THE WEST BENGAL APPELLATE TRIBUNAL RULES, 1972 [See Rule 3(1)]

Mã khối *n* bit (*n*-bit block cipher)

Mã khối với tính chất là các khối của bản rõ và bản mã đều có độ dài n bit.

[Nguồn: TCVN 11367-1:2016 (ISO/IEC 18033-1), 2.23]

3.10

Đệm (Padding)

Các bit bổ sung vào một xâu dữ liệu.

3.11

Bản rõ (plaintext)

Thông tin chưa được mã hóa.

[Nguồn: TCVN 11367-1:2016 (ISO/IEC 18033-1), 2.24]

3.12

Biến khởi tạo (starting variable - *SV*)

Biến có thể được dẫn xuất từ một số giá trị khởi tạo và được sử dụng để xác định điểm bắt đầu của các chế độ hoạt động.

CHÚ THÍCH 1 với đầu vào: Biến khởi tạo (*SV*) được sử dụng trong tài liệu này giống với giá trị khởi tạo hoặc véc tơ khởi tạo (*IV*) được sử dụng trong một vài tiêu chuẩn quốc tế. Nếu biến khởi tạo được gọi đến trong tiêu chuẩn này được dẫn xuất từ một vài giá trị khởi tạo, khi đó nó cần được mô tả trong mọi ứng dụng của các chế độ hoạt động. Một phương pháp để dẫn xuất biến khởi tạo từ giá trị khởi tạo không được định nghĩa trong tiêu chuẩn này.

4 Ký hiệu, chữ viết tắt và chú thích

4.1 Ký hiệu và chữ viết tắt

C	Khối mã.
CTR	Giá trị đếm.
dK	Hàm giải mã của mã khối với khóa K. Quan hệ giải mã được xác định như sau: $$P = dK(C)$$ Trong đó: - P khối bản rõ - C khối bản mã - K khóa
E	Biến trung gian.
eK	Hàm mã hóa của mã khối với khóa K. Quan hệ mã hóa được xác định như sau: $$C = eK(P)$$

	Trong đó: - P khối bản rõ - C khối bản mã - K khóa
F	Biến trung gian.
FB	Bộ đệm phản hồi.
i	Biến lặp.
j	Kích thước của biến bản rõ/bản mã.
K	Khóa mã.
n	Chiều dài tính bằng bit của khối bản rõ/bản mã của mã khối.
m	Số khối bản mã được lưu trữ.
P	Khối bản rõ.
q	Số lượng các biến bản rõ/bản mã.
r	Kích thước của bộ đệm phản hồi.
SV	Biến khởi tạo.(xem Mệnh đề 5)
X	Khối đầu vào mã khối.
Y	Khối đầu ra mã khối.
|	Phép ghép các xâu bit.
$(a_1, a_2, \ldots, a_m)$	Mảng một chiều của các bit. Ví dụ : A $=(a_1, a_2, \ldots, a_m)$ và B $=(b_1, b_2, \ldots, b_m)$ là hai mảng m-bit được đánh số từ 1 đến m. Tất cả các mảng bit được ghi với chỉ số 1 ở vị trí trái nhất. Khi biểu diễn một mảng bit như một số nguyên, bit trái nhất là bit có trọng số cao nhất.
$\oplus$	Toán hạng cộng modulo 2, cũng được biết đến như một hàm or, được ký hiệu bởi $\oplus$. Phép toán này áp dụng đối với mảng A và mảng B có cùng độ dài và được xác định như sau: $A \oplus B = (a_1 \oplus b_1, a_2 \oplus b_2, \ldots, a_m \oplus b_m)$

4.2 Chú thích

$a \bmod n$	Với các số nguyên a và n, $a \bmod n$ kí hiệu phần dư (không âm) trong phép chia n cho a. Một cách tương đương là nếu $b = a \bmod n$ thì b là số nguyên duy

	nhất thỏa mãn các điều kiện sau: - $0 \leq b < n$, và - $(b - a)$ là bội số nguyên của n
~	Toán hạng lựa chọn j bit trái nhất của mảng $A = (a_1, a_2, \ldots, a_m)$ để sinh ra một mảng j bit được viết như sau: $(j \sim A) = (a_1, a_2, \ldots, a_j)$ Toán hạng lựa chọn j bit phải nhất của mảng $A = (a_1, a_2, \ldots, a_m)$ để sinh ra một mảng j bit được viết như sau: $(A \sim j) = (a_{m-j+1}, a_{m-j+2}, \ldots, a_m)$ với $1 \leq j \leq m$
$1(t)$	Xâu t-bit với tất cả các bit có giá trị là 1.
S_t	Cho trước biến X m bit và biến F t bit với $1 \leq t \leq m$, kết quả của hàm dịch $S_t(X\|F)$ để tạo ra biến m-bit $S_t(X\|F) = (x_{t+1}, x_{t+2}, \ldots x_m, f_1, f_2, \ldots, f_t)$ $(t < m)$ $S_t(X\|F) = (f_1, f_2, \ldots, f_t)$ $(t = m)$ Kết quả là dịch các bit của mảng X sang trái t vị trí loại bỏ $x_1, x, \ldots, x_t$ và đặt mảng F vào t vị trí phải nhất của X. Khi $t = m$ thì thay thế hoàn toàn X bởi F.

5 Các yêu cầu

Với tất cả các chế độ được mô tả, bản mã được lựa chọn từ TCVN 11367-3 (ISO/IEC 18033-3) và/hoặc ISO/IEC 29192-2.

Với chế độ sách mã điện tử giá trị các tham số không được thêm vào mà cần được lựa chọn. Với chế độ hoạt động móc xích khối mã (CBC) (xem điều 7), sẽ phải chọn một tham số m. Đối với chế độ hoạt động phản hồi bản mã (CFB) (xem điều 8), cần phải chọn bộ 3 tham số r, j và k. Đối với chế độ phản hồi đầu ra (OFB) (xem điều 9) và chế độ bộ đếm (CTR) (xem điều 10), cần phải chọn tham số j. Khi một trong những chế độ này được sử dụng, cùng một (nhiều) giá trị tham số tương tự được lựa chọn và được sử dụng bởi tất cả các bên giao tiếp. Các tham số này không cần phải giữ bí mật.

Tất cả các chế độ hoạt động được đặc tả trong tiêu chuẩn này yêu cầu các bên mã hóa và giải mã xâu dữ liệu phải chia sẻ khóa bí mật K sử dụng cho mã khối. Tất cả các chế chộ hoạt động ngoài chế độ sách mã điện tử (ECB) cũng yêu cầu các bên chia sẻ một giá trị khởi tạo SV, trong đó độ dài của SV phụ thuộc vào chế độ sử dụng của mã khối. Giá trị của biến khởi tạo thường khác nhau đối với mỗi xâu dữ liệu được mã hóa sử dụng một khóa cụ thể. Cách các khóa và các giá trị khởi tạo được quản lý và phân phối nằm ngoài phạm vi áp dụng của tiêu chuẩn này. Tuy nhiên, hướng dẫn an toàn quan trọng có liên quan đến biến khởi tạo được cung cấp trong Phụ lục B.

Bên mã hóa và tất cả bên giải mã tiềm năng thỏa thuận một phương pháp đệm ngoại trừ thông điệp được mã hóa luôn là bội của các bit độ dài m ($m = n$ trong chế độ ECB và CBC, $m = j$ trong chế độ CFB, OFB và CTR) hoặc ngoại trừ chế độ không yêu cầu đệm. Vì vậy, phương pháp đệm nhận một xâu bit P có độ dài tùy ý làm đầu vào (không rỗng) và đưa ra đầu ra là một chuỗi khối bản rõ m–bit. Ngoại trừ trường hợp độ dài thông điệp rõ được cố định hoặc ngược lại được xác định bởi ứng dụng

khi đó phương pháp đệm có tính chất là không có 2 xâu bit khác biệt đầu vào có thể đưa ra cùng một chuỗi khối *m*-bit, nghĩa là các xâu dữ liệu sau khi được đệm luôn bảo đảm tính duy nhất.

Tiêu chuẩn này khuyến nghị các phương pháp đệm sau: xâu bit *P* (không rỗng) được đệm phải với một đơn bit 1. Xâu kết quả khi đó được đệm phải với một vài bit 0 (có thể rỗng) là cần thiết để nhận được toàn bộ độ dài của một xâu (theo bit) là một số nguyên dương bội của *m*.

CHÚ THÍCH Phương pháp đệm này giống với phương pháp đệm 2 trong cả hai tiêu chuẩn ISO/IEC 9797-1 và TCVN 11816-1:2017 (ISO/IEC 10118-1) ngoại trừ việc chỉ xem xét các xâu bit không rỗng.

6 Chế độ sách mã điện tử (ECB)

6.1 Sơ bộ

Chế độ ECB được sử dụng với phương pháp đệm đã thỏa thuận. Nếu độ dài bit của khối bản rõ có thể không là bội của khối mã khối độ dài *n*.

Các biến được sử dụng bởi chế độ mã hóa ECB là:

a) Các biến đầu vào

1) Một xâu các bit bản rõ *P*.

2) Khóa *K*.

b) Các biến đầu ra là xâu các bit mã *C*

6.2 Mã hóa

Nếu một phương pháp đệm được sử dụng, khi đó trước hết đệm xâu dữ liệu *P* để nhận được một dãy *q* khối bản rõ $P_1, P_2, \dots, P_q$, mỗi khối có độ dài *n*-bit. Nếu không chỉ đơn giản là chia xâu dữ liệu *P* thành một dãy *q* khối bản rõ $P_1, P_2, \dots, P_q$ mỗi khối có độ dài *n*-bit.

Mã hóa của chế độ ECB như sau:

$C_i = eK(P_i)$ với $i = 1,2, \dots, q$.

6.3 Giải mã

Giải mã của chế độ ECB như sau:

$P_i = dK(C_i)$ với $i = 1,2, \dots, q$.

Nếu một phương pháp đệm không được sử dụng, khi đó *P* bằng với phép ghép của $P_1, P_2, \dots, P_q$. Nếu một phương pháp đệm được sử dụng, áp dụng phép nghịch đảo của phương pháp đệm đã được thỏa thuận với chuỗi *q* khối bản rõ $P_1, P_2, \dots, P_q$ để nhận được xâu bản rõ đã giải mã *P*.

CHÚ THÍCH Việc loại bỏ đệm duy nhất được đảm bảo do hạn chế trong việc lựa chọn phương pháp đệm được đưa ra trong Điều 5.

7 Chế độ móc xích khối mã (CBC)

7.1 Sơ bộ

Mã hóa và giải mã trong chế độ CBC được xác định trong 7.2 và 7.3 sử dụng phương pháp đệm đã được thỏa thuận, nếu độ dài bit của bản rõ có thể không phải là bội của khối mã khối độ dài *n*. Nếu không có phương pháp đệm nào được thỏa thuận khi đó bản rõ là toàn bộ số lượng của các khối hoàn chỉnh trừ khi nó được thỏa thuận dùng phương pháp được mô tả trong 7.4.

Chế độ CBC được xác định bởi một tham số trung gian $m > 0$, số lượng các tiến trình mã hóa độc lập có thể được thực hiện song song.

CHÚ THÍCH 1 Giá trị của m có thể nhỏ (thông thường $m = 1$) và hiếm khi vượt quá 1024.

Các biến được sử dụng cho việc mã hóa bởi chế độ CBC:

a) Các biến đầu vào

1) Một xâu các bit bản rõ *P*.

2) Khóa *K*.

3) Dãy m giá trị khởi tạo có độ dài n bit $SV_1, SV, \ldots, SV_m$ mỗi biến n-bit. Hướng dẫn an toàn cho các giá trị của biến khởi tạo được đề cập trong phụ lục B.

CHÚ THÍCH 2 Nếu $m = 1$ không có phương pháp đệm được thêm, sau đó chế độ này tương thích với chế độ CBC được mô tả trong ấn bản lần hai của tiêu chuẩn này.

b) Các biến đầu ra là một xâu các bit bản mã *C*.

7.2 Mã hóa

Nếu một phương pháp đệm được sử dụng, khi đó trước hết đệm xâu dữ liệu *P* để nhận được một dãy *q* khối bản rõ $P_1, P_2, \ldots, P_q$, mỗi khối có độ dài *n*-bit. Nếu không chỉ đơn giản là chia xâu dữ liệu *P* thành một dãy *q* khối bản rõ $P_1, P_2, \ldots, P_q$ mỗi khối có độ dài *n*-bit.

Mã hóa của chế độ CBC như sau:

$$C_i = eK(P_i \oplus SV_i), 1 \leq i \leq min(m, q)$$

Nếu $q > m$, các dãy con khối bản rõ theo sau được mã hóa như sau:

$$C_i = eK(P_i \oplus C_{i-m}), m + 1 \leq i \leq q$$

Tại bất kỳ thời điểm nào trong quá trình tính toán, giá trị của m khối bản mã gần nhất cần được lưu trữ. Ví dụ thường sử dụng "bộ đệm phản hồi" FB (xem hình C.2).

Quá trình này được thể hiện ở bên trái hình C.2.

7.3 Giải mã

Giải mã của chế độ CBC như sau:

$$P_i = dK(C_i) \oplus SV_i, 1 \leq i \leq min(m, q)$$

Nếu $q > m$, các dãy con khối bản mã theo sau được mã hóa như sau:

$$P_i = dK(C_i) \oplus C_{i-m}, m + 1 \leq i \leq q$$

Tại bất kỳ thời điểm nào trong quá trình tính toán, giá trị của m khối bản mã gần nhất cần được lưu trữ. Ví dụ thường sử dụng "bộ đệm phản hồi" FB (xem hình C.2).

Quá trình này được thể hiện ở bên phải hình C.2.

Nếu một phương pháp đệm không được sử dụng, khi đó P bằng với phép ghép của $P_1, P_2, \ldots, P_q$.

Nếu một phương pháp đệm được sử dụng, áp dụng phép nghịch đảo của phương pháp đệm đã được thỏa thuận với chuỗi q khối bản rõ $P_1, P_2, \ldots, P_q$ để nhận được xâu bản rõ đã giải mã P.

CHÚ THÍCH Việc loại bỏ đệm duy nhất được đảm bảo do hạn chế trong việc lựa chọn phương pháp đệm được đưa ra trong Điều 5.

7.4 Vô hiệu hóa việc mở rộng bản mã

7.4.1 Tổng quan

Để tránh việc mở rộng bản mã do đệm có thể thực hiện phương pháp CTS (*ciphertext sealing*). Với phương pháp này, nếu khối bản rõ cuối cùng là một phần, khi đó một vài bit 0 được thêm vào để hoàn chỉnh nó (nghĩa là phương pháp đệm 1 trong ISO/IEC 9797-1:2011) và kết quả bản rõ được mã như trên. Bất cứ mở rộng nào do đệm bit gây ra đều tránh được vì các bit gần cuối của khối bản mã đều được loại bỏ vì chúng có thể được khôi phục từ bản mã của khối mã cuối cùng.

Trường hợp đặc biệt, nếu m = 1 (không có trung gian) và bản rõ đã được đệm thì p bit phải nhất của khối bản rõ P_q là các bit đệm 0. Khi đó, p bit phải nhất của khối bản rõ gần cuối cùng C_{q-1} không được truyền đi nhưng có thể khôi phục như p bit phải nhất của $dK(C_q)$.

CHÚ THÍCH Chế độ CTS không được áp dụng nếu số lượng các bit của bản rõ nhỏ hơn n.

7.4.2 Ba biến thể "*ciphertext stealing*" của CBC

7.4.2.1 Sơ bộ

Tiêu chuẩn này định nghĩa ba biến thể "*ciphertext stealing*" (CBC_CS) của chế độ CBC. Cả ba là các biến thể của chế độ mã hóa và giải mã cơ bản CBC được định nghĩa ở trên sử dụng phương pháp đệm 1 trong tiêu chuẩn ISO/IEC 9797-1:2011 [1]. Các biến thể CBC_CS này chỉ khác nhau với chế độ CBC cơ bản trong cách hai khối mã C_{q-m} và C_q được thực hiện sau giải mã và trước mã hóa.

Mặc dù, biến thể CBC_CS có thể được sử dụng khi tham số trung giam $m>1$ như chú thích trong 7.1, cụ thể, m = 1 trong trường hợp 2 khối bản mã C_{q-m} và C_q là hai khối bản mã cuối cùng C_{q-1}và C_q. Mô tả dưới đây chỉ áp dụng cho trường hợp m = 1 nhưng có thể ngoại suy với trường hợp m > 1.

7.4.2.2 Xử lý sau mã hóa CBC (rút gọn bản mã)

Sau khi mã hóa băn mã $C_1 \mid C_2 \mid \ldots \mid C_{q-1} \mid C_q$ được rút gọn (rút gọn độ dài) bởi số lượng các bit đệm p ($0 \le p < n$). Nếu không đệm (p = 0) khi đó việc rút gọn không xảy ra nhưng trường hợp đặc biệt dưới đây biến thứ ba CBC_CS vẫn sửa đổi bản mã bằng cách hoán đổi hai khối bản mã cuối cùng C_{q-1}và C_q

Với tất cả các biến thể CBC_CS xác định, $C_{q-1}^* = (n-p) \sim C_{q-1}$. Ba biến thể CBC_CS được xác định như sau:

- Biến thể CBC_CS thứ nhất, hai khối bản mã cuối cùng $C_{q-1} \mid C_q$ được thay thế bởi $C_{q-1}^* \mid C_q$.

- Biến thể CBC_CS thứ hai, hai khối bản mã cuối cùng $C_{q-1} \mid C_q$ được thay thế bởi $C_q \mid C_{q-1}^*$ chỉ khi bản rõ được đệm (ngược lại sẽ không thay đổi)

- Biến thể CBC_CS thứ ba, hai khối bản mã cuối cùng $C_{q-1} \mid C_q$ luôn luôn thay thế bởi $C_q \mid C_{q-1}^*$

7.4.2.3 Tiền xử lý giải mã CBC (khai triển bản mã)

Trước khi giải mã, bản mã rút gọn nhận được được khai triển thành toàn bộ các khối n-bit.

Giả sử số lượng các bit của bản mã nhận được là $(q-1)n+d$ trong đó $0 \le d < n$ và nếu $d > 0$ thì giả sử $p = n - d$ (trong trường hợp $p \neq 0$).

Nếu q >1 khi đó $C_{q-1}^{*} = (n-p) \sim C_{q-1}$

- Với biến thể thứ nhất CBC_CS

- Nếu d = 0 thì bản mã nhận được là các khối n bit và tiền xử lý không được yêu cầu.

- Nếu d >0 thì bản mã nhận được không phải là các khối n bit và tiền xử lý được yêu cầu. Trong trường hợp này $n + d$ bit phải nhất của bản mã nhận được được phân tích thành $C_{q-1}^{*} | C_q$, trong đó C_{q-1}^{*} là một khối d-bit và C_q là một khối n-bit và khối n-bit C_{q-1} được tạo thành là $C_{q-1}^{*} | (dK(C_q) \sim p)$.

- Với biến thể thứ hai CBC_CS

- Nếu d = 0 thì bản mã nhận được là toàn bộ các khối n bit và tiền xử lý không được yêu cầu.

- Nếu d >0 thì bản mã nhận được không phải là các khối n bit và tiền xử lý được yêu cầu. Trong trường hợp này $n + d$ bit phải nhất của bản mã nhận được được phân tích thành $C_q | C_{q-1}^{*}$ trong đó C_{q-1}^{*} là một khối d-bit và C_q là một khối n-bit và khối n-bit C_{q-1} được tạo thành là $C_{q-1}^{*} | (dK(C_q) \sim p)$.

- Với biến thể thứ ba CBC_CS, tiền xử lý luôn được yêu cầu

- Nếu d = 0 thì bản mã nhận được là toàn bộ các khối n bit, $2n$ bit của bản mã nhận được được phân tích thành $C_q | C_{q-1}$. Vì vậy tiền xử lý yêu cầu hai khối phải nhất được hoán đổi.

- Nếu d >0 thì bản mã nhận được không phải là các khối n bit và tiền xử lý được yêu cầu. Trong trường hợp này $n + d$ bit phải nhất của bản mã nhận được được phân tích thành $C_q | C_{q-1}^{*}$ trong đó C_{q-1}^{*} là một khối d-bit và C_q là một khối n-bit và khối n-bit C_{q-1} được tạo thành là $C_{q-1}^{*} | (dK(C_q) \sim p)$.

Mã hóa CBC khi mã, bản mã $C_1 | C_2 | \dots | C_{q-1} | C_q$ và nếu d >0 thì xóa bỏ p bit đệm phải nhất trong khối bản rõ kết quả.

CHÚ THÍCH NIST/SP 800-38A định nghĩa ba biến giống nhau của "*ciphertext stealing*" (các biến thể chỉ khác nhau trong trật tự của các bit bản mã)

8 Chế độ phản hồi bản mã (CFB)

8.1 Sơ bộ

Mã hóa và giải mã trong chế độ CFB được xác định trong 8.2 và 8.3 sử dụng phương pháp đệm đã được thỏa thuận. Nếu độ dài bit của bản rõ có thể không phải là bội của tham số j. Nếu không có phương pháp đệm nào được thỏa thuận khi đó bản rõ là toàn bộ số lượng của các khối j bit trừ phương pháp được sử dụng mô tả trong 8.4.

Chế độ CFB được xác định bởi 3 tham số:

- Kích thước của bộ đệm phản hồi, r, trong đó $n \le r \le 1024n$

- Kích thước của biến phản hồi, k, trong đó $1 \le k \le n$

- Kích thước của biến bản rõ, j, trong đó $1 \le j \le k$

CHÚ THÍCH

a) $r - k$ không bị ràng buộc bởi n bằng bất kỳ cách nào, tức là $r - k$ có thể nhỏ hơn, bằng hoặc lớn hơn n. Hình C.3 chỉ ra trường hợp đặc biệt khi $r - k > n$.

b) Nếu $r = n$ thì chế độ này tương thích với phiên bản chế độ CFB được mô tả trong ấn bản lần 1 của tiêu chuẩn này.

c) Cận trên r, tức là $r \leq 1024n$ được chọn vì nó cung cấp một cận trên thực tế dựa trên số lượng bộ vi xử lý phần cứng.

Khuyến cáo trong chế độ CFB cần sử dụng giá trị j và k bằng nhau (xem Điều B.4.1).

Chế độ CFB sẽ được sử dụng với một phương thức đệm đã được thỏa thuận nếu chiều dài tính theo bit của bản rõ không phải là bội số của tham số j.

Số của j-bit khối bản rõ (sau khi thêm nếu một phương thức đệm được sử dụng) là q.

Các biến được sử dụng trong chế độ CFB:

a) Các biến đầu vào

1) Một xâu các bit bản rõ P.

2) Khóa K.

3) Giá trị khởi tạo SV độ dài r bit. Tham khảo phụ lục B các hướng dẫn an toàn liên quan đến giá trị của SV.

b) Các kết quả trung gian:

1) Dãy q khối đầu vào của mã khối độ dài n bit $X_1, X_2, \ldots, X_q$.

2) Dãy q khối đầu ra của mã khối độ dài n bit $Y_1, Y_2, \ldots, Y_q$.

3) Dãy q biến độ dài j bit $E_1, E_2, \ldots, E_q$.

4) Dãy $q - 1$ biến phản hồi độ dài k bit $F_1, F_2, \ldots, F_{q-1}$.

5) Dãy q nội dung bộ đệm phản hồi độ dài r bit $FB_1, FB_2, \ldots, FB_q$.

c) Các biến đầu ra là xâu bit bản mã C.

8.2 Mã hóa

Nếu một phương pháp đệm được sử dụng, khi đó trước hết đệm xâu dữ liệu P để nhận được một dãy q khối bản rõ $P_1, P_2, \ldots, P_q$, mỗi khối có độ dài j bit. Nếu không chỉ đơn giản là chia xâu dữ liệu P thành một dãy q khối bản rõ $P_1, P_2, \ldots, P_q$ mỗi khối có độ dài j bit.

Bộ đệm phản hồi FB được gán giá trị khởi tạo.

$$FB_1 = SV$$

Phép mã hóa mỗi biến bản rõ thực hiện trong 6 bước sau:

a) $X_i = n{\sim}FB_i$ (Lựa chọn n bit trái nhất của FB).

b) $Y_i = eK(X_i)$ (Sử dụng mã khối).

c) $E_i = j{\sim}Y_i$ (Lựa chọn j bit trái nhất của Y_i).

d) $C_i = P_i \oplus E_i$ (Tạo biến bản mã).

e) $F_i = I(k - j)|C_i$ (Tạo biến phản hồi).

f) $FB_{i+1} = S_k(FB_i|F_i)$ (Hàm dịch chuyển trên FB).

Các bước trên lặp với biến đếm $i = 1, 2, ..., q$, trong vòng lặp cuối dừng lại tại bước (d). Quá trình này được thể hiện ở bên trái hình C.3. j bit trái nhất của khối đầu ra Y của mã khối được sử dụng để mã hóa j-bit biến bản rõ theo phép cộng mô-đun 2. Các bit còn lại của Y được bỏ đi. Các bit biến bản rõ/bản mã có chỉ số từ 1 đến j.

Biến bản mã được tăng thêm bằng cách dùng $k - j$ bit ở vị trí trái nhất để tạo biến phản hồi F độ dài k-bit. Sau đó các bit của bộ đệm phản hồi FB được dịch trái đi k vị trí và F được thêm vào vị trí k tận cùng bên phải để tạo ra giá trị mới của bộ đệm phản hồi FB. Trong phép toán dịch, k bit trái nhất của FB được bỏ đi. n bit trái nhất mới của FB được sử dụng như là đầu vào tiếp theo của X trong quá trình mã hóa.

8.3 Giải mã

Các biến được sử dụng trong giải mã cũng giống như trong mã hóa.

Bộ đệm phản hồi FB được gán giá trị khởi tạo.

$$FB_1 = SV$$

Phép giải mã mỗi biến bản mã thực hiện trong 6 bước sau:

a) $X_i = n{\sim}FB_i$ (Lựa chọn n bit trái nhất của FB).

b) $Y_i = eK(X_i)$ (Sử dụng mã khối).

c) $E_i = j{\sim}Y_i$ (Lựa chọn j bit trái nhất của Y_i).

d) $P_i = C_i \oplus E_i$ (Tạo biến bản rõ).

e) $F_i = I(k-j)|C_i$ (Tạo biến phản hồi).

f) $FB_{i+1} = S_k(FB_i|F_i)$ (Hàm dịch chuyển trên FB).

Các bước trên lặp với biến đếm $i = 1, 2, ..., q$, trong vòng lặp cuối dừng lại tại bước (d). Quá trình này được thể hiện ở bên phải hình C.3. j bit trái nhất của khối đầu ra Y của mã khối được sử dụng để giải mã j-bit biến bản mã theo phép cộng modulo 2. Các bit còn lại của Y được bỏ đi. Các bit biến bản rõ/bản mã có chỉ số từ 1 đến j.

Biến bản mã được tăng thêm bằng cách dùng $k - j$ bit ở vị trí trái nhất để tạo biến phản hồi F độ dài k-bit. Sau đó các bit của bộ đệm phản hồi FB được dịch trái đi k vị trí và F được thêm vào vị trí k tận cùng bên phải để tạo ra giá trị mới của bộ đệm phản hồi FB. Trong phép toán dịch, k bit trái nhất của FB được bỏ đi. n bit trái nhất mới của FB được sử dụng như là đầu vào tiếp theo X của quá trình giải mã.

Nếu một phương pháp đệm không được sử dụng, khi đó P bằng với phép ghép của $P_1, P_2, ..., P_q$.

Nếu một phương pháp đệm được sử dụng, áp dụng phép nghịch đảo của phương pháp đệm đã được thỏa thuận với chuỗi q khối bản rõ $P_1, P_2, ..., P_q$ để nhận được xâu bản rõ đã giải mã P.

CHÚ THÍCH Việc loại bỏ đệm duy nhất được đảm bảo do hạn chế trong việc lựa chọn phương pháp đệm được đưa ra trong Điều 5.

8.4 Vô hiệu hóa việc mở rộng bản mã

Sử dụng chế độ CFB có thể tránh được việc mở rộng bản mã trong trường hợp khi bản rõ không phải là toàn bộ các khối có độ dài j bit. Việc áp dụng này, nếu bản rõ không phải là toàn bộ các khối j bit, khi

đó bản rõ được đệm với p bit ($0 \leq p < j$). Vì vậy, bản rõ trở thành toàn bộ các khối j bit nhưng phần đệm thì không nằm trong bản mã. giá trị thực tế của các bit đệm không có liên quan nên chúng có thể là các bit 0 nhị phân. Bản rõ được đệm được mã hóa như trên nhưng các bit p cuối cũng của khối bản mã cuối cùng được loại bỏ vì chúng không cần thiết.

Bên giải mã áp dụng cùng một phương pháp đệm để tạo ra bản mã mà tất cả các khối mã có độ dài j bit, giải mã bản mã ở trên và khi đó loại bỏ p bit cuối cùng của bản rõ.

CHÚ THÍCH 1 Có thể xây dựng phương pháp này mà không cần tham chiếu đến đệm bởi các yêu cầu đơn giản mà biến mã cuối cùng E_q được cắt ngắn để là độ dài bit của bản rõ/mã cuối cùng. Phụ lục B của ấn bản thứ 3 của tiêu chuẩn này chú thích j có thể sửa đổi cho phần cuối cùng của bản rõ. Đệm không được yêu cầu khi một khối bản rõ z bit, $z < j$ được mã bởi phép cộng modulo 2 chỉ với các bit z đầu tiên của biến E tương ứng.

CHÚ THÍCH 2 Nếu phương pháp đệm đã được thỏa thuận trong Điều 5, thì phương pháp này không được sử dụng. Nếu phương pháp đệm không được thỏa thuận thì phương pháp này được sử dụng.

9 Chế độ phản hồi đầu ra (OFB)

9.1 Sơ bộ

Chế độ OFB được xác định bằng một tham số j là kích thước biến bản rõ với $1 \leq j \leq n$.

Mã hóa và giải mã trong chế độ OFB được xác định trong 9.2 và 9.3 sử dụng phương pháp đệm đã được thỏa thuận nếu độ dài bit của bản rõ có thể không phải là bội của tham số j. Nếu không có phương pháp đệm nào được thỏa thuận khi đó bản rõ là toàn bộ số lượng của các khối j bit trừ phương pháp được sử dụng mô tả trong 9.4.

Số lượng của các khối bản rõ j bit là q (sau khi đệm nếu như phương pháp đệm được sử dụng).

Các biến được sử dụng trong chế độ mã hóa của OFB là:

a) Các biến đầu vào

1) Một xâu các bit bản rõ P.

2) Một Khóa K.

3) Giá trị khởi tạo SV độ dài n bit. Tham khảo phụ lục B các hướng dẫn an toàn liên quan đến giá trị của SV.

b) Các kết quả trung gian:

1) Dãy q khối đầu vào của mã khối độ dài n bit $X_1, X_2, \ldots, X_q$.

2) Dãy q khối đầu ra của mã khối độ dài n bit $Y_1, Y_2, \ldots, Y_q$.

3) Dãy q biến độ dài j bit $E_1, E_2, \ldots, E_q$.

c) Các biến đầu ra là xâu bit bản mã C.

9.2 Mã hóa

Nếu một phương pháp đệm được sử dụng, khi đó trước hết đệm xâu dữ liệu P để nhận được một dãy q khối bản rõ $P_1, P_2, \ldots, P_q$, mỗi khối có độ dài j bit. Nếu không chỉ đơn giản là chia xâu dữ liệu P thành một dãy q khối bản rõ $P_1, P_2, \ldots, P_q$ mỗi khối có độ dài j bit.

Khối đầu vào X được gán giá trị khởi tạo

$$X_1 = SV$$

Phép mã hóa mỗi biến bản rõ thực hiện trong 4 bước sau:

a) $Y_i = eK(X_i)$ (Sử dụng mã khối).

b) $E_i = j\sim Y_i$ (Chọn j bit trái nhất).

c) $C_i = P_i \oplus E_i$ (Tạo biến bản mã).

d) $X_{i+1} = Y_i$ (Phép phản hồi).

Các bước trên lặp với biến đếm $i = 1, 2, \ldots, q$, trong vòng lặp cuối dừng lại tại bước (c). Quá trình này được thể hiện ở bên trái hình C.4. Các biến bản mã và bản rõ có các bit với chỉ số từ *1* đến *j*.

Kết quả của mỗi lần sử dụng mã khối là Y_i được đưa trở lại thành giá trị tiếp theo của X, đặt là X_{i+1}. j bit trái nhất của Y_i được sử dụng để mã hóa biến đầu vào.

9.3 Giải mã

Các biến được sử dụng trong giải mã cũng giống như trong mã hóa.

Khối đầu vào X được gán giá trị khởi tạo

$$X_1 = SV$$

Phép giải mã mỗi biến bản mã thực hiện trong 4 bước sau:

a) $Y_i = eK(X_i)$ (Sử dụng mã khối).

b) $E_i = j\sim Y_i$ (Chọn j bit trái nhất).

c) $P_i = C_i \oplus E_i$ (Tạo biến bản mã).

d) $X_{i+1} = Y_i$ (Phép phản hồi).

Các bước trên lặp với biến đếm $i = 1, 2, \ldots, q$, trong vòng lặp cuối dừng lại tại bước (c). Quá trình này được thể hiện ở bên phải hình C.4. Các biến bản mã và bản rõ có các bit với chỉ số từ *1* đến *j*.

Kết quả của mỗi lần sử dụng mã khối là Y_i được đưa trở lại thành giá trị tiếp theo của X, đặt là X_{i+1}. j bit trái nhất của Y_i được sử dụng để giải mã biến đầu vào.

Nếu một phương pháp đệm không được sử dụng, khi đó P bằng với phép ghép của $P_1, P_2, \ldots, P_q$.

Nếu một phương pháp đệm được sử dụng, áp dụng phép nghịch đảo của phương pháp đệm đã được thỏa thuận với chuỗi q khối bản rõ $P_1, P_2, \ldots, P_q$ để nhận được xâu bản rõ đã giải mã P.

CHÚ THÍCH Việc loại bỏ đệm duy nhất được đảm bảo do hạn chế trong việc lựa chọn phương pháp đệm được đưa ra trong Điều 5.

9.4 Vô hiệu hóa việc mở rộng bản mã

Sử dụng chế độ OFB có thể tránh được việc mở rộng bản mã trong trường hợp khi bản rõ không phải là toàn bộ các khối có độ dài j bit. Việc áp dụng này, nếu bản rõ không phải là toàn bộ các khối j bit, khi đó bản rõ được đệm với p bit ($0 \le p < j$). Vì vậy, bản rõ trở thành toàn bộ các khối j bit nhưng phần đệm thì không nằm trong bản mã. giá trị thực tế của các bit đệm không có liên quan nên chúng có thể là các bit 0 nhị phân. Bản rõ được đệm được mã hóa như trên nhưng các bit p cuối cũng của khối bản mã cuối cùng được loại bỏ vì chúng không cần thiết.

Bên giải mã áp dụng cùng một phương pháp đệm để tạo ra bản mã mà tất cả các khối mã có độ dài j bit, giải mã bản mã ở trên và khi đó loại bỏ p bit cuối cùng của bản rõ.

CHÚ THÍCH 1 Có thể xây dựng phương pháp này mà không cần tham chiếu đến đệm bởi các yêu cầu đơn giản mà biến mã hóa cuối cùng E_q được cắt ngắn để là độ dài bit của bản rõ/bản mã cuối cùng. Phụ lục B của ấn bản thứ 3 của tiêu chuẩn này chú thích j có thể sửa đổi cho phần cuối cùng của bản rõ. Đệm không được yêu cầu khi một khối bản rõ z bit, $z < j$ được mã bởi phép cộng modulo 2 chỉ với các bit z đầu tiên của biến E tương ứng.

CHÚ THÍCH 2 Nếu phương pháp đệm đã được thỏa thuận trong Điều 5, thì phương pháp này không được sử dụng. Nếu phương pháp đệm không được thỏa thuận thì phương pháp này được sử dụng

10 Chế độ bộ đếm (CTR)

10.1 Sơ bộ

Chế độ CTR được xác định bằng một tham số j, kích thước biến bản rõ, trong đó $1 \leq j \leq n$.

Mã hóa và giải mã trong chế độ OFB được xác định trong 10.2 và 10.3 sử dụng phương pháp đệm đã được thỏa thuận nếu độ dài bit của bản rõ có thể không phải là bội của tham số j. Nếu không có phương pháp đệm nào được thỏa thuận khi đó bản rõ là toàn bộ số lượng của các khối j bit trừ phương pháp được sử dụng mô tả trong 10.4.

Số lượng của các khối bản rõ j bit là q (sau khi đệm nếu như phương pháp đệm được sử dụng).

Các biến được sử dụng trong chế độ *CTR* là:

a) Các biến đầu vào

1) Một xâu các bit bản rõ P.

2) Một Khóa K.

3) Giá trị khởi tạo SV độ dài n bit. Tham khảo phụ lục B các hướng dẫn an toàn liên quan đến giá trị của SV.

b) Các kết quả trung gian:

1) Dãy q khối đầu vào của mã khối có độ dài n bit $CTR_1, CTR_2, \ldots, CTR_q$.

2) Dãy q khối đầu ra của mã khối có độ dài n bit $Y_1, Y_2, \ldots, Y_q$.

3) Dãy q biến có độ dài j bit $E_1, E_2, \ldots, E_q$.

c) Các biến đầu ra tức là một dãy q biến bản mã $C_1, C_2, \ldots, C_q$, mỗi biến có độ dài j bit.

10.2 Mã hóa

Nếu một phương pháp đệm được sử dụng, khi đó trước hết đệm xâu dữ liệu P để nhận được một dãy q khối bản rõ $P_1, P_2, \ldots, P_q$, mỗi khối có độ dài j bit. Nếu không chỉ đơn giản là chia xâu dữ liệu P thành một dãy q khối bản rõ $P_1, P_2, \ldots, P_q$ mỗi khối có độ dài j bit.

Bộ đếm CTR được gán giá trị khởi tạo

$$CTR_1 = SV$$

Các phép toán để mã hóa mỗi biến bản rõ theo 4 bước sau:

a) $Y_i = eK(CTR_i)$ (Sử dụng mã khối).

b) $E_i = j\sim Y_i$ (Chọn j bit trái nhất của Y_i).

c) $C_i = P_i \oplus E_i$ (Tạo ra biến bản mã).

d) $CTR_{i+1} = (CTR_i + 1) mod\ 2^n$ (Tạo ra giá trị đếm mới CTR).

Các bước trên lặp với biến đếm $i = 1, 2, ..., q$, trong vòng lặp cuối dừng lại tại bước (c) trên chu kỳ cuối. Quá trình này được thể hiện ở bên trái hình C.5. Các biến bản mã và bản rõ có các bit có chỉ số từ *1* đến j.

Giá trị đếm được mã hóa để đưa ra khối đầu ra Y_i và j bit trái nhất của khối đầu ra Y_i được sử dụng để mã hóa giá trị đầu vào. Bộ đếm CTR sau đó được tăng 1 (modulo 2^n) để tạo ra giá trị đếm mới.

10.3 Giải mã

Các biến được sử dụng trong giải mã giống như các biến được sử dụng trong mã hóa.

Bộ đếm CTR được gán giá trị khởi tạo

$$CTR_1 = SV$$

Các phép toán để giải mã mỗi biến bản rõ theo 4 bước sau:

a) $Y_i = eK(CTR_i)$ (Sử dụng mã khối).

b) $E_i = j\text{~}Y_i$ (Chọn j bit trái nhất của Y_i).

c) $P_i = C_i \oplus E_i$ (Tạo ra biến bản rõ).

d) $CTR_{i+1} = (CTR_i + 1)\ mod\ 2^n$ (Tạo ra giá trị đếm mới CTR).

Các bước trên lặp với biến đếm $i = 1, 2, ..., q$, trong vòng lặp cuối dừng lại tại bước (c). Quá trình này được thể hiện ở bên phải hình C.5. Các biến bản mã và bản rõ có các bit có chỉ số từ *1* đến j.

Giá trị đếm được mã hóa để đưa ra khối đầu ra Y_i và j bit trái nhất của khối đầu ra Y_i được sử dụng để mã hóa giá trị đầu vào. CTR sau đó được tăng 1 (modulo 2^n) để tạo ra giá trị đếm mới.

Nếu một phương pháp đệm không được sử dụng, khi đó *P* bằng với phép ghép của $P_1, P_2, ..., P_q$.

Nếu một phương pháp đệm được sử dụng, áp dụng phép nghịch đảo của phương pháp đệm đã được thỏa thuận với chuỗi *q* khối bản rõ $P_1, P_2, ..., P_q$ để nhận được xâu bản rõ đã giải mã *P*.

CHÚ THÍCH Việc loại bỏ đệm duy nhất được đảm bảo do hạn chế trong việc lựa chọn phương pháp đệm được đưa ra trong Điều 5.

10.4 Vô hiệu hóa việc mở rộng bản mã

Sử dụng chế độ CTR có thể tránh được việc mở rộng bản mã trong trường hợp khi bản rõ không phải là toàn bộ các khối có độ dài *j* bit. Việc áp dụng này, nếu bản rõ không phải là toàn bộ các khối *j* bit, khi đó bản rõ được đệm với *p* bit ($0 \le p < j$). Vì vậy, bản rõ trở thành toàn bộ các khối *j* bit nhưng phần đệm thì không nằm trong bản mã. giá trị thực tế của các bit đệm không có liên quan nên chúng có thể là các bit 0 nhị phân. Bản rõ được đệm được mã hóa như trên nhưng các bit *p* cuối cũng của khối bản mã cuối cùng được loại bỏ vì chúng không cần thiết.

Bên giải mã áp dụng cùng một phương pháp đệm để tạo ra bản mã mà tất cả các khối mã có độ dài *j* bit, giải mã bản mã ở trên và khi đó loại bỏ p bit cuối cùng của bản rõ.

CHÚ THÍCH 1 Có thể xây dựng phương pháp này mà không cần tham chiếu đến đệm bởi các yêu cầu đơn giản mà biến mã cuối cũng E_q được cắt ngắn để là độ dài theo bit của bản rõ/mã cuối cùng. Phụ lục B của ấn bản thứ 3 của tiêu chuẩn này chú thích *j* có thể sửa đổi cho phần cuối cùng của bản rõ. Đệm không được yêu cầu khi một khối bản rõ *z* bit, $z < j$ được mã bởi phép cộng modulo 2 chỉ với các bit z đầu tiên của biến *E* tương ứng.

CHÚ THÍCH 2 Nếu phương pháp đệm đã được thỏa thuận trong Điều 5, thì phương pháp này không được sử dụng. Nếu phương pháp đệm không được thỏa thuận thì phương pháp này được sử dụng

Phụ lục A
(Quy định)
Định danh đối tượng

Phụ lục này liệt kê các định danh đối tượng được gán cho các thuật toán được quy định trong tiêu chuẩn này

```
ModesOfOperation {
iso(1) standard(0) modes-of-operation(10116) single-part(0)
asn1-module(0) algorithm-object-identifiers(0) }
DEFINITIONS EXPLICIT TAGS ::= BEGIN

-- Xuất ra tất cả --

BlockAlgorithms
FROM EncryptionAlgorithms-3 { iso(1) standard(0)
encryption-algorithms(18033) part3(3)
asn1-module(0) algorithm-object-identifiers(0) };
OID ::= OBJECT IDENTIFIER -- Alias

-- Đồng nghĩa --

is10116 OID ::= { iso(1) standard(0) modes-of-operation(10116) single-part(0)}
id-mode OID ::= { is10116 mode(1) }
id-pad OID ::= { is10116 pad(2) }
id-pad-null RELATIVE-OID ::= { 0 }    -- không xác định thuật toán đệm --
id-pad-1 RELATIVE-OID ::= { 1 }  --   -- Phương pháp đệm được chỉ định --
                                      -- Trong Phụ lục B, Điều B.2.3 --

id-mode-ecb OID ::= { id-mode ecb(1) }
id-mode-cbc OID ::= { id-mode cbc(2) }
id-mode-cfb OID ::= { id-mode cfb(3) }
id-mode-ofb OID ::= { id-mode ofb(4) }
id-mode-ctr OID ::= { id-mode ctr(5) }
id-mode-cbc_cs1 OID ::= { id-mode cbc_cs1(6) }
id-mode-cbc_cs2 OID ::= { id-mode cbc_cs2(7) }
id-mode-cbc_cs3 OID ::= { id-mode cbc_cs3(8) }

-- Mô tả thuật toán --

ModeOfOperation ::= AlgorithmIdentifier {{ ModeOfOperationAlgorithms
ModeOfOperationAlgorithms ALGORITHM ::= {
{ OID id-mode-ecb PARMS EcbParameters } |
{ OID id-mode-cbc PARMS CbcParameters } |
{ OID id-mode-cfb PARMS CfbParameters } |
{ OID id-mode-ofb PARMS OfbParameters } |
{ OID id-mode-ctr PARMS CtrParameters } |
{ OID id-mode-cbc_cs1 PARMS CbcParameters } |
{ OID id-mode-cbc_cs1 PARMS CbcParameters } |
{ OID id-mode-cbc_cs1 PARMS CbcParameters } ,

-- Các thuật toán bổ sung --
```

```
}
PadAlgo ::= CHOICE {
specifiedPadAlgo      RELATIVE-OID,
generalPadAlgo        OID
}
EcbParameters ::= SEQUENCE {
bcAlgo BlockCipher  OPTIONAL,
padAlgo PadAlgo     DEFAULT specifiedPadAlgo:id-pad-null
}
CbcParameters ::= SEQUENCE {
n       INTEGER       DEFAULT 1,
bcAlgo        BlockCipher   OPTIONAL,
padAlgo       PadAlgo       DEFAULT specifiedPadAlgo:id-pad-1
}
CfbParameters ::= SEQUENCE {
r       INTEGER,      -- n<=r<=1024n where n is the cipher block length
k       INTEGER,      -- 1<=k<=n
j       INTEGER,      -- 1<=j<=k
bc      BlockCipher   OPTIONAL,
padAlgo       PadAlgo       DEFAULT specifiedPadAlgo:id-pad-null


}
OfbParameters ::= SEQUENCE {
j       INTEGER,      -- 1<=j<=n where n is the cipher block length
bc      BlockCipher   OPTIONAL,
padAlgo       PadAlgo       DEFAULT specifiedPadAlgo:id-pad-null
}
CtrParameters ::  SEQUENCE {
j       INTEGER,      -- 1<=j<=n where n is the cipher block length
bc      BlockCipher   OPTIONAL,
padAlgo       PadAlgo       DEFAULT specifiedPadAlgo:id-pad-null
}
```

-- Định nghĩa bổ trợ --

```
BlockCipher ::= AlgorithmIdentifier {{ BlockAlgorithms }}
-- Cryptographic algorithm identification --
ALGORITHM ::= CLASS {
&id OBJECT IDENTIFIER UNIQUE,
&Type OPTIONAL
}
WITH SYNTAX { OID &id [PARMS &Type] }
AlgorithmIdentifier { ALGORITHM:IOSet } ::= SEQUENCE {
algorithm ALGORITHM.&id( {IOSet} ),
parameters ALGORITHM.&Type( {IOSet}{@algorithm} ) OPTIONAL
}
END -- ModesOfOperation --
```

Phụ lục B
(Tham khảo)
Tính chất các chế độ hoạt động và hướng dẫn an toàn quan trọng

B.1 Tổng quan

Như đã mô tả ở trên, các chế độ hoạt động của mã khối này chỉ bảo vệ tính bí mật. Các chứng minh an toàn vẫn đảm bảo cho chế độ CBC, chế độ CFB, chế độ OFB và chế độ CRT. Các chứng minh này giả thiết rằng không thể phân biệt được mã khối từ một hàm giả ngẫu nhiên. Xác suất sai về giả thiết này tăng lên đáng kể khi số lượng các khối được xử lý tăng lên $2^{n/2}$ hoặc lớn hơn. Với lý do này, số lượng các khối bản rõ được mã với cùng một khóa có thể không vượt quá $2^{n/2}$ (xem trong ISO/IEC 29192-2:2012 Phụ lục D và ISO/IEC JTC 1/SC 27, tài liệu tin cậy số 12 (SC 27 SD12) tại địa chỉ http://www.jtc1sc27.din.de/sbe/SD12.

Để đơn giản hóa các thảo luận trong phụ lục này, Các chế độ trong tiêu chuẩn này được xử lý như hoạt động của chúng trên các khối bit có độ dài phù hợp. Nghĩa là trong phụ lục này, thuật ngữ bản rõ được sử dụng để đề cập đến dữ liệu sẽ được mã hóa sau khi bổ sung đệm cần thiết.

Phương pháp đệm được đặc tả trong Điều 5 được khuyến nghị trong trường hợp khi cần thiết phải đệm. Phương pháp đệm này không chỉ đơn giản để thực hiện mà còn chống lại các tấn công nhất định trong phương pháp mã hóa CBC [10],[11],[12] và [13]. Với mỗi phương pháp đệm nhất định khác, nếu kẻ tấn công có thể thao túng các bản mã thông điệp và có thể phát hiện có hay không một bản mã thông điệp gây ra việc giải mã sai cho thiết bị vì quá trình mã hóa phát hiện xâu dữ liệu sai định dạng khi đó việc dò tìm lặp lại của dạng này có thể được sử dụng để tìm kiếm thông tin về bản rõ. Nếu việc thực hiện của chế độ sử dụng đệm khi đó việc kiểm tra tính xác thực của bản mã trước khi mã có thể chống lại tấn công "hàm tiên tri đệm (*padding oracle)*" (xem ISO/IEC 19772)

Việc thực hiện tiêu chuẩn này có thể đạt được mã xác thực bằng cách sử dụng cơ chế 5 (Encryp-then MAC) của ISO/IEC 19772:2009 và sử dụng thuật toán MAC của TCVN 11495-1.

B.2 Hoạt động của chế độ sách mã điện tử (ECB)

B.2.1 Các tính chất và hướng dẫn an toàn

Chế độ sách mã điện tử (ECB) có các tính chất như sau:

a) Mã hóa hoặc giải mã một khối có thể được thực hiện độc lập với các khối khác;

b) Việc sắp xếp lại các khối bản mã sẽ dẫn đến việc sắp xếp lại các khối bản rõ tương ứng;

c) Cùng một khối bản rõ luôn luôn tạo ra cùng một khối bản mã (đối với cùng một khóa) do đó, có thể bị tấn công bằng "tấn công từ điển", ở đó một từ điển được xây dựng với các khối bản rõ và các khối bản mã tương ứng.

Dữ liệu nhị phân trao đổi giữa các máy tính hoặc con người có thể chứa các chuỗi lặp lại hoặc các chuỗi được sử dụng thường xuyên. Trong chế độ ECB, các khối bản rõ giống nhau (đối với cùng một khóa) tạo ra các khối bản mã giống hệt nhau.

Với lý do này, chế độ ECB không khuyến cáo cho các thông điệp dài hơn 1 khối. Việc sử dụng ECB có thể chỉ được đặc tả trong các tiêu chuẩn trong tương lai với các mục đích cụ thể với các đặc trưng lặp

lại được chấp nhận (ví dụ, nếu các khối đầu vào không bao giờ lặp lại) các khối phải được truy cập riêng biệt hoặc ngẫu nhiên.

B.2.2 Lỗi lan truyền

Trong chế độ ECB, một hoặc nhiều bit lỗi trong một khối bản mã sẽ chỉ ảnh hưởng đến việc giải mã khối đó, các khối bản mã khác không bị ảnh hưởng. Giải mã một khối bản mã với một hoặc nhiều bit lỗi sẽ dẫn đến một xác suất xảy ra 50% lỗi cho mỗi bit bản rõ trong khối bản rõ tương ứng.

B.2.3 Đồng bộ hóa

Nếu các giới hạn khối bị mất trong mã hóa/giải mã (ví dụ do mất hoặc chèn thêm một bit bản mã), sự đồng bộ hóa giữa mã hóa và giải mã sẽ bị mất cho đến khi các giới hạn chính xác của các khối được thiết lập lại. Kết quả của tất cả các giải mã sẽ không chính xác trong khi giới hạn các khối bị mất.

B.3 Hoạt động của chế độ móc xích khối mã

B.3.1 Các tính chất và hướng dẫn an toàn

Các tính chất của chế độ CBC là:

a) Phép móc xích làm cho các khối bản mã phụ thuộc vào các khối bản rõ hiện tại và trước đó $P_{i-m}, P_{i-2m}, P_{i-3m}, \ldots$và do đó sắp xếp lại các khối bản mã không dẫn đến việc sắp xếp lại các khối bản rõ tương ứng, do vậy kẻ tấn công không thể thuần túy hoán chuyển các khối bản rõ bằng cách hoán chuyển các khối bản mã.

b) Việc sử dụng các giá trị *SV* khác nhau sẽ ngăn chặn cùng một bản rõ mã hóa thành cùng một bản mã;

c) Lựa chọn $m > 1$ cho phép các toán tử mã hóa của mã khối được thực hiện song song. Với việc sử dụng phần cứng xử lý song song, chẳng hạn như một mạch định hướng đường ống, tạo điều kiện cho việc thực thi dữ liệu thông lượng cao;

d) Việc giải mã khối bản mã bất kỳ có thể không cần giải mã các chuỗi bản mã trước đó; có nghĩa là, chế độ này có thuộc tính "truy cập ngẫu nhiên" cho bản mã;

e) Nếu mã khối có thể được mô phỏng bởi hoán vị giả ngẫu nhiên và *SV* được lựa chọn ngẫu nhiên, chế độ CBC được chứng minh về mặt toán học là an toàn theo nghĩa là mã hóa không có thông tin tính toán về bản rõ, xem viện dẫn [9].

CHÚ THÍCH Có sự hiểu lầm phổ biến rằng chế độ CBC cung cấp tính toàn vẹn dữ liệu - CBC không có tính chất này.

Chế độ CBC tạo ra cùng một bản mã bất cứ khi nào cùng một bản rõ được mã hóa sử dụng cùng khóa và giá trị khởi tạo

Tính an toàn không được đảm bảo nếu không áp dụng một trong những tiến trình tạo *SV* dưới đây khi có nhiều bản rõ được mã hóa với cùng một khóa

- Tạo *SV* một cách ngẫu nhiên từ tập hợp của tất cả các xâu n-bit với một nguồn ngẫu nhiên không dự đoán trước được (xem ISO / IEC 18031)

- Thiết lập khối bản rõ đầu tiên là một định danh duy nhất cho mỗi bản rõ ngoài sự kiểm soát của đối phương, chẳng hạn như tăng một bộ đếm và thiết lập *SV* là một xâu *n-bit* chỉ chứa các số không.

B.3.2 Lỗi lan truyền

Trong chế độ CBC, một hoặc nhiều bit lỗi trong một khối bản mã sẽ chỉ ảnh hưởng đến việc giải mã 2 khối (Khối xuất hiện lỗi và khối thứ *m* sau đó), Lỗi trong khối bản rõ thứ *i* có ảnh hưởng như sau đối với kết quả: khối bản rõ thứ *i* sẽ có xác suất lỗi 50% cho mỗi bit. Khối bản rõ thứ *(i + m)* sẽ có một dạng lỗi tương tự như vậy trong khối bản mã thứ *i*.

B.3.3 Đồng bộ hóa

Nếu các giới hạn khối bị mất trong mã hóa/giải mã (ví dụ do mất hoặc chèn thêm một bit bản mã), sự đồng bộ hóa giữa mã hóa và giải mã sẽ bị mất cho đến khi các giới hạn chính xác của các khối được thiết lập lại. Kết quả của tất cả các giải mã sẽ không chính xác trong khi giới hạn các khối bị mất.

B.4 Hoạt động của chế độ phản hồi bản mã (CFB)

B.4.1 Các tính chất và hướng dẫn an toàn

Các tính chất của chế độ CFB là:

a) Phép móc xích làm cho các khối bản mã phụ thuộc vào các khối bản rõ hiện tại và các khối bản rõ trước đó một số lượng nhất định các biến bản rõ ngay trước đó. Số lượng này phụ thuộc vào việc lựa chọn r, k và j. Do đó, việc sắp xếp lại j bit của các biến bản mã không ảnh hưởng đến việc sắp xếp lại j bit của các biến bản rõ tương ứng, do vậy kẻ tấn công không thể thuần túy hoán chuyển các khối bản rõ bằng cách hoán chuyển các khối bản mã.

b) Sử dụng các giá trị *SV* khác nhau sẽ ngăn chặn cùng một bản rõ được mã hóa thành cùng một bản mã;

c) Quá trình mã hóa và giải mã trong chế độ CFB chỉ sử dụng hàm mã hóa của mã khối;

d) Độ mạnh của chế độ CFB phụ thuộc vào kích thước của k (đạt cực đại nếu $j = k = n$) và kích thước tương ứng của j, k, n và r;

CHÚ THÍCH Lựa chọn j < k sẽ làm tăng xác suất xuất hiện các giá trị lặp lại trong các khối đầu vào. Các giá trị lặp lại như vậy sẽ làm rò rỉ quan hệ tuyến tính giữa các bit bản rõ.

e) Sử dụng chế độ này yêu cầu gấp gần n/j lần các mã hóa mã khối so với chế độ ECB, do đó việc lựa chọn giá trị j nhỏ sẽ làm cho quá trình xử lý phức tạp hơn.

f) Lựa chọn $r \geq n + k$ cho phép thao tác đường ống và liên tục mã khối (giả sử sử dụng phần cứng xử lý song song, ví dụ như một mạch định hướng đường ống); Các các nội dung khởi tạo (*SV*) của bộ đệm phản hồi sẽ chỉ được thay thế hoàn toàn khi số lượng các khối mã được thực hiện lớn hơn hoặc bằng r/ k; và

g) Chứng minh an toàn cho chế độ CFB được trình bày trong [8].

Chế độ CFB tạo ra cùng một bản mã khi một bản rõ bất kỳ được mã hóa sử dụng cùng một khóa và giá trị khởi tạo.

Tính an toàn sẽ không được đảm bảo nếu *SV* không phải là ngẫu nhiên cho mỗi một bản rõ nếu nhiều bản rõ được mã với cùng một khóa, chỉ ra từ tập hợp của tất cả các xâu *r-bit* (xem ISO / IEC 18031)

B.4.2 Lỗi lan truyền

Trong chế độ CFB, các lỗi trong đơn vị j bit bất kỳ của bản mã sẽ ảnh hưởng đến việc giải mã bản mã sau đó cho đến khi các bit lỗi được dịch ra ngoài bộ đệm phản hồi CFB. Một lỗi trong biến bản mã thứ i ảnh hưởng đến kết quả bản rõ như sau: biến bản rõ thứ i có số lỗi bằng với biến bản mã thứ i. Các biến bản rõ sau đó sẽ có xác suất lỗi 50% cho mỗi bit cho đến khi tất các các bit lỗi được dịch ra ngoài bộ đệm phản hồi.

B.4.3 Đồng bộ hóa

Nếu các giới hạn j-bit bị mất trong quá trình mã hóa và giải mã (ví dụ do mất hoặc việc chèn thêm một bit bản mã), việc đồng bộ mật mã sẽ được thiết lập lại cho r bit sau khi thiết lập lại giới hạn j-bit. Nếu một bội j bit bị mất đồng bộ sẽ được thiết lập lại một cách tự động sau r bit. Tương tự, khi $j = k = 1$, chế độ CFB tự động thiết lập lại đồng bộ mật mã sau khi mất hoặc chèn thêm một số lượng bit bản mã bất kỳ.

B.5 Hoạt động của chế độ phản hồi đầu ra (OFB)

B.5.1 Các tính chất và hướng dẫn an toàn

Các tính chất của chế độ OFB là:

a) Sử dụng các giá trị *SV* khác nhau sẽ ngăn chặn cùng một bản rõ được mã hóa thành cùng một bản mã bằng cách tạo ra các dòng khóa khác nhau;

b) Quá trình mã hóa và giải mã trong chế độ OFB chỉ sử dụng hàm mã hóa của mã khối;

c) Chế độ OFB không phụ thuộc vào bản rõ để tạo ra dòng khóa được dùng để cộng modulo 2 với bản rõ;

d) Sử dụng chế độ này yêu cầu gấp gần n/j lần các mã hóa mã khối so với chế độ ECB, do đó việc lựa chọn giá trị j nhỏ sẽ làm cho quá trình xử lý phức tạp hơn;

e) Các chứng minh an toàn cho chế độ OFB tương tự như đối với chế độ CTR.

Chế độ OFB tạo ra cùng một bản mã khi cùng một bản rõ được mã hóa sử dụng cùng một khóa và giá trị khởi tạo. Ngoài ra, trong chế độ OFB, dòng khóa giống nhau (chuỗi kết quả trung gian E_i) được tạo ra khi sử dụng cùng một khóa và *SV*. Tương tự, với các lý do an toàn, một *SV* cụ thể chỉ được sử dụng một lần cho một khóa cho trước.

Tính an toàn không được đảm bảo nếu không áp dụng một trong những tiến trình tạo *SV* dưới đây khi có nhiều bản rõ được mã hóa với cùng một khóa

- Tạo *SV* một cách ngẫu nhiên từ tập hợp của tất cả các xâu n-bit với một nguồn ngẫu nhiên không dự đoán trước được (xem ISO/IEC 18031);

- Chỉ cùng một bản rõ được mã hóa sử dụng cùng một khóa và giá trị khởi tạo. Ngoài ra, trong chế độ OFB, dòng khóa giống nhau (chuỗi kết quả trung gian E_i) được tạo ra khi sử dụng cùng một khóa và *SV*. Tương tự, với các lý do an toàn, một *SV* cụ thể chỉ được sử dụng một lần cho một khóa cho trước.

B.5.2 Lỗi lan truyền

Chế độ OFB không lan truyền các lỗi trong kết quả đầu ra của bản rõ. Mỗi bit lỗi trong bản mã chỉ gây ra một bit lỗi trong bản rõ được giải mã.

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]

[illegible]

[illegible]

[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]