Propagating Cipher Feedback mode

1st revision

By Henrick Hellström, StreamSec HB
Copyright © 2001 StreamSec HB

e-mail: henrick@streamsec.se
home page: http://www.streamsec.com

This document is available at http://www.streamsec.com/pcfb1.pdf

SPECIFICATION

Secrecy

PCFB is a two-way error propagating strictly serialized mode. It might be defined through a comparison with the CFB-m/n mode:

General CFB-m/n, single block encryption (without message authentication):
Input: n-bit vector V, m-bit plain text P
Output: n-bit vector V, m-bit cipher text C
1. T <- F(V)
2. C <- P xor (T mod 2^m)
3. V <- (V>>m) or (C<<(n-m))

General PCFB-m/n, single block encryption (without message authentication):
Input: n-bit vector V, m-bit plain text P
Output: n-bit vector V, m-bit cipher text C
1. T <- F(V)
2. C <- P xor (T mod 2^m)
3. V <- (T>>m) or (C<<(n-m))

Remarks:
If $m = n$, then the modes are identical. It is assumed that $m < n$, and that there is a positive integer k such that $n = m(2^k)$.

Here is the algorithm expanded from block size to message size:

General PCFB-m/n, encryption (without message authentication):
Input: n-bit vector IV, L x n-bit plain text $P = (P_1, P_2, \ldots P_L)$.
Output: n-bit vector IV, L x n-bit cipher text $C = (C_1, C_2, \ldots C_L)$.
1. V_0 <- IV
2. For j <- 1 to L do:
2.1. T <- $E_K(V_{j-1})$
2.2. C_j <- P_j xor (T mod 2^m)
2.3. V_j <- (T>>m) or (C_j<<(n-m))

General PCFB-m/n, decryption (without message authentication):
Input: n-bit vector IV, L x n-bit cipher text $C = (C_1, C_2, \ldots C_L)$.
Output: n-bit vector IV, L x n-bit plain text $P = (P_1, P_2, \ldots P_L)$.
1. V_0 <- IV
2. For j <- 1 to L do:
2.1. T <- $E_K(V_{j-1})$
2.2. P_j <- C_j xor (T mod 2^m)
2.3. V_j <- (T>>m) or (C_j<<(n-m))

Provable security:
Define F_0, F_1 such that $F_0(V) = F(V) \bmod 2^m$, and $F_1(V) = \lfloor F(V)/2^m \rfloor$. Obviously, it would be trivial to transform any distinguisher for either F_0 or F_1 into a distinguisher for F, so both are at least as secure as F. Now, let $G_X(C) = F_0(X||C)$, where X is n-m bits wide and C is m bits wide. It is a reasonable assessment that G_X has a strictly higher security level than F. Furthermore, since F_1 is at least as secure as F, the value X might be modeled as an n-m bits wide random value. Since there is a positive integer k such that $n = m(2^k)$ and F might be assumed to be a secure 128-bit block cipher, this implies that the security level of PCFB mode should be a function of the security level of G_X independent of L, as long as the key data and IV remains secret.

Remarks:

There is no "stateless" version of PCFB mode. Each message must be assigned a secret and probabilistically unique value of IV).

The values of $V_j \bmod 2^{n-m}$ must remain secret, not only to retain the high level of security, but also because an attacker might otherwise manipulate the cipher text and successfully prevent error propagation:

Suppose that an attacker knows the value of $V_L = v$ after the cipher text X has been encrypted. Suppose also that the legitimate sender has earlier transmitted the cipher text $Y||Z$ and that the attacker knows that $V_j \equiv v \pmod{2^{n-m}}$ after Y had been decrypted. If the attacker prevents X from reaching it's destination and makes the recipient decrypt $X'||Z$ instead, where X' is equal to X except that the last m bits are equal to those of Y, then the Z part will still decrypt into the same plain text as it did when the cipher text $Y||Z$ was decrypted. The bit difference between the decryption of X' and the plain text encrypted into X will in this context be equal to the bit difference between the cipher text X' and X.

Scott Fluhrer discovered a known plain text attack based on this fact:

"… [F]irst thing you need to do is find two places in two different messages where V_{j-1} (and hence T) has zero differential. You are aided by two observations:

- The ciphertext for the immediately previous iteration in both places will have zero differential (because the previous ciphertext block contributed to V_{j-1}.

- The plaintext xor the ciphertext for this iteration will have zero differential (because the plaintext xor the ciphertext is a function of T).

Now, you aren't assured that, if you find two places where the above holds, V_{j-1} does have zero differential, however, if F does behave like a random function, the odds are good (p > 0.5, I'm pretty sure). And, since we're the attacker, that's "good enough".

Assuming the V_{j-1} values act as if they were random and independent, we'll need to dig through about 2^{64} of such values before we expect to find such a pair (birthday paradox).

And, if it is such a place, then by pasting the first half of the first ciphertext to the second half of the second ciphertext, we get a message that will decrypt to the first half of the first plaintext pasted onto the second half of the second plaintext." (Private email, March 18, 2001)

It is theoretically possible to mount an adaptive chosen cipher text attack against the error propagation when the values of V_j are unknown, but the expected effort would be $2^{(n-m)/2}(1 + 2^{-m}(n/(2m) - 2^{-(m-n)/2}))$ after the birthday paradox has been accounted for: Let C_j be equal to the last m bits of Y and let C_{j+1} be equal to the first m bits of Z. The probability is 2^{m-n} that V_j has the desired value, and the probability is 2^{-m} that C_{j+1} will decrypt into the desired value. If the recipient decrypts C_{j+1} into the desired plain text, let C_{j+2} be equal to the second m bits of Z. If not, let C_{j+2} be equal to the last m bits of Y and start over. Note that by the birthday paradox the probability is $(1-2^{-(n-m)/2})$ that the recipient decrypts C_{j+1} into the desired plain text even if $(V_j \bmod 2^{n-m})$ happens not to have the desired value, provided that F is a perfect random permutation. It takes an average of $n/(2m)$ trial decryptions to rule out a false hit. Since V_j is expected to have the desired value by chance after $2^{(n-m)/2}(1 + 2^{-m}(n/(2m) - 2^{(m-n)/2}))$ trial decryptions, the proposition follows.

While an effort of approximately 2^{-32} (for PCFB-64/128) might seem very low, one must note that the attack is successful only at the expense of passing an equal amount of garbage plain text blocks to the legitimate recipient.

F is supposed to be a function over $\{0,1\}^n$ (e.g. a keyed encryption function) that fulfills the criteria stated in the security proof. Any secure block cipher with a random key would do, but there might be other choices of F that are of interest.

Authentication

Since PCFB mode is error propagating in both encryption mode and decryption mode, and because there is no feasible way to manipulate the propagation, no additional cryptographic primitives are needed to provide message authentication.

Added Redundancy Explicit Authentication (AREA) PCFB-*m/n*, encryption:

Input: n-bit vector IV, L x m-bit plain text $P = (P_1, P_2, \ldots P_L)$.
Output: n-bit vector IV, L x m-bit cipher text $C = (C_0, C_1, \ldots C_{L+128/m})$.
1. P_0 <- L, $(P_{L+1}||\ldots||P_{L+128/m})$ <- L
2. V_{-1} <- IV
3. For j <- 0 to L+128/m do:
3.1. T <- F(V_{j-1})
3.2. C_j <- P_j xor (T mod 2^m)
3.3. V_j <- (T>>m) or (C_j<<(n-m))
4. IV <- $V_{L+128/m}$
5. Return(*True*)

Added Redundancy Explicit Authentication (AREA) PCFB-*m/n*, decryption and verification:

Input: n-bit vector IV, L x m-bit plain text $C = (C_0, C_1, \ldots C_{L+128/m})$.
Output: n-bit vector IV, L x m-bit cipher text $P = (P_1, P_2, \ldots P_L)$.
1. If L < 1 then Return(*False*)
2. V_{-1} <- IV
3. For j <- 0 to L+128/m do:
3.1. T <- F(V_{j-1})
3.2. P_j <- C_j xor (T mod 2^m)
3.3. V_j <- (T>>m) or (C_j<<(n-m))
4. IV <- $V_{L+128/m}$
5. If $P_0 = L$, $(P_{L+1}||\ldots||P_{L+128/m}) = L$ then Return(*True*) else Return(*False*)

Remark 1:

The main advantage of AREA-PCFB over other integrity authenticating modes is that it only requires a single key set up. Furthermore, the AREA scheme is very simple to understand and to implement, since it relies entirely on the error propagation of PCFB mode.

Remark 2:

The original specification of PCFB-mode mentioned a Plain Text Redundancy Authentication scheme, which has now been discarded from the specification. Instead of sending the value of L encrypted at the beginning and end of the message, this scheme would rely on the recipient to check the redundancy of the plain text in order to detect forgeries. The plain text redundancy authentication scheme is an interesting idea and a subject for further research, but no generalized implementation is yet known.