Reset Request command).

7.3.7. Idle Timeout

The Idle Timeout message element is sent by the AC to the WTP to provide it with the idle timeout that it should enforce on its active mobile station entries.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            Timeout                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 97 for Idle Timeout

Length: 4

Timeout: The current idle timeout to be enforced by the WTP.

7.4. Configuration Update Request

Configure Update Request messages are sent by the AC to provision the WTP while in the Run state. This is used to modify the configuration of the WTP while it is operational.

When an AC receives a Configuration Update Request message it will respond with a Configuration Update Response message, with the appropriate Result Code.

The following subsections define the message elements included in the Configuration Update message.

7.4.1. WTP Name

The WTP Name message element is a variable length bye string. The string is not zero terminated.

```
 0
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+-
| WTP Name ...
+-+-+-+-+-+-+-+-+-
```

Type: 5 for WTP Name

Length: 0

Timeout: A non-zero terminated string containing the WTP name.

7.4.2. Change State Event

The Change State Event message element is defined in Section Section 7.3.2.

7.4.3. Administrative State

The Administrative State message element is defined in Section Section 7.2.1.

7.4.4. Statistics Timer

The Statistics Timer message element is defined in Section Section 7.2.5.

7.4.5. Location Data

The Location Data message elementis a variable length byte string containing user defined location information (e.g. "Next to Fridge"). This information is configurable by the network administrator, and allows for the WTP location to be determined through this field. The string is not zero terminated.

```
     0
     0 1 2 3 4 5 6 7
    +-+-+-+-+-+-+-+-+-
    | Location ...
    +-+-+-+-+-+-+-+-+-
```

Type: 35 for Location Data

Length: 0

Timeout: A non-zero terminated string containing the WTP location.

7.4.6. Decryption Error Report Period

The Decryption Error Report Period message element is defined in Section 7.3.1.

7.4.7. AC IPv4 List

The AC List message element is defined in Section 7.3.4.

7.4.8. AC IPv6 List

The AC List message element is defined in Section 7.3.5.

7.4.9. Add MAC ACL Entry

The Add MAC Access Control List (ACL) Entry message element is used by an AC to add a MAC ACL list entry on a WTP, ensuring that the WTP no longer provides any service to the MAC addresses provided in the message. The MAC Addresses provided in this message element are not expected to be saved in non-volatile memory on the WTP.

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    | Num of Entries|                 MAC Address[]                 |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                 MAC Address[]                 |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 65 for Add MAC ACL Entry

Length: >= 7

Num of Entries: The number of MAC Addresses in the array.

MAC Address: An array of MAC Addresses to add to the ACL.

7.4.10. Delete MAC ACL Entry

The Delete MAC ACL Entry message element is used by an AC to delete a MAC ACL entry on a WTP, ensuring that the WTP provides service to the MAC addresses provided in the message.

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    | Num of Entries|                 MAC Address[]                 |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                 MAC Address[]                 |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 66 for Delete MAC ACL Entry

Length: >= 7

Num of Entries: The number of MAC Addresses in the array.

MAC Address: An array of MAC Addresses to delete from the ACL.

7.4.11. Add Static MAC ACL Entry

The Add Static MAC ACL Entry message element is used by an AC to add a permanent ACL entry on a WTP, ensuring that the WTP no longer provides any service to the MAC addresses provided in the message. The MAC Addresses provided in this message element are expected to be saved in non-volatile memory on the WTP.

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    | Num of Entries|                 MAC Address[]                 |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                 MAC Address[]                 |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 70 for Add Static MAC ACL Entry

Length: >= 7

Num of Entries: The number of MAC Addresses in the array.

MAC Address: An array of MAC Addresses to add to the permanent ACL.

7.4.12. Delete Static MAC ACL Entry

The Delete Static MAC ACL Entry message element is used by an AC to delete a previously added static MAC ACL entry on a WTP, ensuring that the WTP provides service to the MAC addresses provided in the message.

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    | Num of Entries|                 MAC Address[]                 |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                 MAC Address[]                 |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 71 for Delete MAC ACL Entry

Length: >= 7

Num of Entries: The number of MAC Addresses in the array.

MAC Address: An array of MAC Addresses to delete from the static MAC ACL entry.

7.4.13. CAPWAP Timers

The CAPWAP Timers message element is defined in Section 7.3.3.

7.4.14. AC Name with Index

The AC Name with Index message element is defined in Section 7.2.3.

7.4.15. WTP Fallback

The WTP Fallback message element is defined in Section 7.3.6.

7.4.16. Idle Timeout

The Idle Timeout message element is defined in Section 7.3.7.

7.4.17. Timestamp

The Timestamp message element is sent by the AC to to synchronize the WTP’s clock.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Timestamp                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: TBD for Timestamp

Length: 4

Timestamp: The AC’s current time, allowing all of the WTPs to be time synchronized in the format defined by Network Time Protocol (NTP) in RFC 1305 [10].

7.5. Configuration Update Response

The Configuration Update Response message is the acknowledgement message for the Configuration Update Request message.

The Configuration Update Response message is sent by a WTP after receiving a Configuration Update Request message.

When an AC receives a Configure Update Response message the result code indicates if the WTP successfully accepted the configuration.

The following subsections define the message elements that must be present in the Configuration Update message.

7.5.1. Result Code

The Result Code message element value is a 32-bit integer value, indicating the result of the request operation corresponding to the sequence number in the message.

```
      0                   1                   2                   3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                          Result Code                          |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 2 for Result Code

Length: 4

Result Code: The following values are defined:

0 Success

1 Failure (AC List message element MUST be present)

7.6. Change State Event Request

The Change State Event Request message is used by the WTP to inform the AC of a change in the operational state.

The Change State Event Request message is sent by the WTP when it receives a Configuration Response message that includes a Change State Event message element. It is also sent when the WTP detects an operational failure with a radio. The Change State Event Request message may be sent in either the Configure or Run state (see Section 2.2.

When an AC receives a Change State Event message it will respond with a Change State Event Response message and make any necessary modifications to internal WTP data structures.

The following subsections define the message elements that must be present in the Change State Event Request message.

7.6.1. Change State Event

The Change State Event message element is defined in Section 7.3.2.

7.7. Change State Event Response

The Change State Event Response message acknowledges the Change State Event Request message.

A Change State Event Response message is by a WTP after receiving a Change State Event Request message.

The Change State Event Response message carries no message elements.

Its purpose is to acknowledge the receipt of the Change State Event Request message.

The WTP does not need to perform any special processing of the Change State Event Response message.

7.8. Clear Config Indication

The Clear Config Indication message is used to reset a WTP's configuration.

The Clear Config Indication message is sent by an AC to request that a WTP reset its configuration to the manufacturing default configuration. The Clear Config Indication message is sent while in the Run CAPWAP state.

The Clear Config Indication message carries no message elements.

When a WTP receives a Clear Config Indication message it resets its configuration to the manufacturing default configuration.

8. Device Management Operations

This section defines CAPWAP operations responsible for debugging, gathering statistics, logging, and firmware management.

8.1. Image Data Request

The Image Data Request message is used to update firmware on the WTP. This message and its companion response message are used by the AC to ensure that the image being run on each WTP is appropriate.

Image Data Request messages are exchanged between the WTP and the AC to download a new program image to the WTP.

When a WTP or AC receives an Image Data Request message it will respond with an Image Data Response message.

The format of the Image Data and Image Download message elements are described in the following subsections.

8.1.1. Image Download

The image download message element is sent by the WTP to the AC and contains the image filename. The value is a variable length byte string. The string is NOT zero terminated.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Filename ...                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 32 for Image Download

Length: >= 1

Filename: A variable length string containing the filename to download.

8.1.2. Image Data

The image data message element is present in the Image Data Request message sent by the AC and contains the following fields.

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |     Opcode    |           Checksum            |  Image Data   |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                          Image Data ...                       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 33 for Image Data

Length: >= 4 (allows 0 length element if last data unit is 1024 bytes)

Opcode: An 8-bit value representing the transfer opcode. The following values are supported:

3 - Image data is included

5 - An error occurred. Transfer is aborted

Checksum: A 16-bit value containing a checksum of the image data that follows

Image Data: The Image Data field contains 1024 characters, unless the payload being sent is the last one (end of file). If the last block was 1024 in length, an Image Data with a zero length payload is sent.

8.2. Image Data Response

The Image Data Response message acknowledges the Image Data Request message.

An Image Data Response message is sent in response to a received Image Data Request message. Its purpose is to acknowledge the receipt of the Image Data Request message.

The Image Data Response message carries no message elements.

No action is necessary on receipt.

8.3. Reset Request

The Reset Request message is used to cause a WTP to reboot.

A Reset Request message is sent by an AC to cause a WTP to reinitialize its operation.

The Reset Request carries no message elements.

When a WTP receives a Reset Request it will respond with a Reset Response and then reinitialize itself.

8.4. Reset Response

The Reset Response message acknowledges the Reset Request message.

A Reset Response message is sent by the WTP after receiving a Reset Request message.

The Reset Response message carries no message elements. Its purpose is to acknowledge the receipt of the Reset Request message.

When an AC receives a Reset Response message, it is notified that the WTP will reinitialize its operation.

8.5. WTP Event Request

WTP Event Request message is used by a WTP to send information to its AC. The WTP Event Request message may be sent periodically, or sent in response to an asynchronous event on the WTP. For example, a WTP MAY collect statistics and use the WTP Event Request message to transmit the statistics to the AC.

When an AC receives a WTP Event Request message it will respond with a WTP Event Response message.

The WTP Event Request message MUST contain one of the message elements described below, or a message element that is defined for a specific wireless technology.

8.5.1. Decryption Error Report

The Decryption Error Report message element value is used by the WTP to inform the AC of decryption errors that have occurred since the last report. Note that this error reporting mechanism is not used if encryption and decryption services are provided via the AC.

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Radio ID    |Num Of Entries |      Mobile MAC Address       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Mobile MAC Address[]                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 39 for Decryption Error Report

Length: >= 8

Radio ID: The Radio Identifier, which typically refers to an interface index on the WTP

Num Of Entries: An 8-bit unsigned integer indicating the number of mobile MAC addresses.

Mobile MAC Address: An array of mobile station MAC addresses that have caused decryption errors.

8.5.2. Duplicate IPv4 Address

The Duplicate IPv4 Address message element is used by a WTP to inform an AC that it has detected another IP device using the same IP address it is currently using.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          IP Address                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          MAC Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          MAC Address          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 77 for Duplicate IPv4 Address

Length: 10

IP Address: The IP Address currently used by the WTP.

MAC Address: The MAC Address of the offending device.

8.5.3. Duplicate IPv6 Address

The Duplicate IPv6 Address message element is used by a WTP to inform an AC that it has detected another host using the same IP address it is currently using.

```
      0                   1                   2                   3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                          IP Address                           |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                          IP Address                           |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                          IP Address                           |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                          IP Address                           |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                          MAC Address                          |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |          MAC Address          |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 77 for Duplicate IPv6 Address

Length: 22

IP Address: The IP Address currently used by the WTP.

MAC Address: The MAC Address of the offending device.

8.6. WTP Event Response

The WTP Event Response message acknowledges receipt of the WTP Event Request message.

A WTP Event Response message issent by an AC after receiving a WTP Event Request message.

The WTP Event Response message carries no message elements.

8.7. Data Transfer Request

The Data Transfer Request message is used to deliver debug information from the WTP to the AC.

Data Transfer Request messages are sent by the WTP to the AC when the WTP determines that it has important information to send to the AC. For instance, if the WTP detects that its previous reboot was caused by a system crash, it can send the crash file to the AC. The remote debugger function in the WTP also uses the Data Transfer Request message to send console output to the AC for debugging purposes.

When the AC receives a Data Transfer Request message it responds to the WTP ith a Data Transfer Response message. The AC MAY log the information received.

The Data Transfer Request message MUST contain one of the following message element listed below.

8.7.1. Data Transfer Mode

The Data Transfer Mode message element is used by the AC to request information from the WTP for debugging purposes.

```
 0
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|   Data  Type  |
+-+-+-+-+-+-+-+-+
```

Type: 52 for Data Transfer Mode

Length: 1

Data Type: An 8-bit value the type of information being requested. The following values are supported:

1 - WTP Crash Data

2 - WTP Memory Dump

8.7.2. Data Transfer Data

The Data Transfer Data message element is used by the WTP to provide information to the AC for debugging purposes.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Data Type   |  Data Length  |    Data ....
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 53 for Data Transfer Data

Length: >= 3

Data Type: An 8-bit value the type of information being sent. The following values are supported:

1 - WTP Crash Data

2 - WTP Memory Dump

Data Length: Length of data field.

Data: Debug information.

8.8. Data Transfer Response

The Data Transfer Response message acknowledges the Data Transfer Request message.

A Data Transfer Response message is sent in response to a received Data Transfer Request message. Its purpose is to acknowledge receipt of the Data Transfer Request message.

The Data Transfer Response message carries no message elements.

Upon receipt of a Data Transfer Response message, the WTP transmits more information, if more information is available.

9. Mobile Session Management

Messages in this section are used by the AC to create, modify or delete mobile station session state on the WTPs.

9.1. Mobile Config Request

The Mobile Config Request message is used to create, modify or delete mobile session state on a WTP. The message is sent by the AC to the WTP, and may contain one or more message elements. The message elements for this CAPWAP control message include information that is generally highly technology specific. Therefore, please refer to the appropriate binding section or document for the definitions of the messages elements that may be used in this control message.

9.1.1. Add Mobile

The Add Mobile message element is used by the AC to inform a WTP that it should forward traffic for a particular mobile station. The Add Mobile message element will be accompanied by technology specific binding information element which may include security parameters. Consequently, the security parameters must be applied by the WTP for the particular mobile.

Once a mobile station’s policy has been pushed to the WTP through this message element, an AC may change any policies by simply sending a modified Add Mobile message element. When a WTP receives an Add Mobile message element for an existing mobile station, it must override any existing state it may have for the mobile station in question. The latest Add Mobile overrides any previously received messages.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Radio ID   |                  MAC Address                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  MAC Address                 |  VLAN Name...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 29 for Add Mobile

Length: >= 7

Radio ID: An 8-bit value representing the radio

MAC Address: The mobile station's MAC Address

VLAN Name: An optional variable string containing the VLAN Name on which the WTP is to locally bridge user data. Note this field is only valid with WTPs configured in Local MAC mode.

9.1.2. Delete Mobile

The Delete Mobile message element is used by the AC to inform an WTP that it should no longer provide service to a particular mobile station. The WTP must terminate service immediately upon receiving this message element.

The transmission of a Delete Mobile message element could occur for various reasons, including for administrative reasons, as a result of the fact that the mobile has roamed to another WTP, etc.

Once access has been terminated for a given station, any future packets received from the mobile must result in a deauthenticate message, as specified in [6].

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Radio ID   |                  MAC Address                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  MAC Address                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 30 for Delete Mobile

Length: 7

Radio ID: An 8-bit value representing the radio

MAC Address: The mobile station's MAC Address

9.2. Mobile Config Response

The Mobile Configuration Response message is used to acknowledge a previously received Mobile Configuration Request message, and includes a Result Code message element which indicates whether an error occurred on the WTP.

This message requires no special processing, and is only used to acknowledge the Mobile Configuration Request message.

9.2.1. Result Code

The Result Code message element is defined in Section 7.5.1.

10. CAPWAP Security

This version of the CAPWAP protocol uses DTLS with both certificate and shared secret based credentials to secure CAPWAP protocol Control, and (optionally) Data packets. CAPWAP protocol Discovery Request and Discover Response messages are sent in the clear, as they are sent prior to esablishment of a secure DTLS session between the WTP and the AC. Once the DTLS session is established, and the CAPWAP state machine (see Section 2.2) is in the Configure state, all CAPWAP control frames are encrypted.

An in-depth security analysis of threats and risks to AC-AP communication is beyond the scope of this document. The list below provides a summary of the assumptions made in the CAPWAP protocol security design:

- WTP-AC communications may be accessible to a sophisticated attacker.

- When authentication and/or privacy of end to end traffic for which the WTP and AC are intermediaries is required, IPSEC [19] or another end to end security protocol must be used.

- Privacy and authentication for at least some WTP-AC control traffic is required, for example to enable secure delivery of user sessions keys from the AC to the WTP.

10.1. Endpoint Authentication using DTLS

Certificate-based authentication is natively supported in DTLS, and support for preshared keys has been standardized (see [12]). The TLS algorithm suites for each endpoint authentication method are described below.

10.1.1. Authenticating with Certificates

Note that only block ciphers are currently recommended for use with DTLS. To understand the reasoning behind this, see [23]. However,support for AES counter mode encryption is currently progressing in the TLS working group, and once protocol identifiers are available, they will be added below. At present, the following algorithms MUST be supported when using certificates for CAPWAP authentication:

- TLS_RSA_WITH_AES_128_CBC_SHA

- TLS_RSA_WITH_3DES_EDE_CBC_SHA

The following algorithms SHOULD be supported when using certificates:

- o TLS_DH_RSA_WITH_AES_128_CBC_SHA
- o TLS_DH_RSA_WITH_3DES_EDE_CBC_SHA

The following algorithms MAY be supported when using certificates:

- o TLS_RSA_WITH_AES_256_CBC_SHA
- o TLS_DH_RSA_WITH_AES_256_CBC_SHA

10.1.2. Authenticating with Preshared Keys

Pre-shared keys present significant challenges from a security perspective, and for that reason, their use is strongly discouraged. However, [12] defines 3 different methods for authenticating with preshared keys:

- o PSK key exchange algorithm - simplest method, ciphersuites use only symmetric key algorithms
- o DHE_PSK key exchange algorithm - use a PSK to authenticate a Diffie-Hellman exchange. These ciphersuites give some additional protection against dictionary attacks and also provide Perfect Forward Secrecy (PFS).
- o RSA_PSK key exchange algorithm - use RSA and certificates to authenticate the server, in addition to using a PSK. Not susceptible to passive attacks.

The first approach (plain PSK) is susceptible to passive dictionary attacks; hence, while this alorithm MAY be supported, special care should be taken when choosing that method. In particular, user-readable passphrases SHOULD NOT be used, and use of short PSKs should be strongly discouraged. Additionally, DHE_PSK MUST be supported, and RSA_PSK MAY be supported.

The following cryptographic algorithms MUST be supported when using preshared keys:

- o TLS_DHE_PSK_WITH_AES_128_CBC_SHA
- o TLS_DHE_PSK_WITH_3DES_EDE_CBC_SHA

The following algorithms SHOULD be supported when using preshared keys:

- o TLS_DHE_PSK_WITH_AES_256_CBC_SHA

The following algorithms MAY be supported when using preshared keys:

- o TLS_PSK_WITH_AES_128_CBC_SHA
- o TLS_PSK_WITH_AES_256_CBC_SHA
- o TLS_PSK_WITH_3DES_EDE_CBC_SHA
- o TLS_RSA_PSK_WITH_AES_128_CBC_SHA
- o TLS_RSA_PSK_WITH_AES_256_CBC_SHA
- o TLS_RSA_PSK_WITH_3DES_EDE_CBC_SHA

10.2. Refreshing Cryptographic Keys

Since AC-WTP associations will tend to be relatively long-lived, a mechanism is provided to periodically refresh the encryption and authentication keys; this is referred to as "rekeying". When the key lifetime reaches 95% of the configured value, identified in the KeyLifetime timer (see Section 12), a new DTLS seesion SHOULD be initiated (via a CAPWAP implementation API).

10.3. Certificate Usage

Validation of the certificates by the AC and WTP is required so that only an AC may perform the functions of an AC and that only a WTP may perform the functions of a WTP. This restriction of functions to the AC or WTP requires that the certificates used by the AC MUST be distinguishable from the certificate used by the WTP. To accomplish this differentiation, the x.509v3 certificates MUST include the Extensions field [11] and MUST include the NetscapeComment [13] extension.

For an AC, the value of the NetscapeComment extension MUST be the string "CAPWAP AC Device Certificate". For a WTP, the value of the NetscapeComment extension MUST be the string "CAPWAP WTP Device Certificate".

Part of the CAPWAP certificate validation process includes ensuring that the proper string is included in the NetscapeComment extension, and only allowing the CAPWAP session to be established if the extension does not represent the same role as the device validating the certificate. For instance, a WTP MUST NOT accept a certificate whose NetscapeComment field is set to "CAPWAP WTP Device Certificate".

11. IEEE 802.11 Binding

This section defines the extensions required for the CAPWAP protocol to be used with the IEEE 802.11 protocol.

11.1. Division of labor

The CAPWAP protocol, when used with IEEE 802.11 devices, requires a specific behavior from the WTP and the AC, specifically in terms of which IEEE 802.11 protocol functions are handled.

For both the Split and Local MAC approaches, the CAPWAP functions, as defined in the taxonomy specification, reside in the AC.

11.1.1. Split MAC

This section shows the division of labor between the WTP and the AC in a Split MAC architecture. Figure 4 shows the clear separation of functionality among CAPWAP components.

```
 Function                                Location
     Distribution Service                       AC
     Integration Service                        AC
     Beacon Generation                          WTP
     Probe Response                             WTP
     Power Mgmt/Packet Buffering                WTP
     Fragmentation/Defragmentation              WTP
     Assoc/Disassoc/Reassoc                     AC

802.11e
     Classifying                                AC
     Scheduling                                 WTP/AC
     Queuing                                    WTP

802.11i
     802.1X/EAP                                 AC
     Key Management                             AC
     802.11 Encryption/Decryption               WTP or AC
```

Figure 4: Mapping of 802.11 Functions for Split MAC Architecture

The Distribution and Integration services reside on the AC, and therefore all user data is tunneled between the WTP and the AC. As noted above, all real-time 802.11 services, including the control protocol and the beacon and probe response frames, are handled on the WTP.

All remaining IEEE 802.11 MAC management frames are supported on the

AC, including the Association Request which allows the AC to be involved in the access policy enforcement portion of the IEEE 802.11 protocol. The IEEE 802.1X and IEEE 802.11i key management function are also located on the AC.

While the admission control component of IEEE 802.11e resides on the AC, the real time scheduling and queuing functions are on the WTP. Note this does not exclude the AC from providing additional policing and scheduling functionality.

Note that in the following figure, the use of '(-)' indicates that processing of the frames is done on the WTP.

```
          Client                        WTP                          AC

                   Beacon
          <------------------------------
                Probe Request
          ---------------------------( - )------------------------->
                Probe Response
          <------------------------------
                           802.11 AUTH/Association
          <-------------------------------------------------------->
                             Add Mobile (Clear Text, 802.1X Only)
                                         <------------------------->
                 802.1X Authentication & 802.11i Key Exchange
          <-------------------------------------------------------->
                                     Add Mobile (AES-CCMP, PTK=x)
                                         <------------------------->
                            802.11 Action Frames
          <-------------------------------------------------------->
                                802.11 DATA (1)
          <--------------------------( - )------------------------->
```

Figure 5: Split MAC Message Flow

Figure 5 provides an illustration of the division of labor in a Split MAC architecture. In this example, a WLAN has been created that is configured for IEEE 802.11i, using AES-CCMP for privacy. The following process occurs:

- o The WTP generates the IEEE 802.11 beacon frames, using information provided to it through the Add WLAN (see Section Section 11.8.1.1) message element.

- o The WTP processes the probe request and responds with a corresponding probe response. The probe request is then forwarded to the AC for optional processing.

- o The WTP forwards the IEEEE 802.11 Authentication and Association frames to the AC, which is responsible for responding to the client.

- o Once the association is complete, the AC transmits an CAPWAP Add Mobile request to the WTP (see Section Section 9.1.1. In the above example, the WLAN is configured for IEEE 802.1X, and therefore the '802.1X only' policy bit is enabled.

- o If the WTP is providing encryption/decryption services, once the client has completed the IEEE 802.11i key exchange, the AC transmits another Add Mobile request to the WTP, stating the security policy to enforce for the client (in this case AES-CCMP), as well as the encryption key to use. If encryption/decryption is handled in the AC, the Add Mobile request would have the encryption policy set to "Clear Text".

- o The WTP forwards any 802.11 Action frames received to the AC.

- o All client data frames are tunneled between the WTP and the AC. Note that the WTP is responsible for encrypting and decrypting frames, if it was indicated in the Add Mobile request.

11.1.2. Local MAC

This section shows the division of labor between the WTP and the AC in a Local MAC architecture. Figure 6 shows the clear separation of functionality among CAPWAP components.

Function	Location
Distribution Service	WTP
Integration Service	WTP
Beacon Generation	WTP
Probe Response	WTP
Power Mgmt/Packet Buffering	WTP
Fragmentation/Defragmentation	WTP
Assoc/Disassoc/Reassoc	WTP
802.11e	
Classifying	WTP
Scheduling	WTP
Queuing	WTP
802.11i	
802.1X/EAP	AC
Key Management	AC
802.11 Encryption/Decryption	WTP

Figure 6: Mapping of 802.11 Functions for Local AP Architecture

Given the Distribution and Integration Services exist on the WTP, client data frames are not forwarded to the AC, with the exception listed in the following paragraphs.

While the MAC is terminated on the WTP, it is necessary for the AC to be aware of mobility events within the WTPs. As a consequence, the WTP MUST forward the IEEE 802.11 Association Requests to the AC, and the AC MAY reply with a failed Association Response if it deems it necessary.

The IEEE 802.1X and IEEE 802.11i Key Management function resides in the AC. Therefore, the WTP MUST forward all IEEE 802.1X/Key Management frames to the AC and forward the associated responses to the station.

Note that in the following figure, the use of '(-)' indicates that processing of the frames is done on the WTP.

```
             Client                           WTP                       AC

                      Beacon
             <------------------------------
                       Probe
             <----------------------------->
                    802.11 AUTH
             <------------------------------
                                 802.11 Association
             <---------------------------( - )------------------------->
                                Add Mobile (Clear Text, 802.1X Only)
                                             <------------------------->
                    802.1X Authentication & 802.11i Key Exchange
             <--------------------------------------------------------->
                               802.11 Action Frames
             <--------------------------------------------------------->
                                         Add Mobile (AES-CCMP, PTK=x)
                                             <------------------------->
                     802.11 DATA
             <------------------------------>
```

Figure 7: Local MAC Message Flow

Figure 7 provides an illustration of the division of labor in a Local MAC architecture. In this example, a WLAN has been created that is configured for IEEE 802.11i, using AES-CCMP for privacy. The following process occurs:

- o The WTP generates the IEEE 802.11 beacon frames, using information provided to it through the Add WLAN (see Section 11.8.1.1) message element.

- o The WTP processes the probe request and responds with a corresponding probe response.

- o The WTP forwards the IEEE 802.11 Authentication and Association frames to the AC, which is responsible for responding to the client.

- o Once the association is complete, the AC transmits an CAPWAP Add Mobile request to the WTP (see Section Section 9.1.1. In the above example, the WLAN is configured for IEEE 802.1X, and therefore the '802.1X only' policy bit is enabled.

- o The WTP forwards all IEEE 802.1X and IEEE 802.11i key exchange messages to the AC for processing.

o The AC transmits another Add Mobile request to the WTP, stating the security policy to enforce for the client (in this case AES-CCMP), as well as the encryption key to use. The Add Mobile request MAY include a VLAN name, which when present is used by the WTP to identify the VLAN on which the user's data frames are to be bridged.

o The WTP forwards any IEEE 802.11 Action frames received to the AC.

o The WTP optionally may tunnel client data frames to the AC. If client data frames are locally bridged, the WTP will need to provide the necessary encryption and decryption services.

11.2. Roaming Behavior and 802.11 security

It is important that CAPWAP implementations react properly to mobile devices associating to the networks in how they generate Add Mobile and Delete Mobile messages. This section expands upon the examples provided in the previous section, and describes how the CAPWAP control protocol is used in order to provide secure roaming.

Once a client has successfully associated with the network in a secure fashion, it is likely to attempt to roam to another access point. Figure 8 shows an example of a currently associated station moving from its "Old WTP" to a new WTP. The figure is useful for multiple different security policies, including standard IEEE 802.1X and dynamic WEP keys, WPA or even WPA2 both with key caching (where the IEEE 802.1x exchange would be bypassed) and without.

```
          Client              Old WTP              WTP             AC

                          Association Request/Response
           <--------------------------------------( - )-------------->
                                Add Mobile (Clear Text, 802.1X Only)
                                                         <---------------->
           802.1X Authentication (if no key cache entry exists)
           <--------------------------------------( - )-------------->
                           802.11i 4-way Key Exchange
           <--------------------------------------( - )-------------->
                                         Delete Mobile
                                   <---------------------------------->
                                         Add Mobile (AES-CCMP, PTK=x)
                                                         <---------------->
```

Figure 8: Client Roaming Example

11.3. Transport specific bindings

All CAPWAP transports have the following IEEE 802.11 specific bindings:

11.3.1. Payload encapsulation

The CAPWAP protocol defines the data frame, which allows a wireless payload to be encapsulated. For IEEE 802.11, the IEEE 802.11 header and payload is encapsulated (excluding the IEEE 802.11 FCS checksum). The IEEE 802.11 FCS checksum is handled by the WTP. This allows the WTP to validate a frame prior to sending it to the AC. Similarly, when an AC wishes to transmit a frame towards a station, the WTP computes and adds the FCS checksum.

11.3.2. Status and WLANS field

The interpretation of this 16 bit field depends on the direction of transmission of the packet. Refer to the figure in Section 4.1.

Status

When a CAPWAP packet is transmitted from a WTP to an AC, this field is called the status field and indicates radio resource information associated with the frame. When the message is a CAPWAP control message this field is transmitted as zero.

The status field is divided into the signal strength and signal to noise ratio with which an IEEE 802.11 frame was received, encoded in the following manner:

```
 0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     RSSI      |     SNR       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

RSSI: RSSI is a signed, 8-bit value. It is the received signal strength indication, in dBm.

SNR: SNR is a signed, 8-bit value. It is the signal to noise ratio of the received IEEE 802.11 frame, in dB.

WLANs field: When a CAPWAP data message is transmitted from an AC to a WTP, this 16 bit field indicates on which WLANs the encapsulated IEEE 802.11 frame is to be transmitted. For unicast packets, this field is not used by the WTP. For broadcast or multicast packets, the WTP might require this information if it provides encryption services.

Given that a single broadcast or multicast packet might need to be sent to multiple wireless LANs (presumably each with a different broadcast key), this field is defined as a bit field. A bit set indicates a WLAN ID (see Section Section 11.8.1.1) which will be sent the data. The WLANS field is encoded in the following manner:

```
 0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          WLAN ID(s)           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

11.4. BSSID to WLAN ID Mapping

The CAPWAP protocol makes assumptions regarding the BSSIDs used on the WTP. It is a requirement for the WTP to use a contiguous block of BSSIDs. The WLAN Identifier field, which is managed by the AC, is used as an offset into the BSSID list.

For instance, if a WTP had a base BSSID address of 00:01:02:00:00:00, and the AC sent an Add WLAN message with a WLAN Identifier of 2 (see Section Section 11.8.1.1), the BSSID for the specific WLAN on the WTP would be 00:01:02:00:00:02.

The WTP communicates the maximum number of BSSIDs that it supports during the Config Request within the IEEE 802.11 WTP WLAN Radio Configuration message element (see Section 11.9.1).

11.5. Quality of Service for Control Messages

It is recommended that IEEE 802.11 MAC management frames be sent by both the AC and the WTP with appropriate Quality of Service values, ensuring that congestion in the network minimizes occurrences of packet loss. Therefore, a Quality of Service enabled CAPWAP device should use:

802.1P: The precedence value of 6 SHOULD be used for all IEEE 802.11 MAC management frames, except for Probe Requests which SHOULD use 4.

DSCP: The DSCP tag value of 46 SHOULD be used for all IEEE 802.11 MAC management frames, except for Probe Requests which SHOULD use 34.

11.6. Data Message bindings

There are no CAPWAP Data Message bindings for IEEE 802.11.

11.7. Control Message bindings

The IEEE 802.11 binding has the following Control Message definitions.

11.7.1. Mobile Config Request

This section contains the IEEE 802.11 specific message elements that are used with the Mobile Config Request.

11.7.1.1. IEEE 802.11 Mobile

The IEEE 802.11 Mobile message element accompanies the Add Mobile message element, and is used to push the IEEE 802.11 station policy.

The latest IEEE 802.11 Mobile message element overrides any previously received message elements. If the IEEE 802.11 Mobile message element’s EAP Only bit is set, the WTP MUST drop all IEEE 802.11 packets that do not contain EAP packets. Note that when EAP Only is set, the Encryption Policy field MAY be set, and therefore it is possible to inform a WTP to only accept encrypted EAP packets. Once the mobile station has successfully completed EAP authentication, the AC must send a new Add Mobile message element to remove the EAP Only restriction, and optionally push the session key down to the WTP.

If the QoS field is set, the WTP MUST observe and provide policing of the 802.11e priority tag to ensure that it does not exceed the value provided by the AC.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Radio ID   |       Association ID          |     Flags     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Capabilities         |    WLAN ID    |Supported Rates
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: TBD for Add IEEE 802.11 Mobile

Length: >= 8

Radio ID: An 8-bit value representing the radio

Association ID: A 16-bit value specifying the IEEE 802.11 Association Identifier

MAC Address: The mobile station's MAC Address

Capabilities: A 16-bit field containing the IEEE 802.11 capabilities to use with the mobile.

WLAN ID: An 8-bit value specifying the WLAN Identifier

Supported Rates: The variable length field containing the supported rates to be used with the mobile station.

11.7.1.2. IEEE 802.11 Mobile Session Key

The Mobile Session Key Payload message element is sent when the AC determines that encryption of a mobile station must be performed in the WTP. This message element MUST NOT be present without the IEEE 802.11 Mobile (see Section 11.7.1.1) message element, and MUST NOT be sent if the WTP had not specifically advertised support for the requested encryption scheme.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          MAC Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          MAC Address          |E|C|           Flags           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Encryption Policy                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Pairwise TSC                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Pairwise TSC          |         Pairwise RSC          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Pairwise RSC                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Session Key...
+-+-+-+-+-+-+-+-
```

Type: 105 for IEEE 802.11 Mobile Session Key

Length: >= 25

MAC Address: The mobile station's MAC Address

Flags: A 16 bit field, whose unused bits MUST be set to zero. The following bits are defined:

E: The one bit field is set by the AC to inform the WTP that is MUST NOT accept any 802.11 data frames, other than IEEE 802.1X frames. This is the equivalent of the WTP's IEEE 802.1X port for the mobile station to be in the closed state. When set, the WTP MUST drop any non-IEEE 802.1X packets it receives from the mobile station.

C: The one bit field is set by the AC to inform the WTP that encryption services will be provided by the AC. When set, the WTP SHOULD police frames received from stations to ensure that they comply to the stated encryption policy, but does not need to take specific cryptographic action on the frame. Similarly, for transmitted frames, the WTP only needs to forward already encrypted frames.

Encryption Policy: The policy field informs the WTP how to handle packets from/to the mobile station. The following values are supported:

0 - Encrypt WEP 104: All packets to/from the mobile station must be encrypted using standard 104 bit WEP.

1 - Clear Text: All packets to/from the mobile station do not require any additional crypto processing by the WTP.

2 - Encrypt WEP 40: All packets to/from the mobile station must be encrypted using standard 40 bit WEP.

3 - Encrypt WEP 128: All packets to/from the mobile station must be encrypted using standard 128 bit WEP.

4 - Encrypt AES-CCMP 128: All packets to/from the mobile station must be encrypted using 128 bit AES CCMP [7]

5 - Encrypt TKIP-MIC: All packets to/from the mobile station must be encrypted using TKIP and authenticated using Michael [21]

Pairwise TSC: The 6 byte Transmit Sequence Counter (TSC) field to use for unicast packets transmitted to the mobile.

Pairwise RSC: The 6 byte Receive Sequence Counter (RSC) to use for unicast packets received from the mobile.

Session Key: The session key the WTP is to use when encrypting traffic to/from the mobile station. For dynamically created keys, this is commonly known as a Pairwise Transient Key (PTK).

11.7.1.3. Station QoS Profile

The Station QoS Profile Payload message element contains the maximum IEEE 802.11e priority tag that may be used by the station. Any packets received that exceeds the value encoded in this message element must either be dropped or tagged using the maximum value permitted by to the user. The priority tag must be between zero (0) and seven (7).

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          MAC Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          MAC Address          |     802.1P Precedence Tag     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 140 for IEEE 802.11 Station QOS Profile

Length: 8

MAC Address: The mobile station’s MAC Address

802.1P Precedence Tag: The maximum 802.1P precedence value that the WTP will allow in the TID field in the extended 802.11e QOS Data header.

11.7.1.4. IEEE 802.11 Update Mobile QoS

The Update Mobile QoS message element is used to change the Quality of Service policy on the WTP for a given mobile station.

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Radio ID    |                 MAC Address                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          MAC Address          |   DSCP Tag    |  802.1P Tag   |
```

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 106 for IEEE 802.11 Update Mobile QoS

Length: 14

Radio ID: The Radio Identifier, typically refers to some interface index on the WTP

MAC Address: The mobile station's MAC Address.

DSCP Tag: The DSCP label to use if packets are to be DSCP tagged.

802.1P Tag: The 802.1P precedence value to use if packets are to be IEEE 802.1P tagged.

11.7.2. WTP Event Request

This section contains the 802.11 specific message elements that are used with the WTP Event Request message.

11.7.2.1. IEEE 802.11 Statistics

The statistics message element is sent by the WTP to transmit it's current statistics. The value contains the following fields.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Radio ID   |                    Reserved                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Tx Fragment Count                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Multicast Tx Count                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Failed Count                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Retry Count                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Multiple Retry Count                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Frame Duplicate Count                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       RTS Success Count                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       RTS Failure Count                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       ACK Failure Count                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Rx Fragment Count                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Multicast RX Count                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        FCS Error  Count                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Tx Frame Count                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Decryption Errors                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 38 for Statistics

Length: 60

Radio ID: An 8-bit value representing the radio.

Tx Fragment Count: A 32-bit value representing the number of fragmented frames transmitted.

Multicast Tx Count: A 32-bit value representing the number of multicast frames transmitted.

Failed Count: A 32-bit value representing the transmit excessive retries.

Retry Count: A 32-bit value representing the number of transmit retries.

Multiple Retry Count: A 32-bit value representing the number of transmits that required more than one retry.

Frame Duplicate Count: A 32-bit value representing the duplicate frames received.

RTS Success Count: A 32-bit value representing the number of successfully transmitted Ready To Send (RTS).

RTS Failure Count: A 32-bit value representing the failed transmitted RTS.

ACK Failure Count: A 32-bit value representing the number of failed acknowledgements.

Rx Fragment Count: A 32-bit value representing the number of fragmented frames received.

Multicast RX Count: A 32-bit value representing the number of multicast frames received.

FCS Error Count: A 32-bit value representing the number of FCS failures.

Decryption Errors: A 32-bit value representing the number of Decryption errors that occurred on the WTP. Note that this field is only valid in cases where the WTP provides encryption/ decryption services.

11.8. 802.11 Control Messages

This section defines CAPWAP Control Messages that are specific to the IEEE 802.11 binding.

11.8.1. IEEE 802.11 WLAN Config Request

The IEEE 802.11 WLAN Configuration Request is sent by the AC to the WTP in order to change services provided by the WTP. This control message is used to either create, update or delete a WLAN on the WTP.

The IEEE 802.11 WLAN Configuration Request is sent as a result of either some manual admistrative process (e.g., deleting a WLAN), or automatically to create a WLAN on a WTP. When sent automatically to create a WLAN, this control message is sent after the CAPWAP Configuration Request message has been received by the WTP.

Upon receiving this control message, the WTP will modify the necessary services, and transmit an IEEE 802.11 WLAN Configuration Response.

A WTP MAY provide service for more than one WLAN, therefore every WLAN is identified through a numerical index. For instance, a WTP that is capable of supporting up to 16 SSIDs, could accept up to 16 IEEE 802.11 WLAN Configuration Request messages that include the Add WLAN message element.

Since the index is the primary identifier for a WLAN, an AC SHOULD attempt to ensure that the same WLAN is identified through the same index number on all of its WTPs. An AC that does not follow this approach MUST find some other means of maintaining a WLAN Identifier to SSID mapping table.

The following subsections define the message elements that are value for this CAPWAP operation. Only one message MUST be present.

11.8.1.1. IEEE 802.11 Add WLAN

The Add WLAN message element is used by the AC to define a wireless LAN on the WTP. The value contains the following format:

```
        0                   1                   2                   3
        0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |    Radio ID   |         WLAN Capability       |    WLAN ID    |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                      Encryption Policy                        |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                             Key                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                             Key                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                             Key                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                             Key                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                             Key                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                             Key                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                             Key                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                             Key                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |   Key Index   |   Shared Key  | WPA Data Len  |WPA IE Data ...|
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       | RSN Data Len  |RSN IE Data ...| WME Data Len  |WME IE Data ...|
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |  11e Data Len |11e IE Data ...|      QoS      |   Auth Type   |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       | Suppress SSID |    SSID ...
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 7 for IEEE 802.11 Add WLAN

Length: >= 49

Radio ID: An 8-bit value representing the radio.

WLAN Capability: A 16-bit value containing the capabilities to be advertised by the WTP within the Probe and Beacon messages.

WLAN ID: An 8-bit value specifying the WLAN Identifier.

Encryption Policy: A 32-bit value specifying the encryption scheme to apply to traffic to and from the mobile station.

The following values are supported:

0 - Encrypt WEP 104: All packets to/from the mobile station must be encrypted using standard 104 bit WEP.

1 - Clear Text: All packets to/from the mobile station do not require any additional crypto processing by the WTP.

2 - Encrypt WEP 40: All packets to/from the mobile station must be encrypted using standard 40 bit WEP.

3 - Encrypt WEP 128: All packets to/from the mobile station must be encrypted using standard 128 bit WEP.

4 - Encrypt AES-CCMP 128: All packets to/from the mobile station must be encrypted using 128 bit AES CCMP [7]

5 - Encrypt TKIP-MIC: All packets to/from the mobile station must be encrypted using TKIP and authenticated using Michael [21]

6 - Encrypt CKIP: All packets to/from the mobile station must be encrypted using Cisco TKIP.

Key: A 32 byte Session Key to use with the encryption policy.

Key-Index: The Key Index associated with the key.

Shared Key: A 1 byte boolean that specifies whether the key included in the Key field is a shared WEP key. A value of zero is used to state that the key is not a shared WEP key, while a value of one is used to state that the key is a shared WEP key.

WPA Data Len: Length of the WPA IE.

WPA IE: A 32 byte field containing the WPA Information Element.

RSN Data Len: Length of the RSN IE.

RSN IE: A 64 byte field containing the RSN Information Element.

WME Data Len: Length of the WME IE.

WME IE: A 32 byte field containing the WME Information Element.

DOT11E Data Len: Length of the 802.11e IE.

DOT11E IE: A 32 byte field containing the 802.11e Information Element.

QOS: An 8-bit value specifying the QoS policy to enforce for the station.

The following values are supported:

0 - Best Effort

1 - Video

2 - Voice

3 - Background

Auth Type: An 8-bit value specifying the station's authentication type.

The following values are supported:

0 - Open System

1 - WEP Shared Key

2 - WPA/WPA2 802.1X

3 - WPA/WPA2 PSK

Supress SSID: A boolean indicating whether the SSID is to be advertised by the WTP. A value of zero supresses the SSID in the 802.11 Beacon and Probe Response frames, while a value of one will cause the WTP to populate the field.

SSID: The SSID attribute is the service set identifier that will be advertised by the WTP for this WLAN.

11.8.1.2. IEEE 802.11 Delete WLAN

The delete WLAN message element is used to inform the WTP that a previously created WLAN is to be deleted. The value contains the following fields:

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Radio ID   |            WLAN ID            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 28 for IEEE 802.11 Delete WLAN

Length: 3

Radio ID: An 8-bit value representing the radio

WLAN ID: A 16-bit value specifying the WLAN Identifier

11.8.1.3. IEEE 802.11 Update WLAN

The Update WLAN message element is used by the AC to define a wireless LAN on the WTP. The value contains the following format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Radio ID   |             WLAN ID           |Encrypt Policy |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Encryption Policy        |     Key...    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Key ...                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Key Index   |   Shared Key  |       WLAN Capability         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 34 for IEEE 802.11 Update WLAN

Length: 43

Radio ID: An 8-bit value representing the radio.

WLAN ID: A 16-bit value specifying the WLAN Identifier.

Encryption Policy: A 32-bit value specifying the encryption scheme to apply to traffic to and from the mobile station.

The following values are supported:

0 - Encrypt WEP 104: All packets to/from the mobile station must be encrypted using standard 104 bit WEP.

1 - Clear Text: All packets to/from the mobile station do not require any additional crypto processing by the WTP.

2 - Encrypt WEP 40: All packets to/from the mobile station must be encrypted using standard 40 bit WEP.

3 - Encrypt WEP 128: All packets to/from the mobile station must be encrypted using standard 128 bit WEP.

4 - Encrypt AES-CCMP 128: All packets to/from the mobile station must be encrypted using 128 bit AES CCMP [7]

5 - Encrypt TKIP-MIC: All packets to/from the mobile station must be encrypted using TKIP and authenticated using Michael [21]

6 - Encrypt CKIP: All packets to/from the mobile station must be encrypted using Cisco TKIP.

Key: A 32 byte Session Key to use with the encryption policy.

Key-Index: The Key Index associated with the key.

Shared Key: A 1 byte boolean that specifies whether the key included in the Key field is a shared WEP key. A value of zero means that the key is not a shared WEP key, while a value of one is used to state that the key is a shared WEP key.

WLAN Capability: A 16-bit value containing the capabilities to be advertised by the WTP within the Probe and Beacon messages.

11.8.2. IEEE 802.11 WLAN Config Response

The IEEE 802.11 WLAN Configuration Response is sent by the AC to the WTP as an acknowledgement of the receipt of an IEEE 802.11 WLAN Configuration Request.

This CAPWAP control message does not include any message elements.

11.8.3. IEEE 802.11 WTP Event

The IEEE 802.11 WTP Event CAPWAP message is used by the WTP in order to report asynchronous events to the AC. There is no reply message expected from the AC, except that the message is acknowledged via the reliable transport.

When the AC receives the IEEE 802.11 WTP Event, it will take whatever action is necessary, depending upon the message elements present in the message.

The IEEE 802.11 WTP Event message MUST contain one of the following message element described in the next subsections.

11.8.3.1. IEEE 802.11 MIC Countermeasures

The MIC Countermeasures message element is sent by the WTP to the AC to indicate the occurrence of a MIC failure.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Radio ID    |    WLAN ID    |          MAC Address          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          MAC Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 61 for IEEE 802.11 MIC Countermeasures

Length: 8

Radio ID: The Radio Identifier, typically refers to some interface index on the WTP.

WLAN ID: This 8-bit unsigned integer includes the WLAN Identifier, on which the MIC failure occurred.

MAC Address: The MAC Address of the mobile station that caused the MIC failure.

11.8.3.2. IEEE 802.11 WTP Radio Fail Alarm Indication

The WTP Radio Fail Alarm Indication message element is sent by the WTP to the AC when it detects a radio failure.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Radio ID    |     Type      |    Status     |      Pad      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 95 for WTP Radio Fail Alarm Indication

Length: 4

Radio ID: The Radio Identifier, typically refers to some interface index on the WTP

Type: The type of radio failure detected. The following values are supported:

1 - Receiver

2 - Transmitter

Status: An 8-bit boolean indicating whether the radio failure is being reported or cleared. A value of zero is used to clear the event, while a value of one is used to report the event.

Pad: Reserved field MUST be set to zero (0).

11.9. Message Element Bindings

The IEEE 802.11 Message Element binding has the following definitions:

	Conf Req	Conf Resp	Conf Upd	Add Mobile
IEEE 802.11 WTP WLAN Radio Configuration	X	X	X	
IEEE 802.11 Rate Set		X	X	
IEEE 802.11 Multi-domain Capability	X	X	X	
IEEE 802.11 MAC Operation	X	X	X	
IEEE 802.11 Tx Power	X	X	X	
IEEE 802.11 Tx Power Level	X			
IEEE 802.11 Direct Sequence Control	X	X	X	
IEEE 802.11 OFDM Control	X	X	X	
IEEE 802.11 Supported Rates	X	X		
IEEE 802.11 Antenna	X	X	X	
IEEE 802.11 CFP Status	X		X	
IEEE 802.11 Broadcast Probe Mode		X	X	
IEEE 802.11 WTP Mode and Type	X?		X	
IEEE 802.11 WTP Quality of Service		X	X	
IEEE 802.11 MIC Error Report From Mobile			X	
IEEE 802.11 Update Mobile QoS				X
IEEE 802.11 Mobile Session Key				X

11.9.1. IEEE 802.11 WTP WLAN Radio Configuration

The WTP WLAN radio configuration is used by the AC to configure a Radio on the WTP. The message element value contains the following Fields:

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |    Radio ID   |    Reserved   |        Occupancy Limit        |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |    CFP Per    |     CFP Maximum Duration      |     BSS ID    |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                             BSS ID                            |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |     BSS ID    |         Beacon Period         |    DTIM Per   |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                         Country String                        |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    | Num Of BSSIDs |
    +-+-+-+-+-+-+-+-+
```

Type: 8 for IEEE 802.11 WTP WLAN Radio Configuration

Length: 20

Radio ID: An 8-bit value representing the radio to configure.

Reserved: MUST be set to zero

Occupancy Limit: This attribute indicates the maximum amount of time, in TU, that a point coordinator MAY control the usage of the wireless medium without relinquishing control for long enough to allow at least one instance of DCF access to the medium. The default value of this attribute SHOULD be 100, and the maximum value SHOULD be 1000.

CFP Period: The attribute describes the number of DTIM intervals between the start of CFPs.

CFP Maximum Duration: The attribute describes the maximum duration of the CFP in TU that MAY be generated by the PCF.

BSSID: The WLAN Radio’s base MAC Address. For WTPs that support more than a single WLAN, the value of the WLAN Identifier is added to the last octet of the BSSID. Therefore, a WTP that supports 16 WLANs MUST have 16 MAC Addresses reserved for it, and the last nibble is used to represent the WLAN ID.

Beacon Period: This attribute specifies the number of TU that a station uses for scheduling Beacon transmissions. This value is transmitted in Beacon and Probe Response frames.

DTIM Period: This attribute specifies the number of beacon intervals that elapses between transmission of Beacons frames containing a TIM element whose DTIM Count field is 0. This value is transmitted in the DTIM Period field of Beacon frames.

Country Code: This attribute identifies the country in which the station is operating. The first two octets of this string is the two character country code as described in document ISO/IEC 3166-1. The third octet MUST be one of the following:

1. an ASCII space character, if the regulations under which the station is operating encompass all environments in the country,

2. an ASCII 'O' character, if the regulations under which the station is operating are for an outdoor environment only, or

3. an ASCII 'I' character, if the regulations under which the station is operating are for an indoor environment only

Number of BSSIDs: This attribute contains the maximum number of BSSIDs supported by the WTP. This value restricts the number of logical networks supported by the WTP, and is between 1 and 16.

11.9.2. IEEE 802.11 Rate Set

The rate set message element value is sent by the AC and contains the supported operational rates. It contains the following fields.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Radio ID    |                 Rate Set...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 16 for IEEE 802.11 Rate Set

Length: >= 3

Radio ID: An 8-bit value representing the radio to configure.

Rate Set: The AC generates the Rate Set that the WTP is to include in it's Beacon and Probe messages. The length of this field is between 2 and 8 bytes.

11.9.3. IEEE 802.11 Multi-domain Capability

The multi-domain capability message element is used by the AC to inform the WTP of regulatory limits. The value contains the following fields.

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |    Radio ID   |    Reserved   |        First Channel #        |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |      Number of Channels       |       Max Tx Power Level      |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 10 for IEEE 802.11 Multi-Domain Capability

Length: 8

Radio ID: An 8-bit value representing the radio to configure.

Reserved: MUST be set to zero

First Channnel #: This attribute indicates the value of the lowest channel number in the subband for the associated domain country string.

Number of Channels: This attribute indicates the value of the total number of channels allowed in the subband for the associated domain country string.

Max Tx Power Level: This attribute indicates the maximum transmit power, in dBm, allowed in the subband for the associated domain country string.

11.9.4. IEEE 802.11 MAC Operation

The MAC operation message element is sent by the AC to set the 802.11 MAC parameters on the WTP. The value contains the following fields.

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |    Radio ID   |    Reserved   |         RTS Threshold         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |  Short Retry  |  Long Retry   |    Fragmentation Threshold    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                       Tx MSDU Lifetime                        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                       Rx MSDU Lifetime                        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 11 for IEEE 802.11 MAC Operation

Length: 16

Radio ID: An 8-bit value representing the radio to configure.

Reserved: MUST be set to zero

RTS Threshold: This attribute indicates the number of octets in an MPDU, below which an RTS/CTS handshake MUST NOT be performed. An RTS/CTS handshake MUST be performed at the beginning of any frame exchange sequence where the MPDU is of type Data or Management, the MPDU has an individual address in the Address1 field, and the length of the MPDU is greater than this threshold. Setting this attribute to be larger than the maximum MSDU size MUST have the effect of turning off the RTS/CTS handshake for frames of Data or Management type transmitted by this STA. Setting this attribute to zero MUST have the effect of turning on the RTS/CTS handshake for all frames of Data or Management type transmitted by this STA. The default value of this attribute MUST be 2347.

Short Retry: This attribute indicates the maximum number of transmission attempts of a frame, the length of which is less than or equal to RTSThreshold, that MUST be made before a failure condition is indicated. The default value of this attribute MUST be 7.

Long Retry: This attribute indicates the maximum number of transmission attempts of a frame, the length of which is greater than dot11RTSThreshold, that MUST be made before a failure condition is indicated. The default value of this attribute MUST be 4.

Fragmentation Threshold: This attribute specifies the current maximum size, in octets, of the MPDU that MAY be delivered to the PHY. An MSDU MUST be broken into fragments if its size exceeds the value of this attribute after adding MAC headers and trailers. An MSDU or MMPDU MUST be fragmented when the resulting frame has an individual address in the Address1 field, and the length of the frame is larger than this threshold. The default value for this attribute MUST be the lesser of 2346 or the aMPDUMaxLength of the attached PHY and MUST never exceed the lesser of 2346 or the aMPDUMaxLength of the attached PHY. The value of this attribute MUST never be less than 256.

Tx MSDU Lifetime: This attribute speficies the elapsed time in TU, after the initial transmission of an MSDU, after which further attempts to transmit the MSDU MUST be terminated. The default value of this attribute MUST be 512.

Rx MSDU Lifetime: This attribute specifies the elapsed time in TU, after the initial reception of a fragmented MMPDU or MSDU, after which further attempts to reassemble the MMPDU or MSDU MUST be terminated. The default value MUST be 512.

11.9.5. IEEE 802.11 Tx Power

The Tx power message element value is bi-directional. When sent by the WTP, it contains the current power level of the radio in question. When sent by the AC, it contains the power level the WTP MUST adhere to.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Radio ID   |    Reserved   |        Current Tx Power       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 12 for IEEE 802.11 Tx Power

Length: 4

Radio ID: An 8-bit value representing the radio to configure.

Reserved: MUST be set to zero

Current Tx Power: This attribute contains the transmit output power in mW.

11.9.6. IEEE 802.11 Tx Power Level

The Tx power level message element is sent by the WTP and contains the different power levels supported. The value contains the following fields.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Radio ID   |   Num Levels  |        Power Level [n]        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 13 for IEEE 802.11 Tx Power Level

Length: >= 4

Radio ID: An 8-bit value representing the radio to configure.

Num Levels: The number of power level attributes.

Power Level: Each power level fields contains a supported power level, in mW.

11.9.7. IEEE 802.11 Direct Sequence Control

The direct sequence control message element is a bi-directional element. When sent by the WTP, it contains the current state. When sent by the AC, the WTP MUST adhere to the values. This element is only used for 802.11b radios. The value has the following fields.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Radio ID   |    Reserved   | Current Chan  |  Current CCA  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Energy Detect Threshold                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 14 for IEEE 802.11 Direct Sequence Control

Length: 8

Radio ID: An 8-bit value representing the radio to configure.

Reserved: MUST be set to zero

Current Channel: This attribute contains the current operating frequency channel of the DSSS PHY.

Current CCA: The current CCA method in operation. Valid values are:

1 - energy detect only (edonly)

2 - carrier sense only (csonly)

4 - carrier sense and energy detect (edandcs)

8 - carrier sense with timer (cswithtimer)

16 - high rate carrier sense and energy detect (hrcsanded)

Energy Detect Threshold: The current Energy Detect Threshold being used by the DSSS PHY.

11.9.8. IEEE 802.11 OFDM Control

The OFDM control message element is a bi-directional element. When sent by the WTP, it contains the current state. When sent by the AC, the WTP MUST adhere to the values. This element is only used for 802.11a radios. The value contains the following fields:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Radio ID   |    Reserved   | Current Chan  |  Band Support |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         TI Threshold                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 15 for IEEE 802.11 OFDM Control

Length: 8

Radio ID: An 8-bit value representing the radio to configure.

Reserved: MUST be set to zero

Current Channel: This attribute contains the current operating frequency channel of the OFDM PHY.

Band Supported: The capability of the OFDM PHY implementation to operate in the three U-NII bands. Coded as an integer value of a three bit field as follows:

capable of operating in the lower (5.15-5.25 GHz) U-NII band

capable of operating in the middle (5.25-5.35 GHz) U-NII band

capable of operating in the upper (5.725-5.825 GHz) U-NII band

For example, for an implementation capable of operating in the lower and mid bands this attribute would take the value

TI Threshold: The Threshold being used to detect a busy medium (frequency). CCA MUST report a busy medium upon detecting the RSSI above this threshold.

11.9.9. IEEE 802.11 Antenna

The antenna message element is communicated by the WTP to the AC to provide information on the antennas available. The AC MAY use this element to reconfigure the WTP's antennas. The value contains the following fields:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Radio ID    |   Diversity   |    Combiner   |  Antenna Cnt  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Antenna Selection [0..N]                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 41 for IEEE 802.11 Antenna

Length: >= 5

Radio ID: An 8-bit value representing the radio to configure.

Diversity: An 8-bit value specifying whether the antenna is to provide receive diversity. The following values are supported:

0 - Disabled

1 - Enabled (may only be true if the antenna can be used as a receive antenna)

Combiner: An 8-bit value specifying the combiner selection. The following values are supported:

1 - Sectorized (Left)

2 - Sectorized (Right)

3 - Omni

4 - Mimo

Antenna Count: An 8-bit value specifying the number of Antenna Selection fields.

Antenna Selection: One 8-bit antenna configuration value per antenna in the WTP. The following values are supported:

1 - Internal Antenna

2 - External Antenna

11.9.10. IEEE 802.11 Supported Rates

The supported rates message element is sent by the WTP to indicate the rates that it supports. The value contains the following fields.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Radio ID   |               Supported Rates...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 16 for IEEE 802.11 Supported Rates

Length: >= 3

Radio ID: An 8-bit value representing the radio.

Supported Rates: The WTP includes the Supported Rates that it’s hardware supports. The format is identical to the Rate Set message element and is between 2 and 8 bytes in length.

11.9.11. IEEE 802.11 CFP Status

The CFP Status message element is sent to provide the CF Polling configuration.

```
 0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Radio ID    |    Status     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 48 for IEEE 802.11 CFP Status

Length: 2

Radio ID: The Radio Identifier, typically refers to some interface index on the WTP

Status: An 8-bit boolean containing the status of the CF Polling feature. A value of zero disables CFP Status, while a value of one enables it.

11.9.12. IEEE 802.11 Broadcast Probe Mode

The Broadcast Probe Mode message element indicates whether a WTP will respond to NULL SSID probe requests. Since broadcast NULL probes are not sent to a specific BSSID, the WTP cannot know which SSID the sending station is querying. Therefore, this behavior must be global to the WTP.

```
 0
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|    Status     |
+-+-+-+-+-+-+-+-+
```

Type: 51 for IEEE 802.11 Broadcast Probe Mode

Length: 1

Status: An 8-bit boolean indicating the status of whether a WTP shall response to a NULL SSID probe request. A value of zero disables NULL SSID probe response, while a value of one enables it.

11.9.13. IEEE 802.11 WTP Quality of Service

The WTP Quality of Service message element value is sent by the AC to the WTP to communicate quality of service configuration information.

```
 0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Radio ID    |  Tag Packets  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 57 for IEEE 802.11 WTP Quality of Service

Length: >= 2

Radio ID: The Radio Identifier, typically refers to some interface index on the WTP

Tag Packets: An value indicating whether CAPWAP packets should be tagged with for QoS purposes. The following values are currently supported:

0 - Untagged

1 - 802.1P

2 - DSCP

Immediately following the above header is the following data structure. This data structure will be repeated five times; once for every QoS profile. The order of the QoS profiles are Voice, Video, Best Effort and Background.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Queue Depth  |             CWMin             |     CWMax     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     CWMax     |     AIFS      |              CBR              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Dot1P Tag   |   DSCP Tag    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Queue Depth: The number of packets that can be on the specific QoS transmit queue at any given time.

CWMin: The Contention Window minimum value for the QoS transmit queue.

CWMax: The Contention Window maximum value for the QoS transmit queue.

AIFS: The Arbitration Inter Frame Spacing to use for the QoS transmit queue.

CBR: The CBR value to observe for the QoS transmit queue.

Dot1P Tag: The 802.1P precedence value to use if packets are to be 802.1P tagged.

DSCP Tag: The DSCP label to use if packets are to be DSCP tagged.

11.9.14. IEEE 802.11 MIC Error Report From Mobile

The MIC Error Report From Mobile message element is sent by an AC to an WTP when it receives a MIC failure notification, via the Error bit in the EAPOL-Key frame.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Client MAC Address                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Client MAC Address       |             BSSID             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             BSSID                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Radio ID    |    WLAN ID    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: 79 for IEEE 802.11 MIC Error Report From Mobile

Length: 14

Client MAC Address: The Client MAC Address of the station reporting the MIC failure.

BSSID: The BSSID on which the MIC failure is being reported.

Radio ID: The Radio Identifier, typically refers to some interface index on the WTP

WLAN ID: The WLAN ID on which the MIC failure is being reported.

11.10. IEEE 802.11 Message Element Values

This section lists IEEE 802.11 specific values for any generic CAPWAP message elements which include fields whose values are technology specific.

IEEE 802.11 uses the following values:

4 - Encrypt AES-CCMP 128: WTP supports AES-CCMP, as defined in [7].

5 - Encrypt TKIP-MIC: WTP supports TKIP and Michael, as defined in [21].

12. CAPWAP Protocol Timers

A WTP or AC that implements CAPWAP discovery MUST implement the following timers.

12.1. MaxDiscoveryInterval

The maximum time allowed between sending discovery requests from the interface, in seconds. Must be no less than 2 seconds and no greater than 180 seconds.

Default: 20 seconds.

12.2. SilentInterval

The minimum time, in seconds, a WTP MUST wait after failing to receive any responses to its discovery requests, before it MAY again send discovery requests.

Default: 30

12.3. NeighborDeadInterval

The minimum time, in seconds, a WTP MUST wait without having received Echo Responses to its Echo Requests, before the destination for the Echo Request may be considered dead. Must be no less than 2*EchoInterval seconds and no greater than 240 seconds.

Default: 60

12.4. WaitJoin

The maximum time, in seconds, a WTP MUST wait without having received a DTLS Handshake message from an AC. This timer must be greater than TBD seconds.

Default: TBD

12.5. EchoInterval

The minimum time, in seconds, between sending echo requests to the AC with which the WTP has joined.

Default: 30

12.6. DiscoveryInterval

The minimum time, in seconds, that a WTP MUST wait after receiving a

Discovery Response, before initiating a DTLS handshake.

Default: 5

12.7. RetransmitInterval

The minimum time, in seconds, which a non-acknowledged CAPWAP packet will be retransmitted.

Default: 3

12.8. ResponseTimeout

The minimum time, in seconds, which the WTP or AC must respond to a CAPWAP Request message.

Default: 1

12.9. KeyLifetime

The maximum time, in seconds, which a CAPWAP DTLS session key is valid.

Default: 28800

13. CAPWAP Protocol Variables

A WTP or AC that implements CAPWAP discovery MUST allow for the following variables to be configured by system management; default values are specified so as to make it unnecessary to configure any of these variables in many cases.

13.1. MaxDiscoveries

The maximum number of discovery requests that will be sent after a WTP boots.

Default: 10

13.2. DiscoveryCount

The number of discoveries transmitted by a WTP to a single AC. This is a monotonically increasing counter.

13.3. RetransmitCount

The number of retransmissions for a given CAPWAP packet. This is a monotonically increasing counter.

13.4. MaxRetransmit

The maximum number of retransmissions for a given CAPWAP packet before the link layer considers the peer dead.

Default: 5

14. NAT Considerations

There are two specific situations in which a NAT system may be used in conjunction with a CAPWAP-enabled system. The first consists of a configuration where the WTP is behind a NAT system. Given that all communication is initiated by the WTP, and all communication is performed over IP using two UDP ports, the protocol easily traverses NAT systems in this configuration.

The second configuration is one where the AC sits behind a NAT. Two issues exist in this situation. First, an AC communicates its interfaces, and associated WTP load on these interfaces, through the WTP Manager Control IP Address. This message element is currently mandatory, and if NAT compliance became an issue, it would be possible to either:

1. Make the WTP Manager Control IP Address optional, allowing the WTP to simply use the known IP Address. However, note that this approach would eliminate the ability to perform load balancing of WTP across ACs, and therefore is not the recommended approach.

2. Allow an AC to be able to configure a NAT'ed address for every associated AC that would generally be communicated in the WTP Manager Control IP Address message element.

3. Require that if a WTP determines that the AC List message element consists of a set of IP Addresses that are different from the AC's IP Address it is currently communicating with, then assume that NAT is being enforced, and require that the WTP communicate with the original AC's IP Address (and ignore the WTP Manager Control IP Address message element(s)).

Another issue related to having an AC behind a NAT system is CAPWAP's support for the CAPWAP Objective to allow the control and data plane to be separated. In order to support this requirement, the CAPWAP protocol defines the WTP Manager Data IP Address message element, which allows the AC to inform the WTP that the CAPWAP data frames are to be forwarded to a separate IP Address. This feature MUST be disabled when an AC is behind a NAT. However, there is no easy way to provide some default mechanism that satisfies both the data/control separation and NAT objectives, as they directly conflict with each other. As a consequence, user intervention will be required to support such networks.

The CAPWAP protocol allows for all of the ACs identities supporting a group of WTPs to be communicated through the AC List message element. This feature must be disabled when the AC is behind a NAT and the IP Address that is embedded would be invalid.

The CAPWAP protocol has a feature that allows an AC to configure a static IP address on a WTP. The WTP Static IP Address Information message element provides such a function, however this feature SHOULD NOT be used in NAT'ed environments, unless the administrator is familiar with the internal IP addressing scheme within the WTP's private network, and does not rely on the public address seen by the AC.

When a WTP detects the duplicate address condition, it generates a message to the AC, which includes the Duplicate IP Address message element. The IP Address embedded within this message element is different from the public IP address seen by the AC.

15. Security Considerations

The security of the CAPWAP protocol over DTLS is completely dependent on the security of DTLS. Any flaws in DTLS compromise the security of the CAPWAP protocol. In particular, it is critical that the communicating parties verify their peer's credentials. In the case of pre-shared keys, this happens automatically via the key. In the case of certificates, the parties must check the peer's certificate. The appropriate checks are described in Section 10.3.

The use of parallel protected and unprotected channels deserves special consideration, but does not create a threat. There are two potential concerns: attempting to convert protected data into un-protected data and attempting to convert un-protected data into protected data. The use of message authentication makes it impossible for the attacker to forge protected records. The attacker can easily remove protected records from the stream (this is a consequence of unreliability), though not undetectably so. If a non-encrypted cipher suite is in use, the attacker can turn such a record into an un-protected record. However, this attack is really no different from simple injection into the unprotected stream.

Perfect Forward Secrecy is not a requirement for the CAPWAP protocol.

The CAPWAP protocol does not add any new vulnerabilities to IEEE 802.11 infrastructure which uses WEP for encryption. However, implementors SHOULD discourage the use of WEP to allow the market to move towards technically sound cryptographic solutions, such as IEEE 802.11i.

15.1. PSK based Session Key establishment

Use of a fixed shared secret of limited entropy (for example, a PSK that is relatively short, or was chosen by a human and thus may contain less entropy than its length would imply) may allow an attacker to perform a brute-force or dictionary attack to recover the secret.

It is RECOMMENDED that implementations that allow the administrator to manually configure the PSK also provide a functionality for generating a new random PSK, taking RFC 1750 [4] into account.

16. IANA Considerations

A separate UDP port for data channel communications is (currently) the selected demultiplexing mechanism, and a port must be assigned for this purpose.

The Message element type fields must be IANA aassigned, see Section 4.3.2.

17. References

17.1. Normative References

[1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[2] National Institute of Standards and Technology, "Advanced Encryption Standard (AES)", FIPS PUB 197, November 2001, <http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>.

[3] Whiting, D., Housley, R., and N. Ferguson, "Counter with CBC-MAC (CCM)", RFC 3610, September 2003.

[4] Eastlake, D., Crocker, S., and J. Schiller, "Randomness Recommendations for Security", RFC 1750, December 1994.

[5] Manner, J. and M. Kojo, "Mobility Related Terminology", RFC 3753, June 2004.

[6] "Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications", IEEE Standard 802.11, 1999, <http://standards.ieee.org/getieee802/download/802.11-1999.pdf>.

[7] "Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 6: Medium Access Control (MAC) Security Enhancements", IEEE Standard 802.11i, July 2004, <http://standards.ieee.org/getieee802/download/802.11i-2004.pdf>.

[8] Clark, D., "IP datagram reassembly algorithms", RFC 815, July 1982.

[9] Schaad, J. and R. Housley, "Advanced Encryption Standard (AES) Key Wrap Algorithm", RFC 3394, September 2002.

[10] Mills, D., "Network Time Protocol (Version 3) Specification, Implementation", RFC 1305, March 1992.

[11] Housley, R., Polk, W., Ford, W., and D. Solo, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 3280, April 2002.

[12] Eronen, P. and H. Tschofenig, "Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)", RFC 4279, December 2005.

[13] "Netscape Certificate Extensions Specification", <http://wp.netscape.com/eng/security/comm4-cert-exts.html>.

[14] Clancy, C., "Security Review of the Light Weight Access Point Protocol", May 2005, <http://www.cs.umd.edu/˜clancy/docs/lwapp-review.pdf>.

[15] Rescorla et al, E., "Datagram Transport Layer Security", June 2004.

[16] "Recommendation for Block Cipher Modes of Operation: the CMAC Mode for Authentication", May 2005, <http://csrc.ncsl.nist.gov/publications/nistpubs/800-38B/SP_800-38B.pdf>.

17.2. Informational References

[17] Reynolds, J., "Assigned Numbers: RFC 1700 is Replaced by an On-line Database", RFC 3232, January 2002.

[18] Bradner, S., "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, October 1996.

[19] Kent, S. and R. Atkinson, "Security Architecture for the Internet Protocol", RFC 2401, November 1998.

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

[21] "WiFi Protected Access (WPA) rev 1.6", April 2003.

[22] Dierks et al, T., "The TLS Protocol Version 1.1", June 2005.

[23] Modadugu et al, N., "The Design and Implementation of Datagram TLS", Feb 2004.

Editors' Addresses

Pat R. Calhoun
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134

Phone: +1 408-853-5269
Email: pcalhoun@cisco.com

Michael P. Montemurro
Chantry Networks
1900 Minnesota Court, Suite 125
Mississauga, ON L5N 3C9
Canada

Phone: +1 905-363-6413
Email: michael.montemurro@siemens.com

Dorothy Stanley
Aruba Networks
1322 Crossman Ave
Sunnyvale, CA 94089

Phone: +1 630-363-1389
Email: dstanley@arubanetworks.com

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.