Wireless LAN Consortium

802.11 WPA2 Station MAC Conformance Test Suite v2.3 Report

UNH InterOperability Laboratory — 121 Technology Drive, Suite 2 — Durham, NH 03824 — +1-603-862-2263

January 3, 2013
Report Rev. 1.0

Joe Vendor
Magic Wireless Company
52 OFDM Drive
Mimo, NH 010111

Mr. Vendor,

Enclosed are the results from the Wireless WPA2 STA MAC Conformance Test Suite testing performed on the:

Magic Device Name MD-360 802.11a/b/g Station

This testing pertains to a set of standard requirements, put forth in the IEEE Std 802.11-2007 Edition. The tests performed are part of the 802.11 WPA2 Station MAC Conformance Test Suite v2.3, which is available on the UNH-IOL's website:

ftp://ftp.iol.unh.edu/pub/wireless/TestSuites/mac/802.11_STA_WPA2_MAC_Conformance_Test_Suite_v2.3.pdf

Issues Observed While Testing
1.2.4: CCMP PN Replay Detection - part b: The DUT was observed to process a CCMP MSDU containing fragments containing non incrementing PNs.
1.3.3: Key Length Field Processing: The DUT was observed to process EAPoL-Key Messages 1 and 3 containing invalid Key Lengths.
1.3.4: Key Replay Counter Processing- part b: The DUT was observed to process EAPoL-Key Message 3 containing an invalid Key Replay Counter.
1.3.6: Key IV Field Processing: The DUT was observed to process EAPoL-Key Message 1 containing a non-zero Key IV.
1.3.7: Key RSC Field Processing: The DUT was observed to process EAPoL-Key Messages containing non-zero Key RSCs.
1.4.5: Key Nonce Field Formatting: The DUT was observed to transmit EAPoL-Key Message 4 containing a non-zero Key Nonce.

As always, we welcome any comments regarding this Test Suite. If you have any questions about the test procedures or results, please contact me via e-mail at TJ.Tester@iol.unh.edu or by phone at +1-603-862-2263.

Regards,

Touraj Parsa

TJ MCTester

UNH-IOL
2013.01.0
3 16:23:00
-05'00'

DIGITAL SIGNATURE INFORMATION

This document was created using an Adobe Digital signature. A Digital signature helps to ensure the authenticity of the document, but only in this Digital format. For information on how to verify this document's integrity proceed to the following site:

http://www.iol.unh.edu/certifyDoc/

If the document status still indicates "Validity of author NOT confirmed", then please contact the UNH-IOL to confirm the document's authenticity. To further validate the certificate integrity, Adobe 6.0 should report the following fingerprint information:

MD5 Fingerprint: **EEE1 7A82 7806 EB21 AF94 F189 E4BE 361B**
SHA-1 Fingerprint: **ECFB 7FAF AB4A 0832 2408 E965 9F5C E3F2 D784 AAAB**

Table 1 - Result Key - The following table contains possible results and their meanings

Result	**Interpretation**
PASS	The DUT was observed to exhibit conformant behavior.
FAIL	The DUT was observed to exhibit non-compliant behavior.
PASS with Comments	The DUT was observed to exhibit conformant behavior, however, additional explanation of the situation is included.
Warning	The DUT was observed to exhibit behavior that is not recommended.
Informative	Results are for informative purposes only and are not judged on a pass or fail basis.
Refer to Comments	From the observations, a valid pass or fail could not be determined. An additional explanation of the situation is included.
Not Applicable	The DUT does not support the technology required to perform these tests.
Not Available	Due to testing station or time limitations, the tests could not be performed, or were performed in a limited capacity.
Not Tested	Not tested due to time constraint of the test period.
Borderline	The observed values of the specified parameter are valid at one extreme, and invalid at the other.

Table 2 - Setup and Configuration Information

Product	
Manufacturer	Magic Device Machine
Model	MD-360
Hardware Version	3AGE3584
Firmware Version	8.4.2453
MAC Address	00:0M:0A:0C:00:00
Serial Number	FTH25489FE
IOL Label	VN-DUTT-00000123456
PSK	wireless
Test System Hardware	
RF Isolated Environment	USC-26 RF/EMI Isolation Chamber 16'x 8' x 8' @ 100dB
Sniffer	Atheros DK4 Sniffer Station
Test Station	Atheros DK4 Testing Station

In many traces, identification frames were used to simplify result collecting. These frames are identified by their source MAC address (00:00:01:ba:bb:1e). Below is an example of an identification frame found in a trace:

No.	Info	Protocol	Source	Destination
1	Beacon frame, BI=100, SSID=MAC""	IEEE 802.11	Xerox_00:00:11	Broadcast
2	Part a: Non-Acknowledged Frames	Babble Frame	0.0.0.0	255.255.255.255
3	MSDU1 => ICMP Echo Request with Protocol Version > 0	Babble Frame	0.0.0.0	255.255.255.255
4	Echo (ping) request	ICMP	192.168.0.102	aa:bb:cc:dd:ee
5	Acknowledgement	IEEE 802.11		Xerox_00:00:11
6	Clear-to-send	IEEE 802.11		aa:bb:cc:dd:ee
7	Data	IEEE 802.11	Xerox_00:00:11	aa:bb:cc:dd:ee

Rows highlighted in green are Management Frames.
Rows highlighted in blue are Control Frames.
Rows highlighted in white are Data Frames.
Rows highlighted in violet are ICMP Echo Requests/Responses

Fields or Frame excerpts highlighted in this reports are important and directly related to the issue being shown in the trace excerpt.

GROUP 1: CCMP ENCAPSULATION

Test # and Label	Part(s)	Result(s)
1.1.1: CCMP MIC Verification	**a**	**PASS**
	b	**PASS**
	c	**PASS**
	d	**PASS**

Comments on Test Procedure

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

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.

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.

Comments on Test Results

a-d. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.1.2: CCMP Header Format	**a**	**PASS**
	b	**PASS**
	c	**PASS**
	d	**PASS**

Comments on Test Procedure

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

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.

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 4th octet and all bits of the 3rd octet in the CCMP header to 0.
d. increment the PN for every MPDU transmitted.

Comments on Test Results

a-d. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.1.3: CCMP Encryption Verification	a	**PASS**
	b	**PASS**

Comments on Test Procedure

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

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].

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

Comments on Test Results

a-b. There were no issues uncovered during the testing process.

GROUP 2: CCMP DECAPSULATION

Test # and Label	Part(s)	Result(s)
1.2.1: CCMP MIC Processing	**a**	**PASS**

Comments on Test Procedure

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

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.

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

Comments on Test Results

a. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.2.2: CCMP Header Processing	**a**	**PASS**
	b	**PASS**
	c	**PASS**
	d	**Informative**

Comments on Test Procedure

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

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.

The DUT should:
a. accept keys for each Key ID defined.
b. ignore reserved bits b0 to b4 of the 4th octet and all bits of the 3rd 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).

Comments on Test Results

a-d. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.2.3: CCMP Decryption Verification	**a**	**PASS**
	b	**PASS**
	c	**PASS**

Comments on Test Procedure

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

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.

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.

Comments on Test Results

a-c. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.2.4: CCMP PN Replay Detection	**a**	**PASS**
	b	**FAIL**
	c	**PASS**

Comments on Test Procedure

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

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.

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).

Comments on Test Results

a. There were no issues uncovered during the testing process.
b. The DUT was observed to incorrectly process CCMP frames whose constituent PN values were not sequential. The DUT should use the PN from the previously received MPDU to detect replayed frames and discard MSDUs whose constituent MPDU PN values are not sequential. See IEEE Std 802.11™-2007 Edition, subclause 8.3.3.4.3. Please refer to Table 4 for more detailed information regarding this result.
c. There were no issues uncovered during the testing process.

GROUP 3: EAPOL-KEY RECEPTION

Test # and Label	Part(s)	Result(s)
1.3.1: Descriptor Type Processing	**a**	**PASS**
	b	**PASS**

Comments on Test Procedure

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

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.

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

Comments on Test Results

a-b. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.3.2: Key Information Field Processing	**a**	**PASS**
	b	**PASS**
	c	**PASS**

Comments on Test Procedure

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

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].

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).

Comments on Test Results

a-c. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.3.3: Key Length Field Processing	**a**	**FAIL**
	b	**FAIL**

Comments on Test Procedure

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

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 3 – Key Lengths

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

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).

Comments on Test Results

a. The DUT was observed to incorrectly process EAPoL-Key Message-1 containing an invalid Key Length of 0. The DUT should discard all EAPoL-Key Messages containing invalid Key lengths. See IEEE Std 802.11™-2007 Edition, subclause 8.5.3.1. Please refer Table 5 to for more detailed information regarding this result.

b. The DUT was observed to incorrectly process EAPoL-Key Message-3 containing an invalid Key Length of 13. The DUT should discard all EAPoL-Key Messages containing invalid Key lengths. See IEEE Std 802.11™-2007 Edition, subclause 8.5.3.1. Please refer Table 6 to for more detailed information regarding this result.

Test # and Label	Part(s)	Result(s)
1.3.4: Key Replay Counter Processing	**a**	**PASS**
	b	**FAIL**

Comments on Test Procedure

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

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.

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

Comments on Test Results

a. There were no issues uncovered during the testing process.

b. The DUT was observed to process an EAPoL Key Message 3 containing a Key Replay Counter field that is equal to previously received valid EAPoL Key. The DUT should silently discard all invalid EAPoL Key Messages. See IEEE Std. 802.11-2007 subclause 8.5.2. Please refer to Table 7 for more information regarding this result.

Test # and Label	Part(s)	Result(s)
1.3.5: Key Nonce Field Processing	**a**	**PASS**

Comments on Test Procedure

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

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.

a. The DUT should not successfully complete the 4-way handshake.

Comments on Test Results

a. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.3.6: Key IV Field Processing	**a**	**FAIL**

Comments on Test Procedure

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

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.

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

Comments on Test Results

a. The DUT was observed to process an EAPoL Key Message 1 frame containing a non-zero Key IV. The DUT should silently discard all invalid EAPoL Key Messages. See IEEE Std. 802.11-2007 subclause 8.5.2. Please refer to Table 8 for more information regarding this result.

Test # and Label	Part(s)	Result(s)
1.3.7: Key RSC Field Processing	**a**	**FAIL**
	b	**Informative**
	c	**FAIL**

Comments on Test Procedure

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

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.

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.

Comments on Test Results

a. The DUT was observed to process an EAPoL Key Message 1 frame containing a non-zero Key RSC. The DUT should silently discard all invalid EAPoL Key frames. See IEEE Std. 802.11-2007 subclause 8.5.2. Please refer to Table 9 for more information regarding this result.
b. The DUT was observed to ignore the unused octets of the Key RSC, and successfully completed the 4-way handshake.
c. The DUT was observed to process a broadcast CCMP frame containing a PN less than the Key RSC in EAPoL Key Message 3. The DUT should use the Key RSC contained within EAPoL Key Message 3 to determine the PN of the next broadcast CCMP frame. See IEEE Std. 802.11-2007 subclause 8.5.2. Please refer to Table 10 for more information regarding this result.

Test # and Label	Part(s)	Result(s)
1.3.8: Reserved Octets Processing	**a**	**PASS**

Comments on Test Procedure

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

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

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

Comments on Test Results

a. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.3.9: Key MIC Field Processing	**a**	**PASS**

Comments on Test Procedure

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

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.

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

Comments on Test Results

a. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.3.10: Key Data Length Field Processing	**a**	**PASS**
	b	**PASS**

Comments on Test Procedure

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

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 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.

Comments on Test Results

a-b. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.3.11: Key Data Field Processing (Pairwise Message1)	**a**	**PASS**
	b	**PASS**
	c	**PASS**

Comments on Test Procedure

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

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.

The DUT should:
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).

Comments on Test Results

a-c. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.3.12: Key Data Field Processing (Pairwise Message3)	**a**	**PASS**
	b	**PASS**
	c	**PASS**

Comments on Test Procedure

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

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.

The DUT should:

1. discard encrypted Key Data fields that do not contain a GTK KDE (MSDU2).
2. 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).
3. ignore any extraneous IEs or unknown KDEs (MSDU5, MSDU6).

Comments on Test Results

a-c. There were no issues uncovered during the testing process.

GROUP 4: EAPOL-KEY TRANSMISSION

Test # and Label	Part(s)	Result(s)
1.4.1: Descriptor Type Field Formatting	**a**	**PASS**

Comments on Test Procedure

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

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.

a. The DUT should transmit EAPoL-Key frames with a Descriptor Type of 2.

Comments on Test Results

a. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.4.2: Key Information Field Formatting	**a**	**PASS**
	b	**PASS**
	c	**PASS**

Comments on Test Procedure

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

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].

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.

Comments on Test Results

a-c. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.4.3: Key Length Field Formatting	**a**	**PASS**

Comments on Test Procedure

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

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.

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

Comments on Test Results

a. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.4.4: Key Replay Counter Formatting	**a**	**PASS**
	b	**PASS**
	c	**PASS**
	d	**PASS**

Comments on Test Procedure

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

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.

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

Comments on Test Results

a-d. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.4.5: Key Nonce Field Formatting	**a**	**PASS**
	b	**FAIL**

Comments on Test Procedure

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

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.

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.

Comments on Test Results

a. There were no issues uncovered during the testing process.
b. The DUT was observed to not use the value of zero for the Key Nonce within EAPoL-Key Message 4. The DUT should use the value of zero for the Key Nonce within EAPoL-Key Message 4. See IEEE Std. 802.11-2007 subclause 8.5.3.4. Please refer to Table 11 for more information regarding this result.

Test # and Label	Part(s)	Result(s)
1.4.6: Key IV Field Formatting	**a**	**PASS**

Comments on Test Procedure

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

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.

The DUT should:

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

Comments on Test Results

a. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.4.7: Key RSC Field Formatting	**a**	**PASS**

Comments on Test Procedure

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

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.

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

Comments on Test Results

a. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.4.8: Reserved Octets Field Formatting	**a**	**PASS**

Comments on Test Procedure

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

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

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

Comments on Test Results

a. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.4.9: Key MIC Field Formatting	**a**	**PASS**

Comments on Test Procedure

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

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.

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

Comments on Test Results

a. There were no issues uncovered during the testing process.

Test # and Label	Part(s)	Result(s)
1.4.10: Key Data & Length Field Formatting	**a**	**PASS**

Comments on Test Procedure

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

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.

The DUT should:

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.

Comments on Test Results

a. There were no issues uncovered during the testing process.

TRACE EVALUATION:

Table 4

Test #: 1.2.4: CCMP PN Replay Detection part b
From: \traces\wireshark_Traces\1.2.4.apc

No.	Info	Protocol	Source	Destination
1786	Fragmented IEEE 802.11 frame	IEEE 802.11	Xerox_00:00:11	aa:bb:cc:dd:ee
1787	Acknowledgement	IEEE 802.11		Xerox_00:00:11 (RA)
1788	Fragmented IEEE 802.11 frame	IEEE 802.11	Xerox_00:00:11	aa:bb:cc:dd:ee
1789	Fragmented IEEE 802.11 frame	IEEE 802.11	Xerox_00:00:11	aa:bb:cc:dd:ee
1790	Acknowledgement	IEEE 802.11		Xerox_00:00:11 (RA)
1791	Echo (ping) request	ICMP	192.168.0.111	192.168.0.133
1792	SNA device <--> Non-SNA Device	SNA	B76F	D05B
1793	Acknowledgement	IEEE 802.11		Xerox_00:00:11 (RA)
1794	Clear-to-send	IEEE 802.11		aa:bb:cc:dd:ee
1795	Echo (ping) reply	ICMP	192.168.0.133	192.168.0.111
1796	Acknowledgement	IEEE 802.11		aa:bb:cc:dd:ee

```
Frame 1786
IEEE 802.11 Data
    Type/Subtype: Data (0x20)
    Frame Control: 0x4608 (Normal)
    Duration: 208
    Destination address: aa:bb:cc:dd:ee
    BSS Id: Xerox_00:00:11 (00:00:01:00:00:11)
    Source address: Xerox_00:00:11 (00:00:01:00:00:11)
    Fragment number: 0
    Sequence number: 78
    Frame check sequence: 0xee957951 [correct]
    CCMP parameters
        CCMP Ext. Initialization Vector: 0x000000000054
        Key Index: 0
    Reassembled 802.11 in frame: 1791
Data (212 bytes)
    Data: 0846D00002BAD05BB76F0000010000110000010000011E004...
    [Length: 212]

Frame (256 bytes):

0000  08 46 d0 00 02 ba d0 5b b7 6f 00 00 01 00 00 11
0010  00 00 01 00 00 11 e0 04 54 00 00 20 00 00 00 00
0020  0f 54 2b e5 e8 de a8 9a 89 60 b0 cb b5 86 1d 51
0030  f3 57 28 69 0a bc 7d 2a f1 6a 25 aa de 31 f1 c3
0040  4f 53 40 3e 7c 99 d4 02 71 e3 c6 27 7a 78 a0 80
0050  5f 39 29 56 41 0d 54 88 7f 04 1d 22 ac 2c 10 9c
0060  8e 9e db 07 62 6c c4 8c f8 11 0f 5f 34 fd 9a 08
0070  3a c4 01 59 5b 1b a8 47 e1 3d 97 44 04 15 74 4f
0080  b5 54 41 ef 15 ab 94 e7 a4 19 7e bf 24 5d 39 74
0090  a6 96 04 de ed 56 2d 19 36 17 bd d2 ae 56 88 7b
00a0  8e eb 3a 26 af 40 22 41 ca 7e 4a 2f 3b 49 40 e4
00b0  ae 30 b4 8b 10 87 01 ee 26 12 5a f7 f4 e5 55 15
00c0  b0 f2 d4 9b ef 95 12 bb 5b 9e 25 9a de c4 14 94
00d0  a3 63 19 ed 16 3d 9a 6b dd d6 08 79 d9 f4 13 2b
00e0  81 24 5f d0 81 18 45 d9 43 57 0e 90 37 c7 d6 1a
00f0  04 9d e0 c2 9a f9 1f 28 04 1a 09 8a ee 95 79 51
```

```
Frame 1788
IEEE 802.11 Data
    Type/Subtype: Data (0x20)
    Frame Control: 0x4608 (Normal)
    Duration: 144
    Destination address: aa:bb:cc:dd:ee
    BSS Id: Xerox_00:00:11 (00:00:01:00:00:11)
    Source address: Xerox_00:00:11 (00:00:01:00:00:11)
    Fragment number: 1
    Sequence number: 78
    Frame check sequence: 0x35ee17e9 [correct]
    CCMP parameters
        CCMP Ext. Initialization Vector: 0x000000000054
        Key Index: 0
    Reassembled 802.11 in frame: 1791
Data (212 bytes)
    Data: 0846900002BAD05BB76F000001000011000001000011E104...
    [Length: 212]

Frame (256 bytes):

0000  08 46 90 00 02 ba d0 5b b7 6f 00 00 01 00 00 11
0010  00 00 01 00 00 11 e1 04 54 00 00 20 00 00 00 00
0020  15 4f 9a 56 5c 6b 16 2d 74 d9 0b dc 09 3b a3 ee
0030  cc 97 d2 a6 0e d1 bb 82 f9 0b ef e4 1a fc 8e fb
0040  96 2d fd 28 a8 4d 00 d6 ad 3f 1a fb ae ac 74 54
0050  b3 d5 c5 ba b5 f9 a0 7c 83 f8 e1 de 58 d8 e4 68
0060  62 72 37 eb b6 b8 10 58 24 cd d3 83 e0 29 4e dc
0070  16 e8 2d 75 6f 2f 9c 73 dd 01 ab 78 30 21 40 7b
0080  99 78 6d c3 41 ff c0 b3 f8 45 22 e3 70 09 6d 20
0090  ca fa 68 b2 99 22 59 6d 4a 6b c1 ae da 22 fc 0f
00a0  e2 87 56 4a fb 14 76 15 96 22 16 73 6f 1d 14 b0
00b0  82 1c 98 a7 24 b3 35 da 1a 2e 66 cb c0 d1 61 21
00c0  9c de f8 b7 3b 41 c6 6f 87 42 f9 46 0a 10 c0 40
00d0  4f 8f f5 01 e2 c9 6e 9f 21 2a f4 85 2d 00 e7 df
00e0  6d c8 b3 3c 55 cc 91 0d 9f 8b d2 4c e3 13 02 ce
00f0  28 b1 cc ee 2a 11 00 56 67 7c 22 2b 35 ee 17 e9
```

```
Frame 1791
IEEE 802.11 Data
    Type/Subtype: Data (0x20)
    Frame Control: 0x4208 (Normal)
    Duration: 44
    Destination address: aa:bb:cc:dd:ee
    BSS Id: Xerox_00:00:11 (00:00:01:00:00:11)
    Source address: Xerox_00:00:11 (00:00:01:00:00:11)
    Fragment number: 2
    Sequence number: 78
    Frame check sequence: 0x37475079 [correct]
    CCMP parameters
        CCMP Ext. Initialization Vector: 0x000000000055

Frame (56 bytes):

0000  08 42 2c 00 02 ba d0 5b b7 6f 00 00 01 00 00 11
0010  00 00 01 00 00 11 e2 04 55 00 00 20 00 00 00 00
0020  27 d0 7a 47 df 46 09 c7 62 23 dc e0 f8 ee ba b9
0030  55 a6 ea 49 37 47 50 79
```

Table 5

Test #: 1.3.3: Key Length Field Processing part a
From: \traces\wireshark_Traces\1.3.3.apc

No.	Info	Protocol	Source	Destination
414	Key #3	EAPOL	Xerox_00:00:11	Vendor(abcd)
415	Acknowledgement	IEEE 802.11		Xerox_00:00:11 (RA)
416	Key #4	EAPOL	Vendor(abcd)	Xerox_00:00:11
417	Acknowledgement	IEEE 802.11		Vendor(abcd)

```
Frame 414
IEEE 802.11 Data
    Type/Subtype: Data (0x20)
    Frame Control: 0x0208 (Normal)
    Duration: 44
    Destination address: aa:bb:cc:dd:ee
    BSS Id: Xerox_00:00:11 (00:00:01:00:00:11)
    Source address: Xerox_00:00:11 (00:00:01:00:00:11)
    Fragment number: 0
    Sequence number: 51
    Frame check sequence: 0xef7bc0d6 [correct]
802.1X Authentication
    Version: 2
    Type: Key (3)
    Length: 117
    Descriptor Type: EAPOL RSN key (2)
    Key Information: 0x008a
    Key Length: 0
    Replay Counter: 0

0000  08 02 2c 00 02 ba d0 5b b7 6f 00 00 01 00 00 11
0010  00 00 01 00 00 11 30 03 aa aa 03 00 00 00 88 8e
0020  02 03 00 75 02 00 8a 00 00 00 00 00 00 00 00 00
0030  00 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e
0040  0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e
0050  1f 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0070  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0080  00 00 16 dd 14 00 0f ac 04 14 da 62 73 bf 67 4f
0090  0b 6a ed ca 16 d3 69 16 74 ef 7b c0 d6
```

Table 6
Test #: 1.3.3: Key Length Field Processing
From: \traces\wireshark_Traces\1.3.3.apc

No.	Info	Protocol	Source	Destination
4930	Key	EAPOL	Xerox_00:00:11	aa:bb:cc:dd:ee
4931	Acknowledgement	IEEE 802.11		Xerox_00:00:11 (RA)
4932	Key	EAPOL	aa:bb:cc:dd:ee	Xerox_00:00:11
4933	Acknowledgement	IEEE 802.11		aa:bb:cc:dd:ee

```
Frame 4930
IEEE 802.11 Data
    Type/Subtype: Data (0x20)
    Frame Control: 0x0208 (Normal)
    Destination address: aa:bb:cc:dd:ee
    BSS Id: Xerox_00:00:11 (00:00:01:00:00:11)
    Source address: Xerox_00:00:11 (00:00:01:00:00:11)
    Fragment number: 0
    Sequence number: 506
    Frame check sequence: 0x8c869c84 [correct]
802.1X Authentication
    Version: 2
    Type: Key (3)
    Length: 151
    Descriptor Type: EAPOL RSN key (2)
    Key Information: 0x13ca
    Key Length: 13
    Replay Counter: 38

0000  08 02 2c 00 02 ba d0 5b b7 6f 00 00 01 00 00 11
0010  00 00 01 00 00 11 a0 1f aa aa 03 00 00 00 88 8e
0020  02 03 00 97 02 13 ca 00 0d 00 00 00 00 00 00 00
0030  26 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e
0040  0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e
0050  1f 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0070  00 e5 44 a6 19 68 91 31 36 b3 42 b6 ad b7 21 6a
0080  22 00 38 1f 9a 65 ad 2e 5c d3 8f b9 79 a6 b9 fa
0090  20 25 7a ea 83 97 98 8e 05 ab d8 2c 37 92 38 3e
00a0  40 4d fa 73 d1 e2 45 e2 49 de 19 d1 8d 80 c7 45
00b0  f3 3a 3b 0f 9a 1c 16 7e 96 d7 71 8c 86 9c 84
```

Table 7
Test #: 1.3.4: Key Replay Counter Processing part b
From: \traces\wireshark_Traces\1.3.4.apc

No.	Info	Protocol	Source	Destination
1443	Key #1	EAPOL	Xerox_00:00:11	aa:bb:cc:dd:ee
1444	Acknowledgement	IEEE 802.11		Xerox_00:00:11 (RA)
1445	Key #2	EAPOL	aa:bb:cc:dd:ee	Xerox_00:00:11
1446	Acknowledgement	IEEE 802.11		aa:bb:cc:dd:ee
1447	Key #2	EAPOL	aa:bb:cc:dd:ee	Xerox_00:00:11
1448	Acknowledgement	IEEE 802.11		aa:bb:cc:dd:ee
1449	Beacon frame, BI=100, SSID=MAC""	IEEE 802.11	Xerox_00:00:11	Broadcast
1450	Beacon frame, BI=100, SSID=MAC""	IEEE 802.11	Xerox_00:00:11	Broadcast
1451	Key #3	EAPOL	Xerox_00:00:11	aa:bb:cc:dd:ee
1452	Acknowledgement	IEEE 802.11		Xerox_00:00:11 (RA)
1453	Clear-to-send	IEEE 802.11		aa:bb:cc:dd:ee
1454	Key #4	EAPOL	aa:bb:cc:dd:ee	Xerox_00:00:11
1455	Acknowledgement	IEEE 802.11		aa:bb:cc:dd:ee

```
Frame 1443
IEEE 802.11 Data
    Type/Subtype: Data (0x20)
    Frame Control: 0x0208 (Normal)
    Destination address: aa:bb:cc:dd:ee
    BSS Id: Xerox_00:00:11 (00:00:01:00:00:11)
    Source address: Xerox_00:00:11 (00:00:01:00:00:11)
    Fragment number: 0
    Sequence number: 163
    Frame check sequence: 0x0dde4d47 [correct]
802.1X Authentication
    Version: 2
    Type: Key (3)
    Length: 117
    Descriptor Type: EAPOL RSN key (2)
    Key Information: 0x008a
    Key Length: 16
    Replay Counter: 0

0000  08 02 2c 00 02 ba d0 5b b7 6f 00 00 01 00 00 11
0010  00 00 01 00 00 11 30 0a aa aa 03 00 00 00 88 8e
0020  02 03 00 75 02 00 8a 00 10 00 00 00 00 00 00 00
0030  00 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e
0040  0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e
0050  1f 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0070  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0080  00 00 16 dd 14 00 0f ac 04 14 da 62 73 bf 67 4f
0090  0b 6a ed ca 16 d3 69 16 74 0d de 4d 47
```

```
Frame 1451
IEEE 802.11 Data
    Type/Subtype: Data (0x20)
    Frame Control: 0x0208 (Normal)
    Duration: 44
    Destination address: aa:bb:cc:dd:ee
    BSS Id: Xerox_00:00:11 (00:00:01:00:00:11)
    Source address: Xerox_00:00:11 (00:00:01:00:00:11)
    Fragment number: 0
    Sequence number: 165
    Frame check sequence: 0xf04d9d62 [correct]
802.1X Authentication
    Version: 2
    Type: Key (3)
    Length: 151
    Descriptor Type: EAPOL RSN key (2)
    Key Information: 0x13ca
    Key Length: 16
    Replay Counter: 0

0000  08 02 2c 00 02 ba d0 5b b7 6f 00 00 01 00 00 11
0010  00 00 01 00 00 11 50 0a aa aa 03 00 00 00 88 8e
0020  02 03 00 97 02 13 ca 00 10 00 00 00 00 00 00 00
0030  00 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e
0040  0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e
0050  1f 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0070  00 40 68 ed 37 d4 d8 b9 2d 4a bb 2b b3 48 52 a9
0080  cd 00 38 0d f6 95 72 49 e6 18 8d 77 c0 ee c9 63
0090  33 4c e6 2b 97 8b 89 f4 e4 49 cb fc 67 43 1e c2
00a0  5d 13 1a 14 ba f2 ef de e2 42 96 b5 47 f4 43 44
00b0  4c 70 a4 61 94 3f ab 67 67 32 32 f0 4d 9d 62
```

Table 8
Test #: 1.3.6: Key IV Field Processing
From: \traces\wireshark_Traces\1.3.6.apc

No.	Info	Protocol	Source	Destination
497	Key #1	EAPOL	Xerox_00:00:11	Vendor(abcd)
498	Acknowledgement	IEEE 802.11		Xerox_00:00:11 (RA)
499	Key #2	EAPOL	Vendor(abcd)	Xerox_00:00:11
500	Acknowledgement	IEEE 802.11		Vendor(abcd)

```
Frame 497
IEEE 802.11 Data
    Type/Subtype: Data (0x20)
    Frame Control: 0x0208 (Normal)
    Duration: 44
    Destination address: aa:bb:cc:dd:ee
    BSS Id: Xerox_00:00:11 (00:00:01:00:00:11)
    Source address: Xerox_00:00:11 (00:00:01:00:00:11)
    Fragment number: 0
    Sequence number: 65
    Frame check sequence: 0x793ce367 [correct]
802.1X Authentication
    Version: 2
    Type: Key (3)
    Length: 117
    Descriptor Type: EAPOL RSN key (2)
    Key Information: 0x008a
    Key Length: 16
    Replay Counter: 0
    Nonce: 000102030405060708090A0B0C0D0E0F1011121314151617
  Key IV: 2D2E2F303132333435363738393A3B3C


0000  08 02 2c 00 02 ba d0 5b b7 6f 00 00 01 00 00 11
0010  00 00 01 00 00 11 10 04 aa aa 03 00 00 00 88 8e
0020  02 03 00 75 02 00 8a 00 10 00 00 00 00 00 00 00
0030  00 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e
0040  0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e
0050  1f 2d 2e 2f 30 31 32 33 34 35 36 37 38 39 3a 3b
0060  3c 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0070  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0080  00 00 16 dd 14 00 0f ac 04 14 da 62 73 bf 67 4f
0090  0b 6a ed ca 16 d3 69 16 74 79 3c e3 67
```

Table 9
Test#: 1.3.7: Key RSC Field Processing part a
From: \traces\wireshark_Traces\1.3.7.apc

No.	Info	Protocol	Source	Destination
320	Key #1	EAPOL	Xerox_00:00:11	aa:bb:cc:dd:ee
321	Acknowledgement	IEEE 802.11		Xerox_00:00:11 (RA)
322	Key #2	EAPOL	aa:bb:cc:dd:ee	Xerox_00:00:11
323	Acknowledgement	IEEE 802.11		aa:bb:cc:dd:ee

```
Frame 320
IEEE 802.11 Data
    Type/Subtype: Data (0x20)
    Frame Control: 0x0208 (Normal)
    Duration: 44
    Destination address: aa:bb:cc:dd:ee
    BSS Id: Xerox_00:00:11 (00:00:01:00:00:11)
    Source address: Xerox_00:00:11 (00:00:01:00:00:11)
    Fragment number: 0
    Sequence number: 70
    Frame check sequence: 0x8fd1b221 [correct]
802.1X Authentication
    Version: 2
    Type: Key (3)
    Length: 117
    Descriptor Type: EAPOL RSN key (2)
    Key Information: 0x008a
    Key Length: 16
    Replay Counter: 0
    Nonce: 000102030405060708090A0B0C0D0E0F1011121314151617...
    Key IV: 00000000000000000000000000000000
    WPA Key RSC: 3D3E3F4041424344

0000  08 02 2c 00 02 ba d0 5b b7 6f 00 00 01 00 00 11
0010  00 00 01 00 00 11 60 04 aa aa 03 00 00 00 88 8e
0020  02 03 00 75 02 00 8a 00 10 00 00 00 00 00 00 00
0030  00 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e
0040  0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e
0050  1f 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060  00 3d 3e 3f 40 41 42 43 44 00 00 00 00 00 00 00
0070  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0080  00 00 16 dd 14 00 0f ac 04 14 da 62 73 bf 67 4f
0090  0b 6a ed ca 16 d3 69 16 74 8f d1 b2 21
```

Table 10
Test#: 1.3.7: Key RSC Field Processing part c
From: \traces\wireshark_Traces\1.3.7.apc

No.	Info	Protocol	Source	Destination
1409	Key #3	EAPOL	Xerox_00:00:11	aa:bb:cc:dd:ee
1410	Acknowledgement	IEEE 802.11		Xerox_00:00:11 (RA)
1411	Key #4	EAPOL	aa:bb:cc:dd:ee	Xerox_00:00:11
1412	Acknowledgement	IEEE 802.11		aa:bb:cc:dd:ee
1413-1576	…			
1577	Data	IEEE 802.11	Xerox_00:00:11	Broadcast
1578	Clear-to-send	IEEE 802.11		aa:bb:cc:dd:ee
1579	Data	IEEE 802.11	aa:bb:cc:dd:ee	Xerox_00:00:11
1580	Acknowledgement	IEEE 802.11		aa:bb:cc:dd:ee

- Frames 1413-1576 have been removed for clarity.

```
Frame 1409
IEEE 802.11 Data
    Type/Subtype: Data (0x20)
    Frame Control: 0x0208 (Normal)
    Duration: 44
    Destination address: aa:bb:cc:dd:ee
    BSS Id: Xerox_00:00:11 (00:00:01:00:00:11)
    Source address: Xerox_00:00:11 (00:00:01:00:00:11)
    Fragment number: 0
    Sequence number: 187
    Frame check sequence: 0x3d3f5738 [correct]
802.1X Authentication
    Version: 2
    Type: Key (3)
    Length: 151
    Descriptor Type: EAPOL RSN key (2)
    Key Information: 0x13ca
    Key Length: 16
    Replay Counter: 13
    Nonce: 000102030405060708090A0B0C0D0E0F1011121314151617...
    Key IV: 00000000000000000000000000000000
    WPA Key RSC: 0000000000110000

0000  08 02 2c 00 02 ba d0 5b b7 6f 00 00 01 00 00 11
0010  00 00 01 00 00 11 b0 0b aa aa 03 00 00 00 88 8e
0020  02 03 00 97 02 13 ca 00 10 00 00 00 00 00 00 00
0030  0d 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e
0040  0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e
0050  1f 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060  00 00 00 00 00 00 11 00 00 00 00 00 00 00 00 00
0070  00 40 28 0c 34 bd ee ff 02 9f 78 1b 73 00 d5 10
0080  d4 00 38 93 ae c1 34 7f 24 46 c0 40 8e 5d 3c 92
0090  ed 5c df b2 ca 80 e4 0c 81 18 2a 9c 31 22 ff 1b
00a0  f6 e3 f0 8b f8 f1 ff 47 8f 1b 84 c2 91 f7 6a 79
00b0  d0 21 f2 ca c6 b9 0f 09 13 fd 73 3d 3f 57 38
```

```
Frame 1577
IEEE 802.11 Data
    Type/Subtype: Data (0x20)
    Frame Control: 0x4208 (Normal)
    Duration: 0
    Destination address: Broadcast (ff:ff:ff:ff:ff:ff)
    BSS Id: Xerox_00:00:11 (00:00:01:00:00:11)
    Source address: Xerox_00:00:11 (00:00:01:00:00:11)
    Fragment number: 0
    Sequence number: 194
    Frame check sequence: 0x49e711eb [correct]
    CCMP parameters
        CCMP Ext. Initialization Vector: 0x00000000000E
        Key Index: 1

0000  08 42 00 00 ff ff ff ff ff ff 00 00 01 00 00 11
0010  00 00 01 00 00 11 20 0c 0e 00 00 60 00 00 00 00
0020  12 fb af 15 c6 2b 7e 96 c7 5b ae d3 f7 53 66 5d
0030  48 74 df a9 88 a8 e9 c7 c5 ee 4c 74 a8 d9 48 10
0040  24 ab c2 5a 9d d3 35 e3 9d 6c 5c 85 a2 74 5d 7a
0050  68 8b 26 1f 5f ae 19 3b cc 40 e6 84 31 52 47 8f
0060  26 e4 5c 9d 64 c4 78 24 2a ca d7 36 9c 69 aa 69
0070  15 d6 4f 3b e0 25 e2 b0 02 0a 5e 90 1c 87 27 cc
0080  16 36 d3 e0 6b db 38 b8 f4 e9 3f 30 48 f2 e5 7a
0090  64 41 aa a5 64 73 60 cf dc de dd 10 fb e9 dd 7b
00a0  18 b2 e0 a9 02 96 68 e6 2e 24 76 7e 78 f4 ee e8
00b0  c7 6e 8b 22 ca 54 fa 23 33 7f 02 4b 4a 2f b6 af
00c0  06 a6 fe c7 95 04 26 b8 98 4c c5 a0 1e 67 bf 5d
00d0  cc ba b3 29 ee 48 07 73 5a d3 30 58 36 f7 10 1b
00e0  85 62 ac 45 de b0 fc 1c f3 a3 4c 46 40 d5 9f f8
00f0  49 cc ca 8b 3c 18 c2 d5 d7 c4 85 a3 11 e7 8e 50
0100  3f 76 9b 32 58 17 80 d6 cf 6f dc 26 62 27 60 00
0110  61 f1 63 6c 1c 48 ae ac 18 9d ad 2c 3e 13 4c 40
0120  63 40 dc 16 67 4c f7 9e 7e c5 39 76 8e b7 29 cd
0130  e7 ba e0 72 22 5b ef 03 fa 76 f0 67 a3 47 74 89
0140  55 33 79 b4 4d 4b ec 84 ab 5a a7 b8 dc ba 62 56
0150  8b d7 4f 99 c1 94 bf 5f 5b b0 0c 8e 79 1b a8 0c
0160  66 0a e8 ef 20 f0 cb e3 7d 3f 47 74 9f 80 0d 9e
0170  a9 fa fa 79 40 17 0e 04 4a 53 4f 77 f4 6b 56 ed
0180  09 1f dc af 0a 76 8e 8c b2 b5 97 76 9c 38 51 ed
0190  44 ed ff 0a 99 a9 34 3f 06 8b ad 1e e5 5f 2d 5a
01a0  3b b5 86 67 5d ca b7 44 18 d8 d3 15 95 5f 2c d3
01b0  e2 5b 94 c3 38 9f c2 59 1c 8b 99 f0 d9 1a 90 34
01c0  5f 54 ba 25 7e 01 a9 4f 83 20 bf 06 38 c5 cc 29
01d0  73 38 d7 92 67 12 0f 69 58 dd f1 8b a9 73 ff fd
01e0  e1 23 1b f9 a0 35 a7 5a 86 91 7c 04 49 e7 11 eb
```

Table 11
Test #: 1.4.5: Key Nonce Field Formatting part b
From: \traces\wireshark_Traces\1.4.4.apc

No.	Info	Protocol	Source	Destination
90	Key #3	EAPOL	Xerox_00:00:11	aa:bb:cc:dd:ee
91	Acknowledgement	IEEE 802.11		Xerox_00:00:11 (RA)
92	Key #4	EAPOL	aa:bb:cc:dd:ee	Xerox_00:00:11
93	Acknowledgement	IEEE 802.11		aa:bb:cc:dd:ee

```
Frame 92
IEEE 802.11 Data
    Type/Subtype: Data (0x20)
    Frame Control: 0x0108 (Normal)
    Duration: 40
    BSS Id: Xerox_00:00:11 (00:00:01:00:00:11)
    Source address: aa:bb:cc:dd:ee
    Destination address: Xerox_00:00:11 (00:00:01:00:00:11)
    Fragment number: 0
    Sequence number: 897
    Frame check sequence: 0xad2625d6 [correct]
802.1X Authentication
    Version: 2
    Type: Key (3)
    Length: 95
    Descriptor Type: EAPOL RSN key (2)
    Key Information: 0x030a
    Key Length: 0
    Replay Counter: 1
  Nonce: 40E8E882C224E788F4F26EF6EB2483938FD66275C1DEB24A
```

```
0000  08 01 28 00 00 00 01 00 00 11 02 ba d0 5b b7 6f
0010  00 00 01 00 00 11 10 38 aa aa 03 00 00 00 88 8e
0020  02 03 00 5f 02 03 0a 00 00 00 00 00 00 00 00 00
0030  01 40 e8 e8 82 c2 24 e7 88 f4 f2 6e f6 eb 24 83
0040  93 8f d6 62 75 c1 de b2 4a f9 b8 56 f0 45 3a 5a
0050  90 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0070  00 75 af 3a 7d eb a7 09 20 3d 78 05 f5 39 68 32
0080  a9 00 00 ad 26 25 d6
```