(12) BẢN MÔ TẢ SÁNG CHẾ THUỘC BẰNG ĐỘC QUYỀN SÁNG CHẾ

(19) **Cộng hòa xã hội chủ nghĩa Việt Nam (VN)**
CỤC SỞ HỮU TRÍ TUỆ

(11) **1-0053925**

(51)[2020.01] **H04L 9/00**

(13) **B**

(21) 1-2021-01093

(22) 03/03/2021

(45) 25/11/2025 452

(43) 26/04/2021 397A

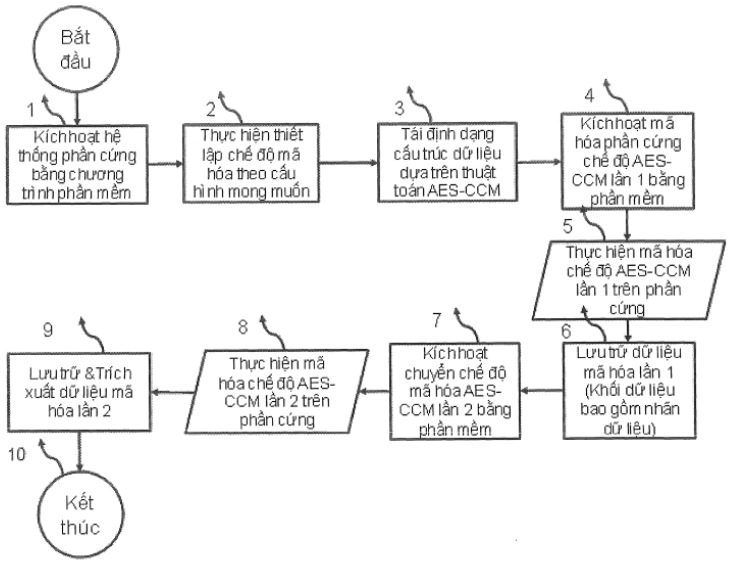
(73) Trường Đại học Công nghệ, Đại học Quốc gia Hà Nội (VN)
Nhà E3, 144 đường Xuân Thủy, quận Cầu Giấy, thành phố Hà Nội

(72) Trần Xuân Tú (VN); Nguyễn Ngô Doanh (VN); Bùi Duy Hiếu (VN).

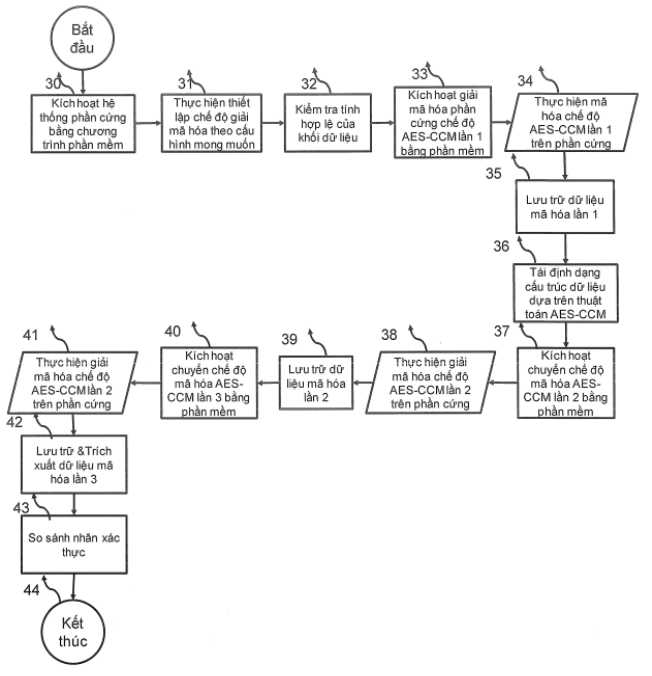
(54) PHƯƠNG PHÁP MÃ HÓA VÀ GIẢI MÃ NHỜ XỬ LÝ ĐỒNG THỜI PHẦN CỨNG VÀ PHẦN MỀM SỬ DỤNG THUẬT TOÁN XÁC THỰC VÀ ĐỊNH DANH AES-CCM

(21) 1-2021-01093

(57) Sáng chế đề xuất phương pháp mã hóa và giải mã, và cụ thể hơn là đến phương pháp mã hóa và giải mã nhờ xử lý đồng thời phần cứng và phần mềm cho mã hóa, giải mã hóa sử dụng thuật toán bảo mật xác thực và định danh AES-CCM nhằm giúp tối ưu hiệu năng, ứng dụng bảo mật cho các thiết bị IoT. Phương pháp mã hóa theo sáng chế bao gồm các bước xử lý đồng thời phần cứng và phần mềm cho mã hoá dữ liệu theo chế độ AES-CCM, và phương pháp giải mã theo sáng chế bao gồm các bước xử lý đồng thời phần cứng và phần mềm cho giải mã hoá dữ liệu theo chế độ AES-CCM.



Hình 1



Hình 4

Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến phương pháp mã hóa và giải mã, cụ thể hơn là đến phương pháp mã hóa và giải mã nhờ xử lý đồng thời phần cứng và phần mềm cho mã hóa, giải mã hóa sử dụng thuật toán bảo mật xác thực và định danh AES-CCM nhằm giúp tối ưu hiệu năng, ứng dụng bảo mật cho các thiết bị IoT.

Tình trạng kỹ thuật của sáng chế

Các ứng dụng mạng Internet kết nối vạn vật (IoT: Internet-of-Things) sử dụng các kỹ thuật định danh và xác thực cho các giao thức bảo mật để nâng cao tính an toàn của hệ thống và dữ liệu. Tuy nhiên, vì các thiết bị IoT thường là các thiết bị nhỏ ngọn, sử dụng pin hoặc nguồn điện không ổn định, có năng lực tính toán, không gian bộ nhớ và không gian lưu trữ có hạn nên các kỹ thuật định danh và xác thực dùng trong các thiết bị IoT cũng phải được thiết kế một cách nhỏ gọn để phù hợp với điều kiện hoạt động nhưng vẫn đáp ứng được yêu cầu của ứng dụng. Các đề xuất cho IoT hiện tại như LoRaWan, IEEE 802.15.4, ZWave, v.v… sử dụng thuật toán AES như là phương pháp bảo mật chính nên các kỹ thuật định danh và xác thực được dùng thường là cũng dựa trên thuật toán AES.

Trong các giao thức bảo mật AES, thuật toán xác thực và định danh CCM là kết hợp của chế độ xác thực CBC (CBC: Cipher Block Chaining) và chế độ mã hóa CTR (Counter Mode). Đầu tiên, CBC-MAC sẽ thực hiện xử lý bản tin *m* để thu được nhãn xác thực *t*. Sau đó, bản tin *m* và nhãn xác thực *t* được mã hóa bằng chế độ CTR. Việc mã hóa và xác thực có thể dùng chung một khóa bí mật mà giá trị của bộ đếm ở chế độ mã hóa CTR không gây xung đột với giá trị của véc-tơ khởi tạo (IV: Initial Vector) dùng trong xác thực. Khóa bí mật được chọn ngẫu nhiên trong tập giá trị khóa $\{0,1\}^{k_0}$ có thể dùng. Quá trình mã hóa dùng thuật toán CCM (hay mã hóa CCM) thực hiện chọn một khối dữ liệu dùng một lần *N (N:*

Nonce) có giá trị độ dài bit cố định $k_n < k_b$, bản tin m và dữ liệu xác thực bổ sung a. Dữ liệu xác thực bổ sung a không dùng để mã hóa mà chỉ để dùng cho xác thực. Trong khi đó, bản tin m phải đồng thời mã hóa và xác thực để thu được nhãn xác thực t, có độ dài bit cố định $k_t \leq k_b$. Cuối cùng, bản tin mới thu được sẽ là chuỗi dữ liệu mã hóa c có độ dài bit $|m| + k_t$.

Phương pháp thực hiện quá trình mã hóa CCM có thể tổng hợp lại như sau.

Đầu tiên, nhãn xác thực t trong chế độ CBC sẽ được xử lý thông qua khối dữ liệu đầu vào $\beta(N, m, a)$, là tổ hợp của khối dữ liệu dùng một lần N, bản tin m và dữ liệu bổ sung a. Sau đó, bản tin m được mã hóa tại chế độ CTR với khối CTR được tạo ra từ khối dữ liệu dùng duy nhất một lần N. Cuối cùng, khối dữ liệu sau mã hóa là kết hợp bởi nhãn xác thực t với khối dữ liệu mã hóa với một khối CTR duy nhất.

Nói chung, phương pháp mã hóa dữ liệu sử dụng thuật toán AES-CCM (hay chế độ AES-CCM) bao gồm các bước theo thứ tự sau:

Bước 1: Xử lý lấy nhãn xác thực dữ liệu thông qua thuật toán mã hóa AES. Trong đó, Bước 1 này có thể được chia nhỏ thành các công đoạn sau:

Công đoạn 1: Phân chia dữ liệu đầu vào $\beta(N, m, a)$, là tổ hợp của khối dữ liệu dùng một lần N, bản tin m và dữ liệu bổ sung a, thành các khối con B_i:

$$\beta(N, m, a) = B_0 . B_1 . \cdots . B_r \quad (1)$$

Công đoạn 2: Tiến hành mã hóa và thu được khối dữ liệu mã hóa đầu tiên như sau:

$$Y_0 = CIPH_k(B_0) \quad (2)$$

Công đoạn 3: Tiến hành mã hóa từ $1 \leq i \leq r$ và thu được khối dữ liệu mã hóa thứ i như sau:

$$Y_i = CIPH_k(Y_{i-1} \oplus B_i) \quad (3)$$

Công đoạn 4: Lấy $Tlen$ bit phía bên trái của Y_r ta thu được nhãn xác thực *t*.

$$t = Y_r \gg Tlen \tag{4}$$

Bước 2: Xử lý mã hóa AES chế độ CTR, được chia thành các công đoạn:

Công đoạn 1: Coi $\mu = \lceil |m|/k_b \rceil$ là số lượng khối dữ liệu có kích thước k_b. Từ $0 \leq i \leq \mu$, tiến hành tổ hợp các khối dữ liệu đầu vào thu được các khối:

$$a'_i = \pi(i, N, |m|, a) \tag{5}$$

Công đoạn 2: Từ $0 \leq i \leq \mu$, tiến hành mã hóa khối dữ liệu a_i với khóa dữ liệu k:

$$S_i = CIPH_k(a'_i). \tag{6}$$

Công đoạn 3: Giá trị s_m sẽ là $|m|$ bit phía bên trái của $S_1.S_2.\cdots.S_\mu$ và S_T sẽ là k_t phía bên trái của S_0.

$$S_T = S_0 \gg k_t \tag{7}$$

$$S_m = S_0 \gg |m| \tag{8}$$

Công đoạn 4: Ghi nhận chuỗi dữ liệu mã hóa *c*

$$c = [m \oplus s_m].[T \oplus S_T]. \tag{9}$$

Bước 3: Xuất bản tin đã mã hóa *c*. Kết thúc phương pháp mã hóa CCM.

Đối với giải mã hóa, dữ liệu bảo mật sẽ được tiếp nhận cùng các dữ liệu bổ sung *a* và khối dữ liệu dùng một lần *N* được sử dụng khi mã hóa. Để thu được dữ liệu trước khi mã hóa, bên nhận tiến hành đảo ngược quá trình ở bên thu. Cụ thể, chế độ CTR sẽ được áp dụng để tìm ra bản tin *m* và mã xác thực bản tin MAC (MAC: Message Authentication Code). Nếu định dạng của bản tin *m*, dữ liệu bổ sung *a* và khối dữ liệu dùng một lần *N* đúng so với quy định của thuật toán CCM

thì bên nhận sẽ tiếp tục tiến hành sử dụng chế độ MAC để tạo ra nhãn xác thực *t*. Quá trình giải mã hóa sẽ kết thúc và trả về bản tin gốc *m* khi nhãn xác thực *t* được xác định là đúng hoặc thông báo bản tin lỗi khi sai nhãn xác thực *t*.

Tương ứng, phương pháp giải mã hóa dữ liệu sử dụng thuật toán AES-CCM này bao gồm các bước sau:

Bước 1: Kiểm tra độ dài khối dữ liệu mã hóa với độ dài nhãn xác thực *t*. Nếu $Clen \leq Tlen$ thì quá trình giải mã hóa kết thúc và thông báo bản tin bị lỗi.

Bước 2: Xử lý khối dữ liệu mã hóa để thu lại bản tin gốc *m* và mã xác thực bản tin MAC thông qua thuật toán mã hóa AES CTR, gồm các công đoạn sau:

Công đoạn 1: Phân chia dữ liệu đầu vào thành các khối con $Ctr_0, Ctr_1, \ldots, Ctr_m$, với $m = \lceil (Clen - Tlen)/128 \rceil$.

Công đoạn 2: Tiến hành mã hóa các khối con bằng thuật toán mã hóa AES CTR:

$$S_j = CIPH_k(Ctr_j), \qquad \forall j \in N, j: 0 \to m \tag{10}$$

Công đoạn 3: Tổng hợp lại các khối dữ liệu sau mã hóa AES CTR thành một khối dữ liệu thống nhất:

$$S = S_1 \parallel S_2 \parallel \cdots \parallel S_m \tag{11}$$

Công đoạn 4: Thu thập bản tin gốc *m* và nhãn của chuỗi dữ liệu mã hóa T:

$$M = \mathrm{MSB}_{Clen-Tlen}(C) \oplus \mathrm{MSB}_{Clen-Tlen}(S) \tag{12}$$

$$T = \mathrm{LSB}_{Clen-Tlen}(C) \oplus \mathrm{MSB}_{Clen-Tlen}(S_0) \tag{13}$$

Bước 4: Kiểm tra tính đúng đắn của khối dữ liệu bổ sung *a*, khối dữ liệu dùng một lần *N*, bản tin gốc *m* và tiến hành mã hóa dùng thuật toán AES CBC nếu các khối dữ liệu phù hợp với chuẩn của thuật toán. Bước 4 này có thể gồm các công đoạn sau:

Công đoạn 1: Phân chia dữ liệu đầu vào $\beta(N, m, a)$, là tổ hợp của khối dữ liệu dùng một lần N, bản tin m và dữ liệu bổ sung a, thành các khối con B_i như biểu diễn ở phương trình (1);

Công đoạn 2: Tiến hành mã hóa và thu được khối dữ liệu mã hóa đầu tiên như biểu diễn ở phương trình (2);

Công đoạn 3: Sau đấy từ $1 \leq i \leq r$, tiến hành mã hóa và thu được khối dữ liệu mã hóa thứ i như biểu diễn ở phương trình (3);

Công đoạn 4: Lấy $Tlen$ bit phía bên trái của Y_r ta thu được nhãn xác thực t' như biểu diễn ở phương trình (4).

Bước 5: So sánh hai nhãn xác thực t và t' để kiểm tra tính đúng đắn của bản tin. Xuất bản tin gốc m nếu nhãn xác thực t và t' giống nhau hoặc thông báo bản tin gốc m bị lỗi nếu hai nhãn xác thực t và t' khác nhau.

Tuy nhiên, trong thực tế khi áp dụng thuật toán mã hóa AES-CCM cho các thiết bị IoT bị giới hạn về tài nguyên như dung lượng phần cứng hay năng lượng cung cấp thì việc triển khai thuật toán thuần trên chương trình phần mềm hay phần cứng sẽ không phù hợp.

Ở đây, việc triển khai thuật toán thuần trên chương trình phần mềm được hiểu là tận dụng kiến trúc tập lệnh ISA của vi xử lý phần cứng để thực hiện các phép tính toán và biến đổi trong mã hóa và giải mã hóa nhằm thu được dữ liệu mong muốn. Trong đó, dữ liệu xử lý sẽ được cấp phát trong vùng bộ nhớ ngăn xếp (stack memory) và dữ liệu thu được sẽ được cấp phát trong vùng bộ nhớ còn lại (heap memory) của thiết bị. Cụ thể, đối với triển khai bằng phần mềm, các tính toán ở phương trình (2), (3), (6), (10) sẽ tốn nhiều thời gian làm mất tính thời gian thực của thiết bị IoT. Đồng thời, các thiết bị IoT vốn chỉ sử dụng các nguồn có điện áp thấp hoặc pin. Do đó, việc thực hiện các phép toán phức tạp bằng chương trình phần mềm sẽ làm tăng năng lượng tiêu thụ, gián tiếp làm giảm tuổi thọ của thiết bị.

Mặt khác, việc triển khai thuật toán thuần trên phần cứng được hiểu là tận dụng các khối chức năng tính toán, biến đổi được thiết kế riêng trong thiết bị để hỗ trợ trong việc mã hóa và giải mã hóa dữ liệu. Trong đó, dữ liệu xử lý và và dữ liệu thu được đã được tối ưu thành các cổng lô-gíc, các thanh ghi với các địa chỉ cố định từ giai đoạn thiết kế. Tuy nhiên, việc triển khai bằng phần cứng cũng có các nhược điểm. Đó là khi thực thi toàn bộ các tính toán ở phương trình (1), (5), (11) sẽ làm tăng kích thước của thiết bị, cũng như không đem lại tính tổng quát sử dụng cho các ứng dụng bảo mật khác. Qua đó, việc thực thi thuật toán bảo mật AES-CCM thuần bằng chương trình phần mềm hoặc phần cứng không phù hợp với các ứng bảo mật cho thiết bị IoT.

Bản chất kỹ thuật của sáng chế

Mục đích của sáng chế này là đề xuất phương pháp xử lý đồng thời phần cứng và phần mềm cho mã hóa, giải mã hóa sử dụng thuật toán bảo mật xác thực và định danh AES-CCM nhằm giúp tối ưu hiệu năng, ứng dụng bảo mật cho các thiết bị IoT.

Để đạt được mục đích trên, phương pháp xử lý đồng thời phần cứng và phần mềm cho mã hóa, giải mã hóa sử dụng thuật toán bảo mật xác thực và định danh AES-CCM theo sáng chế thực hiện xử lý đồng thời phần cứng và phần mềm cho mã hóa dữ liệu theo chế độ mã hóa AES-CCM; và xử lý đồng thời phần cứng và phần mềm cho giải mã hóa theo chế độ giải mã hóa AES-CCM.

Theo một khía cạnh, sáng chế đề xuất phương pháp mã hóa, thực hiện quá trình mã hóa nhờ xử lý phần cứng và phần mềm sử dụng thuật toán xác thực và định danh AES-CCM, phương pháp này bao gồm các bước:

kích hoạt hệ thống phần cứng mã hóa bằng chương trình phần mềm, trong đó câu lệnh kích hoạt sẽ thông qua giao diện truyền thông truyền tới khối vi điều khiển để tác động vào khối mã hóa, giải mã hóa phần cứng;

thiết lập cấu hình mã hóa AES-CCM cho khối phần cứng bằng chương trình phần mềm, trong đó bước này bao gồm các công đoạn:

thiết lập chế độ mã hóa AES-CCM cho khối mã hóa, giải mã hóa phần cứng;

thiết lập độ dài khóa dữ liệu cho khối mã hóa, giải mã hóa phần cứng;

thiết lập độ dài nhãn xác thực của khối dữ liệu cần mã hóa cho khối mã hóa, giải mã hóa phần cứng;

thiết lập kích thước khối dữ liệu cần mã hóa cho khối mã hóa, giải mã hóa phần cứng;

tái định dạng cấu trúc dữ liệu cần mã hóa dựa trên thuật toán AES-CCM bằng chương trình phần mềm theo phương trình dưới đây:

$$\beta(N, m, a) = B_0.B_1.\cdots.B_r \quad (1)$$

trong đó: $\beta(N, m, a)$ là khối dữ liệu đầu vào, B_i là các khối con, N là tổ hợp của khối dữ liệu dùng một lần, m là bản tin, và a là dữ liệu bổ sung,

bước này bao gồm các công đoạn:

xây dựng cờ dữ liệu cho khối dữ liệu cần mã hóa từ độ dài nhãn xác thực $Tlen$, độ dài khối dữ liệu cần được mã hóa $|m|$và sự hiện diện của khối dữ liệu bổ sung a;

xây dựng khối dữ liệu đầu vào $\beta(N, m, a)$, là tổ hợp của cờ dữ liệu, khối dữ liệu dùng một lần N, bản tin m, và dữ liệu bổ sung a;

kích hoạt khối phần cứng sử dụng chế độ mã hóa AES-CCM lần một, bản chất là thực hiện mã hóa theo chế độ AES-CBC, thông qua phần mềm;

thực hiện mã hóa chế độ AES-CCM lần một trên phần cứng;

lưu trữ dữ liệu mã hóa lần một trên phần cứng (khối dữ liệu mã hóa bao gồm cả nhãn xác thực t);

kích hoạt khối phần cứng sử dụng chế độ AES-CCM lần hai, bản chất là thực hiện mã hóa theo chế độ AES-CTR, thông qua phần mềm;

thực hiện mã hóa chế độ AES-CCM lần hai trên phần cứng;

lưu trữ dữ liệu mã hóa lần hai trên phần cứng (khối dữ liệu mã hóa bao gồm dữ liệu bảo mật sau khi mã hóa AES-CCM) và trích xuất khối dữ liệu này ra phần mềm; và

kết thúc quá trình mã hóa dữ liệu theo chế độ mã hóa AES-CCM.

Theo một khía cạnh khác, sáng chế đề xuất phương pháp giải mã hóa, thực hiện quá trình mã hóa nhờ xử lý phần cứng và phần mềm sử dụng thuật toán xác thực và định danh AES-CCM, phương pháp này bao gồm các bước:

kích hoạt hệ thống phần cứng mã hóa bằng chương trình phần mềm, trong đó câu lệnh kích hoạt sẽ thông qua giao diện truyền thông truyền tới khối vi điều khiển để tác động vào khối mã hóa, giải mã hóa phần cứng;

thiết lập cấu hình giải mã hóa cho khối phần cứng bằng chương trình phần mềm, trong đó bước này bao gồm các công đoạn:

thiết lập chế chế độ giải mã hóa AES-CCM cho khối mã hóa, giải mã hóa phần cứng;

thiết lập độ dài khóa dữ liệu cho khối mã hóa, giải mã hóa phần cứng;

thiết lập độ dài nhãn xác thực của khối dữ liệu cần mã hóa cho khối mã hóa, giải mã hóa phần cứng;

thiết lập kích thước khối dữ liệu cần giải mã hóa cho khối mã hóa, giải mã hóa phần cứng;

kiểm tra tính hợp lệ của khối dữ liệu cần giải mã hóa bằng cách so sánh độ dài khối dữ liệu cần giải mã với độ dài nhãn xác thực, trong đó bước này bao gồm các công đoạn:

nếu độ dài khối dữ liệu cần giải mã ngắn hơn độ dài nhãn xác thực được cấu hình thì thông báo dữ liệu cần giải mã không phù hợp và chuyển sang bước so sánh hai nhãn xác thực dữ liệu t và t' dưới đây,

nếu độ dài khối dữ liệu cần mã hóa dài hơn hoặc bằng độ dài dãn nhữ liệu được cấu hình thì chuyển sang bước kích hoạt khối phần cứng sử dụng chế độ giải mã hóa AES-CCM lần một ngay dưới đây;

kích hoạt khối phần cứng sử dụng chế độ giải mã hóa AES-CCM lần một, bản chất là thực hiện mã hóa theo chế độ AES-CTR, thông qua phần mềm;

thực hiện giải mã hóa chế độ AES-CCM lần một trên phần cứng;

lưu trữ dữ liệu giải mã hóa lần một trên phần cứng (khối dữ liệu giải mã hóa bao gồm dữ liệu gốc m trước khi mã hóa vã nhãn xác thực t theo các phương trình dưới đây:

$$m = \mathrm{MSB}_{Clen-Tlen}(C) \oplus \mathrm{MSB}_{Clen-Tlen}(S)$$
$$t = \mathrm{LSB}_{Clen-Tlen}(C) \oplus \mathrm{MSB}_{Clen-Tlen}(S_0)$$

trong đó: $\mathrm{MSB_i}$ là i bit quan trọng nhất (Most Significant Bit), hay i bit có trọng số lớn nhất.

$\mathrm{LSB_i}$ là i bit ít quan trọng nhất (Least Significant Bit), hay i bit có trọng số nhỏ nhất.

và trích xuất ra phần mềm;

tái định dạng cấu trúc dữ liệu cần giải mã hóa dựa trên thuật toán AES-CCM bằng chương trình phần mềm theo phương trình dưới đây:

$$\beta(N, m, a) = B_0.B_1.\cdots.B_r$$

trong đó: $\beta(N, m, a)$ là khối dữ liệu đầu vào, B_i là các khối con, N là tổ hợp của khối dữ liệu dùng một lần, m là bản tin, và a là dữ liệu bổ sung,

bước này bao gồm các công đoạn:

xây dựng cờ dữ liệu cho khối dữ liệu cần giải mã hóa từ độ dài nhãn xác thực $Tlen$, độ dài khối dữ liệu cần được mã hóa $|m|$và sự hiện diện của khối dữ liệu bổ sung a,

xây dựng khối dữ liệu đầu vào $\beta(N, m, a)$, là tổ hợp của cờ dữ liệu, khối dữ liệu dùng một lần N, bản tin m, và dữ liệu bổ sung a;

kích hoạt khối phần cứng sử dụng chế độ giải mã hóa AES-CCM lần hai, bản chất là thực hiện mã hóa theo chế độ AES-ECB, thông qua phần mềm theo phương trình:

$$Y_0 = CIPH_k(B_0)$$

trong đó: Y_o là khối dữ liệu đầu tiên thu được sau khi được mã hóa (CIPH) khối dữ liệu B_0 với khóa dữ liệu k

thực hiện giải mã hóa chế độ AES-CCM lần hai trên phần cứng;

lưu trữ dữ liệu giải mã hóa lần hai trên phần cứng;

kích hoạt khối phần cứng sử dụng chế độ giải mã hóa AES-CCM lần ba, bản chất là thực hiện mã hóa theo chế độ AES-CBC, thông qua phần mềm theo phương trình dưới đây:

$$Y_i = CIPH_k(Y_{i-1} \oplus B_i)$$

trong đó: Y_i là khối dữ liệu thứ i thu được sau khi được mã hóa (CIPH) kết quả của phép XOR giữa khối dữ liệu B_i và khối dữ liệu thu được thứ *i-1* với khóa dữ liệu k

thực hiện giải mã hóa chế độ AES-CCM lần ba trên phần cứng;

lưu trữ dữ liệu giải mã hóa lần ba trên phần cứng (khối dữ liệu giải mã hóa bao gồm nhãn xác thực *t'*) và trích xuất ra phần mềm;

so sánh hai nhãn xác thực dữ liệu *t* và *t'*, bước này bao gồm các công đoạn:

nếu hai nhãn xác thực *t* và *t'* giống nhau thì phản hồi dữ liệu đã giải mã hóa ở bước lưu trữ dữ liệu giải mã hóa lần một trên phần cứng nêu trên,

nếu hai nhãn xác thực *t* và *t'* khác nhau thì thông báo khối dữ liệu cần giải mã hóa bị lỗi; và

kết thúc quá trình giải mã hóa dữ liệu theo chế độ giải mã hóa AES-CCM.

Mô tả vắn tắt các hình vẽ

Hình 1 là sơ đồ khối minh họa phương pháp xử lý đồng thời phần cứng và phần mềm sử dụng thuật toán mã hóa AES-CCM;

Hình 2 là sơ đồ khối minh họa phương pháp con xử lý phần mềm thuật toán mã hóa AES-CCM;

Hình 3 là sơ đồ khối minh họa phương pháp con xử lý phần cứng thuật toán mã hóa AES-CCM;

Hình 4 là sơ đồ khối minh họa phương pháp xử lý đồng thời phần cứng và phần mềm thuật toán giải mã hóa AES-CCM;

Hình 5 là sơ đồ khối minh họa phương pháp con xử lý phần mềm thuật toán giải mã hóa AES-CCM;

Hình 6 là sơ đồ khối minh họa phương pháp con xử lý phần cứng thuật toán giải mã hóa AES-CCM.

Mô tả chi tiết sáng chế

Như được thể hiện trên Hình 1 và Hình 4, việc xử lý đồng thời phần cứng và phần mềm dùng thuật toán AES-CCM theo sáng chế được ứng dụng cho phương pháp riêng biệt là phương pháp mã hóa và phương pháp giải mã hóa.

Như thể hiện trên Hình 1, phương pháp mã hóa sử dụng thuật toán AES-CCM được thực hiện theo các bước như sau:

Bước 1: Kích hoạt hệ thống phần cứng mã hóa bằng chương trình phần mềm. Câu lệnh kích hoạt sẽ thông qua giao diện truyền thông truyền tới khối vi điều khiển để tác động vào khối mã hóa, giải mã hóa phần cứng;

Bước 2: Thiết lập cấu hình mã hóa AES-CCM cho khối phần cứng bằng chương trình phần mềm bao gồm các công đoạn:

Công đoạn 2.1: Thiết lập chế độ mã hóa AES-CCM cho khối mã hóa, giải mã hóa phần cứng;

Công đoạn 2.2: Thiết lập độ dài khóa dữ liệu cho khối mã hóa, giải mã hóa phần cứng với độ dài 128-bit, 192-bit hoặc 256-bit;

Công đoạn 2.3: Thiết lập độ dài nhãn xác thực t của khối dữ liệu cần mã hóa cho khối mã hóa, giải mã hóa phần cứng;

Công đoạn 2.4: Thiết lập kích thước khối dữ liệu cần mã hóa cho khối mã hóa, giải mã hóa phần cứng;

Bước 3: Tái định dạng cấu trúc dữ liệu cần mã hóa dựa trên thuật toán AES-CCM bằng chương trình phần mềm theo phương trình (1) bao gồm:

Công đoạn 3.1: Xây dựng cờ dữ liệu cho khối dữ liệu cần mã hóa từ độ dài nhãn xác thực $Tlen$, độ dài khối dữ liệu cần được mã hóa $|m|$và sự hiện diện của khối dữ liệu bổ sung a;

Công đoạn 3.2: Xây dựng khối dữ liệu đầu vào $\beta(N, m, a)$, là tổ hợp của cờ dữ liệu, khối dữ liệu dùng một lần N, bản tin m, và dữ liệu bổ sung a;

Bước 4: Kích hoạt khối phần cứng sử dụng chế độ mã hóa AES-CCM lần một, bản chất là thực hiện mã hóa theo chế độ AES-CBC, thông qua phần mềm;

Bước 5: Thực hiện mã hóa chế độ AES-CCM lần một trên phần cứng;

Bước 6: Lưu trữ dữ liệu mã hóa lần một trên phần cứng (khối dữ liệu mã hóa bao gồm cả nhãn xác thực t);

Bước 7: Kích hoạt khối phần cứng sử dụng chế độ AES-CCM lần hai, bản chất là thực hiện mã hóa theo chế độ AES-CTR, thông qua phần mềm;

Bước 8: Thực hiện mã hóa chế độ AES-CCM lần hai trên phần cứng;

Bước 9: Lưu trữ dữ liệu mã hóa lần hai trên phần cứng (khối dữ liệu mã hóa bao gồm dữ liệu bảo mật sau khi mã hóa AES-CCM) và trích xuất khối dữ liệu này ra phần mềm;

Bước 10: Kết thúc quá trình mã hóa dữ liệu theo chế độ mã hóa AES-CCM;

Tiếp theo, để hiểu rõ hơn về phương pháp mã hóa theo sáng chế, xử lý đồng thời phần cứng và phần mềm sử dụng thuật toán mã hóa AES-CCM, các bước xử lý phần mềm và các bước xử lý phần cứng độc lập sẽ được tập hợp thành các phương pháp xử lý riêng và được trình bày dưới đây.

Như thể hiện trên Hình 2 các bước thực hiện phương pháp xử lý mã hóa AES-CCM trên phần mềm bao gồm các bước:

Bước 11: Kích hoạt hệ thống phần cứng bằng chương trình phần mềm;

Bước 12: Thiết lập cấu hình mã hóa AES-CCM cho khối mã hóa, giải mã hóa phần cứng;

Bước 13: Tái định dạng cấu trúc dữ liệu cần mã hóa dựa trên thuật toán AES-CCM theo phương trình (1);

Bước 14: Kích hoạt khối mã hóa, giải mã hóa phần cứng sử dụng chế độ mã hóa AES-CCM lần một;

Bước 15: Đợi khối mã hóa, giải mã hóa phần cứng thực thi mã hóa AES-CCM lần một xong;

Bước 16: Kích hoạt khối phần cứng sử dụng chế độ AES-CCM lần hai;

Bước 17: Đợi khối mã hóa, giải mã hóa phần cứng thực thi mã hóa AES-CCM lần hai xong;

Bước 18: Tiếp nhận dữ liệu được trích xuất từ khối phần cứng;

Bước 19: Kết thúc quá trình xử lý phần mềm cho chế độ mã hóa AES-CCM;

Như thể hiện trên Hình 3 các bước thực hiện phương pháp xử lý trên phần cứng bao gồm các bước như sau:

Bước 20: Hoạt động ở thạng thái chờ, đợi tín hiệu kích hoạt từ phần mềm;

Bước 21: Tiếp nhận tín hiệu kích hoạt và cấu hình chế độ mã hóa từ phần mềm;

Bước 22: Đợi phần mềm thực hiện tái định dạng cấu trúc dữ liệu cần mã hóa và tín hiệu kích hoạt mã hóa AES-CCM lần một;

Bước 23: Thực hiện mã hóa dữ liệu lần một với chế độ AES-CCM theo điều khiển của phần mềm;

Bước 24: Lưu trữ dữ liệu mã hóa lần một;

Bước 25: Đợi tín hiệu kích hoạt mã hóa AES-CCM lần hai từ phần mềm;

Bước 26: Tiếp nhận tín hiệu kích hoạt mã hóa AES-CCM lần hai từ phần mềm;

Bước 27: Thực hiện mã hóa dữ liệu lần hai với chế độ AES-CCM theo điều khiển của phần mềm;

Bước 28: Lưu trữ dữ liệu mã hóa lần hai và trích xuất khối dữ liệu này ra phần mềm;

Bước 29: Kết thúc quá trình xử lý phần cứng cho chế độ mã hóa AES-CCM;

Như thể hiện trên Hình 4, phương pháp giải mã hóa sử dụng thuật toán AES-CCM được thực hiện theo các bước như sau:

Bước 30: Kích hoạt hệ thống phần cứng mã hóa bằng chương trình phần mềm. Câu lệnh kích hoạt sẽ thông qua giao diện truyền thông truyền tới khối vi điều khiển để tác động vào khối mã hóa, giải mã hóa phần cứng;

Bước 31: Thiết lập cấu hình giải mã hóa cho khối phần cứng bằng chương trình phần mềm bao gồm:

Công đoạn 31.1: Thiết lập chế chế độ giải mã hóa AES-CCM cho khối mã hóa, giải mã hóa phần cứng;

Công đoạn 31.2: Thiết lập độ dài khóa dữ liệu cho khối mã hóa, giải mã hóa phần cứng với độ dài 128-bit, 192-bit hoặc 256-bit;

Công đoạn 31.3: Thiết lập độ dài nhãn xác thực của khối dữ liệu cần mã hóa cho khối mã hóa, giải mã hóa phần cứng;

Công đoạn 31.3: Thiết lập kích thước khối dữ liệu cần giải mã hóa cho khối mã hóa, giải mã hóa phần cứng;

Bước 32: Kiểm tra tính hợp lệ của khối dữ liệu cần giải mã hóa bằng cách so sánh độ dài khối dữ liệu cần giải mã với độ dài nhãn xác thực:

Công đoạn 32.1: Nếu độ dài khối dữ liệu cần giải mã ngắn hơn độ dài nhãn xác thực được cấu hình thì thông báo dữ liệu cần giải mã không phù hợp và chuyển sang bước 44;

Công đoạn 32.2: Nếu độ dài khối dữ liệu cần mã hóa dài hơn hoặc bằng độ dài dãn nhữ liệu được cấu hình thì chuyển sang bước 33;

Bước 33: Kích hoạt khối phần cứng sử dụng chế độ giải mã hóa AES-CCM lần một, bản chất là thực hiện mã hóa theo chế độ AES-CTR, thông qua phần mềm;

Bước 34: Thực hiện giải mã hóa chế độ AES-CCM lần một trên phần cứng;

Bước 35: Lưu trữ dữ liệu giải mã hóa lần một trên phần cứng (khối dữ liệu giải mã hóa bao gồm bản tin gốc m sau khi được giải mã hóa vã nhãn xác thực t theo phương trình (12) và (13)) và trích xuất ra phần mềm;

Bước 36: Tái định dạng cấu trúc dữ liệu cần giải mã hóa dựa trên thuật toán AES-CCM bằng chương trình phần mềm theo phương trình (1) bao gồm:

Công đoạn 36.1: Xây dựng cờ dữ liệu cho khối dữ liệu cần giải mã hóa từ độ dài nhãn xác thực $Tlen$, độ dài khối dữ liệu cần được mã hóa $|m|$và sự hiện diện của khối dữ liệu bổ sung A;

Công đoạn 36.2: Xây dựng khối dữ liệu đầu vào $\beta(N, m, a)$, là tổ hợp của cờ dữ liệu, khối dữ liệu dùng một lần N, bản tin gốc m đã được giải mã hóa, và dữ liệu bổ sung a;

Bước 37: Kích hoạt khối phần cứng sử dụng chế độ giải mã hóa AES-CCM lần hai, bản chất là thực hiện mã hóa theo chế độ AES-ECB, thông qua phần mềm theo phương trình (2);

Bước 38: Thực hiện giải mã hóa chế độ AES-CCM lần hai trên phần cứng;

Bước 39: Lưu trữ dữ liệu giải mã hóa lần hai trên phần cứng;

Bước 40: Kích hoạt khối phần cứng sử dụng chế độ giải mã hóa AES-CCM lần ba, bản chất là thực hiện mã hóa theo chế độ AES-CBC, thông qua phần mềm theo phương trình (3);

Bước 41: Thực hiện giải mã hóa chế độ AES-CCM lần ba trên phần cứng;

Bước 42: Lưu trữ dữ liệu giải mã hóa lần ba trên phần cứng (khối dữ liệu giải mã hóa bao gồm nhãn xác thực *t'*) và trích xuất ra phần mềm;

Bước 43: So sánh hai nhãn xác thực dữ liệu *t* và *t'*:

Công đoạn 43.1: Nếu hai nhãn xác thực *t* và *t'* giống nhau thì phản hồi dữ liệu đã giải mã hóa ở bước 35;

Công đoạn 43.2: Nếu hai nhãn xác thực *t* và *t'* khác nhau thì thông báo khối dữ liệu cần giải mã hóa bị lỗi;

Bước 44: Kết thúc quá trình giải mã hóa dữ liệu theo chế độ giải mã hóa AES-CCM;

Tương tự như đối với việc mã hóa, để hiểu rõ hơn về phương pháp giải mã hóa xử lý đồng thời phần cứng và phần mềm sử dụng thuật toán giải mã hóa AES-CCM, các bước xử lý phần mềm và các bước xử lý phần cứng độc lập sẽ được tập hợp thành các phương pháp xử lý riêng và được trình bày dưới đây.

Như thể hiện trên Hình 5 các bước thực hiện phương pháp xử lý giải mã hóa AES-CCM trên phần mềm như sau:

Bước 50: Kích hoạt hệ thống phần cứng mã hóa bằng chương trình phần mềm;

Bước 51: Thiết lập cấu hình giải mã hóa AES-CCM cho khối mã hóa, giải mã hóa phần cứng;

Bước 52: Kiểm tra tính hợp lệ của khối dữ liệu cần giải mã hóa bằng cách so sánh độ dài khối dữ liệu cần giải mã với độ dài nhãn xác thực;

Bước 53: Kích hoạt khối mã hóa, giải mã hóa phần cứng sử dụng chế độ giải mã hóa AES-CCM lần một;

Bước 54: Đợi khối mã hóa, giải mã hóa phần cứng thực thi giải mã hóa AES-CCM lần một xong;

Bước 55: Tiếp nhận dữ liệu được trích xuất từ khối phần cứng;

Bước 56: Tái định dạng cấu dữ liệu cần giải mã hóa dữ trên thuật toán AES-CCM theo phương trình (1);

Bước 57: Kích hoạt khối mã hóa, giải mã hóa phần cứng sử dụng chế độ giải mã hóa AES-CCM lần hai;

Bước 58: Đợi khối mã hóa, giải mã hóa phần cứng thực thi giải mã hóa AES-CCM lần hai xong;

Bước 59: Kích hoạt khối mã hóa, giải mã hóa phần cứng sử dụng chế độ giải mã hoá AES-CCM lần ba;

Bước 60: Đợi khối mã hóa, giải mã hóa phần cứng thực thi giải mã hóa AES-CCM lần ba xong;

Bước 61: Tiếp nhận dữ liệu được trích xuất từ khối phần cứng;

Bước 62: So sánh hai nhãn xác thực dữ liệu t và t';

Bước 63: Kết thúc quá trình xử lý phần mềm cho chế độ giải mã hóa AES-CCM.

Như thể hiện trên Hình 6 các bước thực hiện phương pháp xử lý trên phần cứng như sau:

Bước 70: Hoạt động ở trạng thái chờ, đợi tín hiệu kích hoạt từ phần mềm;

Bước 71: Tiếp nhận tín hiệu kích hoạt và cấu hình chế độ giải mã hóa AES-CCM từ phần mềm;

Bước 72: Đợi tín hiệu kích hoạt giải mã hóa AES-CCM lần một từ phần mềm;

Bước 73: Thực hiện giải mã hóa lần một với chế độ AES-CCM theo điều khiển của phần mềm;

Bước 74: Lưu trữ dữ liệu giải mã hóa lần một và trích xuất khối dữ liệu này ra phần mềm;

Bước 75: Đợi phần mềm thực hiện tái định dạng cấu trúc dữ liệu cần giải mã hóa và tín hiệu kích hoạt giải mã hóa AES-CCM lần hai;

Bước 76: Tiếp nhận tín hiệu kích hoạt chế độ giải mã hóa AES-CCM lần hai từ phần mềm;

Bước 77: Thực hiện giải mã hóa lần hai với chế độ AES-CCM theo điều khiển của phần mềm;

Bước 78: Lưu trữ dữ liệu giải mã hóa lần hai trên phần cứng;

Bước 79: Đợi tín hiệu kích hoạt giải mã hóa AES-CCM lần ba từ phần mềm;

Bước 80: Tiếp nhận tín hiệu kích hoạt chế độ giải mã hóa AES-CCM lần ba từ phần mềm;

Bước 81: Thực hiện giả mã hóa lần ba với chế độ AES-CCM theo điều khiển của phần mềm;

Bước 82: Lưu trữ dữ liệu giải mã hóa lần ba và trích xuất khối dữ liệu này ra phần mềm;

Bước 83: Kết thúc quá trình xử lý phần cứng cho chế độ giải mã hóa AES-CCM;

Hiệu quả đạt được từ phương pháp

Với các đặc điểm được mô tả trên đây, phương pháp xử lý đồng thời phần cứng và phần mềm cho thuật toán bảo mật AES-CCM ứng dụng cho các thiết bị IoT theo sáng chế có thể mang lại các hiệu quả dưới đây:

Tận dụng khả năng tính toán song song của phần cứng cho các phép tính sử dụng trong các chế độ mã hóa, giải mã hóa của thuật toán AES-CCM giúp tăng tốc độ xử lý cho thiết bị IoT khi so với thực thi thuần thuật toán bằng phần mềm.

Giảm thiểu thực thi các phép toán phức tạp trên phần cứng giúp tối ưu diện tích và năng lượng tiêu thụ cho thiết bị IoT, đáp ứng các tiêu chí nhỏ gọn và thời gian sống của thiết bị dài.

Áp dụng khối mã hóa, giải mã hóa phần cứng đơn giản làm nền tảng phát triển cho nhiều ứng dụng bảo mật khác cho thiết bị IoT.

Yêu cầu bảo hộ

1. Phương pháp mã hóa, thực hiện quá trình mã hóa nhờ xử lý phần cứng và phần mềm sử dụng thuật toán xác thực và định danh AES-CCM, phương pháp này bao gồm các bước:

kích hoạt hệ thống phần cứng mã hóa bằng chương trình phần mềm, trong đó câu lệnh kích hoạt sẽ thông qua giao diện truyền thông truyền tới khối vi điều khiển để tác động vào khối mã hóa, giải mã hóa phần cứng;

thiết lập cấu hình mã hóa AES-CCM cho khối phần cứng bằng chương trình phần mềm, trong đó bước này bao gồm các công đoạn:

thiết lập chế độ mã hóa AES-CCM cho khối mã hóa, giải mã hóa phần cứng;

thiết lập độ dài khóa dữ liệu cho khối mã hóa, giải mã hóa phần cứng;

thiết lập độ dài nhãn xác thực của khối dữ liệu cần mã hóa cho khối mã hóa, giải mã hóa phần cứng;

thiết lập kích thước khối dữ liệu cần mã hóa cho khối mã hóa, giải mã hóa phần cứng;

tái định dạng cấu trúc dữ liệu cần mã hóa dựa trên thuật toán AES-CCM bằng chương trình phần mềm theo phương trình dưới đây:

$$\beta(N, m, a) = B_0.B_1.\cdots.B_r$$

trong đó: $\beta(N, m, a)$ là khối dữ liệu đầu vào, B_i là các khối con, N là tổ hợp của khối dữ liệu dùng một lần, m là bản tin, và a là dữ liệu bổ sung,

bước này bao gồm các công đoạn:

xây dựng cờ dữ liệu cho khối dữ liệu cần mã hóa từ độ dài nhãn xác thực $Tlen$, độ dài khối dữ liệu cần được mã hóa $|m|$và sự hiện diện của khối dữ liệu bổ sung a;

xây dựng khối dữ liệu đầu vào $\beta(N, m, a)$, là tổ hợp của cờ dữ liệu, khối dữ liệu dùng một lần N, bản tin m, và dữ liệu bổ sung a;

kích hoạt khối phần cứng sử dụng chế độ mã hóa AES-CCM lần một, bản chất là thực hiện mã hóa theo chế độ AES-CBC, thông qua phần mềm;

thực hiện mã hóa chế độ AES-CCM lần một trên phần cứng;

lưu trữ dữ liệu mã hóa lần một trên phần cứng (khối dữ liệu mã hóa bao gồm cả nhãn xác thực t);

kích hoạt khối phần cứng sử dụng chế độ AES-CCM lần hai, bản chất là thực hiện mã hóa theo chế độ AES-CTR, thông qua phần mềm;

thực hiện mã hóa chế độ AES-CCM lần hai trên phần cứng;

lưu trữ dữ liệu mã hóa lần hai trên phần cứng (khối dữ liệu mã hóa bao gồm dữ liệu bảo mật sau khi mã hóa AES-CCM) và trích xuất khối dữ liệu này ra phần mềm;

kết thúc quá trình mã hóa dữ liệu theo chế độ mã hóa AES-CCM.

2. Phương pháp giải mã hóa, thực hiện quá trình mã hóa nhờ xử lý phần cứng và phần mềm sử dụng thuật toán xác thực và định danh AES-CCM, phương pháp này bao gồm các bước:

kích hoạt hệ thống phần cứng mã hóa bằng chương trình phần mềm, trong đó câu lệnh kích hoạt sẽ thông qua giao diện truyền thông truyền tới khối vi điều khiển để tác động vào khối mã hóa, giải mã hóa phần cứng;

thiết lập cấu hình giải mã hóa cho khối phần cứng bằng chương trình phần mềm, trong đó bước này bao gồm các công đoạn:

thiết lập chế chế độ giải mã hóa AES-CCM cho khối mã hóa, giải mã hóa phần cứng;

thiết lập độ dài khóa dữ liệu cho khối mã hóa, giải mã hóa phần cứng;

thiết lập độ dài nhãn xác thực của khối dữ liệu cần mã hóa cho khối mã hóa, giải mã hóa phần cứng;

thiết lập kích thước khối dữ liệu cần giải mã hóa cho khối mã hóa, giải mã hóa phần cứng;

kiểm tra tính hợp lệ của khối dữ liệu cần giải mã hóa bằng cách so sánh độ dài khối dữ liệu cần giải mã với độ dài nhãn xác thực, trong đó bước này bao gồm các công đoạn:

nếu độ dài khối dữ liệu cần giải mã ngắn hơn độ dài nhãn xác thực được cấu hình thì thông báo dữ liệu cần giải mã không phù hợp và chuyển sang bước so sánh hai nhãn xác thực dữ liệu *t* và *t'* dưới đây,

nếu độ dài khối dữ liệu cần mã hóa dài hơn hoặc bằng độ dài dãn nhữ liệu được cấu hình thì chuyển sang bước kích hoạt khối phần cứng sử dụng chế độ giải mã hóa AES-CCM lần một ngay dưới đây;

kích hoạt khối phần cứng sử dụng chế độ giải mã hóa AES-CCM lần một, bản chất là thực hiện mã hóa theo chế độ AES-CTR, thông qua phần mềm;

thực hiện giải mã hóa chế độ AES-CCM lần một trên phần cứng;

lưu trữ dữ liệu giải mã hóa lần một trên phần cứng (khối dữ liệu giải mã hóa bao gồm bản tin gốc *m* trước khi mã hóa vã nhãn xác thực *t* theo các phương trình dưới đây:

$$m = \mathrm{MSB}_{Clen-Tlen}(C) \oplus \mathrm{MSB}_{Clen-Tlen}(S)$$
$$t = \mathrm{LSB}_{Clen-Tlen}(C) \oplus \mathrm{MSB}_{Clen-Tlen}(S_0)$$

trong đó: MSB_i là *i* bit quan trọng nhất (Most Significant Bit), hay *i* bit có trọng số lớn nhất.

LSB_i là *i* bit ít quan trọng nhất (Least Significant Bit), hay *i* bit có trọng số nhỏ nhất.

và trích xuất ra phần mềm;

tái định dạng cấu trúc dữ liệu cần giải mã hóa dựa trên thuật toán AES-CCM bằng chương trình phần mềm theo phương trình dưới đây:

$$\beta(N, m, a) = B_0.B_1.\cdots.B_r$$

trong đó: $\beta(N, m, a)$ là khối dữ liệu đầu vào, B_i là các khối con, N là tổ hợp của khối dữ liệu dùng một lần, m là bản tin, và a là dữ liệu bổ sung,

bước này bao gồm các công đoạn:

xây dựng cờ dữ liệu cho khối dữ liệu cần giải mã hóa từ độ dài nhãn xác thực $Tlen$, độ dài khối dữ liệu cần được mã hóa $|m|$và sự hiện diện của khối dữ liệu bổ sung a,

xây dựng khối dữ liệu đầu vào $\beta(N, m, a)$, là tổ hợp của cờ dữ liệu, khối dữ liệu dùng một lần N, bản tin m, và dữ liệu bổ sung a;

kích hoạt khối phần cứng sử dụng chế độ giải mã hóa AES-CCM lần hai, bản chất là thực hiện mã hóa theo chế độ AES-ECB, thông qua phần mềm theo phương trình:

$$Y_0 = CIPH_k(B_0)$$

trong đó: Y_o là khối dữ liệu đầu tiên thu được sau khi được mã hóa (CIPH) khối dữ liệu B_0 với khóa dữ liệu k;

thực hiện giải mã hóa chế độ AES-CCM lần hai trên phần cứng;

lưu trữ dữ liệu giải mã hóa lần hai trên phần cứng;

kích hoạt khối phần cứng sử dụng chế độ giải mã hóa AES-CCM lần ba, bản chất là thực hiện mã hóa theo chế độ AES-CBC, thông qua phần mềm theo phương trình dưới đây:

$$Y_i = CIPH_k(Y_{i-1} \oplus B_i)$$

trong đó: Y_i là khối dữ liệu thứ i thu được sau khi được mã hóa (CIPH) kết quả của phép XOR giữa khối dữ liệu B_i và khối dữ liệu thu được thứ i-1 với khóa dữ liệu k;

thực hiện giải mã hóa chế độ AES-CCM lần ba trên phần cứng;

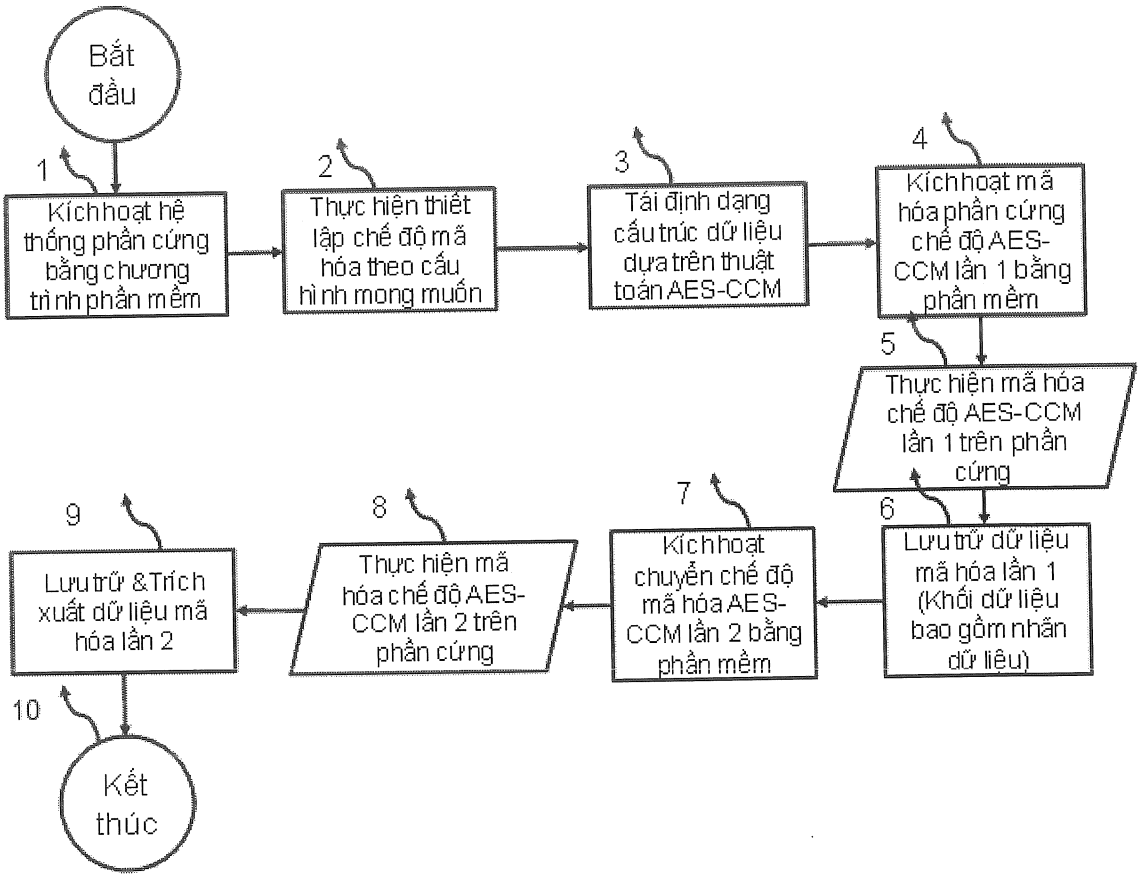
lưu trữ dữ liệu giải mã hóa lần ba trên phần cứng (khối dữ liệu giải mã hóa bao gồm nhãn xác thực *t'*) và trích xuất ra phần mềm;

so sánh hai nhãn xác thực dữ liệu *t* và *t'*, bước này bao gồm các công đoạn:

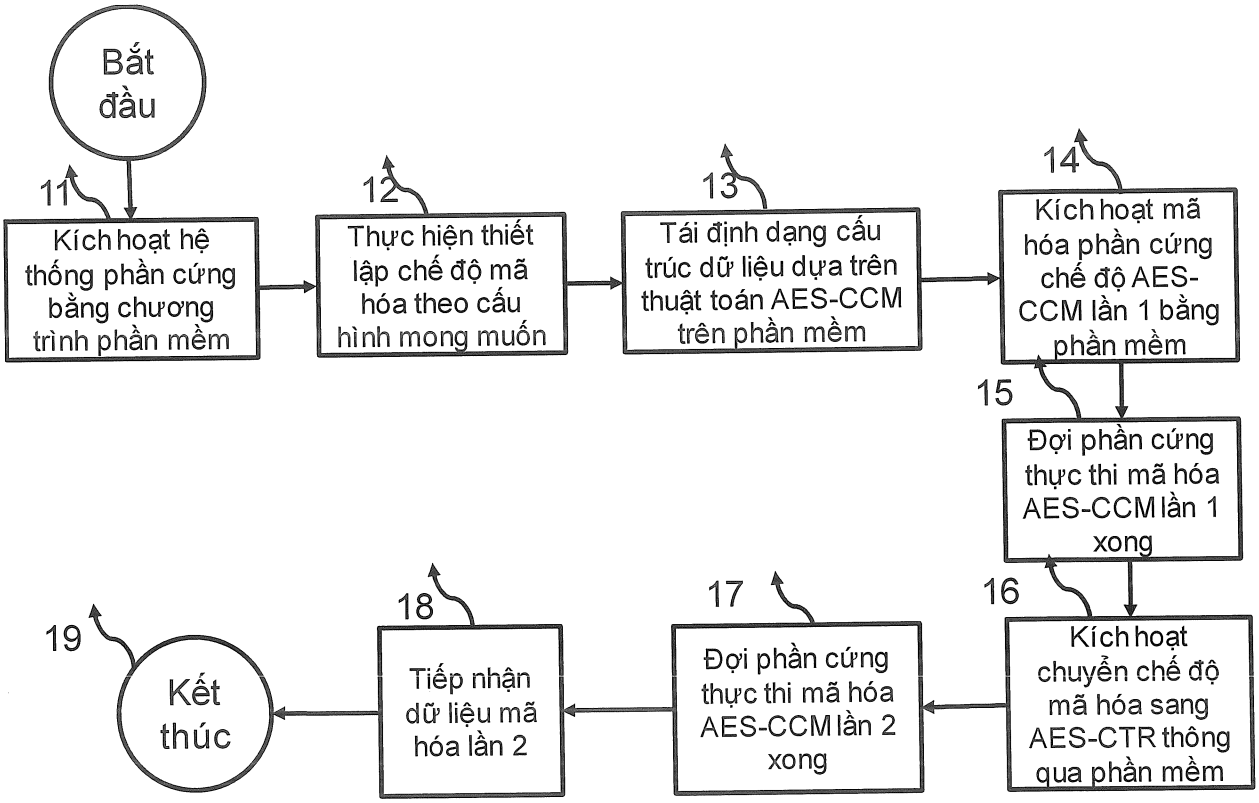
nếu hai nhãn xác thực *t* và *t'* giống nhau thì phản hồi dữ liệu đã giải mã hóa ở bước lưu trữ dữ liệu giải mã hóa lần một trên phần cứng nêu trên,

nếu hai nhãn xác thực *t* và *t'* khác nhau thì thông báo khối dữ liệu cần giải mã hóa bị lỗi; và

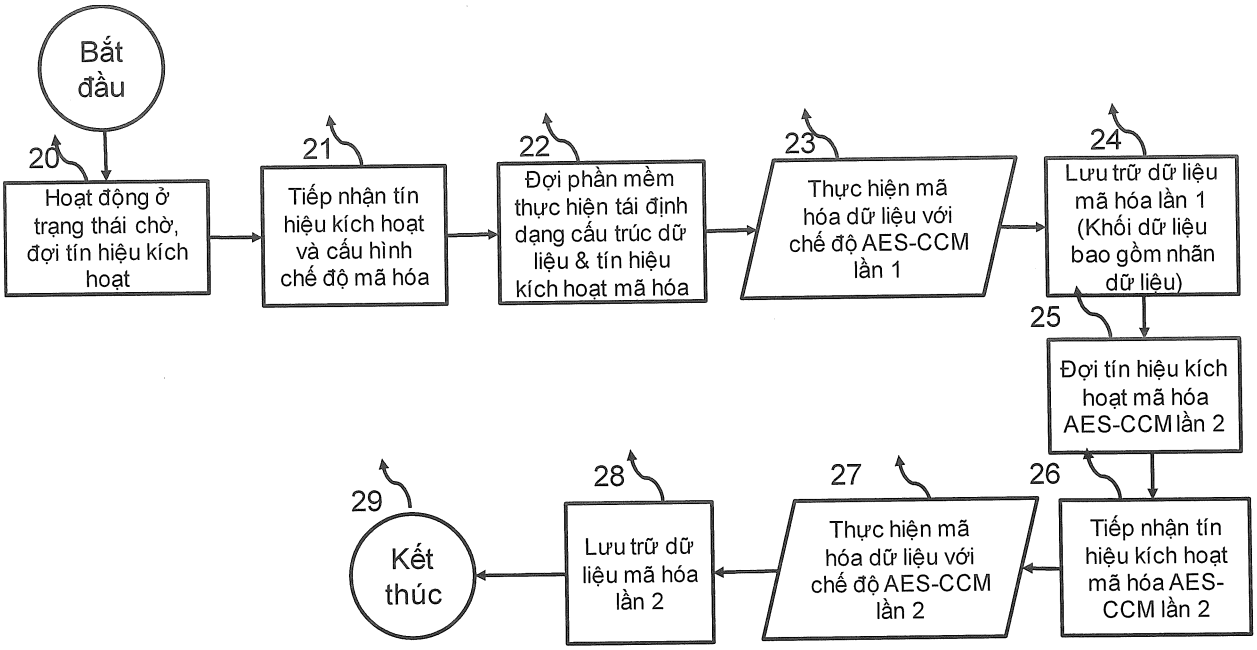
kết thúc quá trình giải mã hóa dữ liệu theo chế độ giải mã hóa AES-CCM.


Bắt đầu
1
Kích hoạt hệ thống phần cứng bằng chương trình phần mềm
2
Thực hiện thiết lập chế độ mã hóa theo cấu hình mong muốn
3
Tái định dạng cấu trúc dữ liệu dựa trên thuật toán AES-CCM
4
Kích hoạt mã hóa phần cứng chế độ AES-CCM lần 1 bằng phần mềm
5
Thực hiện mã hóa chế độ AES-CCM lần 1 trên phần cứng
6
Lưu trữ dữ liệu mã hóa lần 1 (Khối dữ liệu bao gồm nhãn dữ liệu)
7
Kích hoạt chuyển chế độ mã hóa AES-CCM lần 2 bằng phần mềm
8
Thực hiện mã hóa chế độ AES-CCM lần 2 trên phần cứng
9
Lưu trữ &Trích xuất dữ liệu mã hóa lần 2
10
Kết thúc

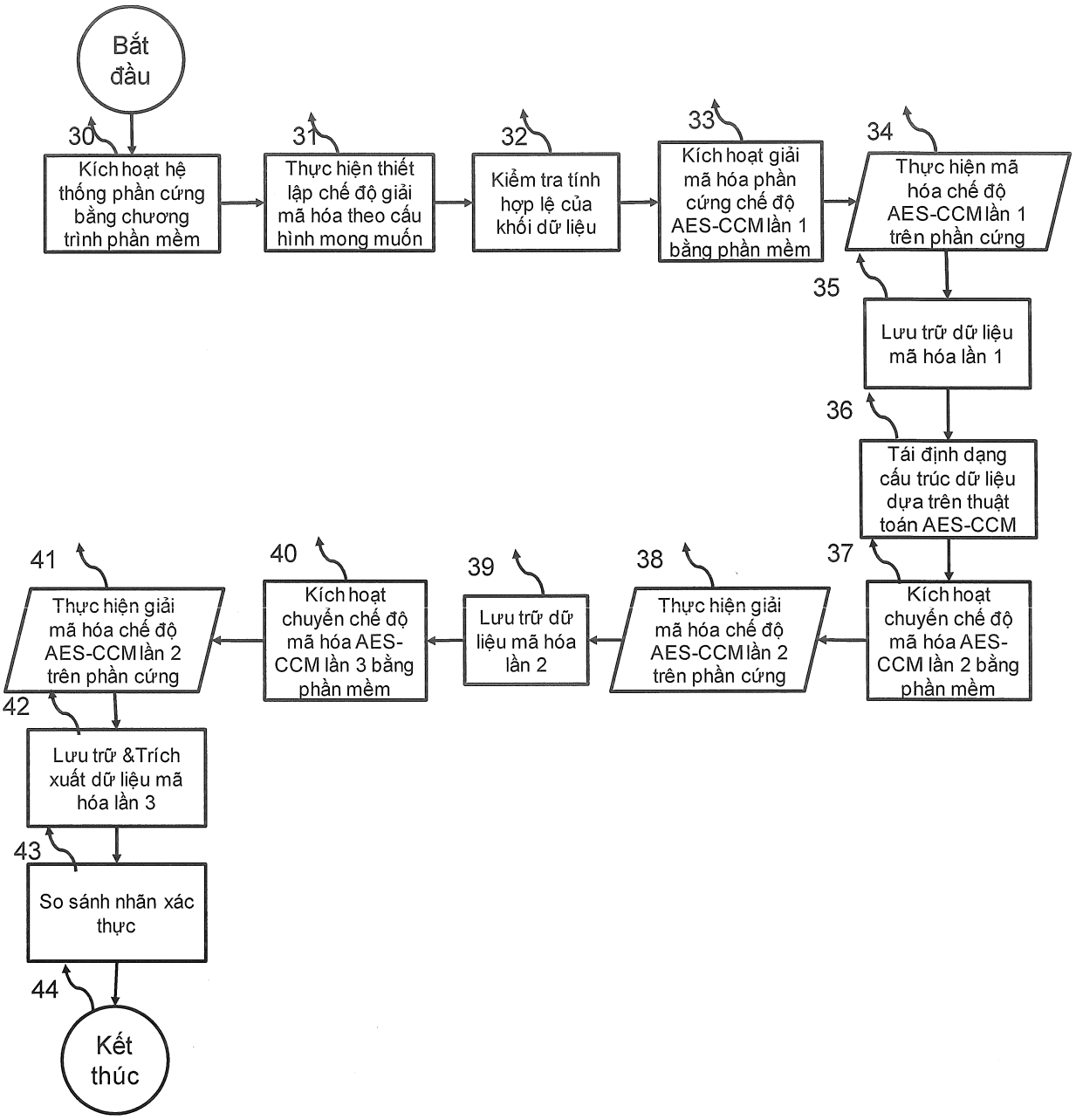
Hình 1


Bắt đầu
11
Kích hoạt hệ thống phần cứng bằng chương trình phần mềm
12
Thực hiện thiết lập chế độ mã hóa theo cấu hình mong muốn
13
Tái định dạng cấu trúc dữ liệu dựa trên thuật toán AES-CCM trên phần mềm
14
Kích hoạt mã hóa phần cứng chế độ AES-CCM lần 1 bằng phần mềm
15
Đợi phần cứng thực thi mã hóa AES-CCM lần 1 xong
16
Kích hoạt chuyển chế độ mã hóa sang AES-CTR thông qua phần mềm
17
Đợi phần cứng thực thi mã hóa AES-CCM lần 2 xong
18
Tiếp nhận dữ liệu mã hóa lần 2
19
Kết thúc

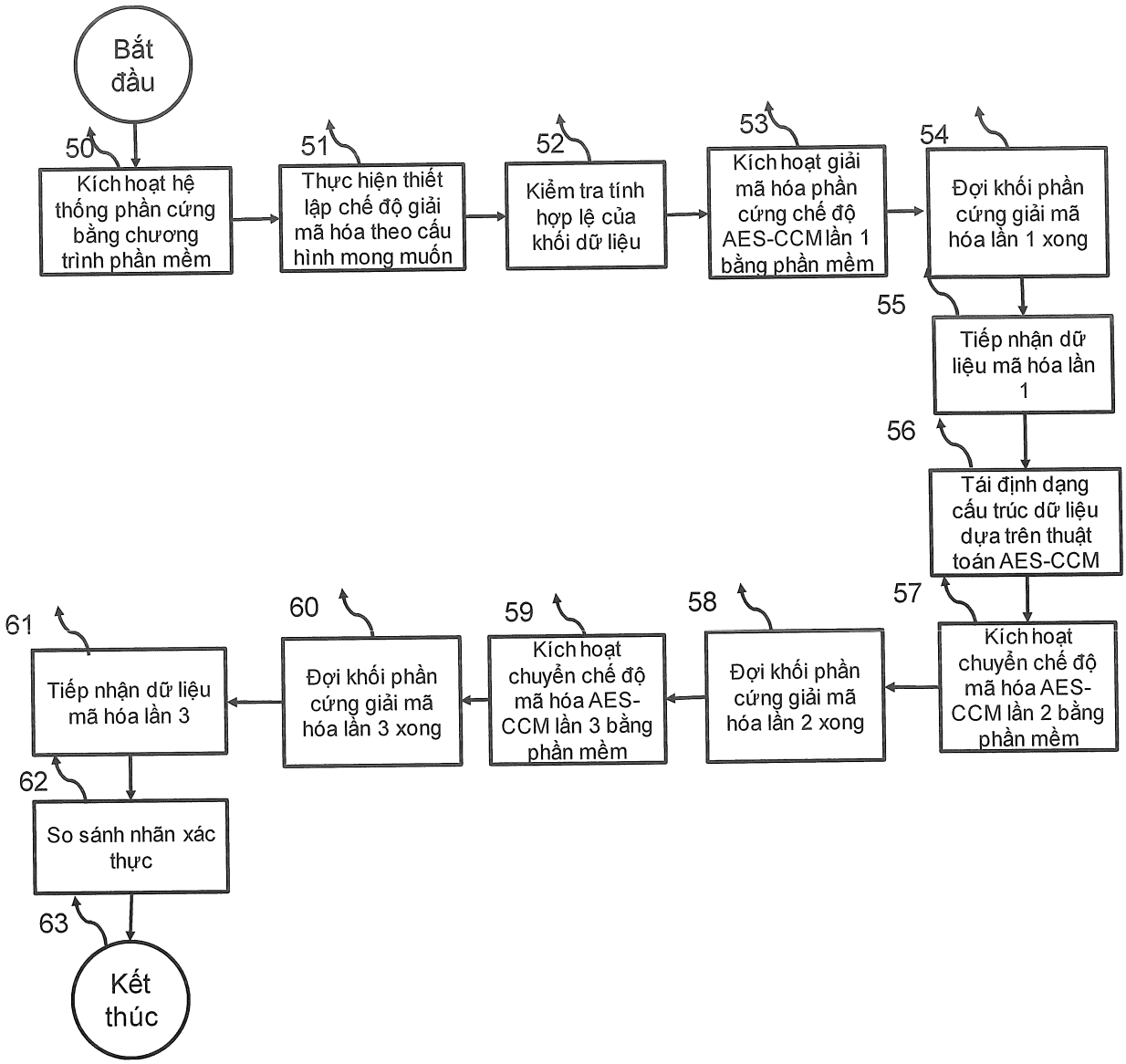
Hình 2


Bắt đầu
20
Hoạt động ở trạng thái chờ, đợi tín hiệu kích hoạt
21
Tiếp nhận tín hiệu kích hoạt và cấu hình chế độ mã hóa
22
Đợi phần mềm thực hiện tái định dạng cấu trúc dữ liệu & tín hiệu kích hoạt mã hóa
23
Thực hiện mã hóa dữ liệu với chế độ AES-CCM lần 1
24
Lưu trữ dữ liệu mã hóa lần 1 (Khối dữ liệu bao gồm nhãn dữ liệu)
25
Đợi tín hiệu kích hoạt mã hóa AES-CCM lần 2
26
Tiếp nhận tín hiệu kích hoạt mã hóa AES-CCM lần 2
27
Thực hiện mã hóa dữ liệu với chế độ AES-CCM lần 2
28
Lưu trữ dữ liệu mã hóa lần 2
29
Kết thúc

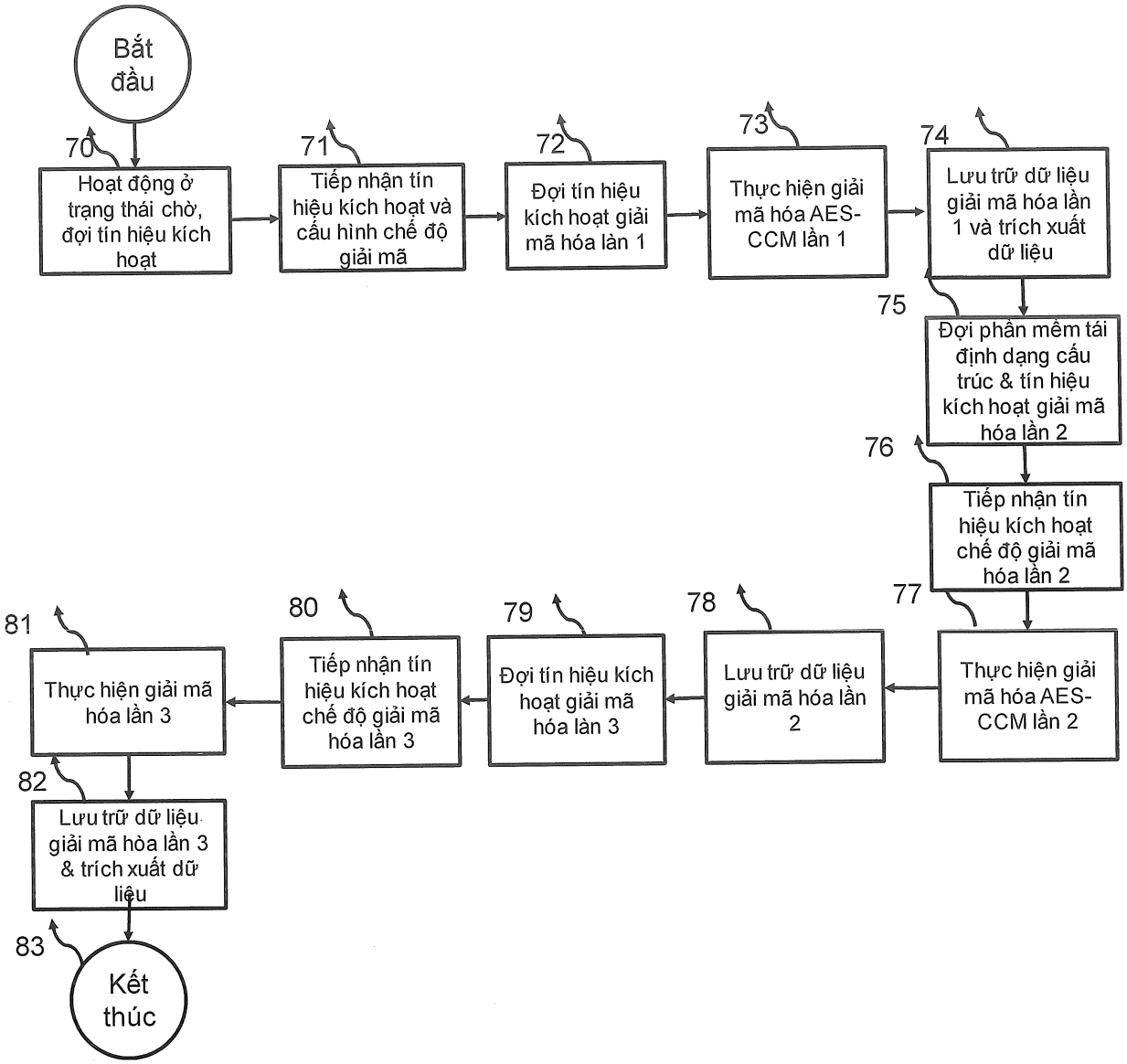
Hình 3


Bắt đầu
30
Kích hoạt hệ thống phần cứng bằng chương trình phần mềm
31
Thực hiện thiết lập chế độ giải mã hóa theo cấu hình mong muốn
32
Kiểm tra tính hợp lệ của khối dữ liệu
33
Kích hoạt giải mã hóa phần cứng chế độ AES-CCM lần 1 bằng phần mềm
34
Thực hiện mã hóa chế độ AES-CCM lần 1 trên phần cứng
35
Lưu trữ dữ liệu mã hóa lần 1
36
Tái định dạng cấu trúc dữ liệu dựa trên thuật toán AES-CCM
37
Kích hoạt chuyển chế độ mã hóa AES-CCM lần 2 bằng phần mềm
38
Thực hiện giải mã hóa chế độ AES-CCM lần 2 trên phần cứng
39
Lưu trữ dữ liệu mã hóa lần 2
40
Kích hoạt chuyển chế độ mã hóa AES-CCM lần 3 bằng phần mềm
41
Thực hiện giải mã hóa chế độ AES-CCM lần 2 trên phần cứng
42
Lưu trữ &Trích xuất dữ liệu mã hóa lần 3
43
So sánh nhãn xác thực
44
Kết thúc

Hình 4


Bắt đầu
50
Kích hoạt hệ thống phần cứng bằng chương trình phần mềm
51
Thực hiện thiết lập chế độ giải mã hóa theo cấu hình mong muốn
52
Kiểm tra tính hợp lệ của khối dữ liệu
53
Kích hoạt giải mã hóa phần cứng chế độ AES-CCM lần 1 bằng phần mềm
54
Đợi khối phần cứng giải mã hóa lần 1 xong
55
Tiếp nhận dữ liệu mã hóa lần 1
56
Tái định dạng cấu trúc dữ liệu dựa trên thuật toán AES-CCM
57
Kích hoạt chuyển chế độ mã hóa AES-CCM lần 2 bằng phần mềm
58
Đợi khối phần cứng giải mã hóa lần 2 xong
59
Kích hoạt chuyển chế độ mã hóa AES-CCM lần 3 bằng phần mềm
60
Đợi khối phần cứng giải mã hóa lần 3 xong
61
Tiếp nhận dữ liệu mã hóa lần 3
62
So sánh nhãn xác thực
63
Kết thúc

Hình 2


Bắt đầu
70
Hoạt động ở trạng thái chờ, đợi tín hiệu kích hoạt
71
Tiếp nhận tín hiệu kích hoạt và cấu hình chế độ giải mã
72
Đợi tín hiệu kích hoạt giải mã hóa làn 1
73
Thực hiện giải mã hóa AES-CCM lần 1
74
Lưu trữ dữ liệu giải mã hóa lần 1 và trích xuất dữ liệu
75
Đợi phần mềm tái định dạng cấu trúc & tín hiệu kích hoạt giải mã hóa lần 2
76
Tiếp nhận tín hiệu kích hoạt chế độ giải mã hóa lần 2
77
Thực hiện giải mã hóa AES-CCM lần 2
78
Lưu trữ dữ liệu giải mã hóa lần 2
79
Đợi tín hiệu kích hoạt giải mã hóa làn 3
80
Tiếp nhận tín hiệu kích hoạt chế độ giải mã hóa lần 3
81
Thực hiện giải mã hóa lần 3
82
Lưu trữ dữ liệu giải mã hòa lần 3 & trích xuất dữ liệu
83
Kết thúc

Hình 3