ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

Mai Mạnh Trừng

NGHIÊN CỨU MẬT MÃ NHẸ, ỨNG DỤNG TRÊN THIẾT BỊ THÔNG MINH

Chuyên ngành: Khoa học máy tính

Mã số: 9480101.01

TÓM TẮT LUẬN ÁN TIẾN SĨ KHOA HỌC MÁY TÍNH

Hà Nội – 2022

Công trình được hoàn thành tại: Trường Đại học Công nghệ, Đại học Quốc gia Hà Nội

Người hướng dẫn khoa học:

1. PGS.TS. Đỗ Trung Tuấn

2. TS. Lê Phê Đô

Phản biện: .....

.....

Phản biện: .....

.....

Phản biện: .....

.....

Luận án sẽ được bảo vệ trước Hội đồng cấp Đại học Quốc gia chấm luận án tiến sĩ họp tại .....

vào hồi giờ ngày tháng năm

Có thể tìm hiểu luận án tại:

- Thư viện Quốc gia Việt Nam

- Trung tâm Thông tin - Thư viện, Đại học Quốc gia Hà Nội

Mục Lục

MỞ ĐẦU

Bảo mật và đảm bảo an toàn thông tin luôn là vấn đề được các nhà nghiên cứu quan tâm, là một chủ đề rộng có liên quan đến nhiều lĩnh vực. Trong thực tế có rất nhiều phương pháp được thực hiện để đảm bảo an toàn thông tin dữ liệu từ thời xưa, từ những dấu niêm phong trên bao thư cho đến những mật mã thay thế đơn giản dùng để mã hóa, ẩn giấu thông tin.

Ngày nay với sự phát triển của khoa học công nghệ máy tính, thì việc lưu trữ và bảo mật thông tin được thực hiện một cách phổ biến trên máy vi tính hay các thiết bị công nghệ số như điện thoại di động. Đặc biệt với các công nghệ điện tử bán dẫn, công nghệ mạng không dây,... thì ngày càng xuất hiện nhiều thiết bị có cấu hình khá cao với năng lực tính toán lớn cho phép triển khai dễ dàng các thuật toán mã hóa như DES, RSA, AES... để mã hóa, bảo mật thông tin.

Tuy nhiên, nhu cầu sử dụng các thiết bị có kích cỡ nhỏ, khả năng tính toán thấp phục vụ các công việc và giải quyết bài toán chuyên dụng, đơn giản, điển hình như các thẻ thông minh (smartcard), vi điều khiển (microcontroller) ngày càng tăng. Trong khi đó, các hệ mật mã truyền thống hiện có khó có thể sử dụng đa năng cho mọi kiểu thiết bị (bộ vi xử lý), do sự phức tạp, sử dụng nhiều tài nguyên, năng lượng. Một hệ mật mã truyền thống cũng khó có thể cài đặt hiệu quả trên các thiết bị có năng lực và tài nguyên hạn chế (như các bộ vi điều khiển 4 bit, 8 bit, có kích cỡ RAM nhỏ, tần số thấp, v.v...). Vì vậy, nhu cầu cần có các hệ mật mã (mã khối, mã dòng, hàm băm, mã xác thực, ECC) riêng, áp dụng cho các thiết bị, hệ thống bị hạn chế (thông tin cần phải bảo vệ không cần bảo mật quá cao) đã và đang được đặt ra trong những năm qua, và mật mã hạng nhẹ (Lightweight Cryptograhy) được ra đời từ các nhu cầu đó. Theo tiêu chuẩn ISO/IEC 29192-1 đã đưa ra khái niệm về mật mã hạng nhẹ phù hợp với những môi trường hạn chế.

Mục tiêu của mật mã nhẹ là một loạt các ứng dụng cho các thiết bị hiện đại, như các thiết bị đo thông minh, hệ thống an ninh xe, hệ thống giám sát bệnh nhân không dây, hệ thống giao thông thông minh (lntelligent Transport System - ITS) và Internet of Things (IoT),…. Ở những trường hợp này, các thuật toán mật mã truyền thống là không phù hợp bởi các lý do như chi phí quá lớn, độ an toàn quá cao so với mức cần thiết, năng lượng tiêu thụ quá lớn,... . Do đó sử dụng các thuật toán mật mã hạng nhẹ, thứ phù hợp với các thiết bị hạn chế là vấn đề cần được quan tâm nghiên cứu. Vậy nghiên cứu sinh đã lựa chọn luận án *"Nghiên cứu mật mã nhẹ, ứng dụng trên thiết bị thông minh"*.

Ngoài phần mở đầu và kết luận, bố cục của luận án chia thành 04 chương như sau:

Chương 1 của luận án trình bày những kiến thức chung về mật mã và mật mã hạng nhẹ, trình bày các thiết bị thông minh, thiết bị có năng lực tính toán hạn chế. Chương này cũng trình bày các tham số để đánh giá một thuật toán mật mã là thuật toán mật mã hạng nhẹ và mục tiêu của luận án giải quyết các vấn đề này.

Chương 2 trình bày một số hệ mật tiêu biểu như Present và Grain có đánh giá theo các tham số.

Chương 3 của luận án trình bày cơ sở toán học của hệ mật đường cong Elliptic, đề xuất thuật toán mật mã đường cong Elliptic ứng dụng trên các thiết bị di động

Chương 4 của luận án trình bày các kết quả nghiên cứu đối với bài toán mật mã nhẹ trên các thiết bị hạn chế dựa trên kết quả thực nghiệm.

CHƯƠNG 1. TỔNG QUAN VỀ MẬT MÃ NHẸ VÀ ỨNG DỤNG TRÊN THIẾT BỊ THÔNG MINH

1.1. Bài toán nghiên cứu

Trên cơ sở thực tiễn nhiều hệ mật mã nhẹ được đề xuất nhưng chưa được đánh giá đầy đủ và cần phải tạo ra hệ mật mã nhẹ mới có hiệu suất cao, luận án đề ra cho mình nhiệm vụ đánh giá một số hệ mật mã mới, phổ biến và đề xuất hệ mật mã nhẹ mới cho thiết bị thông minh.

Mật mã nhẹ hướng tới việc tạo ra các giải pháp cài đặt rất gọn nhẹ nhưng không làm giảm đi quá nhiều về tính an toàn. Nó là một giải pháp đưa ra thỏa hiệp giữa độ an toàn và tính hiệu quả trong cài đặt của các thuật toán mật mã. Ngày càng nhiều các sản phẩm được nâng cấp thành các thiết bị thâm nhập khắp nơi nhờ năng lực tính toán nhúng. Quan hệ mật mã thiết giữa các thiết bị này dẫn đến triển vọng rằng tính toán khắp nơi và sẽ là mô hình tiếp theo trong công nghệ thông tin. Một nhân tố làm tăng nguy cơ mất an toàn là các thiết bị thâm nhập khắp nơi thường triển khai trong một môi trường không được kiểm soát mà đây lại là một môi trường mà đối phương có thể truy cập vật lý tới thiết bị hoặc điều khiển thiết bị. Vì thế, khả năng an toàn và bảo mật của các thiết bị cần được hết sức chú trọng. Với những lý do trên, luận án nghiên cứu "*Nghiên cứu mật mã nhẹ, ứng dụng trên thiết bị thông minh*".

1.2. Mật mã học

Mật mã là ngành khoa học nghiên cứu các kỹ thuật toán học nhằm cung cấp các dịch vụ bảo vệ thông tin. Đây là ngành khoa học quan trọng, có nhiều ứng dụng trong đời sống xã hội. Khoa học mật mã đã ra đời từ thời cổ đại, trong suốt nhiều thế kỷ, mật mã có vai trò quan trọng trong lĩnh vực quân sự, chính trị, ngoại giao,…Tuy nhiên, trong lĩnh vực dân sự thông thường của đời sống-xã hội thì hầu như không được ứng dụng.

Ngày nay, các ứng dụng mã hóa và bảo mật thông tin được được sử dụng ngày càng phổ biến trong các lĩnh vực khác nhau trên thế giới, từ các lĩnh vực như trên cho đến các lĩnh vực dân sự như giao dịch mua bán, thương mại điện tử, tài chính, ngân hàng, …

Định nghĩa 1.1. Mật mã

Mật mã là những quy tắc, quy ước riêng dùng để thay đổi hình thức biểu hiện thông tin nhằm bảo đảm bí mật, xác thực, toàn vẹn của nội dung thông tin.

Một sơ đồ hệ thống mật mã là một bộ năm thành phần: $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ thỏa mãn các điều kiện sau đây:

(i) $\mathcal{P}$ là một tập hữu hạn các ký tự bản rõ,

(ii) $\mathcal{C}$ là một tập hữu hạn các ký tự bản mã,

(iii) $\mathcal{K}$ là một tập hữu hạn các khóa,

(iv) $\mathcal{E}$ là một ánh xạ từ $\mathcal{K}$ x $\mathcal{P}$ vào $\mathcal{C}$, được gọi là phép lập mật mã,

(v) $\mathcal{D}$ là một ánh xạ từ $\mathcal{K}$ x $\mathcal{C}$ vào $\mathcal{P}$, được gọi là phép giải mã.

Với mỗi $K \in \mathcal{K}$, ta định nghĩa e_K: $\mathcal{P} \to \mathcal{C}$, $d_K : \mathcal{C} \to \mathcal{P}$ là hai hàm cho bởi: $\forall x \in \mathcal{P} : e_K(x) = \mathcal{E}(K, x)$; $\forall y \in \mathcal{C} : d_K(y) = \mathcal{D}(K, y)$. e_K, d_K được gọi lần lượt là hàm lập mã và giải mã ứng với khóa mật mã K. Các hàm đó phải thảo mãn hệ thức: $\forall x \in \mathcal{P} : d_K(e_K(x)) = x$.

1.3. Thiết bị thông minh và mạng vạn vật kết nối

1.3.1. Thiết bị thông minh

Định nghĩa 1.2. Thiết bị thông minh

Thiết bị thông minh là một thiết bị điện tử có khả năng thực hiện tính toán tự động và kết nối với các thiết bị khác thông qua mạng không dây Wifi, 3G, 4G,… [53].

Một số loại thiết bị tiêu biểu như điện thoại thông minh, máy tính bảng, đồng hồ thông minh, vòng đeo thông minh và chuỗi khóa thông minh...

Các thiết bị thông minh này có vai trò quan trọng trong cuộc sống hàng ngày của con người. Những thiết bị thông minh này có năng lực tính toán hạn chế, nguồn pin hạn chế,... Do vậy, để đảm bảo bảo mật cho các thiết bị này người ta cần một phần mềm với thuật toán phù hợp để những thiết bị thông minh có thể sử dụng. Các thuật toán truyền thống thường phù hợp với thiết bị có cấu hình lớn như máy để bàn, máy chủ…

1.3.2. Mạng vạn vật kết nối

Mạng vạn vật IoT là đề cập đến hàng tỷ thiết bị vật lý trên khắp thế giới được kết nối với Internet, thu thập và chia sẻ dữ liệu. Nhờ bộ xử lý bên trong cùng mạng không dây, người ta có thể biến mọi thứ trở nên chủ động và thông minh hơn. Ta có thể bắt gặp IoT từ hệ thống cửa tự động cho tới máy bay tới xe tự lái đã trở thành một phần phổ biến của IoT. Điều này bổ sung một mức độ thông minh kỹ thuật số cho các thiết bị thụ động, cho phép chúng giao tiếp dữ liệu thời gian thực mà không cần con người tham gia, hợp nhất hiệu quả thế giới kỹ thuật số và vật lý.

Định nghĩa 1.3. Mạng vạn vật kết nối

Mạng vạn vật kết nối là là mạng kết nối các đồ vật và thiết bị thông qua cảm biến, phần mềm và các công nghệ khác, cho phép các đồ vật và thiết bị thu thập và trao đổi dữ liệu với nhau.

1.4. Mật mã nhẹ

Với các hệ thống IoT sử dụng dữ liệu trong thế giới thực, việc thu thập dữ liệu từ các thiết bị cũng có thể là mục tiêu của các cuộc tấn công mạng. Chính vì điều này mà các biện pháp đối phó dựa trên mã hóa hiện đang trở nên quan trọng. Mật mã nhẹ là một phương pháp mã hóa cài đặt gọn nhẹ và với độ phức tạp tính toán thấp. Nó nhằm mục đích mở rộng các ứng dụng của mật mã cho các thiết bị bị hạn chế và đáp ứng tiêu chuẩn quốc tế. Mã hóa xác thực đạt được cả tính bảo mật và tính toàn vẹn đã thu hút sự chú ý đặc biệt và có một cuộc thi công nghệ mang tên CAESAR đã được tổ chức. NEC đã phát triển mật mã khối nhẹ TWINE và OTR.

Các yêu cầu của mật mã nhẹ với các yếu tố đặc trưng bao gồm: (i) kích thước (kích thước mạch, kích thước ROM/RAM); (ii) nguồn; (iii) sự tiêu thụ năng lượng; (iv) tốc độ xử lý (thông lượng, độ trễ). Yếu tố đầu tiên xác định khả năng thực hiện trong một thiết bị là kích thước. Tiếp theo, nguồn điện đặc biệt quan trọng với RFID và các thiết bị thu năng lượng trong khi mức tiêu thụ điện năng quan trọng với các thiết bị chạy bằng pin. Thông lượng cao là cần thiết cho các thiết bị có truyền dữ liệu lớn như máy ảnh hoặc cảm biến rung, trong khi độ trễ thấp là quan trọng đối với quá trình xử lý điều khiển thời gian thực của hệ thống điều khiển ô tô,…

Mật mã chia theo khóa thì chia thành (i) mật mã khóa đối xứng và (ii) khóa công khai. Mật mã khóa đối xứng sử dụng cùng một khóa bí mật để mã hóa và giải mã. Với quá trình xử lý dữ liệu tương đối nhẹ, nó được sử dụng trong mã hóa và xác thực dữ liệu. Mặt khác, mật mã khóa công khai sử dụng khóa bí mật trong giải mã và khóa công khai khác với khóa bí mật trong mã hóa, và khá khó để đoán khóa bí mật từ khóa công khai. Độ phức tạp tính toán của mật mã khóa công khai thường cao gấp hơn nhiều lần so với mật mã khóa đối xứng, nhưng công nghệ này được sử dụng để chia sẻ khóa bí mật được sử dụng trong mật mã khóa đối xứng và chữ ký số, nhờ đặc tính bất đối xứng.

Mật mã nhẹ đã được bắt đầu vào khoảng năm 2004 với một dự án ở Châu Âu. Tiêu chuẩn quốc tế ISO/IEC 29192 "Mật mã nhẹ" được thành lập tại ISO/IEC JTC 1/SC 27. Viện Tiêu chuẩn và Công nghệ Quốc gia Hoa Kỳ NIST ban hành các hướng dẫn về công nghệ mật mã đã bắt đầu dự án mật mã nhẹ vào năm 2013 và công bố công khai cho các ứng dụng của tiền mã hóa nhẹ vào năm 2017.

Tại Nhật Bản, Ủy ban nghiên cứu và đánh giá mật mã CRYPTREC có nhiệm vụ đánh giá các hệ mật mã do chính phủ khuyến nghị và theo dõi các xu hướng của công nghệ mật mã. Tại đây mật mã nhẹ đã phát triển các hoạt động của mình từ năm 2013. Chúng bao gồm việc đánh giá việc triển khai các mật mã khối nhẹ, mã dòng nhẹ, hệ mật đường cong elliptic cũng như các cuộc khảo sát bảo mật và nghiên cứu việc sử dụng hiệu quả các mật mã nhẹ.

Định nghĩa 1.4. Mật mã nhẹ

Mật mã nhẹ là một thuật toán mật mã hoặc giao thức mật mã để triển khai trong các môi trường hạn chế bao gồm thẻ RFID, thiết bị cảm biến, đồng hồ thông minh, thiết bị chăm sóc sức khỏe,…

1.4.1. Mật mã khối nhẹ

Trong kỷ nguyên kỹ thuật số, thông tin được mã hóa một cách có hệ thống thành các chuỗi nhị phân, trong đó mỗi đơn vị nhị phân (bit) giữ một giá trị bằng 0 hoặc 1. Kích thước hoặc chiều dài là tổng số bit trong chuỗi nhị phân. Nói chung, người ta có thể phân biệt hai nhóm lớn của thuật toán mã hóa đó là (i) mật mã khối và (ii) mật mã dòng. Mật mã khối hoạt động trên khối thông điệp có độ dài cố định và khóa bí mật.

SIMON và SPECK là hai họ mật mã khối nhẹ, được thiết kế bởi các nhà nghiên cứu của NSA vào năm 2013. Mỗi SIMON và SPECK chứa mười phiên bản với các kích thước khối và kích thước khóa khác nhau. SIMON và SPECK cung cấp hiệu suất tốt trên cả nền tảng phần cứng và phần mềm, chẳng hạn như ASIC, FPGA và vi điều khiển 4/8/16/32-bit, và được thiết kế để hoạt động tốt trên toàn bộ các ứng

dụng nhẹ. SIMON được tối ưu hóa cho việc triển khai phần cứng và SPECK được điều chỉnh để có hiệu suất tối ưu trong phần mềm. Cấu trúc của SIMON và SPECK dựa trên cấu trúc Feistel, nhưng với PRESENT dựa trên cấu trúc SPN.

Định nghĩa 1.5. Mật mã khối nhẹ

Mật mã khối nhẹ là một nhóm mật mã nhẹ, mà thuật toán mã hóa là mật mã khối.

Người ta đánh giá thuật toán mã hóa khối nhẹ qua các tiêu chí: độ trễ xử lý, số lượng cổng tương đương, năng lượng tiêu thụ, độ an toàn.

1.4.2. Mật mã dòng nhẹ

Mật mã dòng nhẹ là nhánh của mật mã nhẹ sử dụng khoá bí mật, đối xứng. Ý tưởng của thuật toán mã hoá theo dòng là bản nguồn sẽ được phân rã thành các bit; từng bit này sẽ được mã hoá bằng mỗi bit khoá để cho kết quả là một bit mã. Việc mã hoá bit được thực hiện bằng hàm logic XOR.

Định nghĩa 1.6. Mật mã dòng nhẹ

Mật mã dòng nhẹ là cơ chế mã hóa sử dụng dòng khóa để mã hóa bản rõ theo cách từng bit hoặc từng khối bit trên các thiết bị có năng lực tính toán hạn chế.

1.4.3. Mật mã đường cong elliptic

Nghiên cứu về các đường cong elliptic của các nhà đại số, các nhà lý thuyết số có từ giữa thế kỷ XIX. Mật mã đường cong elliptic được phát hiện vào năm 1985 bởi Neil Koblitz và Victor Miller. Chúng có thể được xem như các đường cong elliptic của các hệ mật mã logarit rời rạc. Trong đó nhóm $\boldsymbol{Z_P^*}$ được thay thế bằng nhóm các điểm trên một đường cong elliptic trên một trường hữu hạn. Cơ sở toán học cho tính bảo mật của các hệ thống mật mã đường cong elliptic là tính hấp dẫn tính toán của bài toán logarit rời rạc đường cong elliptic (ECDLP).

Mật mã đường cong elliptic (ECC) là một trong những loại mật mã được ứng dụng rộng rãi hiện nay. Tại nhiều công ty tại Mỹ, như CloudFlare, đã sử dụng rộng rãi ECC để bảo mật mọi thứ từ các kết nối HTTPS của khách hàng đến cách thức chuyển dữ liệu giữa các trung tâm dữ liệu của họ. ECC được ứng dụng trong thương mại điện tử với tài nguyên hạn chế, trong công nghệ nhận dạng đối tượng bằng sóng vô tuyến hiệu quả và an toàn, trong các mạng cảm biến không dây sử dụng phép biến đổi lý thuyết số.

Định nghĩa 1.7. Mật mã đường cong elliptic

Mật mã đường cong elliptic là sử dụng các tính chất toán học của đường cong elliptic để tạo ra các hệ thống mật mã khóa đối xứng, khóa công khai để mã hóa dữ liệu.

1.5. Các tham số đánh giá hệ mật mã nhẹ

Để đánh giá một hệ mật mã nhẹ người ta cần xét đến các tham số của chúng. Tham số này được đánh giá trên góc độ phần cứng và phần mềm. Các tham số bao gồm: Bảo mật; Diện tích chip; Thông lượng; Độ trễ; Nguồn và năng lượng; Tính hiệu suất cài đặt; Kích thước mã; Mức sử dụng bộ nhớ.

CHƯƠNG 2. HỆ MẬT MÃ KHỐI NHẸ, HỆ MẬT MÃ DÒNG NHẸ

2.1. Hệ mật mã khối nhẹ

2.1.1. Các phương pháp tấn công

Định nghĩa 2.1. Tấn công mật mã khối nhẹ

Tấn công mật mã khối nhẹ là cách tấn công vào hệ thống mã khối nhẹ.

Mệnh đề 2.1. Các loại tấn công mật mã nhẹ

Để tấn công một hệ mật mã khối trong mật mã nhẹ có các phương pháp tấn công (i) tấn công vét cạn; (ii) tấn công đại số; (iii) tấn công khôi phục khóa; (iv) tấn công thống kê bão hòa.

2.1.2. Mật mã khối PRESENT

Mật mã PRESENT được giới thiệu vào năm 2007 với mục đích hoạt động trên các môi trường hạn chế. Khi được triển khai trên một thiết bị, PRESENT có thể bao gồm cả mã hóa và giải mã, hoặc nó có thể chỉ là mã hóa còn với việc giải mã được xử lý ở một nơi khác. Vì thiết kế thuật toán PRESENT dựa trên mô hình mạng SPN, nên thuật toán PRESENT có 3 phép biến đổi chính là (i) phép toán trộn khóa; (ii) phép toán thay thế; (iii) phép toán hoán vị.

2.1.3. Một số hệ mật mã khối nhẹ khác

Họ mã khối *NOEKEON* được đề xuất vào năm 2000 bởi bốn nhà khoa học Daemen, Peeters, Van Assche và Rijme. Một trong các ưu điểm của NOEKEON đó là nhỏ gọn, nhanh chóng triển khai trên phần cứng chuyên dụng. Ngoài ra, nó còn yêu cầu về bộ nhớ RAM rất thấp và hoạt động hiệu quả trên đa nền tảng. Nhưng trong thuật toán họ NOEKEON này sử dụng rất nhiều các ánh xạ tuyến tính và sử dụng khóa trực tiếp để mã hóa mà không thông qua những phép biến đổi đối với khóa. Do đó, NOEKEON dường như khá mong manh trước những dạng thám mã tuyến tính.

Họ mã khối *LED* được thiết kế và đề xuất bởi Guo, Peyrin, Poschmann và Robshaw vào năm 2011, thời điểm này đây là một trong những thuật toán mã hóa nhẹ mới. Kích thước khối tin 64 bit với hai biến thể của khóa lần lượt là 64 bit và 128 bit. Thiết kế của LED có nhiều điểm tương đồng với thuật toán AES.

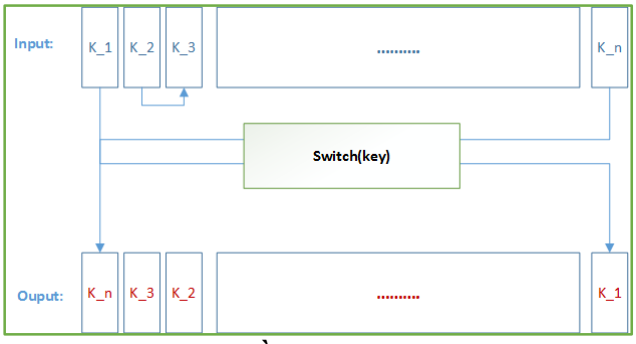
2.1.4. Đề xuất cải tiến NOEKEON và LED

NOEKEON cải tiến

Đối với NOKEON cải tiến thì sử dụng thêm 1 hàm Switch (key) để làm việc với khóa 128 bit trước khi sử dụng hàm Theta, nhằm mục đích xáo trộn dữ liệu khóa K, tăng độ mật cho thuật toán. Đảm bảo cho dù thám mã ra giá trị khóa K cũng sẽ không thám mã được nội dung bản rõ một cách dễ dàng. Chi tiết hàm Switch (key) như sau:

- Input: Khóa K có độ dài 128 bit.
- Xử lý: Đổi chỗ vị trí bit thứ 1 và vị trí thứ n của khóa. Từ vị trí thứ 2, thực hiện đổi chỗ của cặp bit thứ (i; i+1).

- Output: Khóa K’ có độ dài 128 bit.



Hình 2.1. Sơ đồ thuật toán Switch (key)

Như vậy, Hình 2.1 đối với hàm *Switch (key)* có thể sử dụng trong cả mã hóa và giải mã mà không cần thay đổi. Giá trị output K’ sẽ được sử dụng tại NOEKEON cải tiến:

```
Round (Key, STATE, Constant1, Constant2)
{
 STATE[0] ^= Constant1;
 Theta (Switch (Key), STATE);
 STATE[0] ^= Constant2;
 Pi1 (STATE);
 Gamma (STATE);
 Pi2 (STATE);
}
```

LED cải tiến

Hộp S và vị trí của từng thành phần hộp S:

Bảng 2.1. Vị trí thành phần hộp S

Pos		S[x]	x	
0	12	C	0	
1		5	1	
2		6	2	
3	11	B	3	
4		9	4	
5		0	5	
6	10	A	6	
7	13	D	7	
8		3	8	
9	14	E	9	
10	15	F	A	10
11		8	B	11
12		4	C	12
13		7	D	13
14		1	E	14
15		2	F	15

LED là mật mã khối nhẹ có độ an toàn cao, do đó ta có thể tiến hành cải tiến LED để áp dụng cho các thiết bị IoT có giới hạn lưu trữ và xử lý ở tầm trung cho đến tầm cao. Và những thiết bị đó yêu cầu một thuật toán mã hóa có độ bảo mật tốt. Để cải tiến LED ta nên đi theo hướng bù đắp các yếu điểm của thuật toán mã hóa khối. Có 3 hướng chính là (i) tăng kích thước khối hoặc khóa; (ii) tăng độ hỗn loạn; (iii) tăng tính khuếch tán.

Hiện tại độ dài khối và khóa của LED đang ở mức rất tốt là kích thước khối 64 bit và kích thước khóa 128 bit do đó ta không nên phá vỡ cấu trúc này. Lựa chọn phương pháp cải tiến là tăng độ hỗn loạn

và khuếch tán để kẻ tấn công khó khăn trong việc dịch ngược đoạn mã. Vị trí cải tiến là hộp S và phương pháp cải tiến là áp dụng công thức của mã Caesar trong việc thay thế từng thành phần trong hộp S. Áp dụng vào hộp S ta có công thức mã hóa: *Pos' = (Pos + k) mod 16*: Trong đó Pos là vị trí hiện tại của từng thành phần trong hộp S được đánh số như Bảng 2.2. Pos' là vị trí thay thế. Sau khi có Pos' ta chuyển đổi trả lại về mã Hecxa, k là số đơn vị dịch chuyển để thay thế. Tương tự như vậy, ta có công thức giải mã: *Pos' = (Pos - k) mod 16*:

Bảng 2.2. Sự biến đổi hộp S và vị trí Pos'

	S[x]'	Pos'	Pos
F	15	10	0
	8	11	1
	4	12	2
	7	13	3
	1	14	4
	2	15	5
C	12	0	6
	5	1	7
	6	2	8
B	11	3	9
	9	4	10
	0	5	11
A	10	6	12
D	13	7	13
	3	8	14
E	14	9	15

Mặt khác LED sử dụng hộp S của PRESENT là cố định nên ta cũng nên cố định k khi thiết kế. Mục đích tăng sự phức tạp khi có kẻ tấn công nhưng đồng thời không làm phức tạp quá trình giải mã. Luận án chọn k bằng 10 theo tỉ lệ phù hợp 16/10 của toán học. Như vậy thu được Pos, Pos' và hộp S' sau khi thực hiện quá trình gây nhiễu theo công thức Ceasar.

Với việc sử dụng thay thế Ceasar không làm tăng độ phức tạp của thuật toán nhưng lại gây ra sự nhiễu loạn không nhỏ trong giải mã khiến cho kẻ tấn công khó khăn với ý định giải mã để xem lén thông tin.

So sánh hệ mật mã PRESENT với các hệ mật mã khác

Hệ mật mã PRESENT với kích thước khối 64 bit và kích thước khóa là 80 bit thì độ bảo mật là 0.84. Có thể nói rằng với cùng kích thước khối, kích thước khóa nhỏ hơn so với các hệ mật mã khối nhẹ khác nhưng độ bảo mật của PRESENT là tốt hơn.

Mệnh đề 2.2. So sánh PRESENT với một số hệ mật mã khối khác

Thuật toán PRESENT với kích thước khối và kích thước khóa tương đương cho mức bảo mật tốt hơn.

Bảng 2.3. Bảng so sánh hệ mật mã PRESENT với các hệ mật mã khác

Hệ mật mã	Kích thước khối (bit)	Kích thước khóa (bit)	Mức độ bảo mật
PRESENT	64	80	0.84
HIGHT	64	128	0.69
Piccolo	64	80	0.56

Hệ mật mã	Kích thước khối (bit)	Kích thước khóa (bit)	Mức độ bảo mật
PRINCE	64	128	0.83
RC5	64	128	0.90
TWINE	64	80	0.64

Bảng 2.4. Bảng so sánh hệ mật mã PRESENT với hệ mật mã LED, NOEKEON

Hệ mật mã	Kích thước khối (bit)	Kích thước khóa (bit)	Mức độ bảo mật
PRESENT	64	80	0.84
LED	64	128	0.86
NOEKEON	128	128	0.65

2.1.4. Đề xuất thiết kế thuật toán mật mã khối nhẹ mới

Một câu hỏi mà tất cả các nhà thiết kế cần giải quyết trong khi thiết kế bất kỳ mã pháp nào là *"độ an toàn bao nhiêu thì được coi là đủ an toàn"*. Do đó, nếu một cơ chế an toàn được triển khai không được sử dụng đầy đủ khả năng của nó sẽ dẫn tới việc lãng phí tài nguyên. Một ví dụ, ta đều biết rằng AES đã được phân tích rộng rãi đối với độ an toàn của nó. Cho đến nay, nó đã được chứng minh kháng lại rất nhiều tấn công. Do đó, thật lý tưởng khi các nhà cung cấp phát triển được thuật toán AES trong các thiết bị của họ. Tuy nhiên, một vấn đề gặp phải đối với AES là nó rất cồng kềnh và cần rất nhiều tài nguyên cho việc cài đặt. Ngoài ra, nó cung cấp độ an toàn nhiều hơn những gì cần thiết cho việc sử dụng.

Vì vậy, ta cần thấy rằng để thiết kế một hệ mật mã phù hợp với các hạn chế về tài nguyên của các thiết bị nhỏ và cùng lúc các hệ mật mã này cũng cung cấp độ an toàn đầy đủ cho việc sử dụng. Đây cũng chính là một trong những nguyên nhân chính thúc đẩy mật mã nhẹ. Bây giờ, ta xem xét khía cạnh kỹ thuật của thiết kế mật mã khối nhẹ, sau khi quyết định chọn lựa các tham số đầu vào phù hợp việc tiếp theo mà người thiết kế quan tâm chính là hàm vòng. Đặc biết đối với mật mã khối nhẹ, hàm vòng phải thật đơn giản đối với việc cài đặt phần cứng. Một hàm vòng chứa (i) một hàm phi tuyến và (ii) một hàm tuyến tính. Hàm phi tuyến được gọi là tầng xáo trộn còn hàm tuyến tính được gọi là tầng khuếch tán. Do vậy, ta dựa vào hai khía cạnh quan trọng là (i) xáo trộn và (ii) khuếch tán trong việc xây dựng hàm vòng. Mục đích của hai hàm này được phát biểu cụ thể như sau:

1. *Xáo trộn (confusion):* Sự phụ thuộc của bản mã đối với bản rõ phải thực phức tạp để gây rắc rối, cảm giác hỗn loạn đối với kẻ thù có ý định phân tích tìm qui luật để phá mã. Quan hệ hàm số của mã tin là phi tuyến (non-linear).
2. *Khuếch tán (diffusion*)**:** Làm khuếch tán những mẫu văn bản mang đặc tính thống kê (gây ra do dư thừa của ngôn ngữ) lẫn vào toàn bộ văn bản. Nhờ đó tạo ra khó khăn cho kẻ thám mã trong việc dò phá mã trên cơ sở thống kê các mẫu lặp lại cao. Sự thay đổi của một bit trong một khối bản rõ phải dẫn tới sự thay đổi hoàn toàn trong khối mã tạo ra.

Ngoài ra, hai khía cạnh xáo trộn và khuếch tán thì với ý tưởng thuật toán mật mã khối mới cần

chia khối bit bản rõ cho phù hợp, lựa chọn khóa cho phù hợp, chọn cấu trúc SPN hoặc Feistel cho phù hợp, lựa chọn hộp S cho phù hợp.

2.2. Hê mật mã dòng nhẹ

2.2.1. Mật mã dòng GRAIN

GRAIN là hệ mật mã dòng được công bố trên eSTREAM bởi Martin Hell, Thomas Johansson và Willi Meier năm 2004 với phiên bản đầu tiên GRAIN-V0 [81] được tổ chức EU ECRYPT thông qua. Sau đó hệ mật mã này tiếp tục được phát triển thành GRAIN-v1 là một trong bảy dự án được eSTREAM đưa vào các danh mục đầu tư từ 09/09/2008. Cùng với GRAIN-V1 là một phiên bản mật mã với khóa bí mật 128 bit GRAIN-128 cũng được áp dụng rộng rãi hiện nay. Luận án chủ yếu tập trung phân tích hệ mã nguyên thủy GRAIN-V0, GRAIN-V1, GRAIN-128 và một phiên bản xác thực GRAIN-128a cùng những cuộc tấn công đã được thực hiện trên các hệ mã này.

Một số cải tiến hệ mật mã GRAIN

Người ta cải tiến theo cách tăng tốc thông lượng nhờ công nghệ lượng tử. Việc sử dụng công nghệ ôtomat di động lượng tử cho việc thiết kế các mạch logic đã cho thấy tăng tốc độ truyền dữ liệu lên đến 2 THz. Trong công nghệ này các mạch được thiết kế để có một kích thước đặc biệt siêu nhỏ cũng như tiêu thụ điện năng cực thấp. GRAIN-128 là một trong những mật mã dòng tốt nhất trong danh sách cuối cùng của dự án eSTREAM.

Đánh giá các phiên bản GRAIN

Luận án đưa ra những so sánh các thông số thiết kế khác nhau cho các thành viên của họ GRAIN. Trong Bảng 2.5 người ta có thể thấy sự khác nhau về độ dài khóa, kích thước IV và padding sử dụng trong các thành viên của GRAIN.

Bảng 2.5. Độ dài khóa và IV của họ GRAIN

Hệ mật mã	Chiều dài khóa	Kích thước IV	Padding với IV
GRAIN-V0	80	64	FFFF
GRAIN-V1	80	64	FFFF
GRAIN-128	128	96	FFFFFFFF
GRAIN-128a	128	96	FFFFFFFE

Chỉ có phiên bản cuối cùng của họ GRAIN là GRAIN-128a với phần đệm được hoàn thiện bởi việc thay bit bên phải của LFSR bởi giá trị 0 để tránh cuộc tấn công đồng bộ hóa, do Kucuk đề xuất. Trong tất cả các phiên bản khác, phần đệm được thực hiện với tất cả các bit.

Trong Bảng 2.6 so sánh các thành viên của họ GRAIN trên cơ sở thời gian thiết lập khóa, thời gian thiết lập IV và tốc độ mã hóa. Các tốc độ mã hóa này được đo bằng các bộ xử lý Pentium 4, 2.80 Ghz cho hai loại dữ liệu, một cho các luồng dài và một cho các luồng dữ liệu ngắn dưới 40 bytes.

Bảng 2.6. Hiệu suất của họ GRAIN

Hệ mật mã	Thời gian thiết lập khóa	Thời gian thiết lập IV	Tốc độ mã hóa	
			Luồng dài	Luồng bytes
GRAIN-V0	29.27	73408.44	3729.79	5545.83
GRAIN-V1	31.14	1498.23	57.31	102.95
GRAIN-128	38.89	1098.61	31.16	70.30

Có thể thấy GRAIN-128 có sự vượt trội về tốc độ tính toán, và hiệu quả phần cứng cao nhất trong họ GRAIN.

2.2.2. Một số mật mã dòng nhẹ khác

Mật mã dòng MICKEY

Mật mã dòng nhẹ MICKEY sử dụng khóa 80 bit cố định và từ 0 đến 80 bit IV. Bộ tạo dòng khóa của MICKEY bao gồm hai thanh ghi dịch R và S 100 bit, trong đó mỗi thanh ghi được được điều khiển phản hồi tương ứng. Sau khi khóa và IV được tạo, hệ thống sẽ được quay với chu kỳ 100 vòng trước khi tạo ra dòng khóa có thể sử dụng được. Mỗi cặp khóa và cặp IV có thể tạo ra tối đa 2^{40} bit của dòng khóa duy nhất.

Mật mã dòng TRIVIUM

Mật mã dòng TRIVIUM sử dụng khóa 80 bit và 80 bit IV. Bộ tạo dòng khóa chứa một thanh ghi dài 288 bit, trong đó các thanh ghi cụ thể được đọc và đưa trở lại hệ thống khi nó được tạo xung nhịp theo chu kỳ. Khóa được tải trong 93 bit đầu tiên và IV được tải trong 84 bit tiếp theo. Hệ thống có thể tạo ra tối đa 2^{64} bit dòng khóa duy nhất.

So sánh GRAIN với một số hệ mật mã dòng, khối nhẹ khác

Khi thiết kế một hệ mật mã, người thiết kế cần phải tập trung vào một số điểm đặc biệt của giải thuật. Không thể thực hiện một thiết kế hoàn hảo, đáp ứng được tất cả các mong muốn của một hệ mật mã như đáp ứng được tất cả các độ dài bản rõ, tất cả các ứng dụng, tất cả các hạn chế về bộ nhớ… GRAIN được thiết kế cho một phần cứng rất nhỏ, sử dụng ít cổng nhất có thể trong khi vẫn đáp ứng được an ninh cao. Hệ mật mã này được sử dụng trong môi trường có cổng tương đương, điện năng tiêu thụ, bộ nhớ cần thiết là rất nhỏ. Ngoài ra GRAIN vẫn có thể được sử dụng trong phần mềm nói chung nếu thực hiện tăng tốc độ cho nó. Nhưng việc so sánh hiệu suất của GRAIN trong các ứng dụng phần mềm là không có ý nghĩa. Luận án chỉ so sánh hiệu suất của GRAIN với một số giải thuật khác trong việc ứng dụng vào phần cứng.

Hệ mật mã dòng GRAIN cho phép thực hiện song song 16 mã hóa khác nhau, cho phép triển khai nhanh hơn, với chi phí sử dụng ít hơn nhưng đem lại hiệu quả cao hơn. Tính hiệu quả của phần cứng là tỷ lệ thông lượng với điện tích sử dụng trong thuật toán. Nhìn vào Bảng 2.7 thống kê dưới, ta có thể thấy thuật toán GRAIN có tính hiệu quả phần cứng cao hơn TRIVIUM (77.28 > 38.48).

Bảng 2.7. So sánh mật mã nhẹ

Hệ mật mã	Số bit khóa	Số bit khối	Chu kỳ xung nhịp trên một khối	Thông lượng ở 100 MHz (Kbps)	Xử lý logic (µm)	Diện tích (GE)
Mật mã khối						
PRESENT	80	64	32	200	0.18	1570
HIGHT	128	64	34	188	0.25	3048
mCrypton	96	64	13	492	0.13	2681
Mật mã dòng						
TRIVIUM	80	1	1	100	0.13	2599
GRAIN	80	1	1	100	0.13	1294
MICKEY	80	1	1	100	0.13	2183

CHƯƠNG 3. ĐỀ XUẤT CÁC HỆ MẬT MÃ TRÊN ĐƯỜNG CONG ELLIPTIC VÀ ỨNG DỤNG

3.1. Thuật toán sinh chuỗi

Mục đích của thuật toán này là yêu cầu người ta tạo ra một chuỗi các vectơ được gọi là S_i. Các bước của thuật toán được trình bày ở thuật toán 3.1. Bước dịch vòng một phần tử của hàng sang bên phải sẽ làm xáo trộn các bit, điều này làm cho kẻ thám mã khó phát hiện, dò tìm ra bản rõ. Sau đó, kết quả của quá trình này được áp dụng cho đầu ra của ECC.

Thuật toán 3.1. Sinh chuỗi

Input: Tham số đường cong elliptic
Output: Chuỗi các bit

Bước 1:

- Tính tổng số điểm (n) trên đường cong elliptic
- Xác định điểm P là điểm sinh của phương trình đã cho
- Đưa ra tập các điểm trên đường cong elliptic từ điểm sinh P

Bước 2:

- Chuyển đổi tổng số điểm (n) trong cơ số 3
- Lấy m sẽ là số chữ số chuyển tổng số điểm sang cơ số 3

Bước 3:

- Lập ma trận M có kích thước (n + 1) * m. Ở đây (n + 1) là số hàng, m là số cột cũng chính là số chữ số trong một hàng.

$$M = \begin{pmatrix} a_{0,0}a_{0,1} \dots \dots a_{0,m} \\ a_{1,0}a_{1,1} \dots \dots a_{1,m} \\ a_{2,0}a_{2,1} \dots \dots a_{2,m} \\ \dots \dots \\ a_{n,0}a_{n,1} \dots \dots a_{n,m} \end{pmatrix}$$

Bước 4:

- Dịch chuyển theo vòng hàng của ma trận ở bước 3 theo một phần tử sang bên phải: $[a_{i,0}\ a_{i,1}\ a_{i,2} \dots a_{i,m-1}] \rightarrow [a_{i,m-1}\ a_{i,0}\ a_{i,1}\ a_{i,2} \dots a_{i,m-2}]$

Bước 5:

- Chuỗi được hình thành là: S: $[S_0 = [a_{0,m-1}\ a_{0,0}\ a_{0,1}\ a_{0,2} \dots a_{0,m-2}]$, $S_1 = [a_{1,m-1}\ a_{1,0}\ a_{1,1}\ a_{1,2} \dots a_{2,m-2}]$, …, $S_n = [a_{n,m-1}\ a_{n,0}\ a_{n,1}\ a_{n,2} \dots a_{n,m-2}]]$

3.2. Thuật toán đề xuất CECC

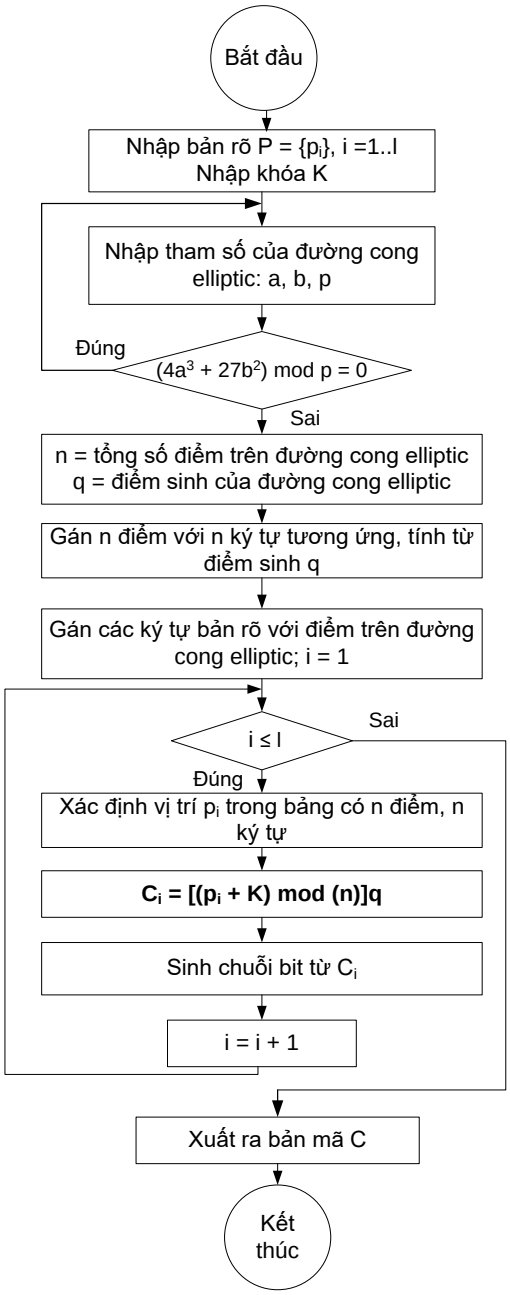
3.2.1. Ý tưởng thuật toán CECC

Thuật toán CECC dựa vào phép toán trên đường cong elliptic kết hợp với hệ mật mã dịch chuyển Caesar cải tiến. Thuật toán CECC đề xuất này còn dựa vào thuật toán tạo chuỗi dữ liệu làm cơ sở xây dựng thuật toán mã hóa bằng cách sử dụng đường cong elliptic trên trường hữu hạn có khóa đối xứng để mã hóa văn bản tiếng Việt này.

3.2.2. Trình bày thuật toán CECC

Lưu đồ thuật toán mã hóa CECC

Lưu đồ thuật toán mã hóa của hệ mật mã CECC trên đường cong elliptic trình bày như Hình 3.1 dưới đây.



Hình 3.1. Lưu đồ thuật toán mã hóa CECC

Thuật toán mã hóa CECC

Các bước của thuật toán mã hóa CECC được trình bày ở thuật toán 3.2 như dưới đây:

Thuật toán 3.2. Mã hóa CECC

```
BEGIN
Input: P = {pi} i= 1..l; Khóa K;
Do
      Begin
            Input (a, b, p);
      End;
While ((4a3 + 27b2) mod p = 0)
```

```
      n = Tổng số điểm trên đường cong elliptic;
      q = Điểm sinh trên đường cong elliptic;
      Gán n điểm với n ký tự tương ứng, tính từ điểm sinh q;
      Gán các ký tự bản rõ với điểm trên đường cong elliptic;
      i = 1;
      While (i<=l) do
            Begin
                  Xác định vị trí pi trong bảng có n điểm, n ký tự;
                  Ci = [(pi + K) mod (n)]q;
                  Sinh chuỗi bit từ Ci
                  i = i + 1;
            End;
Output: Xuất ra bản mã C;
END.
```

Thuật toán giải mã CECC

Thuật toán 3.3. Giải mã CECC

```
BEGIN
Input: C = {Ci} i= 1..l; Khóa K;
       a, b, p tham số đường cong elliptic;
Xác định q = điểm sinh của đường cong elliptic;
Tính n = tổng số điểm trên đường cong elliptic;
while (i<=l) do
      Begin
            Xét đoạn gồm m chữ số của bản mã;
            Sử dụng phép dịch trái một phần tử;
            Chuyển đổi sang thập phân;
            Hiển thị mã điểm trên đường cong elliptic;
            Xác định vị trí Ci trong bảng của điểm trên đường cong
            elliptic;
            Pi = [(Ci – K) mod (n)]P;
            Hiển thị rõ điểm trên đường cong elliptic;
            Hiển thị ký tự bản rõ ứng với điểm trên đường cong
            elliptic;
            i = i + 1;
      End;
Output: Xuất ra bản rõ P;
END.
```

3.3. Thuật toán đề xuất AECC

3.3.1. Ý tưởng thuật toán AECC

Thuật toán AECC dựa vào phép toán trên đường cong elliptic kết hợp với hệ mật mã Affine cải tiến. Thuật toán này dựa vào thuật toán tạo chuỗi dữ liệu, rồi sử dụng ý tưởng mật mã Affine làm cơ sở để xây dựng thuật toán mã hóa bằng cách sử dụng đường cong elliptic trên trường hữu hạn có khóa đối xứng.

3.3.2. Trình bày thuật toán AECC

Thuật toán mã hóa AECC

Thuật toán 3.4. Mã hóa AECC

```
BEGIN
Input: P = {pi} i= 1..l; Khóa K;
Do
      Begin
            Input (a, b, p);
      End;
While ((4a^3 + 27b^2) mod p = 0)
      n = Tổng số điểm trên đường cong elliptic;
      q = Điểm sinh trên đường cong elliptic;
      Gán n điểm với n ký tự tương ứng, tính từ điểm sinh q;
      Gán các ký tự bản rõ với điểm trên đường cong elliptic;
      i = 1;
      While (i<=l) do
            Begin
                  Xác định vị trí pi trong bảng có n điểm, n ký tự;
                  Ci = [(u*pi + v) mod (n)]q;
                  Sinh chuỗi bit từ Ci
                  i = i + 1;
            End;
Output: Xuất ra bản mã C;
END.
```

Thuật toán giải mã AECC

Thuật toán 3.5. Giải mã AECC

```
BEGIN
Input: C = {Ci} i= 1..l; Khóa K;
       a, b, p tham số đường cong elliptic;
Xác định q = điểm sinh của đường cong elliptic;
Tính n = tổng số điểm trên đường cong elliptic;
      while (i<=l) do
            Begin
                  Xét đoạn gồm m chữ số của bản mã;
                  Sử dụng phép dịch trái một phần tử;
                  Chuyển đổi sang thập phân;
                  Hiển thị mã điểm trên đường cong elliptic;
                  Xác định vị trí Ci trong bảng của điểm trên đường cong
                  elliptic;
                  Pi = [u^-1(Ci – v) mod (n)]q;
                  Hiển thị rõ điểm trên đường cong elliptic;
                  Hiển thị ký tự bản rõ ứng với điểm trên đường cong
                  elliptic;
                  i = i + 1;
            End;
Output: Xuất ra bản rõ P;
```

```
END.
```

3.4. Thuật toán VECC

3.4.1. Ý tưởng thuật toán VECC

Thuật toán VECC dựa trên ý tưởng cơ bản của phép toán trên đường cong elliptic. Thuật toán VECC đề xuất này không dựa vào thuật toán tạo chuỗi dữ liệu, quá trình tạo chuỗi dữ liệu này thì độ mật sẽ tốt hơn. Tuy nhiên, mặt hạn chế là sẽ (i) tốn nhiều thời gian và (ii) dung lượng bộ nhớ. Luận án sử dụng sử dụng ý tưởng mật mã Vigenere làm cơ sở xây dựng thuật toán mã hóa bằng cách sử dụng đường cong elliptic trên trường hữu hạn thông qua vị trí điểm của đường cong elliptic để mã hóa dữ liệu.

3.4.2. Trình bày thuật toán VECC

Thuật toán mã hóa VECC

Tiếp theo, trình bày nội dung thuật toán mã hóa hệ mật mã VECC trên đường cong elliptic với thuật toán 3.6 như dưới đây:

Thuật toán 3.6. Mã hóa VECC

```
BEGIN
     Input: P = {pi} i= 1..l; Khóa K = {kj} j= 1..d;
          Do
               Begin
                    Input (a, b, p);
               End;
          While ((4a^3 + 27b^2) mod p = 0)
          n = Tổng số điểm trên đường cong elliptic;
          q = Điểm sinh trên đường cong elliptic;
          Gán n điểm với n ký tự tương ứng, tính từ điểm sinh q;
          Gán các ký tự bản rõ với điểm trên đường cong elliptic;
          i = 1;
          j = 1;
          While (i<=l) and (j<=d) do
             Begin
               Xác định vị trí pi,kj trong bảng có n điểm, n ký tự;
               Ci = [(pi + kj) mod (n)]q;
               Tìm được mã điểm trên đường cong elliptic;
               Tìm được mã ký tự theo mã điểm;
               i = i + 1;
               if (j >= d)
                    Begin
                         j = 1;
                    End;
               else
                    Begin
                         j = j + 1;
             End;
```

```
       Output: Xuất ra bản mã C;
END.
```

Thuật toán giải mã VECC

Thuật toán 3.7. Giải mã VECC

```
BEGIN
Input: C = {Ci} i= 1..l; Khóa K = {kj} j= 1..d;
       a, b, p tham số đường cong elliptic;
Tính n = tổng số điểm trên đường cong elliptic;
Xác định q = điểm sinh của đường cong elliptic;
while (i<=l)and (j<=d) do
     Begin
         Xác định vị trí Ci, kj trong bảng của điểm trên đường cong
         elliptic;
         Pi = [(Ci – kj) mod (n)]q;
         Hiển thị rõ điểm trên đường cong elliptic;
         Hiển thị ký tự bản rõ ứng với điểm trên đường cong elliptic;
         i = i + 1;
         if (j >= d)
                Begin
                     j = 1;
                End;
          else
                Begin
                     j = j + 1;
                End;
     End;
 Output: Xuất ra bản rõ P;
END.
```

3.5. Đánh giá ba thuật toán CECC, AECC, VECC

Độ an toàn và bảo mật của thuật toán phụ thuộc vào tham số của đường cong, phụ thuộc vào điểm trên đường cong elliptic, phụ thuộc độ dài của khóa K. Độ phức tạp thuật toán mật mã đường cong elliptic CECC, AECC, VECC có độ phức tạp thuật toán là: O(n*logn).

CHƯƠNG 4. THỰC NGHIỆM MẬT MÃ NHẸ TRONG HỆ THỐNG CÁC THIẾT BỊ THÔNG MINH

4.1. Thực hiện hệ mật mã khối nhẹ

4.1.1. Tổng hợp mật mã khối nhẹ

Bảng 4.1. Tổng hợp một số hệ mật mã khối nhẹ

Hệ mật mã	Độ dài khóa - bit	Kích thước khối tin - bit	Số vòng mã hóa	Cấu trúc
AES	128, 192, 256	128	10, 12, 14	SPN
NOEKEON	128	128	16	SPN
PRESENT	80, 128	64	31	SPN
MINI-AES	64	64	10	SPN
PRINTcipher	80, 160	48, 96	48, 96	SPN
KLEIN	64, 80, 96	64	12, 16, 20	SPN
LED	64, 80, 96, 128	64	32, 48	SPN
LBLOCK	80	64	32	Feistel
PRINCE	128	64	12	SPN
SIMON	64, 72, 96, 128 144, 192, 256	32, 48, 64, 96, 128	32, 36, 42, 44, 52, 54, 68, 69, 72	Feistel
RECTANGLE	80, 128	64	25	SPN

4.1.2. Đánh giá các thuật toán

Luận án thực hiện đánh giá thuật toán mã hóa qua các tiêu chí: độ trễ xử lý, số lượng cổng tương đương, năng lượng tiêu thụ.

4.2. Thực nghiệm hệ mật mã dòng nhẹ trên phần cứng

Để đo hiệu suất của mật mã dòng nhẹ trên phần cứng eStream đã đưa ra năm tiêu chí để đánh giá: (i) tính nhỏ gọn về diện tích; (ii) hiệu suất về thông lượng; (iii) mức tiêu thu năng lượng/nguồn; (iv) tính linh hoạt; (v) tính đơn giản. Trong đó ba tiêu chí đầu thì tương đối dễ định lượng. Tuy nhiên, tiêu chí còn lại mang tính chủ quan nhiều hơn và không xét ở đây. Kết quả được tổng hợp như Bảng 4.2.

Bảng 4.2. Kết quả trên phần cứng

Mật mã dòng hạng nhẹ	Khóa (bit)	Diện tích (GE)	Nguồn (µW)	Năng lượng (pJ/bit)	Thông lượng (Mbps)
GRAIN-0	80	1294	2.22	10.73	725
GRAIN-V1	80	1678	3.24	3.08	2778
GRAIN-128	128	2191	4.63	1.83	5063

Mật mã dòng hạng nhẹ	Khóa (bit)	Diện tích (GE)	Nguồn (µW)	Năng lượng (pJ/bit)	Thông lượng (Mbps)
GRAIN-128a	128	3239	7.40	1.21	9877
MICKEY	128	5039	8.14	30.28	414
TRIVIUM	80	2599	3.84	17.74	358

4.3. Phát triển hệ mật mã trên đường cong elliptic

4.3.1. Phát triển hệ mật mã sử dụng khóa công khai

Ý tưởng thuật toán: Với nghiên cứu người ta đã sử dụng khóa công khai trên đường cong elliptic để mã hóa văn bản tiếng Anh. Luận án đề xuất áp dụng thuật toán mã hóa này với văn bản tiếng Việt.

4.3.2. Phát triển hệ mật mã sử dụng khóa đối xứng

Ý tưởng thuật toán: Với nghiên cứu người ta đã sử dụng khóa đối xứng trên đường cong elliptic để mã hóa văn bản tiếng Anh. Trong nghiên cứu người ta sử dụng thuật toán sinh chuỗi dữ liệu rồi sử dụng hệ mật mã trên đường cong ellliptic để mã hóa dữ liệu. Luận án đề xuất áp dụng thuật toán mã hóa này với văn bản tiếng Việt.

4.4. Thực nghiệm các hệ mật mã đã được đề xuất trên thiết bị thông minh

Các hệ mật mã trên đương cong elliptic đã được đề xuất và trình bày trong chương 3. Gồm các (i) hệ mật mã đường cong elliptic – CECC; (ii) hệ mật mã đường cong elliptic – AECC và (iii) hệ mật mã đường cong elliptic – VECC. Các thuật toán mã hóa và giải mã các hệ mật mã này được trình bày trong chương 3 với các thuật toán gồm: (i) thuật toán 3.2 mã hóa CECC; (ii) thuật toán 3.3 giải mã CECC; (iii) thuật toán 3.4 mã hóa AECC; (iv) thuật toán 3.5 giải mã AECC; (v) thuật toán 3.6 mã hóa VECC và (vi) thuật toán 3.7 giải mã VECC.

Các thuật toán đề xuất này được cài đặt thực nghiệm trên thiết bị thông minh với các thông số của thiết bị thông minh như sau: Phần mềm: Hệ điều hành android 12, phiên bản samsung galaxy S22; Phần cứng: RAM:8 GB; ROM: 128 GB; PIN: 3700 mAh

Kết quả thực nghiệm thuật toán đề xuất trên thiết bị thông minh trình bày trong phần phụ lục của luận án.

4.5. Thực nghiệm trên các thiết bị hạn chế

Sử dụng hệ mật mã nhẹ đã đề xuất và được thử nghiệm trên hệ thống IoT với các thiết bị bản mạch Arduino, bản mạch vi điều khiển ESP8266. Hệ thống có ba con Robot 1, Robot 2, Robot 3. Robot 1 và Robot 3 muốn trao đổi dữ liệu với nhau, cần con Robot 2 làm trung chuyển. Để đảm bảo bí mật và an toàn dữ liệu, trước khi dữ liệu gửi đi thì dữ liệu được mã hóa để Robot 2 không biết được nội dung của dữ liệu gửi.

KẾT LUẬN

- Luận án đề xuất các thuật toán mật mã trên đường cong elliptic.
  1) Thuật toán mật mã trên đường cong elliptic – CECC

     Ý tưởng của thuật toán dựa trên ý tưởng mật mã CAESAR, sử dụng khóa đối xứng K, khóa K là một giá trị số. Luận án kết hợp giữa ý tưởng này với đặc điểm, phép toán trên đường cong elliptic để đề xuất thuật toán mật mã trên đường cong elliptic – CECC.
  2) Thuật toán mật mã trên đường cong elliptic – AECC

     Ý tưởng của thuật toán dựa trên ý tưởng mật mã AFFINE, sử dụng khóa đối xứng K, khóa K là một cặp khóa K(u, v). Luận án kết hợp giữa ý tưởng này với đặc điểm, phép toán trên đường cong elliptic để đề xuất thuật toán mật mã trên đường cong elliptic – AECC.
  3) Thuật toán mật mã trên đường cong elliptic – VECC

     Ý tưởng của thuật toán dựa trên ý tưởng mật mã Vigenere, sử dụng khóa đối xứng K, khóa K là một giá trị chuỗi văn bản. Luận án kết hợp giữa ý tưởng này với đặc điểm, phép toán trên đường cong elliptic để đề xuất thuật toán mật mã trên đường cong elliptic – VECC.
- Luận án đề xuất cải tiến thuật toán mật mã khối LED và NOEKEON;
- Luận án đã phát triển hai hệ mật mã trên đường cong elliptic để mã hóa văn bản tiếng Việt;
- Luận án đã thử nghiệm thuật toán đề xuất trên thiết bị thông minh.
- Luận án đã thử nghiệm thuật toán đề xuất trên các thiết bị có năng lực tính toán hạn chế.

DANH MỤC CÔNG TRÌNH KHOA HỌC CỦA TÁC GIẢ

LIÊN QUAN ĐẾN LUẬN ÁN

1. [MMTRUNG1] **Mai Manh Trung**, Le Phe Do, Do Trung Tuan, Nguyen Van Tanh, Ngo Quang Tri, *"Design a cryptosystem using elliptic curves cryptography and sysmetry key"*, International Journal of Electrical and Computer Engineering (IJECE), Vol. 13, No. 2, pp. 1734~1743, ISSN: 2088-8708, DOI: 10.11591/ijece.v13i2.pp1734-1743, April 2023 **(SCOPUS).**
2. [MMTRUNG2] **Mai Manh Trung**, Le Phe Do, Le Trung Thuc, Dao Thi Phuong Anh, "*Proposing an elliptic curve cryptosystem with the symmetric key for Vietnamese text encryption and decryption*", International Journal of Advanced Trends in Computer Science and Engineering**,** ISSN 2278-3091, Volume 9, No.3, May - June 2020 **(SCOPUS).**
3. [MMTRUNG3] **Mai Manh Trung**, Do Trung Tuan, Le Phe Do, "*Building an elliptic curve cryptography to encode and decode Vietnamese texts*", Computer Science and Communication Engineering, VNU Journal of Science, 44-51, Vol.36, No. 2, 2020.
4. [MMTRUNG4] Nguyen Van Tanh, Ngo Quang Tri, **Mai Manh Trung**, "*The solution to improve information security for IoT networks by combining lightweight encryption protocols",* Indonesian Journal of Electrical Engineering and Computer Science, Vol. 23, No. 3, ISSN: 2502-4752, DOI: 10.11591/ijeecs.v23.i3.pp1727-1735, September 2021 **(SCOPUS).**
5. [MMTRUNG5] **Mai Manh Trung**, Do Trung Tuan, Le Phe Do, "*Building elliptic curve cryptography with public key to encrypt Vietnamese text*", Journal of science and technology on information security, Special Issue CS (I5), pp 119-126, 2022.
6. [MMTRUNG6] **Mai Mạnh Trừng**, Đỗ Trung Tuấn, Lê Phê Đô, Lê Trung Thực, Đào Thị Phương Anh, "*Xây dựng hệ mật mã đường cong elliptic với khóa đối xứng Affine để mã hóa giải mã văn bản tiếng Việt*", Kỷ yếu Hội nghị KHCN Quốc gia lần thứ XIII về Nghiên cứu cơ bản và ứng dụng Công nghệ thông tin (FAIR), 724-732, Nha Trang, ngày 8-9/10/2020.
7. [MMTRUNG7] Lê Phê Đô, **Mai Mạnh Trừng**, Lê Trung Thực, Nguyễn Thị Hằng, Vương Thị Hạnh, Nguyễn Khắc Hưng, Định Thị Thúy, Lê Thị Len, *"Nghiên cứu một số hệ mật mã nhẹ và ứng dụng trong IoT"*, Tạp chí Nghiên cứu khoa học và Công nghệ Quân Sự, chủ đề "Những tiến bộ Khoa học trong lĩnh vực An ninh-An toàn thông tin", 137-147, số đặc san 5-2017.
8. [MMTRUNG8] Lê Phê Đô, **Mai Mạnh Trừng**, Nguyễn Khắc Hưng, Trần Văn Mạnh, Lê Trung Thực, Lê Thị Len, Nguyễn Thị Hằng *"Cải tiến mã khối nhẹ LED và NOEKEON"*, Kỷ yếu hội thảo Quốc gia lần thứ XX, "Một số vấn đề chọn lọc của CNTT & TT", 8-12, 2017.
9. [MMTRUNG9] **Mai Mạnh Trừng**, Lê Phê Đô, Lê Trung Thực, Trần Văn Mạnh, Lê Thị Len, Nguyễn Thị Hằng, Nguyễn Khắc Hưng, *"Nghiên cứu các cuộc tấn công hệ mật mã nhẹ PRESENT",* Kỷ yếu Hội thảo lần thứ II, "Một số vấn đề chọn lọc về An toàn An ninh Thông tin", 1-6, Thành phố H.C.M, 2017.
10. [MMTRUNG10] **Mai Mạnh Trừng**, Lê Trung Thực, Đào Thị Phương Anh, *"Ứng dụng phân tích dữ liệu và phân lớp giám sát NAÏVE BAYES phát hiện gian lận trong thanh toán trực tuyến"*, TNU Journal of Science and Technology, ISSN: 1859-2171 e-ISSN: 2615-9562, 225 (06): 157-164, 2020.