IEEE P802.11

Wireless Access Method and Physical Layer Specification

# Exorcising the Ghosts of Connections (and RT Encryption)

**Michael Fischer**
**Digital Ocean, Inc.**
**4242–3 Medical Drive**
**San Antonio, TX 78229**
**Telephone: +1–210–614–4096**
**Facsimile: +1–210–614–8192**
**email: mfischer@CHILD.com**

## Summary

This submission contains modifications to the contents of several sections of the D2.0 draft to remove the vestiges of the connection establishment management frames. This removal was approved in the closing plenary of the July, 1995 meeting (Motion #31, vote 33–3–10), but not fully reflected in the D2.0 text. This document also contains modfications to sections 5.2. and 5.3 to update the text and figures for the WEP changes from document 95–138 plus IV alighment changes approved in the closing plenary of the July, 1995 meeting (Motion 28, vote 23–0–24), but not fully reflected in the D2.0 text. Voters favoring this proposal can cite this document as the source of replacement text for their D2.0 letter ballot comments.

NOTE TO EDITORS: These draft updates are all "major editorial" changes, since they complete items approved by plenary vote prior to release of the D2.0 document for letter ballot.

## Updated Section 4.1.2.3

The Duration/ID field shall be 16 bits in length. The contents of the this field shall be as follows:

a) In Data Type frames transmitted during the contention free period that have frame body information associated with a ~~time-bounded~~ connection, the Duration/ID field shall carry the connection identity (CID) of the time-bound connection in the 14 least-significant bits, with the 2 most-significant bits set to '10'. The value of the CID shall be in the range 1 - 16383. Connection and establishment~~This~~ usage shall be reserved for future standardization.

b) In Control Type frames of SubType PS-Poll, the Duration/ID field shall carry the station identity (SID) of the station that transmitted the frame in the 14 least-significant bits, with the 2 most-significant bits set to '11'. The value of the SID shall be in the range 1 - 16383.

c) In all other frames the Duration/ID field shall contain a duration value. For frames transmitted during the contention period the duration value shall be set to the time in microseconds from the end of the current frame to the end of the next anticipated frame of Type Control and Subtype ACK. For frames transmitted during the contention free period the duration value shall be set to 32768. Whenever the contents of the Duration/ID field are less than 32768, the duration value shall be used to update the Net Allocation Vector according to the procedures defined in 6.

The encoding of the Duration/ID field is given in table 4-4.

| Bit 15 | Bit 14 | Bits 13-0 | Usage |
|---|---|---|---|
| 0 | 0 - 32767 | | Duration (in microseconds from end of this frame) |

| | | | |
|---|---|---|---|
| 1 | 0 | 0 | CF frames that do not need a CID or an SID |
| 1 | 0 | 1 - 16383 | CID in ~~TBS~~ frames using an established connection |
| 1 | 1 | 1 - 16383 | SID in PS-Poll frames (under either PCF or DCF) |

**Table 4-4: Duration/ID Field Encoding**

## Updated Section 4.2.3.11 — section to be removed and not replaced

~~The Frame Body of a Management frame of Subtype Connection Request shall contain the following information:~~

| ~~Order~~ | ~~Information~~ | ~~Note~~ |
|---|---|---|
| ~~TBD~~ | | |

## Updated Section 4.2.3.12 — section to be removed and not replaced

~~The Frame Body of a Management frame of Subtype Grant Connection shall contain the following information:~~

| ~~Order~~ | ~~Information~~ | ~~Note~~ |
|---|---|---|
| ~~TBD~~ | | |

## Updated Section 4.2.3.13 — section to be removed and not replaced

~~The Frame Body of a Management frame of Subtype End Connection shall contain the following information:~~

| ~~Order~~ | ~~Information~~ | ~~Note~~ |
|---|---|---|
| ~~TBD~~ | | |

## Updated Section 5.2.3

The process of disguising (binary) data in order to hide its information content is called **encryption**[1]. Data that is not enciphered is called **plaintext** (denoted by *P*) and data that is enciphered is called **ciphertext** (denoted by *C*). The process of turning ciphertext back into plaintext is called **decryption**. A **cryptographic algorithm**, or cipher, is a mathematical function used for enciphering or deciphering data. Modern cryptographic algorithms use a key sequence (denoted by *k*) to modify their output. The encryption function *E* operates on *P* to produce *C*:

$$E_k(P) = C$$

In the reverse process, the decryption function *D* operates on *C* to produce *P*:

$$D_k(C) = P$$

As illustrated in Figure 5-1, note that if the same key is used for encryption and decryption then

[1]Bruce Schneier, "Applied Cryptography, Protocols, Algorithms and Source Code in C", John Wiley & Sons, Inc. 1994

$D_k(E_k(P)) = P$

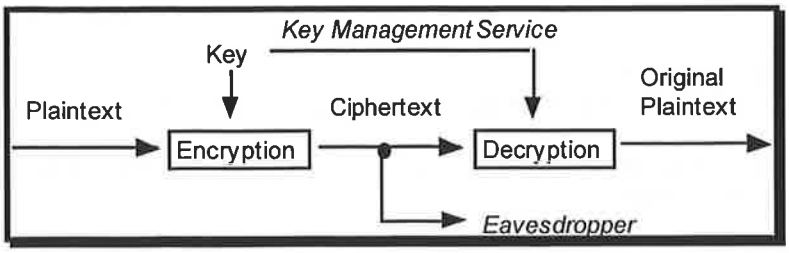



**Figure 5-1: A Confidential Data Channel**

The WEP algorithm is a form of electronic code book in which a block of plaintext is bitwise XOR'd with a pseudo random key sequence of equal length. The key sequence is generated by the WEP algorithm.

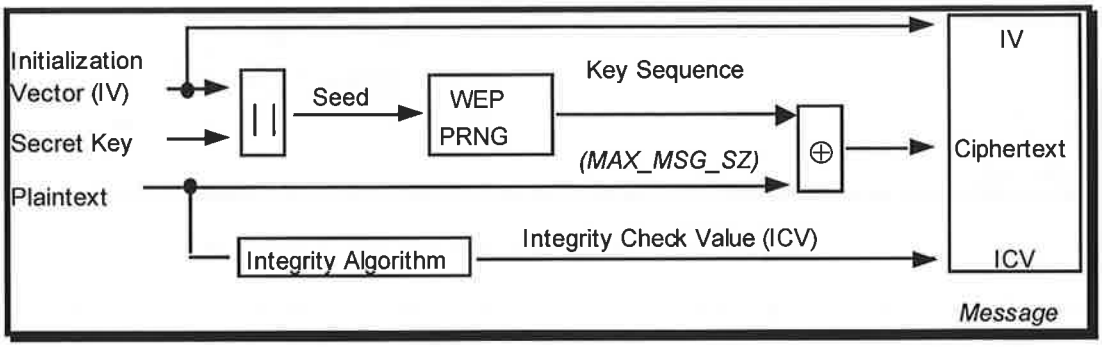



**Figure 5-2: WEP Encipherment Block Diagram**

Referring to Figure 5-2 and following from left to right, encipherment begins with a **secret key** that has been distributed to cooperating stations by an external key management service. WEP is a symmetric algorithm in which the same key is used for encipherment and decipherment.

The secret key is concatenated with an **initialization vector** (IV) and the resulting **seed** is input to a **pseudo random number generator** (PRNG). The PRNG outputs a **key sequence** *k* of pseudo-random bits equal in length to the largest possible MSDU. Two processes are applied to the plaintext MSDU. To protect against unauthorized data modification, an integrity algorithm operates on *P* to produce an **integrity check value** (ICV). Encipherment is then accomplished by mathematically combining the key sequence with *P*. The output of the process is a **message** containing the resulting ciphertext, the IV, and the ICV.

The WEP PRNG is the critical component of this process, since it transforms a relatively short secret key into an arbitrarily long key sequence. This greatly simplifies the task of key distribution as only the secret key needs to be communicated between stations. The IV extends the useful lifetime of the secret key and provides the self-synchronous property of the algorithm. The secret key remains constant while the IV changes periodically. Each new IV results in a new seed and key sequence, thus there is a one-to-one correspondence between the IV and *k*. The IV may be changed as frequently as every MSDU and, since it travels with the message, the receiver will always be able to decipher any message. The IV may be transmitted in the clear since it does not provide an attacker with any information about the secret key.

Because IV and the ICV must be transmitted with the MSDU, fragmentation may be invoked. The WEP algorithm is applied to an MSDU. The {IV, MSDU, ICV} triplet forms the actual data to be sent in the data frame.

For WEP protected frames, the first four octets of the frame contain the IV field for the MSDU. This field shall contain two sub-fields: A ~~1-octet field that contains the confidentiality algorithm ID, followed by a~~ 3-octet field that contains the initialization vector followed by a 1-octet pad field to maintain even-octet alignment of the encrypted payload~~The WEP IV is 16 bits.~~ The 64-bit PRNG seed is formed using the secret key as the most significant 40 bits and the initialization vector as the least significant 24 bits. The WEP IV is 24~~16~~ bits. The IV is followed by the MSDU, which is followed by the ICV. The WEP ICV is 32 bits. The WEP Integrity Check algorithm is CRC-32.

The entire {IV, MSDU, ICV} package may be split into several fragments (depending on the realtive values of the MSDU and the active MPDU size).

As stated previously, WEP combines *k* with *P* using bitwise XOR.

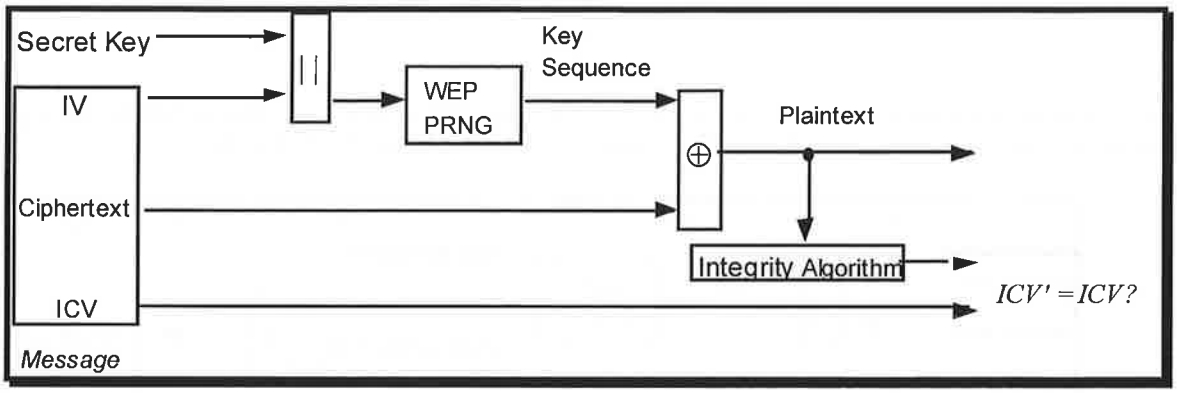



**Figure 5-3: WEP Decipherment Block Diagram**

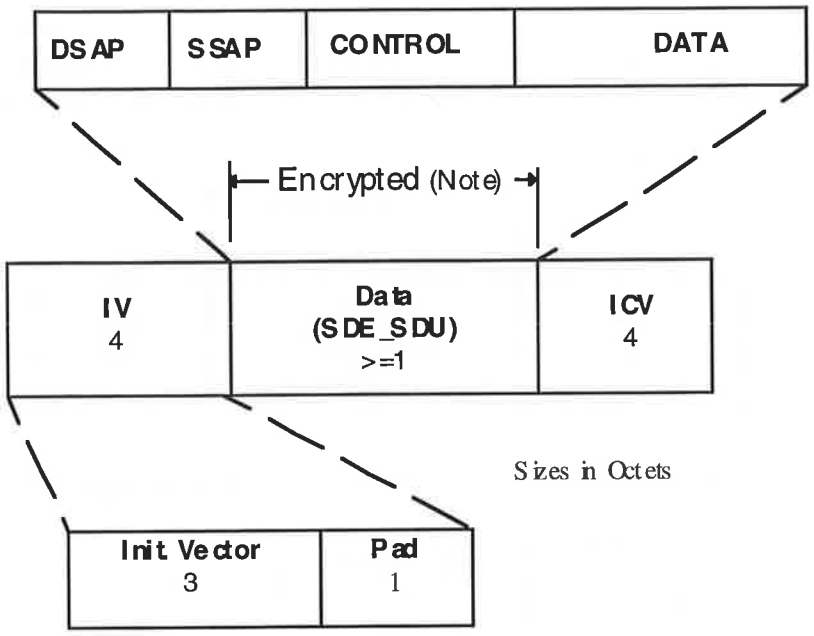
Referring to Figure 5-3 and following from left to right, decipherment begins with the arrival of a message. The IV of the incoming message is used to generate the key sequence necessary to decipher the incoming message. Combining the ciphertext with the proper key sequence yields the original plaintext. Correct decipherment is verified by performing the integrity algorithm on the recovered plaintext and comparing the output ICV' to the ICV transmitted with the message. If ICV' is not equal to ICV, the received MSDU is not passed to LLC and and error indication is sent to MAC management to be counted in aICV_Error_Count.

## Updated Section 5.2.5

Figure 5-4 shows the encrypted MSDU as constructed by the WEP.

The WEP ICV = 32 bits. The expanded MSDU shall include a 32 bit IV field immediately preceding the MSDU. This field shall contain two~~one~~ sub-fields: A 3~~4~~-octet field that contains the initialization vector and a 1-octet pad field to maintain even-octet alignment of the encrypted payload.

The WEP mechanism is invisible to entities outside the 802.11 MAC.




Note: The encipherment process has expanded the original MSDU by 8 Octets, 4 for the Initialization Vector (IV) field and 4 for the Integrity Check Value (ICV). The ICV is calculated on the Data field only.

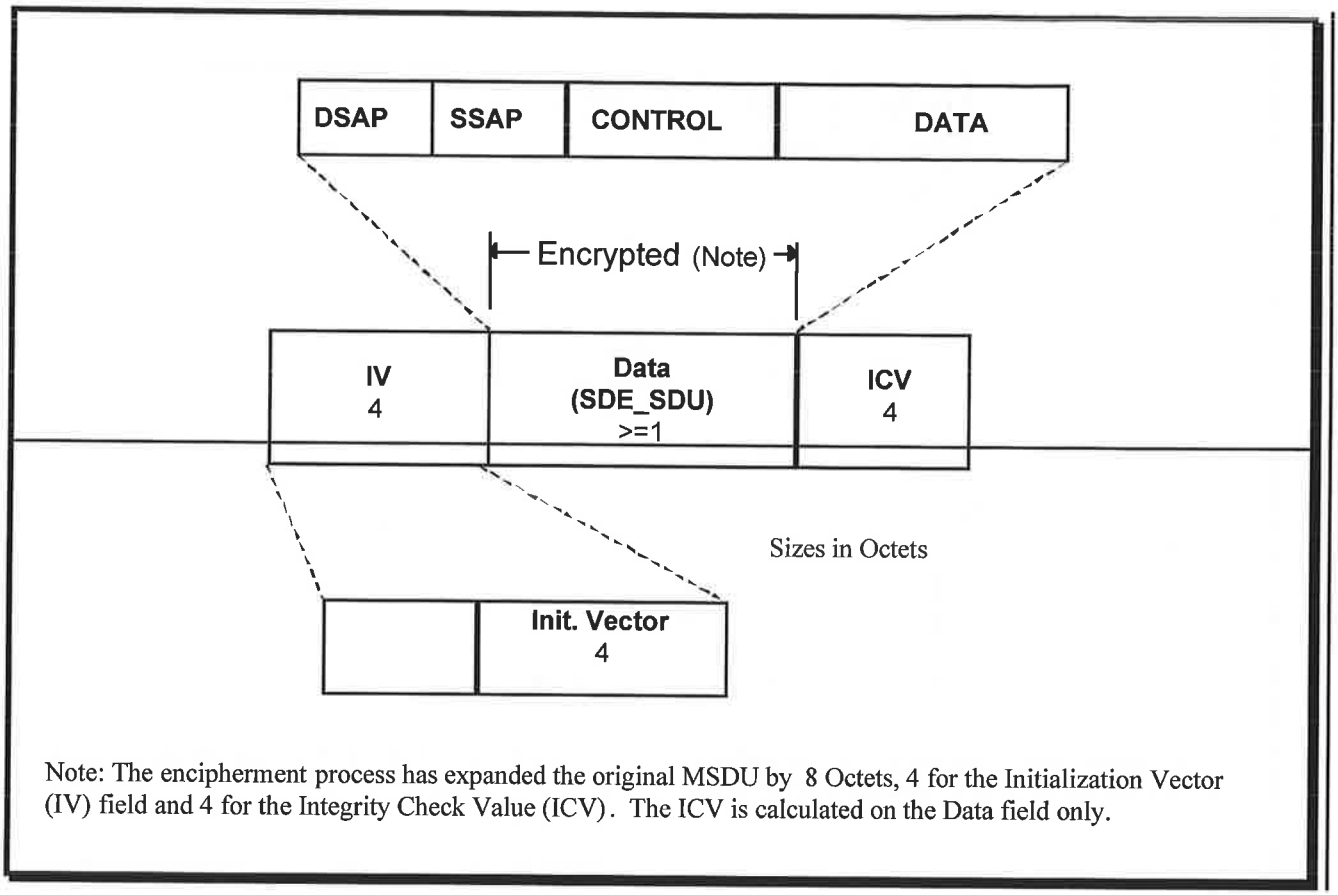



**Figure 5-4: Construction of expanded WEP MSDU**

## Updated Section 5.3

The 802.11 security mechanisms are controlled via the MAC management path and related MIB variables. This section gives an overview of the security related MIB variables and how they are used. For details of the MIB variable definitions, refer to section 8.4.~~7.X.~~

## Updated Section 5.3.1

The type of authentication invoked when authentication is attempted is controlled by the MIB variable aAuthentication_Type. This variable may have the following values:

1 = Open System
2 = Shared Key
All other values are reserved.