# WLAN

The Wireless Local Area Network Consortium

WPA2
Station MAC Conformance Test Suite
*Version 2.4*

*Technical Document*

*Last Updated:* February 10, 2013


***Wireless LAN Consortium***
***InterOperability Laboratory***
***University of New Hampshire***

***121 Technology Drive, Suite 2***
***Durham, NH 03824***
***Phone: +1-603- 862-2263***
***Fax: +1-603- 862-4181***
http://www.iol.unh.edu/consortiums/wireless/

## MODIFICATION RECORD

- February 2013 updated all IEEE 802.11 references to the 802.11 2012 standard
- November 2012 Changed name of the test plan to WPA2
- January 2010 Updated all IEEE 802.11-2007 subclause references. Test 1.2.2 - changed observable results for MSDU 4 (ICMP Echo Request containing the Key ID in the CCMP header set to its complement) to informative.
- November 2008 Test 1.1.2 - changed 3rd to 4th and 4th to 3rd. Changed the observable results to include which MSDU is related: 1.2.2, 1.2.2, 1.2.4, 1.3.1, 1.3.2, 1.3.4, 1.3.9, 1.3.11, 1.3.12 (dsr)
- August 2007 Minor grammatical editing (2.1)
- June 2007 Rewritten for station. Unused procedures and groups were removed. Rewrote test procedures, discussions and observable results for clarity. 1.2.3 was added. (2.0)
- January 2006 Fixed error in 1.3.4, part a observable results (1.0)
- August 2005 Third version for external review (0.90)
- May 2005 Second version for external review (0.86)
- March 2005 Initial version for external review (0.7)

**ACKNOWLEDGMENTS**

**The University of New Hampshire would like to acknowledge the efforts of the following individuals in the development of this test suite.**

| | |
|---|---|
| Chris Kane | University of New Hampshire |
| Kevin Karcz | University of New Hampshire |
| Jeremy Kent | University of New Hampshire |
| Jonathan Zink | University of New Hampshire |
| Anthony Murabito | University of New Hampshire |
| Daniel Reynolds | University of New Hampshire |

## INTRODUCTION

### Overview

The University of New Hampshire's InterOperability Laboratory (UNH-IOL) is an institution designed to improve the interoperability of standards based products by providing an environment where a product can be tested against other implementations of a standard. This particular suite of tests has been developed to help implementers evaluate WPAv2 AES-PSK encryption on their STAs.

These tests are designed to determine if a product conforms to specifications defined in IEEE Std 802.11™-2012. Successful completion of all tests contained in this suite does not guarantee that the tested device will operate with other devices. However, combined with satisfactory operation in the IOL's interoperability test bed, these tests provide a reasonable level of confidence that the DUT will function properly in many RSNA environments.

### Organization of Tests

The tests contained in this document are organized to simplify the identification of information related to a test and to facilitate in the actual testing process. Each test contains an identification section that describes the test and provides cross-reference information. The discussion section covers background information and specifies why the test is to be performed. Tests are grouped in order to reduce setup time in the lab environment. Each test contains the following information:

### Test Number

The Test Number associated with each test follows a simple grouping structure. Listed first is the Test Group Number followed by the test's number within the group. This allows for the addition of future tests to the appropriate groups of the test suite without requiring the renumbering of the subsequent tests.

### Purpose

The purpose is a brief statement outlining what the test attempts to achieve. The test is written at the functional level.

### References

The references section lists cross-references to the IEEE Std 802.11™-2012 standards and other documentation that might be helpful in understanding and evaluating the test results.

### Resource Requirements

The requirements section specifies the test hardware and/or software needed to perform the test. This is generally expressed in terms of minimum requirements, however in some cases specific equipment manufacturer/model information may be provided.

### Last Modification

This specifies the date of the last modification to this test.

### Discussion

The discussion covers the assumptions made in the design or implementation of the test, as well as known limitations. Other items specific to the test are covered here.

### Test Setup

The setup section describes the initial configuration of the test environment. Small changes in the configuration should be included in the test procedure.

### Procedure

The procedure section of the test description contains the systematic instructions for carrying out the test. It provides a cookbook approach to testing, and may be interspersed with observable results.

### Observable Results

This section lists the specific observables that can be examined by the tester in order to verify that the DUT is operating properly. When multiple values for an observable are possible, this section provides a short discussion on how to interpret them. The determination of a pass or fail outcome for a particular test is often based on the successful (or unsuccessful) detection of a certain observable.

**Possible Problems**

This section contains a description of known issues with the test procedure, which may affect test results in certain situations. It may also refer the reader to test suite appendices and/or whitepapers that may provide more detail regarding these issues.

## TABLE OF CONTENTS

**LIST OF ABBREVIATIONS**

**Table 1 - Abbreviations**

| Abbreviation | Description |
|---|---|
| AAD | Additional Authentication Data |
| AES | Advanced Encryption Standard |
| AKMP | Authentication and Key Management Protocol |
| AP | Access Point |
| ARP | Address Resolution Protocol |
| CBC-MAC | Cipher Block Chaining Message Authentication Code |
| CCMP | Counter-Mode/CBC-MAC protocol |
| DLS | Direct Link Setup |
| DS | Distribution System |
| DUT | Device Under Test |
| EAP | Extensible Authentication Protocol |
| EAPoL | Extensible Authentication Protocol over LANs |
| FCS | Frame Check Sequence |
| GTK | Group Temporal Key |
| GTKSA | Group Temporal Key Security Association |
| ICMP | Internet Control Message Protocol |
| ICV | Integrity Check Value |
| ID | Identifier |
| IE | Information Element |
| IV | Initialization Vector |
| KDE | Key Data Encapsulation |
| KEK | EAPOL-Key Encryption Key |
| MAC | Media Access Control |
| MIC | Message Integrity Code |
| MPDU | MAC Protocol Data Unit |
| MSDU | MAC Service Data Unit |
| OUI | Organizationally Unique Identifier |
| PMKID | Pairwise Master Key IDentifier |
| PN | Packet Number |
| PSK | Pre-Shared Key |
| PTK | Pairwise Transient Key |
| PTKSA | Pairwise Transient Key Security Association |
| QoS | Quality of Service |
| RSC | Receive Sequence Counter |
| RSN | Robust Security Network |
| RSNA | Robust Security Network Association |
| RX | Receive |
| STA | STA |
| STSL | Station to Station Link |
| TK | Transmit Key |
| TKIP | Temporal Key Integrity Protocol |
| TS | Testing Station |
| TSC | Transmit Sequence Counter |
| TX | Transmit |
| WEP | Wired Equivalent Privacy |

## GROUP 1: CCMP Encapsulation

**Scope:** The following tests cover MAC security operations specific to the CCMP encapsulation process.

**Overview:** These tests are designed to verify that the DUT properly encrypts the payload of a plaintext MPDU and encapsulates the resulting cipher text. The MAC security functions explored are defined in Subclause 11.4.3 of IEEE Std 802.11™-2012.

**Test Setup:** These tests should be run with CCMP encryption using a PSK and all other settings as default unless specified.

**Test # 1.1.1: CCMP MIC Verification**

**Purpose:** To verify that CCMP encrypted data frames transmitted by the DUT contain a properly constructed MIC.

**References:**

[1] IEEE Std 802.11™-2012 Edition, Subclause 11.4.3.2 and 11.4.3.3

**Resource Requirements:**

- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** June 2007

**Discussion:** In an RSN, the encrypted MPDU uses a MIC to validate whether the frame has been received unaltered. The MIC is a function of the Nonce, TK, AAD, and plaintext data. CCMP processing expands the original MPDU size by 16 octets, 8 octets for the CCMP Header field and 8 octets for the MIC field. The MIC is calculated using the AES algorithm with CBC-MAC. This differs from the encryption of plaintext that uses the Counter Mode instead. Once the MIC is calculated, it is appended to the MPDU payload and finally the appended MPDU is encrypted.

**Test Setup:**

Table 2 - Configuration Parameter(s)

| Parameter | Value |
|---|---|
| Fragmentation Threshold | 256 |

Table 3 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | ICMP Echo Request of length 1500-bytes. |
| MSDU2 | ICMP Echo Request of length 257-bytes. |
| MSDU3 | ICMP Echo Request of length 256-bytes. |

**Procedure:**

1. Configure the DUT to the settings defined in the test setup above.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit MSDU1-3 to the DUT each with a payload consisting of 0xffff…
4. Repeat step 3 with payloads consisting of 0xaaaa…, 0x0000…, and 0x0123…
5. Observe transmissions from the DUT.

**Observable Results:**

The DUT should:

a. properly compute the cipher text and MIC using the TK, AAD, Nonce, and MPDU payload.
b. calculate the MIC transmitted to a unicast receiver address with the PTK.
c. should append the MIC to the MPDU payload with exactly 8 bytes after fragmentation occurs.
d. should append the MIC to the MPDU prior to its encryption.

**Possible Problems:** None

**Test # 1.1.2: CCMP Header Format**

**Purpose:** To verify that CCMP encrypted data frames transmitted from the DUT format the CCMP header properly.

**References:**

[1] IEEE Std 802.11™-2012 Edition, Subclause 11.4.3.2 and 11.4.3.3

**Resource Requirements:**

- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** November 2008

**Discussion:** In an RSN, the encrypted MPDU inserts an 8 octet CCMP header after the MAC header and before the encrypted payload. The CCMP header consists of the Key ID, ExtIV and PN values. The Extended IV bit is always set. The PN is a 48-bit number that is incremented for each MPDU transmitted by the DUT and should never be repeated while the same TK is being used. All other bits are reserved and should be ignored on reception.

Table 4 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | ICMP Echo Request of length 1500-bytes. |
| MSDU2 | ICMP Echo Request of length 257-bytes. |
| MSDU3 | ICMP Echo Request of length 256-bytes. |

**Procedure:**

1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit MSDU1-3 to the DUT.
3. Observe transmission from the DUT.

**Observable Results:**

The DUT should:

a. add exactly 8 bytes to the MPDU after fragmentation for the CCMP header.
b. set the Extended IV bit to 1.
c. set reserved bits b0 to b4 of the $4^{th}$ octet and all bits of the $3^{rd}$ octet in the CCMP header to 0.
d. increment the PN for every MPDU transmitted.

**Possible Problems:** None

**Test # 1.1.3: CCMP Encryption Verification**

**Purpose:** To verify that CCMP encryption on frames transmitted by the DUT is implemented properly.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.4.3.2 and 11.4.3.3

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** June 2007

**Discussion:** CCMP encrypts the payload of a plaintext MPDU and encapsulates the resulting cipher text using the following steps. The PN is incremented to obtain a fresh PN for each MPDU so that the PN never repeats for the same temporal key. Note that retransmitted MPDUs are not modified on retransmission. Using the fields in the MPDU header construct the AAD for CCM. The CCM algorithm provides integrity protection for the fields included in the AAD. MPDU header fields that may change when retransmitted are muted by being masked to 0 when calculating the AAD. The CCM Nonce block is constructed from the PN, A2, and the Priority field of the MPDU where A2 is MPDU Address 2. The new PN and the key identifier are placed into the 8-octet CCMP header. Using the temporal key, AAD, nonce, and MPDU data the cipher text and MIC are computed. This step is known as CCM originator processing. The encrypted MPDU is formed by combining the original MPDU header, the CCMP header, the encrypted data and MIC, as described in [1].

Table 5 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | ICMP Echo Request of length 1500-bytes. |
| MSDU2 | ICMP Echo Request of length 257-bytes. |
| MSDU3 | ICMP Echo Request of length 256-bytes. |

**Procedure:**
1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit MSDU1-3 to the DUT.
3. Observe transmission from the DUT.

**Observable Results:**
The DUT should:
a. properly compute the cipher text MPDU payload.
b. encrypt data transmitted to a unicast receiver address with the PTK.

**Possible Problems:** None

### GROUP 2: CCMP Decapsulation

**Scope:** The following tests cover MAC security operations specific to the CCMP decapsulation process.

**Overview:** These tests are designed to verify that the DUT properly decrypts the payload of a cipher text MPDU and decapsulates a plaintext MPDU. The MAC security functions explored are defined in Subclause 11.4.3 of IEEE Std 802.11™-2012 Edition.

**Test Setup:** These tests should be run with CCMP encryption using a PSK and all other settings as default unless specified.

**Test # 1.2.1: CCMP MIC Processing**

**Purpose:** To verify that the DUT correctly calculates the MIC when decrypting CCMP encrypted data.

**References:**

[1] IEEE Std 802.11™-2012 Edition, Subclause 11.4.3.3 and 11.4.3.4.3

**Resource Requirements:**

- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** November 2008

**Discussion:** In an RSN, the encrypted MPDU uses a MIC to validate whether the frame has been received unaltered. The MIC is a function of the Nonce, TK, AAD, and plaintext data. CCMP processing expands the original MPDU size by 16 octets, 8 octets for the CCMP Header field and 8 octets for the MIC field. The MIC is calculated using the AES algorithm with CBC-MAC. This differs from the encryption of plaintext that uses the Counter Mode instead. Once the MIC is calculated, it is appended to the MPDU payload and finally the appended MPDU is encrypted.

Table 6 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | ICMP Echo Request of length 512-bytes containing the MIC set to all zeroes before appending to the MPDU payload. |
| MSDU2 | ICMP Echo Request of length 512-bytes containing the MIC set to one less than the calculated value. |
| MSDU3 | ARP Request of length 512-bytes containing the MIC set to all zeros before appending to the MPDU payload. |
| MSDU4 | ICMP Echo Request of length 1500-bytes fragmented at 256-bytes. |
| MSDU5 | ICMP Echo Request of length 1500-bytes, fragmented at 256-bytes, containing the MIC set to one less than the calculated value. |

**Procedure:**

1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit MSDU1-2 to the DUT.
3. Instruct the TS to transmit MSDU3 to the broadcast address requesting the DUT's address.
4. Instruct the TS to transmit MSDU4-5 to the DUT.
5. Observe transmissions from the DUT.

**Observable Results:**

a. The DUT should discard received MPDUs containing invalid MICs (MSDU1, MSDU2, MSDU3, MSDU5).

**Possible Problems:** None.

**Test # 1.2.2: CCMP Header Processing**

**Purpose:** To verify that the DUT processes the CCMP header properly on received CCMP encrypted data frames.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.4.3.2 and 11.4.3.3

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** January 2010

**Discussion:** In an RSN, the encrypted MPDU inserts an 8 octet CCMP header after the MAC header and before the encrypted payload. The CCMP header consists of the Key ID, ExtIV and PN values. The Extended IV bit is always set. The PN is a 48-bit number that is incremented for each MPDU transmitted by the DUT and should never be repeated while the same TK is being used. All other bits are reserved and should be transmitted as 0.

Table 7 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | ICMP Echo Request of length 256-bytes. |
| MSDU2 | ICMP Echo Request containing all reserved bits in the CCMP header set to 1. |
| MSDU3 | ICMP Echo Request containing the Extended IV in the CCMP header muted to 0. |
| MSDU4 | ICMP Echo Request containing the Key ID in the CCMP header set to its complement. |

**Procedure:**
1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit MSDU1 to the DUT twice.
3. Instruct the TS to transmit MSDU2-4 to the DUT.
4. Repeat the preceding steps 1-3, a total of four times, each time using a new value for the Key ID assigned in the 4-way handshake.
5. Observe transmissions from the DUT.

**Observable Results:**
The DUT should:
a. accept keys for each Key ID defined.
b. ignore reserved bits b0 to b4 of the 4$^{th}$ octet and all bits of the 3$^{rd}$ octet in the CCMP header (MSDU2).
c. discard any CCMP encrypted frame with the Extended IV bit set to 0 (MSDU3).
d. may discard any frame containing an invalid Key ID (MSDU4).

**Possible Problems:** None.

**Test # 1.2.3: CCMP Decryption Verification**

**Purpose**: To verify that CCMP decryption on frames is implemented properly.

**References:**

[1] IEEE Std 802.11™-2012 Edition, Subclause 11.4.3.4

**Resource Requirements:**

- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** June 2007

**Discussion:** CCMP decrypts the payload of a cipher text MPDU and decapsulates a plaintext MPDU using the following steps. The encrypted MPDU is parsed to construct the AAD and nonce values. The AAD is formed from the MPDU header of the encrypted MPDU. The Nonce value is constructed from the A2, PN, and Priority Octet fields. The MIC is extracted for use in the CCM integrity checking. The CCM recipient processing uses the temporal key, AAD, nonce, MIC, and MPDU cipher text data to recover the MPDU plaintext data as well as to check the integrity of the AAD and MPDU plaintext data. The received MPDU header and the MPDU plaintext data from the CCM recipient processing may be concatenated to form a plaintext MPDU. The decryption processing prevents replay of MPDUs by validating that the PN in the MPDU is greater than the replay counter maintained for the session.

Table 8 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | ICMP Echo Request of length 256-bytes. |
| MSDU2 | ICMP Echo Request of length 512-bytes. |
| MSDU3 | ICMP Echo Request of length 1024-bytes. |
| MSDU4 | ICMP Echo Request of length 1500-bytes. |
| MSDU5 | ICMP Echo Request of length 256-bytes containing the reserved bits of the Priority field within the Nonce set to 1. |
| MSDU6 | ICMP Echo Request of length 256-bytes containing the Subtype bits within the AAD set to 1. |
| MSDU7 | ICMP Echo Request of length 256-bytes containing the Retry bit within the AAD set to 1. |
| MSDU8 | ICMP Echo Request of length 256-bytes containing the PwrMgt bit within the AAD set to 1. |
| MSDU9 | ICMP Echo Request of length 256-bytes containing the MoreData bit within the AAD set to 1. |
| MSDU10 | ICMP Echo Request of length 256-bytes containing the Protected Frame bit within the AAD set to 0. |
| MSDU11 | ICMP Echo Request of length 256-bytes containing the Sequence Number subfield bits within the AAD set to 1. |

**Procedure:**

*Part a: Valid Frame Decryption*

1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit MSDU1-4 to the DUT.
3. Observe transmissions from the DUT.

*Part b: Reserved Nonce Construction*

1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit MSDU5 to the DUT.
3. Observe transmissions from the DUT.

*Part c: AAD Field Masking*

1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake with the DUT.
2. Instruct the TS to transmit MSDU6-11 to the DUT.
3. Observe transmissions from the DUT.

**Observable Results:**
The DUT should:

a. properly decrypt and respond to MSDU1-4.
b. not be able to decrypt the frame and not respond to MSDU5.
c. not be able to decrypt the frame and not respond to MSDU6-11.

**Possible Problems:** None.

**Test # 1.2.4: CCMP PN Replay Protection**

**Purpose**: To verify that the DUT properly implements the PN packet replay procedure.

**References:**

[1] IEEE Std 802.11™-2012 Edition, Subclause 11.4.3.4 and 11.4.3.4.4

**Resource Requirements:**

- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** November 2008

**Discussion:** To effect replay detection, the receiver extracts the PN from the CCMP Header. This PN value shall be a 48-bit monotonically incrementing non-negative integer, initialized to one when the TK is initialized or refreshed. The PN values sequentially number each MPDU. A separate set of PN replay counters for each PTKSA, GTKSA, and STAKeySA shall exist, and be initialized to zero whenever the TK is reset for a peer.

A receiver shall discard an MSDU if the constituent MPDU PN values are not sequential. A receiver shall discard any MPDU that is received with a PN less than or equal to the replay counter, and then shall increment the value of dot11RSNAStatsCCMPReplays for the key.

Table 9 - Test Frame(s)

| Frame Label | Description | PN |
|---|---|---|
| MSDU1 | ICMP Echo Request of length 256-bytes containing the specified PN. | p |
| MSDU2 | ICMP Echo Request of length 256-bytes containing the specified PN and an invalid FCS. | p+2 |
| MSDU3 | ICMP Echo Request of length 256-bytes containing the specified PN. | p+1 |
| MSDU4 | ICMP Echo Request of length 256-bytes containing the specified PN and an invalid MIC. | p+3 |
| MSDU5 | ICMP Echo Request of length 256-bytes containing the specified PN. | p+2 |
| MSDU6 | ICMP Echo Request of length 256-bytes containing the specified PN. | p+3 |
| MSDU7a | Fragment 0 of an ICMP Echo Request of length 256-bytes containing the specified PN. | p+4 |
| MSDU7b | Fragment 1 of an ICMP Echo Request of length 256-bytes containing the specified PN. | p+4 |
| MSDU7c | Fragment 1 of an ICMP Echo Request of length 256-bytes containing the specified PN. | p+5 |
| MSDU8 | ICMP Echo Request of length 256-bytes containing the specified PN. | p+6 |
| MSDU9 | ICMP Echo Request of length 256-bytes containing the specified PN. | $(p+7)+ 2^{16}$ |
| MSDU10 | ICMP Echo Request of length 256-bytes containing the specified PN. | p+8 |
| MSDU11 | ICMP Echo Request of length 256-bytes containing the specified PN. | $(p+9)+ 2^{16}$ |

| | | |
|---|---|---|
| MSDU12 | ICMP Echo Request of length 256-bytes containing the specified PN. | $(p+10)+ 2^{47}$ |
| MSDU13 | ICMP Echo Request of length 256-bytes containing the specified PN. | $(p+11)+ 2^{16}$ |
| MSDU14 | ICMP Echo Request of length 256-bytes containing the specified PN. | $(p+11)+ 2^{47}$ |
| MSDU15 | ICMP Echo Request of length 256-bytes containing the specified PN. | $2^{48}-1$ |
| MSDU16 | ICMP Echo Request of length 256-bytes containing the specified PN. | 0 |
| MSDU17 | ARP Request of length 256-bytes to the broadcast address containing the specified PN. | p |
| MSDU18 | ARP Request of length 256-bytes to the broadcast address containing the specified PN. | p+1 |
| MSDU19 | ARP Request of length 256-bytes to the broadcast address containing the specified PN. | p+2 |

**Procedure:**

*Part a: Invalid PN Processing*

1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit MSDU1-MSDU5 to the DUT.
3. For each received MPDU, read the value of the PN replay counter field.
4. Observe transmissions from the DUT.

*Part b: PN Replay Processing*

1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit MSDU6-MSDU16 to the DUT.
3. For each received MPDU, read the value of the PN replay counter field.
4. Observe transmissions from the DUT.

*Part c: Separate PN replay counters per TK*

1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake
2. Instruct the TS to transmit MSDU1 and MSDU3 to the DUT.
3. Instruct the TS to transmit MSDU17-19 to the broadcast address.
4. Instruct the TS to transmit MSDU5 to the DUT.
5. Observe transmissions from the DUT.

**Observable Results:**

The DUT should:

a. only update its PN replay counter for valid CCMP MPDUs (MSDU1, MSDU3, MSDU5).
b. use the PN from the received MPDU to detect replayed frames and discard MSDUs whose constituent MPDU PN values are not sequential (MSDU7, MSDU10, MSDU13, MSDU16).
c. use the PN from the received MPDU to detect replayed frames for each unique TK independently (i.e. no frames are replayed).

**Possible Problems:** None.

### GROUP 3: EAPoL-Key Reception

**Scope:** The following tests cover MAC security operations specific to the reception of EAPoL-Key frames.

**Overview:** These tests are designed to verify that the DUT properly processes each field of an EAPoL-Key frame. The MAC security functions explored are defined in Clause 11 of IEEE Std 802.11™-2012.

The EAPoL-Key frame is used in the 4-way pairwise handshake, the group key handshake and the station key handshake. In addition to handling properly formatted EAPoL-Key frames, there are two rules that the DUT must follow for all other EAPoL-Key frames. EAPoL-Key frames containing invalid fields shall be silently discarded, and fields containing reserved bits shall ignore the values of those bits.

A variety of EAPoL-Key frames will be generated to test the DUT's conformance to these rules. These improperly formatted frames may be encountered as future revisions of the standard are ratified, other devices may implement proprietary protocols, another device may generate them due to failure of that device or they may be willfully generated in an active attack on the DUT.

The EAPoL-key frames that a DUT receives are dependent upon its role as either Supplicant or Authenticator. The EAPoL-Key frame types are listed in the following table.

Table 10 – EAPoL-Key frame types

| | EAPoL-Key frame | Supplicant | Authenticator |
|---|---|---|---|
| 1 | Pairwise message 1 | RX | |
| 2 | Pairwise message 3 | RX | |
| 3 | Group key message 1 | RX | |
| 4* | STAKey message 1 | RX | |
| 5 | Pairwise message 2 | | RX |
| 6 | Pairwise message 4 | | RX |
| 7 | Group key message 2 | | RX |
| 8 | Re-key request | | RX |
| 9* | STAKey Request | | RX |
| 10* | STAKey message 2 | | RX |

**Test Setup:** These tests should be run with CCMP encryption using a PSK and all other settings as default unless specified.

**Test # 1.3.1: Descriptor Type Processing**

**Purpose:** To verify that the DUT can properly process the Descriptor Type field present in EAPoL-key frames.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2, 11.6.6.2 and 11.6.6.4
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** November 2008

**Discussion:** The Descriptor Type field is one octet in length, taken to represent an unsigned binary number. The value defines the type of the Key Descriptor, which in turn defines how the Descriptor Body is used and interpreted. For 802.11 the Descriptor Type is 2.

Table 11 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | EAPoL-Key Message 1 containing the Descriptor Type set to 1. |
| MSDU2 | EAPoL-Key Message 1 containing the Descriptor Type set to 0. |
| MSDU3 | EAPoL-Key Message 1 containing the Descriptor Type set to 3. |
| MSDU4 | EAPoL-Key Message 3 containing the Descriptor Type set to 1. |
| MSDU5 | EAPoL-Key Message 3 containing the Descriptor Type set to 0. |
| MSDU6 | EAPoL-Key Message 3 containing the Descriptor Type set to 3. |

**Procedure:**
*Part a: Valid Descriptor Type*
1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
3. Observe transmissions from the DUT.

*Part b: Invalid Descriptor Type*
1. Instruct the TS to use MSDU1 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Repeat steps 1-5 with MSDU2-6.
7. Observe transmissions from the DUT.

**Observable Results:**
The DUT should:
a. successfully complete the 4-way handshake.
b. silently discard invalid EAPoL-Key frames. (MSDU1-6)

**Possible Problems:** None.

**Test # 1.3.2: Key Information Field Processing**

**Purpose:** To verify that the DUT can properly process the Key Information field present in EAPoL-key frames.

**References:**

[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2, 11.6.6.2 and 11.6.6.4
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**

- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** November 2008

**Discussion:** The Key Information Field is 2 octets in length and specifies characteristics of the key. The Key Information Field is comprised of the following fields, Key Descriptor Version, Key Type, Reserved, Install, Key MIC, Secure, Error, Request, Encrypted Key Data, SMK Message, and another Reserved. The values that should be contained within each field of the Key Information Field are specified within [1].

Table 12 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | EAPoL-Key Message 1 containing reserved bit 4 of the Key Information field set. |
| MSDU2 | EAPoL-Key Message 1 containing reserved bit 5 of the Key Information field set. |
| MSDU3 | EAPoL-Key Message 1 containing reserved bit 13 of the Key Information field set. |
| MSDU4 | EAPoL-Key Message 1 containing reserved bit 14 of the Key Information field set. |
| MSDU5 | EAPoL-Key Message 1 containing reserved bit 15 of the Key Information field set. |
| MSDU6 | EAPoL-Key Message 3 containing reserved bit 4 of the Key Information field set. |
| MSDU7 | EAPoL-Key Message 3 containing reserved bit 5 of the Key Information field set. |
| MSDU8 | EAPoL-Key Message 3 containing reserved bit 13 of the Key Information field set. |
| MSDU9 | EAPoL-Key Message 3 containing reserved bit 14 of the Key Information field set. |
| MSDU10 | EAPoL-Key Message 3 containing reserved bit 15 of the Key Information field set. |
| MSDU11 | EAPoL-Key Message 1 containing the Key Descriptor Version type set to 0. |
| MSDU12 | EAPoL-Key Message 1 containing the Key Descriptor Version type set to 3. |
| MSDU13 | EAPoL-Key Message 1 containing the Key Descriptor Version type set to 4. |
| MSDU14 | EAPoL-Key Message 1 containing the Key Descriptor Version type set to 5. |
| MSDU15 | EAPoL-Key Message 1 containing the Key Descriptor Version type set to 6. |
| MSDU16 | EAPoL-Key Message 1 containing the Key Descriptor Version type set to 7. |
| MSDU17 | EAPoL-Key Message 3 containing the Key Descriptor Version type set to 0. |
| MSDU18 | EAPoL-Key Message 3 containing the Key Descriptor Version type set to 3. |
| MSDU19 | EAPoL-Key Message 3 containing the Key Descriptor Version type set to 4. |
| MSDU20 | EAPoL-Key Message 3 containing the Key Descriptor Version type set to 5. |
| MSDU21 | EAPoL-Key Message 3 containing the Key Descriptor Version type set to 6. |
| MSDU22 | EAPoL-Key Message 3 containing the Key Descriptor Version type set to 7. |
| MSDU23 | EAPoL-Key Message 3 containing the Error bit set. |

**Procedure:**

*Part a: Reserved Bits Processing*

1. Instruct the TS to use MSDU1 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Repeat steps 1-5 with MSDU2-10.
7. Observe transmissions from the DUT.

*Part b: Key DescriptorVersion Processing*

1. Instruct the TS to use MSDU11 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Repeat steps 1-5 with MSDU12-22.
7. Observe transmissions from the DUT.

*Part c: Error Bit Processing*

1. Instruct the TS to use MSDU23 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Observe transmissions from the DUT.

**Observable Results:**

The DUT should:

a. ignore Key Information field reserved bits (MSDU1-10) and successfully complete the 4-way handshake.
b. silently discard all EAPoL-Key frames containing invalid Key Descriptor values (MSDU11-22).
c. silently discard all incorrectly formatted EAPoL-Key frames (MSDU23).

**Possible Problems:** None.

**Test # 1.3.3: Key Length Field Processing**

**Purpose:** To verify that the DUT can properly process the Key Length field present in EAPoL-key frames.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2, 11.6.6.2 and 11.6.6.4
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** June 2007

**Discussion:** The Key Length Field is 2 octets in length, represented as an unsigned binary number. The value defines the length, in octets, of the PTK to configure into IEEE Std 802.11.

Table 13 – Key Lengths

| Cipher Suite | CCMP | TKIP | WEP40 | WEP104 |
|---|---|---|---|---|
| Key Length (octets) | 16 | 32 | 5 | 13 |

Table 14 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | EAPoL-Key Message 1 containing a Key Length field of 0. |
| MSDU2 | EAPoL-Key Message 1 containing a Key Length field of 32. |
| MSDU3 | EAPoL-Key Message 1 containing a Key Length field of 16. |
| MSDU4 | EAPoL-Key Message 1 containing a Key Length field of 13. |
| MSDU5 | EAPoL-Key Message 1 containing a Key Length field of 5. |
| MSDU6 | EAPoL-Key Message 3 containing a Key Length field of 0. |
| MSDU7 | EAPoL-Key Message 3 containing a Key Length field of 32. |
| MSDU8 | EAPoL-Key Message 3 containing a Key Length field of 16. |
| MSDU9 | EAPoL-Key Message 3 containing a Key Length field of 13. |
| MSDU10 | EAPoL-Key Message 3 containing a Key Length field of 5. |

**Procedure:**
1. Instruct the TS to use MSDU1 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way Handshake using the default EAPoL-key frame values.
6. Repeat steps 1-5 with MSDU2-10.
7. Observe transmissions from the DUT.

**Observable Results:**
The DUT should:

a. silently discard all EAPoL-Key Message 1 frames with invalid Key Lengths (Valid values: CCMP:16, TKIP:32. WEP-40:5, WEP-104:13).
b. silently discard all EAPoL-Key Message 3 frames with invalid Key Lengths (Valid values: CCMP:16, TKIP:32. WEP-40:5, WEP-104:13).

**Possible Problems:** None.

**Test # 1.3.4: Key Replay Counter Processing**

**Purpose:** To verify that the DUT can properly process the Key Replay Counter field present in EAPoL-key frames.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2 , 11.6.6.3 and 11.6.6.5
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** November 2008

**Discussion:** The Key Replay Counter Field is 8 octets, represented as an unsigned binary number, and is initialized to 0 when the PMK is established. The Supplicant shall use the key replay counter in the received EAPOL-Key frame when responding to an EAPOL-Key frame. It carries a sequence number that the protocol uses to detect replayed EAPOL-Key frames. The Supplicant and Authenticator shall track the key replay counter per security association. The Key Replay Counter shall be initialized to 0 on (re)association. The Authenticator shall increment the key replay counter on each successive EAPOL-Key frame. When replying to a message from the Authenticator, the Supplicant shall use the Key Replay Counter field value from the last valid EAPOL-Key frames received from the Authenticator. The Authenticator should use the key replay counter to identify invalid messages to silently discard. The Supplicant should also use the Key Replay Counter and ignore EAPOL-Key frames with a Key Replay Counter field value smaller than or equal to any received in a valid message. The local Key Replay Counter field should not be updated until the after EAPOL-Key MIC is checked and is valid. In other words, the Supplicant never updates the Key Replay Counter field for Message 1 in the 4-Way Handshake, as it includes no MIC. This implies the Supplicant must allow for retransmission of Message 1 when checking for the key replay counter of Message 3.

Table 15 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | EAPoL-Key Message 1 containing a Key Replay Counter of p. |
| MSDU2 | EAPoL-Key Message 3 containing a Key Replay Counter of p+3. |
| MSDU3 | EAPoL-Key Message 1 containing a Key Replay Counter of p. |
| MSDU4 | EAPoL-Key Message 3 containing a Key Replay Counter of p. |

**Procedure:**
*Part a: Non-Sequential Counter*
1. Instruct the TS to use MSDU1-2 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Observe transmissions from the DUT.

*Part b: Non-incrementing Counter*
1. Instruct the TS to use MSDU3-4 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.

4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Observe transmissions from the DUT.

**Observable Results:**

a. The DUT should:
   - use the Key Replay Counter from the received EAPoL-Key frame when responding (Message 2, Message 4).
   - successfully complete the 4-way handshake.

b. The DUT should:
   - silently discard any EAPoL-Key frames received with a Key Replay Counter field that is less than or equal to any received in a valid message.
   - not successfully complete the 4-way handshake.

**Possible Problems:** None.

**Test # 1.3.5: Key Nonce Field Processing**

**Purpose:** To verify that the DUT can properly process the Key Nonce field present in EAPoL-key frames.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2, 11.6.5, 11.6.6.2 and 11.6.6.4
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** June 2007

**Discussion:** The Key Nonce Field is 32 octets. It conveys the ANonce from the Authenticator and the SNonce from the Supplicant. [1] states that the ANonce and SNonce shall be random or pseudo-random values that shall not repeat for any security association. Choosing the nonces randomly helps prevent precomputation attacks. With unpredictable nonces, a man-in- the-middle attack that uses the Supplicant to precompute messages to attack the Authenticator cannot progress beyond Message 2, and a similar attack against the Supplicant cannot progress beyond Message 3.

Table 16 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | EAPoL-Key Message 1 containing a Key Nonce of 0. |
| MSDU2 | EAPoL-Key Message 3 containing a Key Nonce of 0. |

**Procedure:**
1. Instruct the TS to use MSDU1 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Repeat steps 1-5 with MSDU2.
7. Observe transmissions from the DUT.

**Observable Results:**
a. The DUT should not successfully complete the 4-way handshake.

**Possible Problems:** None.

**Test # 1.3.6: EAPoL-Key IV Field Processing**

**Purpose:** To verify that the DUT can properly process the Key IV field present in EAPoL-key frames.

**References:**

[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2, 11.6.6.2 and 11.6.6.4
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**

- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** June 2007

**Discussion:** The Key IV field is 16 octets. It contains the IV used with the KEK. It shall contain 0 when an IV is not required. It should be initialized by taking the current value of the global key counter and incrementing it. Note that only the lower 16 octets of the counter value will be used.

Table 17 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | EAPoL-Key Message 1 using the CCMP cipher containing a non-zero EAPoL-Key IV. |
| MSDU2 | EAPoL-Key Message 3 using the CCMP cipher containing a non-zero EAPoL-Key IV. |

**Procedure:**

1. Configure the DUT to use the CCMP cipher.
2. Instruct the TS to use MSDU1 within the 4-way handshake with the DUT.
3. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
4. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
5. Instruct the TS to transmit a deauthentication frame to the DUT.
6. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
7. Repeat steps 2-6 with MSDU2.
8. Observe transmissions from the DUT.

**Observable Results:**

a. The DUT should silently discard all EAPoL-Key frames containing invalid EAPoL-Key IV fields.

**Possible Problems:** None.

**Test # 1.3.7: Key RSC Field Processing**

**Purpose:** To verify that the DUT can properly process the Key RSC field present in EAPoL-key frames.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2, 11.6.6.2 and 11.6.6.4
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** June 2007

**Discussion:** The Key RSC Field is 8 octets in length. It contains the RSC for the GTK being installed in IEEE Std 802.11. It is used in Message 3 of the 4-Way Handshake and Message 1 of the Group Key Handshake, where it is used to synchronize the IEEE 802.11 replay state. It may also be used in the Michael MIC Failure Report frame, to report the TSC field value of the frame experiencing a MIC failure. It shall contain 0 in other messages. The Key RSC field gives the current message number for the GTK, to allow a STA to identify replayed MPDUs. If the Key RSC field value is less than 8 octets in length, the remaining octets shall be set to 0. The least significant octet of the TSC or PN should be in the first octet of the Key RSC field.

Table 18 – Key RSC Field

| Key RSC0 | Key RSC1 | Key RSC2 | Key RSC3 | Key RSC4 | Key RSC5 | Key RSC6 | Key RSC7 |
|---|---|---|---|---|---|---|---|
| PN0 | PN1 | PN2 | PN3 | PN4 | PN5 | 0 | 0 |

Table 19 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | EAPoL-Key Message 1 containing a non-zero Key RSC. |
| MSDU2 | EAPoL-Key Message 3 containing a KeyRSC 6 and KeyRSC 7 set to 0xff. |
| MSDU3 | EAPoL-Key Message 3 containing a KeyRSC of p. |
| MSDU4 | ARP Request of length 512-bytes with the PN set to p-3. |

**Procedure:**
*Part a: Invalid Non-Zero Key RSC*
1. Instruct the TS to use MSDU1 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Observe transmissions from the DUT.

*Part b: Unused Octet Processing*
1. Instruct the TS to use MSDU2 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.

5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Observe transmissions from the DUT.

*Part c: Valid RSC Processing*

1. Instruct the TS to use MSDU3 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit MSDU4 to the broadcast address requesting the DUT's address.
5. Instruct the TS to transmit a deauthentication frame to the DUT.
6. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
7. Observe transmissions from the DUT.

**Observable Results:**

a. The DUT should silently discard all EAPoL-Key frames containing invalid Key RSCs.
b. INFORMATIVE: The DUT may ignore the unused octets of the Key RSC.
c. The DUT should discard MSDU4 as a replayed frame.

**Possible Problems:** None.

**Test # 1.3.8: Reserved Octets Processing**

**Purpose:** To verify that the DUT can properly process the Reserved Octets 61-68 present in EAPoL-key frames of Descriptor Type 2.

**References:**

[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**

- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** June 2007

**Discussion:** [1] states that reserved bits should be set to 0 upon transmission and ignored upon reception.

Table 20 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | EAPoL-Key Message 1 containing all bits set within reserved octets 61-68. |
| MSDU2 | EAPoL-Key Message 3 containing all bits set within reserved octets 61-68. |

**Procedure:**

1. Instruct the TS to use MSDU1 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Repeat steps 1-5 with MSDU2.
7. Observe transmissions from the DUT.

**Observable Results:**

a. The DUT should ignore reserved bits set within EAPoL-Key Messages.

**Possible Problems:** None.

**Test # 1.3.9: Key MIC Field Processing**

**Purpose:** To verify that the DUT can properly process the Key MIC field present in EAPoL-key frames.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2, 11.6.6.2 and 11.6.6.4
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** November 2008

**Discussion:** The Key MIC Field is 16 octets in length when the Key Descriptor Version subfield is 1 or 2. The EAPOL-Key MIC is a MIC of the EAPOL-Key frames, from and including the EAPOL protocol version field to and including the Key Data field, calculated with the Key MIC field set to 0. If the Encrypted Key Data subfield (of the Key Information field) is set, the Key Data field is encrypted prior to computing the MIC.

Table 21 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | EAPoL-Key Message 1 containing a non-zero Key MIC. |
| MSDU2 | EAPoL-Key Message 3 containing a Key MIC of 0. |
| MSDU3 | EAPoL-Key Message 3 containing a Key MIC of one less than calculated. |

**Procedure:**
1. Instruct the TS to use MSDU1 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Repeat steps 1-5 with MSDU2-3.
7. Observe transmissions from the DUT.

**Observable Results:**
a. The DUT should silently discard all EAPoL-Key frames with incorrect Key MIC fields (MSDU2, MSDU3).

**Possible Problems:** None.

**Test # 1.3.10: Key Data Length Field Processing**

**Purpose:** To verify that the DUT can properly process the Key Data Length present in EAPoL-key frames.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** June 2007

**Discussion:** The Key Data Length Field is 2 octets in length, taken to represent an unsigned binary number. This represents the length of the Key Data field in octets. If the Encrypted Key Data subfield (of the Key Information field) is set, the length is the length of the Key Data field after encryption, including any padding.

Table 22 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | EAPoL-Key Message 1 containing the most significant bit of the Key Data Length field set. |
| MSDU2 | EAPoL-Key Message 3 containing the most significant bit of the Key Data Length field set. |
| MSDU3 | EAPoL-Key Message 3 containing 2 octets of padding in the Key Data Length field. |
| MSDU4 | EAPoL-Key Message 3 containing 8 octets of padding in the Key Data Length field. |
| MSDU5 | EAPoL-Key Message 3 containing 15 octets of padding in the Key Data Length field. |

**Procedure:**
*Part a: Incorrect Key Data Length*
1. Instruct the TS to use MSDU1 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Repeat steps 1-5 with MSDU2.
7. Observe transmissions from the DUT.

*Part b: Key Data Padding*
1. Instruct the TS to use MSDU3 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Repeat steps 1-5 with MSDU4-5.
7. Observe transmissions from the DUT.

**Observable Results:**
The DUT should:

a. receive EAPoL-Key frames with invalid Key Data Lengths without failure and discard the frame.
b. discard any pad bytes appended to an Encrypted Key Data field and included in the Key Data Length field.

**Possible Problems:** None.

**Test # 1.3.11: Key Data Field Processing (Pairwise Message1)**

**Purpose:** To verify that the DUT can properly process encrypted Key Data present in EAPoL-key frames of the 4-way handshake.

**References:**

[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**

- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** November 2008

**Discussion:** The Key Data Field is a variable-length field that is used to include any additional data required for the key exchange that is not included in the fields of the EAPOL-Key frame. The additional data may be zero or more information element(s) (such as the RSN information element) and zero or more key data cryptographic encapsulation(s) (KDEs) (such as GTK(s) or PMKID(s)). Information elements sent in the Key Data field include the Element ID and Length subfields. KDEs shall be encapsulated.

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

Table 23 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | EAPoL-Key Message 1 containing the correct PMKID KDE and a reserved IE. |
| MSDU2 | EAPoL-Key Message 1 containing the correct PMKID KDE and an unknown vendor KDE. |
| MSDU3 | EAPoL-Key Message 1 containing the correct PMKID KDE with 7 bytes of padding. |
| MSDU4 | EAPoL-Key Message 1 containing the correct PMKID KDE but an incorrect PMKID (0x00). |
| MSDU5 | EAPoL-Key Message 1 containing the correct PMKID KDE but an incorrect PMKID (0xFF). |

**Procedure:**

1. Instruct the TS to use MSDU1 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Repeat steps 1-5 with MSDU2-5.
7. Observe transmissions from the DUT.

**Observable Results:**
The DUT:

a. should ignore any IEs or KDEs that are unknown (MSDU1, MSDU2).
b. should ignore any pad bytes appended to an Encrypted Key Data field (MSDU3).
c. should accept the incorrect PMKIDs (MSDU4, MSDU5).

**Possible Problems:** None.

**Test # 1.3.12: Key Data Field Processing (Pairwise Message3)**

**Purpose:** To verify that the DUT can properly process encrypted Key Data present in EAPoL-key frames of the 4-way handshake.

**References:**

[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**

- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.

**Last Modification:** November 2008

**Discussion:** The Key Data Field is a variable-length field that is used to include any additional data required for the key exchange that is not included in the fields of the EAPOL-Key frame. The additional data may be zero or more information element(s) (such as the RSN information element) and zero or more key data cryptographic encapsulation(s) (KDEs) (such as GTK(s) or PMKID(s)). Information elements sent in the Key Data field include the Element ID and Length subfields. KDEs shall be encapsulated.

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

Table 24 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | EAPoL-Key Message 3 containing the correct RSN IE from Beacons and a GTK KDE. |
| MSDU2 | EAPoL-Key Message 3 containing the correct RSN IE from Beacons and no GTK KDE. |
| MSDU3 | EAPoL-Key Message 3 containing an empty payload. |
| MSDU4 | EAPoL-Key Message 3 containing a modified RSN IE from Beacons and a GTK KDE. |
| MSDU5 | EAPoL-Key Message 3 containing the correct RSN IE from Beacons, a reserved IE and a GTK KDE. |
| MSDU6 | EAPoL-Key Message 3 containing the correct RSN IE from Beacons, a GTK KDE, and a reserved KDE. |

**Procedure:**

1. Instruct the TS to use MSDU1 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Repeat steps 1-5 with MSDU2-6.
7. Observe transmissions from the DUT.

**Observable Results:**
The DUT should:

a. discard encrypted Key Data fields that do not contain a GTK KDE (MSDU2).
b. discard EAPoL-Key frames if the RSN IE contained in the Key Data field does not bitwise match the RSN IE transmitted by the TS in its Beacon and Probe Response frames (MSDU3, MSDU4).
c. ignore any extraneous IEs or unknown KDEs (MSDU5, MSDU6).

**Possible Problems:** None.

### GROUP 4: EAPoL-Key Transmission

**Scope:** The following tests cover MAC security operations specific to the transmission of EAPoL-Key frames.

**Overview:** These tests are designed to verify that the DUT properly transmits each field of an EAPoL-Key frame. The MAC security functions explored are defined in Clause 11 of IEEE Std 802.11™-2012.

**Test Setup:** These tests should be run with CCMP encryption using a PSK and all other settings as default unless specified.

**Test # 1.4.1: Descriptor Type Field Formatting**

**Purpose:** To verify that the DUT uses the proper Descriptor Type in EAPoL-key frames.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2, 11.6.6.3 and 11.6.6.5
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.
- A wired station on the DS that can respond to ICMP Echo Request frames.

**Last Modification:** June 2007

**Discussion:** The Descriptor Type field is one octet in length, taken to represent an unsigned binary number. The value defines the type of the Key Descriptor, which in turn defines how the Descriptor Body is used and interpreted. For 802.11 the Descriptor Type is 2.

**Procedure:**
1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
3. Instruct the TS to transmit a deauthentication frame to the DUT.
4. Observe transmissions from the DUT.

**Observable Results:**
a. The DUT should transmit EAPoL-Key frames with a Descriptor Type of 2.

**Possible Problems:** None.

**Test # 1.4.2: Key Information Field Formatting**

**Purpose:** To verify that the DUT properly formats the Key Information field present in EAPoL-key frames.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2, 11.6.6.3 and 11.6.6.5
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.
- A wired station on the DS that can respond to ICMP Echo Request frames.

**Last Modification:** June 2007

**Discussion:** The Key Information Field is 2 octets in length and specifies characteristics of the key. The Key Information Field is comprised of the following fields, Key Descriptor Version, Key Type, Reserved, Install, Key MIC, Secure, Error, Request, Encrypted Key Data, SMK Message, and another Reserved. The values that should be contained within each field of the Key Information Field are specified within [1].

**Procedure:**
1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
3. Instruct the TS to transmit a deauthentication frame to the DUT.
4. Observe transmissions from the DUT.

**Observable Results:**
The DUT should:
a. set the Key Descriptor Version to 2.
b. set the following bits in the following frames:

| Frame | Result |
|---|---|
| Pairwise Key Message 2 | Key Type and Key MIC subfields should be set to 1. |
| Pairwise Key Message 4 | Key Type, the Key MIC and the Secure subfields should all be set to 1. |

c. All other bits in the Key Info field should be set to 0.

**Possible Problems:** None.

**Test # 1.4.3: Key Length Field Formatting**

**Purpose:** To verify that the DUT properly formats the Key Length field present in EAPoL-key frames.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2, 11.6.6.3 and 11.6.6.5
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.
- A wired station on the DS that can respond to ICMP Echo Request frames.

**Last Modification:** June 2007

**Discussion:** The Key Length Field is 2 octets in length, represented as an unsigned binary number. The value defines the length, in octets, of the PTK to configure into IEEE Std 802.11.

Table 25 – Key Lengths

| Cipher Suite | CCMP | TKIP | WEP40 | WEP104 |
|---|---|---|---|---|
| Key Length (octets) | 16 | 32 | 5 | 13 |

**Procedure:**
1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
3. Instruct the TS to transmit a deauthentication frame to the DUT.
4. Observe transmissions from the DUT.

**Observable Results:**
a. The DUT should transmit EAPoL-Key Message 2 and 4 with a Key Length Field of 0.

**Possible Problems:** None.

**Test # 1.4.4: Key Replay Counter Formatting**

**Purpose:** To verify that the DUT properly formats the Key Replay Counter field present in EAPoL-key frames.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2, 11.6.6.3 and 11.6.6.5
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.
- A wired station on the DS that can respond to ICMP Echo Request frames.

**Last Modification:** June 2007

**Discussion:** The Key Replay Counter Field is 8 octets, represented as an unsigned binary number, and is initialized to 0 when the PMK is established. The Supplicant shall use the key replay counter in the received EAPOL-Key frame when responding to an EAPOL-Key frame. It carries a sequence number that the protocol uses to detect replayed EAPOL-Key frames. The Supplicant and Authenticator shall track the key replay counter per security association. The Key Replay Counter shall be initialized to 0 on (re)association. The Authenticator shall increment the key replay counter on each successive EAPOL-Key frame. When replying to a message from the Authenticator, the Supplicant shall use the Key Replay Counter field value from the last valid EAPOL-Key frames received from the Authenticator. The Authenticator should use the key replay counter to identify invalid messages to silently discard. The Supplicant should also use the Key Replay Counter and ignore EAPOL-Key frames with a Key Replay Counter field value smaller than or equal to any received in a valid message. The local Key Replay Counter field should not be updated until the after EAPOL-Key MIC is checked and is valid. In other words, the Supplicant never updates the Key Replay Counter field for Message 1 in the 4-Way Handshake, as it includes no MIC. This implies the Supplicant must allow for retransmission of Message 1 when checking for the key replay counter of Message 3.

Table 26 - Test Frame(s)

| Frame Label | Description |
|---|---|
| MSDU1 | EAPoL-Key Message 1 containing a Key Replay Counter of 0. |
| MSDU2 | EAPoL-Key Message 1 containing a Key Replay Counter of 5. |
| MSDU3 | EAPoL-Key Message 3 containing a Key Replay Counter of 1. |
| MSDU4 | EAPoL-Key Message 3 containing a Key Replay Counter of 2. |

**Procedure:**
*Part a: Correct Initialization*
1. Instruct the TS to use MSDU1 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Observe transmissions from the DUT.

*Part b: Incorrect Initialization*
1. Instruct the TS to use MSDU2 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.

3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Observe transmissions from the DUT.

*Part c: Incrementing by One*

1. Instruct the TS to use MSDU3 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Observe transmissions from the DUT.

*Part d: Incrementing by Two*

1. Instruct the TS to use MSDU4 within the 4-way handshake with the DUT.
2. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
3. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
4. Instruct the TS to transmit a deauthentication frame to the DUT.
5. Allow the DUT to authenticate, associate, and successfully complete a 4-way handshake using the default EAPoL-key frame values.
6. Observe transmissions from the DUT.

**Observable Results:**

a-d. The DUT should always use the Key Replay Counter from the received EAPoL-Key frame when responding.

**Possible Problems:** None.

**Test # 1.4.5: Key Nonce Field Formatting**

**Purpose:** To verify that the DUT properly formats the Key Nonce field present in EAPoL-key frames.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2, 11.6.5, 11.6.6.3, and 11.6.6.5
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.
- A wired station on the DS that can respond to ICMP Echo Request frames.

**Last Modification:** June 2007

**Discussion:** The Key Nonce Field is 32 octets. It conveys the ANonce from the Authenticator and the SNonce from the Supplicant. [1] states that the ANonce and SNonce shall be random or pseudo-random values that shall not repeat for any security association. Choosing the nonces randomly helps prevent precomputation attacks. With unpredictable nonces, a man-in- the-middle attack that uses the Supplicant to precompute messages to attack the Authenticator cannot progress beyond Message 2, and a similar attack against the Supplicant cannot progress beyond Message 3.

**Procedure:**
1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
3. Instruct the TS to transmit a deauthentication frame to the DUT.
4. Observe transmissions from the DUT.

**Observable Results:**
The DUT should:
a. use a random or pseudo-random value for the Key Nonce within EAPoL-Key Message 2.
b. use the value of zero for the Key Nonce within EAPoL-Key Message 4.

**Possible Problems:** None.

**Test # 1.4.6: Key IV Field Formatting**

**Purpose:** To verify that the DUT properly formats the Key IV field present in EAPoL-key frames.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2, 11.6.6.3 and 11.6.6.5
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.
- A wired station on the DS that can respond to ICMP Echo Request frames.

**Last Modification:** June 2007

**Discussion:** This field is 16 octets. It contains the IV used with the KEK. It shall contain 0 when an IV is not required. It should be initialized by taking the current value of the global key counter and then incrementing the counter. Note that only the lower 16 octets of the counter value will be used.

**Procedure:**
1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
3. Instruct the TS to transmit a deauthentication frame to the DUT.
4. Observe transmissions from the DUT.

**Observable Results:**
a. The DUT should transmit EAPoL-Key Message 2 and 4 with an EAPoL-Key IV value of zero.

**Possible Problems:** None.

**Test # 1.4.7: Key RSC Field Formatting**

**Purpose:** To verify that the DUT properly formats the Key RSC field present in EAPoL-key frames.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2, 11.6.6.3 and 11.6.6.5
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.
- A wired station on the DS that can respond to ICMP Echo Request frames.

**Last Modification:** June 2007

**Discussion:** The Key RSC Field is 8 octets in length. It contains the RSC for the GTK being installed in IEEE Std 802.11. It is used in Message 3 of the 4-Way Handshake and Message 1 of the Group Key Handshake, where it is used to synchronize the IEEE 802.11 replay state. It may also be used in the Michael MIC Failure Report frame, to report the TSC field value of the frame experiencing a MIC failure. It shall contain 0 in other messages. The Key RSC field gives the current message number for the GTK, to allow a STA to identify replayed MPDUs. If the Key RSC field value is less than 8 octets in length, the remaining octets shall be set to 0. The least significant octet of the TSC or PN should be in the first octet of the Key RSC field.

**Procedure:**
1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
3. Instruct the TS to transmit a deauthentication frame to the DUT.
4. Observe transmissions from the DUT.

**Observable Results:**
a. The DUT should transmit EAPoL-Key Message 2 and 4 with a RSC value of zero.

**Possible Problems:** None.

**Test # 1.4.8: Reserved Octets Field Formatting**

**Purpose:** To verify that the DUT properly formats reserved octets present in EAPoL-key frames.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2, 11.6.6.3 and 11.6.6.5
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.
- A wired station on the DS that can respond to ICMP Echo Request frames.

**Last Modification:** June 2007

**Discussion:** [1] states that reserved bits should be set to 0 upon transmission and ignored upon reception.

**Procedure:**
1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
3. Instruct the TS to transmit a deauthentication frame to the DUT.
4. Observe transmissions from the DUT.

**Observable Results:**
a. The DUT should transmit EAPoL-Key Message 2 and 4 with all reserved field values set to zero.

**Possible Problems:** None.

**Test # 1.4.9: Key MIC Field Formatting**

**Purpose:** To verify that the DUT properly formats the MIC field present in EAPoL-key frames.

**References:**
[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2, 11.6.6.3 and 11.6.6.5
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**
- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.
- A wired station on the DS that can respond to ICMP Echo Request frames.

**Last Modification:** June 2007

**Discussion:** The Key MIC Field is 16 octets in length when the Key Descriptor Version subfield is 1 or 2. The EAPOL-Key MIC is a MIC of the EAPOL-Key frames, from and including the EAPOL protocol version field to and including the Key Data field, calculated with the Key MIC field set to 0. If the Encrypted Key Data subfield (of the Key Information field) is set, the Key Data field is encrypted prior to computing the MIC.

**Procedure:**
1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
3. Instruct the TS to transmit a deauthentication frame to the DUT.
4. Observe transmissions from the DUT.

**Observable Results:**
a. The DUT should set the Key MIC to the correct calculated value in EAPoL-Key Message 2 and 4.

**Possible Problems:** None.

**Test # 1.4.10: Key Data & Length Field Formatting**

**Purpose:** To verify that the DUT properly formats the Key Data Length and Key Data fields present in EAPoL-key frames.

**References:**

[1] IEEE Std 802.11™-2012 Edition, Subclause 11.6.2, 11.6.6.3 and 11.6.6.5
[2] IEEE Std 802.1X™-2004 Edition, Subclause 7.6

**Resource Requirements:**

- A TS that is capable of transmitting user defined MAC frames and does not participate in the MAC protocol.
- A monitor configured for capturing and analyzing MAC frames.
- A wired station on the DS that can respond to ICMP Echo Request frames.

**Last Modification:** June 2007

**Discussion:** The Key Data Length Field is 2 octets in length, taken to represent an unsigned binary number. This represents the length of the Key Data field in octets. If the Encrypted Key Data subfield (of the Key Information field) is set, the length is the length of the Key Data field after encryption, including any padding.

The Key Data Field is a variable-length field that is used to include any additional data required for the key exchange that is not included in the fields of the EAPOL-Key frame. The additional data may be zero or more information element(s) (such as the RSN information element) and zero or more key data cryptographic encapsulation(s) (KDEs) (such as GTK(s) or PMKID(s)). Information elements sent in the Key Data field include the Element ID and Length subfields. KDEs shall be encapsulated.

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

**Procedure:**

1. Allow the DUT to authenticate, associate, and successfully complete the 4-way Handshake.
2. Instruct the TS to transmit multiple ICMP Echo Requests to the DUT.
3. Instruct the TS to transmit a deauthentication frame to the DUT.
4. Observe transmissions from the DUT.

**Observable Results:**

a. The DUT should transmit the Key Data field within EAPoL-Key Message 2 with an RSN IE that is a bit-wise match of the RSN IE found in Association Request frames transmitted by the DUT.

**Possible Problems:** None.

## Appendix A: 802.11 EAPoL-Key values

| | 4-Way Msg 1 | 4-Way Msg 2 | 4-Way Msg 3 | 4-Way Msg 4 | Group Key Msg 1 | Group Key Msg 2 | STAKey Request | STAKey Msg 1 | STAKey Msg 2 |
|---|---|---|---|---|---|---|---|---|---|
| Clause | **8.5.3.1** | **8.5.3.2** | **8.5.3.3** | **8.5.3.4** | **8.5.4.1** | **8.5.4.2** | **8.5.5.1** | **8.5.5.2** | **8.5.5.3** |
| Tx by | Auth | Supp | Auth | Supp | Auth | Supp | Supp | Auth | Supp |
| | Descriptor Type | | | | | | | | |
| 802.11 | 2 | | | | | | | | |
| **WPA1** | **254** | | | | | | | | |
| Key Info | *(rsvd 15\|14\|13, enc key data12, request11, error10, secure9, mic8 \|\| ack7, install6, rsvd5\|4, type3, desc2\|1\|0)* | | | | | | | | |
| CCMP | 0x008a | 0x010a | 0x13ca/138a[1] | 0x030a | 0x1382 | **0x0302** | 0x0b02 | 0x13c2 | **0x0302** |
| TKIP | 0x0089 | **0x0109** | 0x13c9/1389[1] | 0x0309 | 0x1381 | **0x0301** | 0x0b01 | 0x13c1 | **0x0301** |
| **WPA1** | | | | 0x0309 **0x0109** | | | | | |
| | Key Length | | | | | | | | |
| CCMP | 16 | 0 | 16 | 0 | 0 **(WPA1: 16, 32, 5, 13)** | 0 | 0 | 16 | 0 |
| TKIP | 32 | | 32 | | | | | 32 | |
| WEP40 | 5 | | 5 | | | | | 5 | |
| WEP104 | 13 | | 13 | | | | | 13 | |
| | Key Replay Counter | | | | | | | | |
| | *n* | | *n+1* | | *n*+2 | | *r* | *n*+3 [2] | *n*+3 |
| Key Nonce | ANonce | SNonce | same as Msg 1 | 0 | 0 | | 0 | | |
| | Key IV | | | | | | | | |
| CCMP | 0 | | | | | | | | |
| TKIP | 0 | 0 | **random** | 0 | **random** | 0 | 0 | **random** | **random** |
| | Key RSC | | | | | | | | |
| | 0 | 0 | starting seq number Authenticator will use in MPDUs protected by GTK | 0 | last transmit sequence number for the GTK | 0 | 0 | | |
| Key MIC | 0 | MIC (KCK, EAPOL) | | | | | MIC (initiator STA's KCK, EAPOL) | MIC (Peer STA's KCK, EAPOL) | |
| Key Data Length | 22 | length of included RSN IE | Length of included RSN IEs and GTK | 0 | Length of Key Data field | 0 | Length of Key Data field | Length of Key Data field | Length of Key Data field |
| Key Data | PMKID for the PMK being used during this exchange. | the sending STA's RSN IE | Encrypted, Encapsulated Beacon/Probe RSN IE [3]<br>Opt: 2nd RSN<br>Opt: **GTK**<br>*Pad bytes?* | None required | encrypted, encap-sulated<br>**GTK**<br>*Pad bytes?* | none required | Peer MAC Address KDE | Encrypted Initiator MAC Address KDE and **STAKey** *Pad bytes?* | Initiator MAC Address KDE |

[1] 0 only if the AP does not support key mapping keys, or if the STA has the "No Pairwise" capability bit set, and only the group key will be used.
[2] assuming this follows the Group Key Handshake between the Peer STA
[3] and, optionally, a second RSN IE that is the Authenticator's Pairwise cipher suite assignment, and, if a Group cipher has been negotiated, the encapsulated GTK and the GTK's Key ID