Một số thuật ngữ, từ viết tắt và ký hiệu

AddRoundKey	: thao tác cộng khóa của chu kỳ vào khối đang mã hóa trong AES và XAES
AES	: chuẩn mã hóa Advanced Encryption Standard
ALH	: bao tuyến tính
Branch Number	: Một tiêu chuẩn đánh giá mức độ khuếch tán thông tin của biến đổi tuyến tính
CRYPTREC	: Cryptography Research and Evaluation Committee, do Chính phủ Nhật thành lập từ năm 2000
DES	: chuẩn mã hóa Data Encryption Standard
DC	: Phương pháp sai phân để phân tích mã (Differential Cryptanalysis)
DIFF	: Tập vết sai phân
DP	: Xác suất sai phân (Differential Probability)
KeySchedule	: Hàm phát sinh các khóa sử dụng cho từng chu kỳ mã hóa từ khóa chính
LC	: Phương pháp tuyến tính để phân tích mã (Linear Cryptanalysis)
LP	: Xác suất tuyến tính (Linear Probability)
MDS-code	: Maximum Distance Separable code
MixColumns	: Một phép biến đổi tuyến tính trong AES và XAES
NIST	: National Institute of Standards and Technologies (Viện Tiêu chuẩn và Công nghệ Hoa Kỳ)
NESSIE	: New European Schemes for Signature, Integrity and Encryption là dự án nghiên cứu của E.U.
SAC	: Strict Avalanche Criterion

S-box	: Bảng thay thế
SDS	: Kiến trúc hàm mã hóa gồm tầng thay thế - tầng khuếch tán – tầng thay thế (Substitution – Diffusion – Subsitution)
ShiftRows	: Một phép biến đổi tuyến tính trong AES và XAES
SPN	: Kiến trúc thuật toán mã hóa khối "Mạng thay thế - hoán vị" (Substitution – Permutation Network)
SubBytes	: Phép biến đổi phi tuyến trong AES và XAES
Wide Trail Strategy	: Chiến lược vết rộng
$\bullet$	: Tích vô hướng của hai vector
$\oplus$	: Phép toán XOr
$\otimes$	: Phép nhân 2 đa thức (mỗi đa thức có bậc tối đa là $Nw - 1$ và có hệ số trên GF(2^m)) modulo cho đa thức $x^{Nw} + 1$.
m	: số lượng bit trong mỗi nhóm đơn vị dữ liệu
Nw	: số lượng nhóm (m bit) trong mỗi từ
Nk	: số lượng từ trong khóa chính
Nb	: số lượng từ trong khối
Nr	: số lượng chu kỳ mã hóa
k^r	: khóa sử dụng trong chu kỳ mã hóa thứ r
π	: biến đổi ShiftRows
θ	: biến đổi MixColumns
σ	: biến đổi AddRoundKey
φ	: biến đổi SubBytes
$\mathcal{B}$	: Branch Number

Mục lục

Danh sách hình

Danh sách bảng

Mở đầu

Tóm tắt:

✍ *Nội dung phần mở đầu trình bày tổng quan về luận án, mục tiêu và các đóng góp chính trong luận án. Nội dung tóm tắt của từng chương trong luận án được trình bày ở cuối phần này.*

Tổng quan

Mật mã học là ngành nghiên cứu các kỹ thuật Toán học nhằm cung cấp các dịch vụ an toàn thông tin [64][85]. Mặc dù khoa học mật mã đã ra đời từ hàng nghìn năm nhưng trải qua nhiều thế kỷ, các kết quả của Mật mã học chủ yếu chỉ được sử dụng trong lĩnh vực quân sự, chính trị, ngoại giao... Ngày nay, các ứng dụng mã hóa và bảo mật thông tin được sử dụng ngày càng phổ biến trong các lĩnh vực khác nhau trên Thế giới, từ việc bảo mật nội dung các tài liệu điện tử, bảo vệ an toàn các giao dịch thương mại điện tử, đấu giá trên mạng, bầu cử trực tuyến… đến ứng dụng trong các hệ thống thẻ thông minh, mạng cảm ứng không dây, hệ thống ubiquitous…

Cho đến đầu thập niên 1970, hầu hết các nghiên cứu và ứng dụng của Mật mã học tập trung vào việc bảo mật thông tin [48]. Từ giữa thập niên 1970 đến nay, phạm vi nghiên cứu của Mật mã học được mở rộng, các ứng dụng của Mật mã học ngày càng đa dạng và phong phú. Tùy vào đặc thù của mỗi hệ thống bảo vệ thông tin mà ứng dụng sẽ có các tính năng với đặc trưng riêng. Dưới đây là một số tính năng chính của hệ thống bảo vệ thông tin [64]:

- ***Bảo mật thông tin***: hệ thống đảm bảo thông tin được giữ bí mật. Thông tin có thể bị phát hiện, ví dụ như trong quá trình truyền nhận, nhưng người tấn công không thể hiểu được nội dung thông tin bị đánh cắp này.
- ***Toàn vẹn thông tin***: hệ thống bảo đảm tính toàn vẹn thông tin trong liên lạc hoặc giúp phát hiện rằng thông tin đã bị sửa đổi.

- ***Xác thực*** các đối tác trong liên lạc và xác thực nội dung thông tin trong liên lạc.
- ***Chống từ chối trách nhiệm***: hệ thống đảm bảo một đối tác bất kỳ trong hệ thống không thể từ chối trách nhiệm về hành động mà mình đã thực hiện

Các ứng dụng đầu tiên và phổ biến nhất của Mật mã học là bảo mật nội dung thông tin sử dụng ***hệ thống mã hóa đối xứng*** (hay còn gọi là hệ thống mã hóa quy ước). Trong hệ thống này, quá trình mã hóa và giải mã một thông điệp sử dụng cùng một khóa gọi là ***khóa bí mật*** (secret key) hay ***khóa đối xứng*** (symmetric key). Do đó, vấn đề bảo mật thông tin đã mã hóa hoàn toàn phụ thuộc vào việc giữ bí mật nội dung của khóa đã được sử dụng.

Hầu hết các thuật toán mã hóa đối xứng ra đời từ nửa cuối thế kỷ XX đều là các thuật toán mã hóa theo khối. Các thuật toán này được xây dựng dựa trên nguyên lý của C. Shannon về sự ***hỗn loạn*** (*confusion*) và ***khuếch tán*** (*diffusion*) thông tin [78]. Tính hỗn loạn giúp phá vỡ mối quan hệ giữa thông điệp nguồn và thông điệp đã mã hóa, còn sự khuếch tán sẽ phá vỡ và phân tán các phần tử trong các mẫu xuất hiện trong thông điệp nguồn để không thể phát hiện ra các mẫu này trong thông điệp sau khi mã hóa. Shannon đề xuất phương án sử dụng ***phép thay thế*** và ***biến đổi tuyến tính*** để tạo ra sự hỗn loạn và khuếch tán thông tin. Hiện nay, ***hai kiến trúc chính của các phương pháp mã hóa theo khối*** là mạng Feistel [25] và mạng thay thế - hoán vị (Substitution-Permutation Network [43] – SPN)

Từ giữa thập niên 1970, với sự ra đời của chuẩn mã hóa DES [31] được xây dựng theo kiến trúc mạng Feistel, các nghiên cứu tập trung khá nhiều vào kiến trúc này. Nhiều giải thuật mã hóa theo kiến trúc Feistel đã được đề xuất, ví dụ như RC2 [53], FEAL [82], TEA [99], BlowFish [81], CAST-128 [1], MARS [9], RC6 [76], TwoFish [80], Camellia [55]…

Thập niên 1990 đánh dấu những kết quả quan trọng trong lĩnh vực phân tích mã. Phương pháp sai phân (differential cryptanalysis [6]) do E. Biham và A. Shamir đề xuất năm 1991 cùng với phương pháp tuyến tính (linear cryptanalysis [62]) của M. Matsui đề xuất năm 1993 được đánh giá là hai phương pháp hiệu quả trong việc phân tích, tấn công các thuật toán mã hóa khối, kể cả chuẩn DES. Vấn đề an toàn đối với

phương pháp tấn công sai phân và tuyến tính ***trở thành tiêu chuẩn khi thiết kế và đánh giá các thuật toán mã hóa theo khối***. Chúng tôi đã vận dụng tiêu chuẩn này khi xây dựng và chứng minh tính an toàn của giải thuật XAES được đề xuất trong luận án này.

Trước tình hình phương pháp DES không còn đủ mức độ an toàn để bảo mật các thông tin quan trọng, năm 1997, Viện Tiêu chuẩn và Công nghệ Quốc gia Hoa Kỳ (NIST) đã kêu gọi các nhà nghiên cứu xây dựng các thuật toán mã hóa theo khối an toàn hơn để chọn ra thuật toán chuẩn mã hóa nâng cao - Advanced Encryption Encryption, gọi tắt là AES. Ngày 2 tháng 10 năm 2000, phương pháp Rijndael [16] của hai tác giả người Bỉ là Vincent Rijmen và Joan Daemen, được xây dựng theo kiến trúc SPN, đã chính thức được chọn trở thành chuẩn AES.

Trong số 5 giải thuật được lọt vào vòng chung kết, có 3 giải thuật được xây dựng theo kiến trúc Feistel, gồm MARS [9], TwoFish [80] và RC6 [76]; 2 giải thuật còn lại là Rijndael và Serpent [2] được xây dựng theo kiến trúc SPN. Điều đáng chú ý là giải thuật Serpent được đánh giá xếp thứ hai, ngay sau thuật toán Rijndael.

Mặc dù đã có những nghiên cứu đạt được các kết quả quan trọng bước đầu về kiến trúc SPN trước năm 2000 như chiến lược vết rộng (Wide Trail strategy) [19], thuật toán Shark [90], thuật toán Square [18], nhưng sau khi phương pháp Rijndael được chính thức trở thành chuẩn AES, ngày càng có nhiều công trình nghiên cứu, khảo sát, phân tích về thuật toán Rijndael nói riêng, về kiến trúc SPN nói chung và một số vấn đề liên quan. Một số phương pháp mã hóa khối theo kiến trúc SPN hoặc tựa-Rijndael đã được đề xuất từ năm 2000 đến nay, ví dụ như Aria [54], Crypton [59], Anubis[3], Khazad [91], GrandCru [8]. Chính vì vậy, trong phạm vi luận án này, chúng tôi chọn hướng tiếp cận theo kiến trúc SPN và giải thuật Rijndael để đề xuất giải thuật XAES.

Lý do thực hiện luận án

Cùng với việc NIST chính thức công bố chuẩn AES, nhiều thuật toán mã hóa khối được đề xuất để được đánh giá và chọn làm chuẩn mã hóa trong dự án NESSIE của Châu Âu và CRYPTREC của chính phủ Nhật Bản. Bên cạnh đó, nhiều tập đoàn lớn như cũng đầu tư nghiên cứu để xây dựng các hệ mã riêng của mình, ví dụ như các thuật toán Hierocrypt [95] của Toshiba, thuật toán Camellia [55] của NTT và Mitsubishi. Điều này khẳng định ***nhu cầu thực tế*** của các quốc gia cũng như giới công nghiệp về các chuẩn mật mã riêng, đặc biệt là các ***thuật toán mã hóa khối mới***.

Trên thực tế, hầu hết các thuật toán mã hóa khối được đề xuất đều được đặc tả "cứng" với một số kích thước khóa, kích thước khối cố định, các hằng số cố định và không nêu ra tường minh khả năng ***mở rộng*** và tạo ra các ***biến thể*** của giải thuật.

Khả năng mở rộng của thuật toán mã hóa khối

Khả năng mở rộng của thuật toán được đề cập đến trong phạm vi luận án này được xét trên hai góc độ sau:

- ***Tính dễ mở rộng về kích thước khóa và kích thước khối***: khác với thuật toán khóa công khai (ví dụ như RSA [75]) cho phép sử dụng khóa và khối có kích thước linh hoạt tùy ý, mỗi thuật toán mã hóa khối được xây dựng ứng với một số kích thước khóa và một số kích thước khối cố định. Khi cần nâng mức an toàn lên, ví dụ việc sử dụng khóa 64 bit lên khóa 128 bit, cần đề xuất thuật toán mã hóa khối mới. Nếu thuật toán mã hóa khối có kiến trúc tốt và có khả năng được tổng quát hóa để phù hợp với kích thước khóa và kích thước khối lớn thì các phiên bản mở rộng của thuật toán có thể được dùng để thay thế cho phiên bản hiện tại khi yêu cầu về mức an toàn của thuật toán được nâng lên.
- ***Tính dễ mở rộng để tương thích các kiến trúc xử lý khác nhau***: Hầu hết các thuật toán mã hóa khối hiện nay thường được xây dựng theo kiến trúc xử lý dựa trên byte (8 bit). Do đó, các thuật toán này chưa thật sự phù hợp trong các hệ thống mà đơn vị dữ liệu được xử lý không phải là byte, ví dụ như trong một số hệ thống mạng cảm ứng không dây hoặc ubiquitous, dữ liệu được xử lý theo từng nhóm 4 bit hoặc 6 bit.

Khả năng tạo ra các biến thể của thuật toán mã hóa khối

Biến thể *của một thuật toán mã hóa là thuật toán có cùng cấu trúc nhưng khác hằng số*. Thông thường, bộ hằng số được trình bày trong đặc tả của thuật toán mã hóa không phải là bộ hằng số tối ưu duy nhất. Nếu trong đặc tả của thuật toán nêu rõ cách tự xây dựng bộ hằng số để dùng trong thuật toán thì mỗi cá nhân hay tổ chức có thể dễ dàng tạo ra các biến thể của thuật toán. Việc tạo ra biến thể mang đến một số lợi ích sau:

- Mỗi tổ chức, cá nhân có thể tự tạo ra các biến thể của thuật toán để sử dụng riêng trong hệ thống của mình. Ngoài thông tin bí mật của khóa, thông tin về giá trị của bộ hệ số được giữ bí mật, giúp tăng cường độ an toàn của hệ thống,
- Giúp giải quyết mối lo về các "bẫy" (trap door) được đưa vào trong bộ hệ số được trình bày trong đặc tả của thuật toán gốc.

Tham số hóa thuật toán

Trong đặc tả của thuật toán Rijndael [16], V. Rijmen và J. Daemen đã sử dụng 3 tham số, gồm: *Nk* là số từ (32-bit) trong khóa, *Nb* là số từ (32-bit) trong khối và *Nr* là số chu kỳ mã hóa. Mục đích chủ yếu khi các tác giả của Rijndael sử dụng các tham số này nhằm phát biểu đặc tả thuật toán ở dạng tổng quát với một số (hữu hạn) giá trị khác nhau cụ thể của mỗi tham số. Khả năng mở rộng (không giới hạn) kích thước khối và kích thước khóa, cũng như khả năng tương thích với các kiến trúc xử lý khác nhau vẫn chưa được giải quyết đối với thuật toán Rijndael. Bên cạnh đó, khả năng sử dụng các giá trị tham số khác bên ngoài tập giá trị định sẵn của mỗi tham số được nêu ra trong [16][30] như một hướng mở cho việc nghiên cứu.

Trong luận án này, chúng tôi cũng ứng dụng ý tưởng tham số hóa trong đặc tả thuật toán nhưng với mục tiêu nhằm giải quyết vấn đề tính dễ mở rộng và khả năng tạo ra biến thể cho thuật toán. Chúng tôi đề xuất thuật toán mã hóa khối được tham số hóa (XAES) trên cơ sở mở rộng và tham số hóa chuẩn mã hóa AES. Giải thuật XAES được chúng tôi xây dựng với hai nhóm tham số chính như sau:

- ***Các tham số cấu trúc***: cho phép định nghĩa cấu trúc của thuật toán một cách tổng quát, giúp giải quyết vấn đề mở rộng (không giới hạn) kích thước khối, kích thước khóa và tương thích với các kiến trúc xử lý khác nhau.
- ***Các tham số xử lý***: cho phép tham số hóa các hằng số trong mỗi thành phần mã hóa/giải mã của thuật toán, cho phép tạo ra các biến thể của thuật toán.

Với chiến lược tham số hóa này, giải thuật XAES được đề nghị không phải là một thuật toán mã hóa khối cụ thể mà xác định một họ các thuật toán mã hóa khối tựa-Rijndael gồm vô hạn các thuật toán cụ thể tương ứng với các kích thước khóa, kích thước khối lớn không giới hạn, tương thích với các kiến trúc xử lý khác nhau, và tương ứng với những bộ hằng số trong thuật toán khác nhau. Tính an toàn của XAES đối với các phương pháp phân tích hiện nay được chứng minh tổng quát, độc lập với giá trị cụ thể của các tham số cấu trúc và tham số xử lý. Vì vậy, mỗi thuật toán mã hóa khối - thể hiện cụ thể của XAES - đều an toàn đối với các phương pháp phân tích mã hiện tại tương ứng với kích thước khóa.

Mục tiêu và đóng góp của luận án

Mục tiêu của luận án nhằm đề xuất một thuật toán mã hóa khối được tham số hóa dựa trên việc mở rộng thuật toán AES nhằm giải quyết vấn đề tính dễ mở rộng và khả năng tạo ra các biến thể; trên cơ sở đó, chứng minh tổng quát tính an toàn của thuật toán mã hóa khối được tham số hóa đã xây dựng và đề xuất một số giải thuật để tạo ra các bộ hệ số được sử dụng trong thuật toán.

Các đóng góp chính của luận án:

1. Đề xuất ý tưởng về việc xây dựng thuật toán mã hóa được tham số hóa (với hai loại tham số: tham số cấu trúc và tham số xử lý) nhằm giải quyết vấn đề tính dễ mở rộng và khả năng tạo ra các biến thể. Đây là cầu nối giữa kiến trúc thuật toán ở mức trừu tượng với các thuật toán mã hóa cụ thể.
2. Đề xuất một thuật toán mã hóa được tham số hóa được đặt tên là XAES trên cơ sở tổng quát hóa và tham số hóa các thành phần trong giải thuật Rijndael (AES).

3. Chứng minh tổng quát tính an toàn của XAES đối với phương pháp tuyến tính và phương pháp sai phân trong việc phân tích mã theo hướng tiếp cận truyền thống dựa trên vết sai phân đơn và vết tuyến tính đơn
4. Chứng minh tổng quát tính an toàn của XAES đối với phương pháp tuyến tính và phương pháp sai phân trong việc phân tích mã theo hướng tiếp cận mới dựa trên tập vết sai phân và bao tuyến tính.
5. Đề xuất thuật giải kiểm tra sơ bộ và kiểm tra ngẫu nhiên để kiểm tra các bộ hệ số mạnh (bao gồm bộ hệ số khuếch tán tối đa hoặc gần tối đa) dùng cho biến đổi MixColumns trong XAES
6. Xây dựng thành công một cải tiến cho S-box trong AES bằng cách sử dụng bước chuyển đổi sang mã Gray làm tiền xử lý cho S-box trong AES. Kết quả tạo ra S-box có biểu diễn đại số gồm đầy đủ 255 đơn thức có hệ số khác 0 và vẫn bảo toàn các tính chất mật mã ưu điểm của S-box trong AES.

Nội dung luận án

Nội dung của luận án được trình bày gồm:

- **Phần mở đầu** trình bày tổng quan về luận án; phân tích nhu cầu thực tế về việc đề xuất các thuật toán mã hóa khối mới có độ an toàn cao, vấn đề tính dễ mở rộng và khả năng tùy biến thuật toán; từ đó, nêu lên mục tiêu của luận án.
- **Chương 1** trình bày tổng quan về mối liên hệ giữa kiến trúc thuật toán mã hóa khối với các thuật toán cụ thể và vai trò cầu nối của XAES; phân tích các hướng tiếp cận trong việc xây dựng các giải thuật tựa-Rijndael và các mở rộng; trình bày thuật toán AES và phân tích giải pháp tổng quát hóa AES để xây dựng XAES.
- **Chương 2** trình bày chi tiết về thuật toán mã hóa khối được tham số hóa XAES, trong đó nêu rõ các tham số và cách xây dựng các thành phần mã hóa trong XAES.

- **Chương 3** trình bày kết quả chứng minh tính an toàn của XAES đối với phương pháp sai phân và phương pháp tuyến tính sử dụng hướng tiếp cận truyền thống với vết lan truyền (sai phân/tuyến tính) đơn.
- **Chương 4** trình bày kết quả chứng minh tính an toàn của XAES đối với phương pháp sai phân và phương pháp tuyến tính sử dụng hướng tiếp cận với tập vết lan truyền (sai phân/tuyến tính).
- **Chương 5** trình bày giải thuật kiểm tra sơ bộ và giải thuật kiểm tra ngẫu nhiên nhằm tối ưu hóa việc tạo ra các bộ hệ số cho biến đổi tuyến tính trong biến đổi MixColumns của XAES.
- **Chương 6** trình bày kết quả cải tiến cho S-box trong AES bằng cách sử dụng bước chuyển đổi sang mã Gray làm tiền xử lý cho S-box. Kết quả tạo ra 1 S-box có biểu diễn đại số gồm đầy đủ 255 đơn thức có hệ số khác 0 và vẫn bảo toàn các tính chất mật mã ưu điểm của S-box trong AES. Kết quả này nhằm minh họa cho kiến trúc S-box được sử dụng XAES với 2 ánh xạ affine.
- **Phần kết luận và hướng phát triển**
- **Phụ lục A** trình bày tóm tắt một số quy trình ứng dụng các thuật toán có độ an toàn cao vào việc bảo vệ thông tin, bao gồm quy trình nhúng thông tin mật vào dữ liệu multimedia, hệ thống dịch vụ multimedia tích hợp mã hóa bảo mật nội dung và chứng thực người dùng.
- **Phụ lục B** trình bày tất cả các bộ hệ số tối ưu cho biến đổi MixColumns của XAES trong trường hợp m=8 và Nw = 2, 3, …, 8.

Chương 1

Kiến trúc mã hóa khối và AES

Tóm tắt chương:

✍ *Nội dung chương 1 trình bày các vấn đề chính sau:*

- ❖ *Giới thiệu và phân tích quá trình phát triển của kiến trúc thuật toán mã hóa khối, xuất phát từ ý tưởng của C. Shannon đến kiến trúc mạng Feistel [31], kiến trúc mạng thay thế - hoán vị (SPN [43]), chiến lược vết rộng (wide trail strategy [20]). Từ đó, đi đến kết luận XAES là một bước tiếp nối giữa các kiến trúc mã hóa khối với các thuật toán mã hóa cụ thể.*
- ❖ *Trình bày và phân tích các thuật toán mã hóa khối tựa-Rijndael và các mở rộng đã được đề xuất, từ đó rút ra các kết luận về những hướng tiếp cận trong việc tạo ra những thuật toán tựa-Rijndael cũng như các phiên bản mở rộng.*
- ❖ *Trình bày về thuật toán AES và phân tích hướng tiếp cận của chúng tôi trong việc tổng quát hóa các thành phần của AES để xây dựng XAES.*

1.1 Từ kiến trúc thuật toán mã hóa khối đến XAES

1.1.1 Kiến trúc thuật toán mã hóa khối

Trong bài viết "Communication Theory of Secrecy Systems" xuất bản năm 1949, C. Shannon đã đề xuất ***một phương án tổng quát*** để xây dựng thuật toán mã hóa khối an toàn bằng cách sử dụng kết hợp các thao tác mã hóa tạo ra tính hỗn loạn và tính khuếch tán thông tin [78].

- ***Tính hỗn loạn*** giúp phá vỡ mối quan hệ giữa bản rõ và bản mã, tạo ra mối quan hệ phức tạp và chặt chẽ giữa khóa với bản mã.
- ***Sự khuếch tán*** giúp phá vỡ và phân tán các phần tử trong các mẫu xuất hiện trong bản rõ để không thể phát hiện ra các mẫu này trong bản mã.

Ý tưởng của Shannon được xem là một phương án tổng quát đầu tiên cho việc xây dựng các thuật toán mã hóa khối hiện đại. Xuất phát từ ý tưởng của Shannon, một số ***kiến trúc mã hóa khối*** đã được đề xuất. Trong số đó, ***mạng Feistel*** [31] và ***mạng thay thế - hoán vị*** (Substitution-permutation-network - SPN) [43] là hai kiến trúc mã hóa khối được sử dụng phổ biến trong việc tạo ra các thuật toán mã hóa khối hiện đại.

1.1.2 "Chiến lược vết rộng"

Chiến lược ***"Wide Trail Strategy",*** tạm dịch là "chiến lược vết rộng", được J. Daemen đề xuất trong [19] và được phân tích chi tiết trong [20][21]. "Chiến lược vết rộng" được đề xuất để cụ thể hóa cách xây dựng một lớp các thuật toán mã hóa khối theo kiến trúc SPN. Trong chiến lược vết rộng, tác giả đã đề xuất một kiến trúc trừu tượng cho thuật toán mã hóa khối dựa trên kiến trúc SPN, đồng thời chứng minh cách xác định giới hạn để kiểm tra tính an toàn đối với phương pháp tấn công cho các thuật toán được xây dựng theo chiến lược vết rộng.

Thuật toán mã hóa tham số hóa XAES mà chúng tôi đề xuất được xây dựng dựa trên chiến lược vết rộng. Do đó, trong phần dưới đây, chúng tôi trình bày tóm tắt về các thành phần của chiến lược này.

Trong chiến lược vết rộng, bản rõ được chia thành các khối dữ liệu có kích thước bằng nhau cố định. Mỗi khối được mã hóa với khóa chính k cho trước và tạo ra một khối có cùng kích thước. Quá trình mã hóa gồm Nr chu kỳ biến đổi. Trong chu kỳ r, $(1 \le r \le Nr)$, khóa của chu kỳ, ký hiệu là k^r, được phát sinh từ khóa chính k thông qua hàm sinh khóa $KeySchedule$.

Mỗi chu kỳ mã hóa r $(1 \le r \le Nr)$ gồm 2 bước xử lý:

- Biến đổi độc lập khóa (ký hiệu là ρ^r): gồm một số biến đổi bool độc lập khóa,
- Cộng khóa (ký hiệu là σ): mỗi bit của trạng thái hiện tại của khối dữ liệu đang được mã hóa sẽ được XOr với bit tương ứng trong khóa k^r của chu kỳ r.

Trong chiến lược vết rộng, thuật toán mã hóa C sử dụng khóa chính k bắt đầu bằng thao tác cộng khóa, tiếp theo là Nr chu kỳ mã hóa.

$$C[k] = \sigma[k^{Nr}] \circ \rho^{Nr} \circ \sigma[k^{Nr-1}] \circ \rho^{Nr-1} \circ \ldots \circ \sigma[k^1] \circ \rho^1 \circ \sigma[k^0] \qquad (1.1)$$

Đặt $\zeta^r[k^r] = \sigma[k^r] \circ \rho^r$ là thủ tục mã hóa trong chu kỳ r, thuật toán mã hóa C với khóa chính k được biểu diễn lại như sau:

$$C[k] = \zeta[k^{Nr}] \circ \zeta[k^{Nr-1}] \circ \ldots \circ \zeta[k^1] \circ \sigma[k^0] \qquad (1.2)$$

Phép biến đổi độc lập khóa ρ^r được xây dựng bằng cách kết hợp hai thao tác biến đổi khả nghịch sau:

- φ: phép thay thế phi tuyến cục bộ. Tính chất cục bộ của φ được hiểu là các bit đầu vào (và bit đầu ra) được xử lý cục bộ theo từng nhóm gồm m bit [20]
- λ: phép biến đổi trộn tuyến tính có khả năng tạo ra tính khuếch tán cao sau một số chu kỳ mã hóa. Tính chất này sẽ được phân tích chi tiết trong phần 3.2.

So với kiến trúc SPN, chiến lược vết rộng đã tiến thêm một bước trong việc cụ thể hóa cách xây dựng thuật toán mã hóa khối. Tuy nhiên, chiến lược vết rộng vẫn dừng lại ở mức trừu tượng. Trong chiến lược này chưa nêu ra cách cụ thể để xây dựng từng thành phần mã hóa, ví dụ như hàm KeySchedule để phát sinh khóa cho từng chu kỳ từ khóa chính k cho trước, các hàm biến đổi độc lập khóa (φ và λ). Mỗi nhóm nghiên cứu mật mã sẽ tự đề xuất cách xây dựng cụ thể các thành phần để gắn vào khung thuật toán tổng quát này. Giải thuật Rijndael là một thuật toán cụ thể đã hiện thực hóa thành công chiến lược vết rộng. Ngoài ra, còn có nhiều thuật toán mã hóa khối khác được đề xuất trên cơ sở cụ thể hóa chiến lược vết rộng. Các thuật toán này sẽ được giới thiệu và phân tích trong phần 1.2-Các thuật toán mã hóa khối tựa-Rijndael.

1.1.3 Chiến lược vết rộng và XAES

Thuật toán XAES được chúng tôi đề xuất cũng theo hướng tiếp cận nhằm hiện thực hóa chiến lược vết rộng. Chúng tôi quyết định tổng quát hóa cách xây dựng các thành phần mã hóa trong Rijndael để gắn vào khung tổng quát của chiến lược vết rộng do các thành phần mã hóa trong thuật toán Rijndael đã được các chuyên gia mật mã nghiên cứu trong nhiều năm gần đây và các tính chất quan trọng của những thành phần này đã được khảo sát và phân tích kỹ. Tất cả các thành phần mã hóa trong XAES không dừng lại ở mức trừu tượng như trong chiến lược vết rộng mà đều được đặc tả chi tiết cách xây dựng.

Tuy nhiên, mục tiêu của việc đề xuất XAES không phải là xây dựng một thuật toán mã hóa cụ thể theo chiến lược vết rộng như các thuật toán được trình bày trong phần 1.2.1 mà nhằm đề xuất một ***phương pháp cụ thể*** để tạo ra ***một lớp các thuật toán mã hóa khối*** theo chiến lược vết rộng. Vì thế, chúng tôi đã đề xuất việc tham số hóa các thành phần mã hóa trong XAES.

So với chiến lược vết rộng, XAES đã tiến thêm một bước trong việc cụ thể hóa cách xây dựng một lớp các thuật toán mã hóa khối. Trong XAES, cách xây dựng và xử lý trong các thành phần mã hóa đều được tham số hóa nên có thể chứng minh được công thức tổng quát cho độ an toàn của XAES đối với các phương pháp phân tích mã hiện nay. Trong khi đó, với chiến lược vết rộng nói riêng và các kiến trúc thuật toán mã hóa nói chung, các thành phần mã hóa được đề xuất ở mức trừu tượng nên cần phải chứng minh tính an toàn của từng thuật toán cụ thể.

So với các thuật toán mã hóa cụ thể, XAES có mức độ trừu tượng cao hơn. Với các thuật toán cụ thể, việc chứng minh tính an toàn đối với các phương pháp phân tích mã được thực hiện với các giá trị cụ thể . Đối với XAES, tính an toàn đối với các phương pháp phân tích mã được chứng minh tổng quát, không phụ thuộc vào giá trị cụ thể của các tham số mà chỉ sử dụng các tính chất, ràng buộc trên các tham số.

Có thể xem XAES như một cầu nối giữa chiến lược trừu tượng với các giải thuật mã hóa cụ thể. Tập hợp các thể hiện của XAES là một tập con (vô hạn) của tập hợp các thuật toán mã hóa khối xây dựng theo chiến lược vết rộng.

1.2 Các thuật toán mã hóa khối tựa-Rijndael và các mở rộng

1.2.1 Các thuật toán mã hóa khối tựa-Rijndael

Trong phần này, chúng tôi trình bày và phân tích các điểm tương đồng và khác biệt giữa Rijndael với một số thuật toán mã hóa khối tựa-Rijndael, bao gồm GrandCru [8], Khazad [91], Anubis [3]. Trong mỗi thuật toán này, các tác giả đều tái sử dụng, một phần hay toàn bộ, một số thành phần mã hóa của AES. Các thành phần còn lại được thay thế bằng các thành phần tương đương về tính năng và thỏa mãn một số tiêu chí riêng.

Trong thuật toán GrandCru, kích thước khối và kích thước khóa được giữ nguyên là 128 bit như thuật toán Rijndael và sử dụng lại hàm phát sinh khóa trong Rijndael. Điểm khác biệt giữa thuật toán GrandCru và Rijndael là việc thay thế các thao tác không sử dụng khóa trong Rijndael bằng các thao tác sử dụng khóa.

Đối với Anubis, mục tiêu chính là giảm tối đa sự khác biệt trong quy trình mã hóa với quy trình giải mã để có thể tái sử dụng các thành phần của module mã hóa trong việc giải mã, nhằm tiết kiệm chi phí cài đặt trên phần cứng. Do đó, Anubis tái sử dụng toàn bộ cấu trúc thuật toán của Rijndael và lần lượt thay thế từng thành phần trong quy trình mã hóa bằng thành phần tương đương có tính chất xoắn (biến đổi f trên miền D được gọi là có tính chất xoắn nếu $f(f(x)) = x, \forall x \in D$):

- Biến đổi MixColumns trong Anubis được tạo ra bằng cách thay thế mã MDS [8,4,5] trong Rijndael bằng một mã MDS [8,4,5] khác sao cho biến đổi MixColumns có tính xoắn.
- Biến đổi ShiftRows trong Anubis được thay thế bằng phép chuyển vị ma trận vuông, đảm bảo tính chất phân tán tất cả các byte trên mỗi cột của khối dữ liệu sang các cột khác nhau.
- S-box trong Anubis có tính xoắn và được xây dựng với cấu trúc đệ quy. Mặc dù S-box này không có được các tính chất mật mã tối ưu như S-box trong Rijndael (được xây dựng dựa trên ánh xạ nghịch đảo trên GF(2^8)) nhưng khi kết hợp với các thành phần mật mã khác trong Anubis vẫn đảm bảo độ an toàn đối với phương pháp phân tích mã sai phân và tuyến tính.

Thuật toán Khazad hỗ trợ kích thước khối 64 bit và kích thước khóa 128 bit. Tương tự Anubis, Khazad cũng hướng đến việc giảm thiểu sự khác biệt giữa quy trình mã hóa với quy trình giải mã. Do đó, trong thuật toán Khazad cũng sử dụng S-box xoắn được xây dựng theo cấu trúc đệ quy để thay thế S-box của Rijndael, đồng thời thay thế mã MDS [8, 4, 5] trong Rijndael bằng mã MDS [16, 8, 9].

Từ những phân tích trên đây, chúng ta có thể rút ra một số kết luận sau:

- Có thể tạo ra thuật toán mã hóa khối đáp ứng một số yêu cầu hay tiêu chí mới bằng cách thay thế một số thành phần mã hóa bằng các thành phần có tính năng tương đương và thỏa mãn yêu cầu hay tiêu chí mới.
- Để có thể đáp ứng các yêu cầu hay tiêu chí mới, từng thành phần mã hóa được chọn không nhất phải đạt ngưỡng tối đa đối với các tính chất mật mã. Ví dụ như

để thỏa mãn tính xoắn, S-box trong Anubis hoặc Khazad không đạt được giá trị tối đa về mức đồng nhất sai phân [69] như S-box trong Rijndael. Vấn đề chính là thành phần mã hóa được chọn, khi kết hợp với các thành phần khác, qua nhiều chu kỳ mã hóa, đảm bảo tính an toàn cho hệ mã đối với các phương pháp phân tích mã.

- Có nhiều cách khác nhau để tạo ra mỗi thành phần trong quy trình mã hóa/giải mã mà vẫn đảm bảo vai trò và tính chất của thành phần này trong hệ mã. Điều này cho phép tạo ra các biến thể của cùng một thuật toán gốc mà vẫn đảm bảo tính an toàn của thuật toán gốc.

Cùng với các thuật toán tiền thân của Rijndael, bao gồm Shark [90] và Square [18], các thuật toán tựa-Rijndael như GrandCru, Khazad và Anubis cũng là thành viên của họ thuật toán mã hóa tựa-Rijndael. Các thuật toán cụ thể được tạo ra từ giải thuật XAES cũng là thành viên của họ thuật toán này. Tập hợp các thể hiện của XAES là một tập con (vô hạn) của họ thuật toán mã hóa tựa-Rijndael.

1.2.2 Các mở rộng của AES

Cấu trúc của AES được các chuyên gia trong lĩnh vực mật mã rất quan tâm và một số mở rộng của AES đã được đề xuất. Tuy nhiên, hầu hết các công trình này đều xuất phát từ góc độ phân tích mã: các phiên bản mở rộng được đề xuất nhằm phục vụ việc tìm hiểu các tính chất bên trong cấu trúc AES với hi vọng khai thác các tính chất này trong việc tấn công AES:

- Trong thuật toán BES [68] do Murphy và Robshow đề xuất, tất cả các thao tác mã hóa đều được thực hiện trên $GF(2^8)$ nhằm đơn giản hóa việc khảo sát hoạt động của thuật toán (trong AES sử dụng kết hợp thao tác trên $GF(2^8)$ với thao tác trên $GF(2)^8$).
- Các mở rộng với kích thước nhỏ của AES được đề xuất trong [13] nhằm khảo sát kiến trúc tương tự AES ở kích thước nhỏ, hướng đến khả năng thể hiện dưới dạng tham số các tính chất của AES với hi vọng có thể áp dụng để tấn công AES.

- Monnerat và Vaudenay đề xuất hai mở rộng CES và Big-BES trong [66] nhằm phản bác việc đề xuất BES. Theo Monnerat và Vaudenay, các kết quả khảo sát dựa trên BES ít có ảnh hưởng đến việc tấn công AES.

Trong luận án này, chúng tôi cũng tập trung nghiên cứu cấu trúc của AES nhưng từ góc độ xây dựng một họ các thuật toán mã hóa khối tựa-Rijndael. Các thể hiện với kích thước nhỏ của XAES có thể được dùng trong thiết bị ubiquitous hay các thiết bị cảm ứng, đồng thời có thể được sử dụng trong việc khảo sát các tính chất của AES phục vụ phân tích mã. Các thể hiện với kích thước lớn có thể được dùng trong tương lai khi đòi hỏi về độ dài khóa tăng.

1.3 Từ AES đến XAES

Trong phần dưới đây, chúng tôi trình bày và phân tích thuật toán AES để nêu lên hướng tiếp cận của chúng tôi trong việc mở rộng AES thành XAES. Đầu tiên, chúng tôi trình bày cấu trúc và các biến đổi trong AES. Sau đó, chúng tôi phân tích và đề xuất giải pháp dựa trên AES để tạo ra thuật toán tham số hóa XAES. Phần đặc tả chi tiết của thuật toán XAES sẽ được trình bày chi tiết trong Chương 2.

1.3.1 Biểu diễn khối và khóa

AES là thuật toán xử lý trên byte, dữ liệu được xử lý theo từng nhóm gồm m =8 bit. Mỗi byte được xem là một phần tử của trường Galois GF (2^8) xác định bởi đa thức bất khả quy $\mu(x) = x^8 + x^4 + x^3 + x + 1$. Trường Galois được xác định bởi $\mu(x)$ còn được gọi là trường Galois của Rijndael [17]. Khi thay thế $\mu(x)$ bằng một đa thức bất khả quy khác sẽ tạo ra thuật toán đối ngẫu của Rijndael [4]. Hai thuật toán đối ngẫu hoàn toàn tương đương nhau về các tính chất mật mã [4][92].

Trong Rijndael, khối dữ liệu có kích thước 128, 192, hoặc 256 bit; khóa có kích thước 128, 192 hoặc 256 bit. Trong AES, NIST đã giới hạn lại kích thước khối là 128 bit. Trong phạm vi luận án này, thuật ngữ AES và Rijndael cùng được dùng để chỉ một thuật toán, trong trường hợp cần nhấn mạnh sự khác biệt giữa AES và Rijndael, chúng tôi sẽ ghi chú rõ trong nội dung trình bày.

Mỗi khối dữ liệu được biểu diễn bằng ma trận 4 × *Nb* byte với *Nb* = 4, 6 hay 8. Mỗi vector gồm 4 byte được xem là một từ, do đó, mỗi khối có thể được xem là một vector gồm *Nb* từ. Khóa chính cũng được biểu diễn bằng ma trận 4 × *Nk* byte hoặc một vector gồm *Nk* từ với *Nk* =4, 6 hay 8.

Để tham số hóa cấu trúc cho XAES, chúng tôi kế thừa ý tưởng sử dụng tham số m trong chiến lược vết rộng [19] để thể hiện số lượng bit cho mỗi nhóm dữ liệu được xử lý, đồng thời, chúng tôi đề xuất thêm tham số Nw là số nhóm (m bit) trong mỗi từ. Như vậy, tương ứng với hai đơn vị dữ liệu cơ bản trong AES là byte (nhóm 8 bit) và từ (vector gồm 4 byte), chúng tôi sử dụng hai tham số cấu trúc cho XAES như sau:

- số lượng bit trong mỗi nhóm, ký hiệu là m,
- số lượng nhóm (m bit) trong mỗi từ, ký hiệu là Nw.

Tham số m cho phép XAES tương thích với các hệ thống không sử dụng đơn vị dữ liệu byte, ví dụ như trong các thiết bị cảm ứng 4-bit. Trong ứng dụng thực tế nên chọn giá trị $m \geq 4$ để thuật toán có thể đạt độ an toàn đối với tấn công sai phân và tuyến tính (xem phần 3.2). Tham số Nw cho phép định nghĩa các thuật toán mã hóa với kích thước khối và khóa lớn không giới hạn, đồng thời khai thác đặc điểm của các kiến trúc xử lý khác nhau, ví dụ như bộ xử lý 64 bit có thể hỗ trợ Nw = 8 (với m = 8). Chi tiết về cấu trúc của XAES sẽ được trình bày trong phần 2.1.

Trong phạm vi luận án này, chúng tôi sử dụng ký hiệu $\{xy\}$ để biểu diễn giá trị ở dạng thập lục phân trên trường Galois GF(2^8).

1.3.2 Thuật toán mã hóa

Trong AES, quy trình mã hóa gồm $Nr = \max\{Nb, Nk\} + 6$ chu kỳ mã hóa. Các phép biến đổi trong mỗi chu kỳ mã hóa là sự hiện thực hóa các phép biến đổi trừu tượng đã được đề xuất trong "chiến lược vết rộng" (xem phần 1.1.2):

- Biến đổi φ trở thành SubBytes (xem phần 1.3.3)
- Biến đổi λ được xây dựng bằng cách kết hợp 2 biến đổi tuyến tính: ShiftRows, ký hiệu là π (xem phần 1.3.4), và MixColumns, ký hiệu là θ (xem phần 1.3.5).
- Biến đổi σ trở thành AddRoundKey (xem phần 1.3.6).

Quy trình mã hóa AES gồm:

- Thực hiện thao tác cộng khóa đầu tiên $\sigma[k^0]$
- Thực hiện $Nr-1$ chu kỳ mã hóa sử dụng cùng thủ tục mã hóa, ký hiệu là ζ. Mỗi chu kỳ mã hóa gồm có 4 thao tác biến đổi: SubBytes, ShiftRows, MixColumns và AddRoundKey

$$\zeta[k^r] = \sigma[k^r] \circ \theta \circ \pi \circ \varphi \quad \text{với } 1 \le r < Nr \tag{1.3}$$

  với k^r là khóa của chu kỳ thứ r ($1 \le r < Nr$).

- Thực hiện chu kỳ mã hóa cuối cùng. Trong chu kỳ này bỏ qua thao tác MixColumns:

$$\zeta[k^{Nr}] = \sigma[k^{Nr}] \circ \pi \circ \varphi \tag{1.4}$$

Như vậy, thuật toán mã hóa AES với khóa chính k được biểu diễn như sau:

$$\text{AES}[k] = \sigma[k^{Nr}] \circ \pi \circ \varphi \circ \zeta^{Nr-1}[k^{Nr-1}] \circ \ldots \circ \zeta^2[k^2] \circ \zeta^1[k^1] \circ \sigma[k^0] \tag{1.5}$$

1.3.3 Biến đổi SubBytes trong AES

Mỗi byte y được thay thế sử dụng bảng thay thế (cố định) S-box được xác định như sau (xem Hình 2.3):

- Lấy nghịch đảo $z = y^{-1} \in \text{GF}(2^8)$ với quy ước $0^{-1} = 0$.
- Cho $(z_0, z_1, \ldots, z_7)$ là biểu diễn nhị phân của z. Thực hiện ánh xạ affine trên trường $\text{GF}(2^8)$ với biểu diễn nhị phân z. Kết quả $t = (t_0, t_1, \ldots, t_7)$ được xác định như sau :

$$\begin{pmatrix} t_0 \\ t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \\ t_7 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} z_0 \\ z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \\ z_7 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} \tag{1.6}$$

Thành phần chính của S-box trong AES là ***ánh xạ nghịch đảo*** trên trường $GF(2^m)$ với m = 8. Ý tưởng về việc xây dựng S-box bằng ánh xạ nghịch đảo trên $GF(2^m)$ được K. Nyberg đề xuất trong [69]. Sử dụng ánh xạ nghịch đảo, cả hai tính chất mật mã quan trọng của S-box là ***chặn trên tối thiểu của tương quan đầu vào – đầu ra*** [20] và ***chặn trên tối thiểu của lan truyền sai phân*** [20] đều đạt giá trị ngưỡng tối ưu (về mặt lý thuyết). Điều này giúp S-box đạt được tính an toàn tối ưu đối với phương pháp phân tích mã sai phân [6] và phương pháp phân tích mã tuyến tính [62]. Vì vậy, chúng tôi cũng chọn ánh xạ nghịch đảo để xây dựng S-box cho giải thuật XAES (xem phần 2.2.1). Cần lưu ý là trong XAES, ánh xạ nghịch đảo được định nghĩa trên $GF(2^m)$ thay vì $GF(2^8)$ trong trường hợp AES.

Trong AES, ánh xạ affine trên $GF(2)^8$ được sử dụng làm bước hậu xử lý nhằm loại bỏ các điểm bất biến ($0 \rightarrow 0$, $1 \rightarrow 1$) trong ánh xạ nghịch đảo. Trong XAES sử dụng $m \times m$ S-box (S-box với m bit đầu vào và m bit đầu ra), chúng tôi thay thế ánh xạ affine trong AES bằng ánh xạ affine bất kỳ trên $GF(2)^m$.

Bên cạnh những ưu điểm, S-box trong AES có một tính chất không mong muốn là tính đơn giản trong biểu diễn đại số trên trường $GF(2^8)$ [90]. Biểu diễn đại số của S-box trong AES chỉ gồm 9 đơn thức khác 0 và điều này có khả năng dẫn đến việc tấn công đại số [14][27] hay tấn công nội suy [41]. Do đó, đối với XAES, chúng tôi đề xuất kiến trúc xây dựng S-box bằng cách bổ sung thêm một ánh xạ affine trên trường $GF(2)^m$ làm bước tiền xử lý trước khi thực hiện ánh xạ nghịch đảo nhằm nâng cao độ phức tạp đại số [28] đối với các phương pháp phân tích mã hiện nay (xem phần 2.2.1-Biến đổi SubBytes trong XAES).

Chúng tôi đã chứng minh trong phần 3.2.3 rằng với mọi cặp ánh xạ affine trên $GF(2)^m$, S-box trong XAES đều bảo toàn các tính chất mật mã tối ưu của ánh xạ nghịch đảo, bao gồm ***chặn trên tối thiểu của tương quan đầu vào – đầu ra*** [17] và ***chặn trên tối thiểu của lan truyền sai phân***[17]. Vì vậy, trong phần 2.2.1-Biến đổi SubBytes trong XAES, chúng tôi đề xuất việc tham số hóa hai ánh xạ affine này cho biến đổi SubBytes của XAES.

1.3.4 Biến đổi ShiftRows trong AES

Trong biến đổi ShiftRows (xem Hình 2.4), mỗi byte trên dòng i được quay trái i vị trí với i =0,1,2,3[1]. Tính chất chính của thao tác này là tất cả các byte trên mỗi cột được phân tán đến càng nhiều cột khác nhau càng tốt, và mỗi cột kết quả đều nhận được các byte từ càng nhiều cột khác nhau càng tốt. Từ đó suy ra giá trị offset của mỗi dòng phải phân biệt. Để đảm bảo điều này, số cột trong khối dữ liệu phải nhiều hơn hay bằng số dòng. Chính vì vậy, chúng tôi đã đề nghị ràng buộc này trong kích thước của khối dữ liệu trong XAES (xem phần 2.1.1-Biểu diễn khối và khóa). Khi đó, danh sách các giá trị offset cho mỗi dòng có thể được xem là tham số của biến đổi ShiftRows trong XAES và giá trị offset cho mỗi dòng được chọn ngẫu nhiên sao cho không trùng nhau từ tập chỉ số cột (xem phần 2.2.2-Biến đổi ShiftRows trong XAES).

1.3.5 Biến đổi MixColumns trong AES

Trước khi trình bày về biến đổi MixColumns trong AES, chúng ta cần một số định nghĩa sau:

Định nghĩa 1.1 [16]: *Branch Number* $\mathcal{B}$ *của biến đổi tuyến tính* θ *được định nghĩa như sau:*

$$\mathcal{B}(\theta) = \min_{a \neq \vec{0}} \{wt(a) + wt(\theta(a))\} \tag{1.7}$$

với wt là trọng số Hamming của một vector (được xác định bằng số lượng thành phần khác 0 trong vector)

Nhận xét [16]: nếu θ được định nghĩa trên không gian n chiều thì $\mathcal{B}(\theta)$ bị chặn trên bởi $n + 1$.

Định nghĩa 1.2 [22]: *Ma trận luân hoàn là ma trận vuông, trong đó, mỗi dòng thứ* $i > 0$ *được xây dựng bằng hoán vị vòng quanh các phần tử của dòng thứ* $i - 1$ *sang phải* 1 *vị trí.*

---

[1] Trong thuật toán Rijndael với *Nb* = 8, các byte trên dòng 2 và dòng 3 lần lượt được quay trái 3 và 4 vị trí.

Cho ma trận luân hoàn $\mathcal{M}$ (kích thước $n \times n$) với $\mathcal{M}_{i,j}$ là phần tử tại dòng i, cột j. Ta có:

$$\mathcal{M}_{i,j} = \mathcal{M}_{0,(j-i) \bmod n} \quad , \quad 1 \le i < n, 0 \le j < n \tag{1.8}$$

❑ Ví dụ: $\begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 2 & 3 & 1 \end{pmatrix}$ là 1 ma trận luân hoàn.

Biến đổi MixColumns trong AES sử dụng ánh xạ tuyến tính trên GF(2^8)^4 tương ứng với ma trận luân hoàn – ma trận vuông trong đó mỗi dòng được tạo bằng cách xoay giá trị dòng liền trên sang phải một vị trí.

Hình 2.5 minh họa biến đổi MixColumns. Mỗi cột trong khối dữ liệu được trộn với cùng một biến đổi tuyến tính trên GF(2^8)^4. Mỗi cột được biểu diễn bằng một ma trận với các hệ số trên trường GF(2^8) và nhân (modulo $x^4 + 1$) với 1 đa thức cố định:

$$c(x) = 3x^3 + x^2 + x + 2 \tag{1.9}$$

Biến đổi MixColumns trong AES sử dụng ánh xạ tuyến tính trên GF(2^8)^4 có Branch Number đạt được giá trị chặn trên tối ưu (trên lý thuyết) bằng 5. Việc chứng minh tính an toàn của AES trong [16][17] chỉ sử dụng tính chất này của đa thức $c(x)$, độc lập với giá trị cụ thể của các hệ số trong $c(x)$. Vì vậy, mọi đa thức tương ứng với branch number là 5 đều có thể dùng để thay thế đa thức $c(x)$ được chọn trong biến đổi MixColumns của AES. Điều này dẫn đến khả năng tạo ra các biến thể của AES bằng cách thay thế đa thức $c(x)$ bằng đa thức khác tương ứng với branch number là 5.

Đối với XAES, mỗi cột gồm Nw phần tử trên trường GF(2^m). Tương tự AES, mỗi cột trong khối của XAES được biểu diễn dưới dạng một đa thức bậc $Nw - 1$ có các hệ số trên trường GF(2^m). Đa thức này được nhân (module $x^m + 1$) với đa thức có bậc Nw–1 có các hệ số trên trường GF(2^m). Tuy nhiên, đối với XAES, trong phần 3.2, chúng tôi chứng minh rằng để đảm bảo tính an toàn của XAES đối với phương pháp phân tích sai phân và tuyến tính, branch number của ánh xạ tuyến tính được dùng không cần phải đạt ngưỡng tối đa (trên lý thuyết) là $Nw + 1$ mà chỉ cần đạt giá trị Nw.

Điều này giúp mở rộng khả năng chọn lựa các đa thức an toàn cho biến đổi MixColumns trong XAES.

Ngoài ra, do mỗi cột của khối trong biến đổi MixColumns được xử lý riêng nên có thể chọn sử dụng đa thức khác nhau cho từng cột. Chính vì vậy, chúng tôi đề nghị tham số hóa danh sách các đa thức được dùng để biến đổi từng cột trong MixColumns của XAES (xem phần 2.2.3-Biến đổi MixColumns trong XAES).

1.3.6 Biến đổi AddRoundKey và hàm sinh khóa KeySchedule trong AES

Từ khóa chính k cho trước, thuật toán AES phát sinh dãy gồm $Nr + 1$ khóa cho mỗi chu kỳ, mỗi khóa của chu kỳ gồm Nb từ. Trong cho kỳ thứ r, mỗi byte trong trạng thái hiện hành được XOr với byte tương ứng trong khóa k^r của chu kỳ. Do đó, cần phát sinh mảng khóa mở rộng gồm $(Nr+1)\times Nb$ từ với chất liệu đầu vào là Nk từ của khóa k cho trước.

Từ khóa k, hàm sinh khóa KeySchedule của AES lần lượt phát sinh và lưu lại các vector $H^{(i)}$ gồm Nk từ vào mảng khóa mở rộng cho đến khi có đủ $(Nr+1) \times Nb$ từ.

Chúng tôi đã tổng quát hóa hàm phát sinh khóa của AES để xây dựng hàm phát sinh khóa trong XAES: mỗi phần tử trong XAES được xử lý là nhóm gồm m bit (thay vì một byte gồm 8 bit như trong AES) và mỗi từ gồm Nw phần tử m bit (thay vì mỗi từ gồm 4 byte như trong AES). Trong phần này, chúng tôi không trình bày chi tiết về hàm sinh khóa của AES. Hàm sinh khóa của AES hoàn toàn có thể được suy ra từ hàm sinh khóa của XAES (trình bày trong phần 2.2.5-Hàm phát sinh khóa trong XAES) bằng cách thay thế giá trị m =8 và Nw =4.

1.4 Kết luận

Trong chương 2, chúng tôi đã phân tích mối liên hệ giữa kiến trúc thuật toán mã hóa khối và các thuật toán mã hóa cụ thể, từ đó đề xuất ý tưởng về việc xây dựng thuật toán mã hóa khối được tham số hóa làm bước chuyển tiếp giữa kiến trúc thuật toán ở mức trừu tượng với các thuật toán mã hóa cụ thể. Mỗi thuật toán mã hóa khối

được tham số hóa xác định một ***lớp các thuật toán mã hóa khối*** có ***cùng kiến trúc*** và ***chiến lược xây dựng các thành phần mã hóa***.

Việc tham số hóa thuật toán Rijndael đã được các tác giả của thuật toán này đề xuất trong đặc tả của thuật toán Rijndael [16] và được nhắc lại trong phần 6.2 của tài liệu FIPS 197 [30] của NIST công bố chuẩn AES, bao gồm 3 tham số: Nk là số từ trong khóa, Nb là số từ trong khối và Nr là số chu kỳ mã hóa. Tuy nhiên, việc tham số hóa này được xây dựng trong ngữ cảnh bị giới hạn: cần phải giữ cố định số lượng bit trong mỗi phần tử dữ liệu được xử lý là 8 và số lượng phần tử trong mỗi từ là 4. Trong XAES, chúng tôi đã tiếp tục tham số hóa thuật toán bằng việc đề xuất bổ sung 2 tham số cấu trúc là số bit trong mỗi đơn vị dữ liệu được xử lý (ký hiệu là m) và số phần tử trong mỗi từ (ký hiệu là Nw). Hai tham số này cho phép thay đổi hoàn toàn cách xây dựng các thành phần mã hóa trong XAES và cho phép mở rộng không giới hạn kích thước khối và kích thước khóa. Ngoài ra, chúng tôi còn đề xuất việc tham số hóa các hệ số trong mỗi thành phần mã hóa để tạo ra một lớp các thuật toán mã hóa khối có cùng kiến trúc và chiến lược xây dựng các thành phần mã hóa.

Trong các chương tiếp theo, chúng tôi sẽ trình bày về thuật toán XAES, một thuật toán mã hóa được tham số hóa được xây dựng theo kiến trúc SPN. Do các thành phần mã hóa trong Rijndael đã được nghiên cứu kỹ trong những năm gần đây, các tính chất mật mã quan trọng của Rijndael đã được khảo sát chi tiết, chúng tôi quyết định chọn phương án tổng quát hóa các thành phần trong Rijndael để xây dựng các thành phần trong XAES.

Thông qua việc trình bày và phân tích giải thuật AES, chúng tôi đã đề xuất 2 tham số cấu trúc cho XAES, gồm số lượng bit trong mỗi nhóm dữ liệu được xử lý (ký hiệu là m) và số lượng nhóm (m bit) trong mỗi từ (ký hiệu là Nw). Đồng thời, trên cơ sở trình bày và phân tích vai trò của từng biến đổi trong AES, chúng tôi đã đề xuất các tham số xử lý cho từng biến đổi trong XAES. Chi tiết về các tham số xử lý được trình bày trong Chương 2.

Chương 2

XAES - Thuật toán mã hóa khối được tham số hóa

Tóm tắt chương:

✍ *Nội dung của chương 2 trình bày thuật toán mã hóa được tham số hóa XAES:*

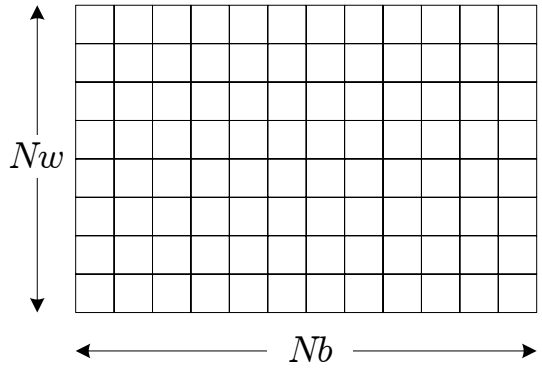
- *Trình bày cấu trúc thuật toán và chi tiết về từng biến đổi (được tham số hóa) trong XAES. Với mỗi biến đổi, chúng tôi trình bày rõ ý nghĩa các tham số xử lý được đề xuất cùng với cách xây dựng cụ thể các tham số này.*
- *Khảo sát mối quan hệ giữa độ an toàn (tính theo số bit của khóa và khối). tốc độ xử lý của các thể hiện của XAES theo các tham số cấu trúc.*

2.1 Cấu trúc thuật toán XAES

2.1.1 Biểu diễn khối và khóa

Mỗi khối dữ liệu được biểu diễn dưới dạng ma trận gồm $Nb \times Nw$ phần tử m-bit với $Nw \le Nb \le 2Nw$ (xem Hình 2.1). Điều kiện chặn dưới của Nb nhằm đảm bảo tính khuếch tán tối đa của biến đổi ShiftRows (đã phân tích trong phần 1.3.4), điều kiện chặn trên của Nb được dùng trong quá trình chứng minh tổng quát tính an toàn của XAES đối với phương pháp phân tích mã sai phân [6] và phương pháp phân tích mã tuyến tính [62]. Tương tự, khóa chính cũng được biểu diễn dưới dạng ma trận $Nw \times Nk$ phần tử m-bit với $Nw \le Nk \le 2Nw$. Ma trận biểu diễn một khối hay khóa có thể được xử lý dưới dạng mảng 1 chiều các từ, mỗi từ (gồm Nw phần tử m-bit) tương ứng với 1 cột của ma trận.

Vậy, giải thuật XAES xử lý khối dữ liệu có kích thước $m \times Nw \times Nb$ bit sử dụng khóa có kích thước $m \times Nw \times Nk$ bit.



Hình 2.1. Khối dữ liệu trong XAES gồm *Nw* dòng và *Nb* cột

Trong Chương 3, chúng tôi chứng minh được rằng:

- Với 8 chu kỳ mã hóa đủ đảm bảo tính an toàn cho XAES đối với phương pháp sai phân và phương pháp tuyến tính trong trường hợp $Nb = Nw = Nk$,
- Khi tăng cường thêm 4 chu kỳ mã hóa nữa thì đảm bảo tính an toàn cho XAES trong trường hợp $\max\{Nb, Nk\} = 2Nw$ và $m > 4$
- Khi tăng cường thêm 8 chu kỳ mã hóa nữa thì đảm bảo tính an toàn cho XAES trong trường hợp $\max\{Nb, Nk\} = 2Nw$ và $m = 4$

Trong các thuật toán mã hóa khối hiện nay thường tăng cường thêm từ 1 đến 2 chu kỳ mã hóa gọi là biên an toàn [49]. Vì thế, chúng tôi cũng chọn bổ sung thêm 2 chu kỳ mã hóa để tạo biên an toàn cho XAES.

Đặt:

$$\eta = \max\{Nb/Nw, Nk/Nw\} \tag{2.1}$$

Số chu kỳ mã hóa (Nr) của XAES được xác định như sau:

$$Nr = \begin{cases} 8 + 2\lceil 2\eta \rceil, \text{ khi } m > 4 \\ 2 + 4\lceil 2\eta \rceil, \text{ khi } m = 4 \end{cases} \tag{2.2}$$

Bảng 2.1 và Bảng 2.2 lần lượt thể hiện một số ví dụ về số lượng chu kỳ mã hóa trong XAES trong trường hợp $m = 4$ và $m > 4$.

Bảng 2.1. Một số ví dụ về số lượng chu kỳ mã hóa trong XAES ($m = 4$)

Nw	max $\{Nb, Nk\}$	Nr	Nw	max $\{Nb, Nk\}$	Nr
4	4	10	6	6	10
4	5, 6	14	6	7, 8, 9	14
4	7, 8	18	6	10, 11, 12	18
5	5	10	7	7	10
5	6, 7	14	7	8, 9, 10	14
5	8, 9, 10	18	7	11, 12, 13, 14	18

Bảng 2.2. Một số ví dụ về số lượng chu kỳ mã hóa trong XAES (m > 4)

Nw	max $\{Nb, Nk\}$	Nr
4	4	10
4	5, 6	12
4	7, 8	14
5	5	10
5	6, 7	12
5	8, 9, 10	14

Nw	max $\{Nb, Nk\}$	Nr
6	6	10
6	7, 8, 9	12
6	10, 11, 12	14
7	7	10
7	8, 9, 10	12
7	11, 12, 13, 14	14

2.1.2 Quy trình mã hóa

Để xây dựng XAES, chúng tôi chọn phương án tổng quát hóa cách mà J. Daemen và V. Rijmen đã sử dụng trong AES để hiện hóa các biến đổi mã hóa được đề xuất ở mức trừu tượng trong "chiến lược vết rộng". Vì vậy, trong quy trình mã hóa của XAES sử dụng 4 phép biến đổi như sau:

1. Biến đổi SubBytes, ký hiệu $\varphi_{\langle m,Nw\rangle}$, là phép thay thế phi tuyến 1 phần tử m-bit trong trạng thái hiện hành sử dụng bảng thay thế (S-box) cố định (xem phần 2.2.1)
2. Biến đổi ShiftRow , ký hiệu $\pi_{\langle m,Nw\rangle}$, thực hiện dịch chuyển xoay vòng từng dòng của trạng thái hiện hành với offset riêng cho từng dòng (xem phần 2.2.2)
3. Biến đổi MixColumns, ký hiệu $\theta_{\langle m,Nw\rangle}$, thực hiện trộn thông tin của từng cột trong trạng thái hiện hành. Mỗi cột được xử lý độc lập (xem phần 2.2.3).
4. Biến đổi AddRoundKey, ký hiệu $\sigma_{\langle m,Nw\rangle}$, thực hiện việc cộng ($\oplus$) khóa của chu kỳ vào trạng thái hiện hành (xem phần 2.2.4). Độ dài khóa của chu kỳ bằng với kích thước của trạng thái. Hàm phát sinh khóa KeySchedule trong XAES được trình bày trong phần 2.2.5.

Ký hiệu $\langle m, Nw\rangle$ nhấn mạnh việc toàn bộ các biến đổi trong XAES đều được tham số hóa theo 2 tham số cấu trúc là m và Nw. Để đơn giản trong việc trình bày, chúng tôi không ghi lại ký hiệu này trong các phần trình bày sau này.

Mỗi phép biến đổi thao tác trên trạng thái hiện hành a. Kết quả b của mỗi phép biến đổi sẽ trở thành đầu vào của biến đổi kế tiếp trong quy trình mã hóa.

Quy trình mã hóa XAES gồm các bước sau:

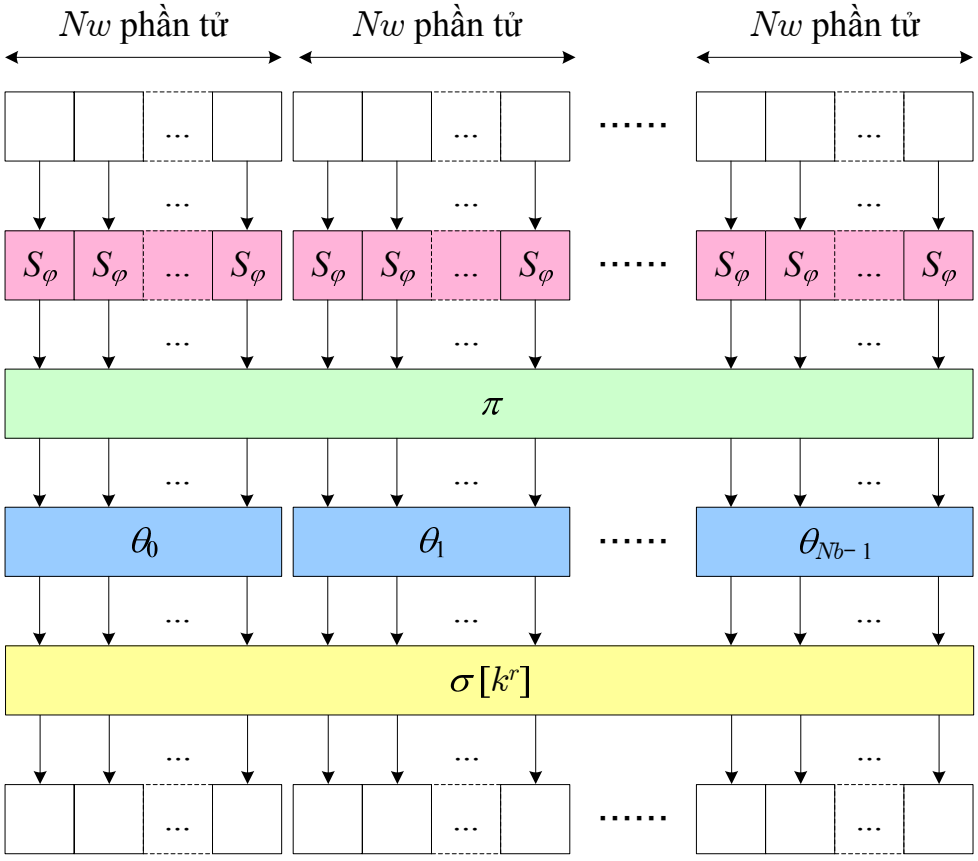
1. Thực hiện thao tác AddRoundKey trước khi thực hiện các chu kỳ mã hóa.
2. Nr–1 chu kỳ mã hóa bình thường: mỗi chu kỳ bao gồm 4 bước biến đổi liên tiếp: SubBytes, ShiftRows, MixColumns và AddRoundKey (xem Hình 2.2). Đặt:

$$\zeta[k^r] = \sigma[k^r] \circ \theta \circ \pi \circ \varphi,\ 1 \le r < Nr \qquad (2.3)$$

   với k^r là khóa của chu kỳ r được phát sinh từ khóa chính k.
3. Thực hiện chu kỳ mã hóa cuối cùng: gồm 3 bước biến đổi SubBytes, ShiftRows và AddRoundKey.

Như vậy, thuật toán mã hóa XAES với khóa k có thể biểu diễn dưới dạng tích của các biến đổi như sau:

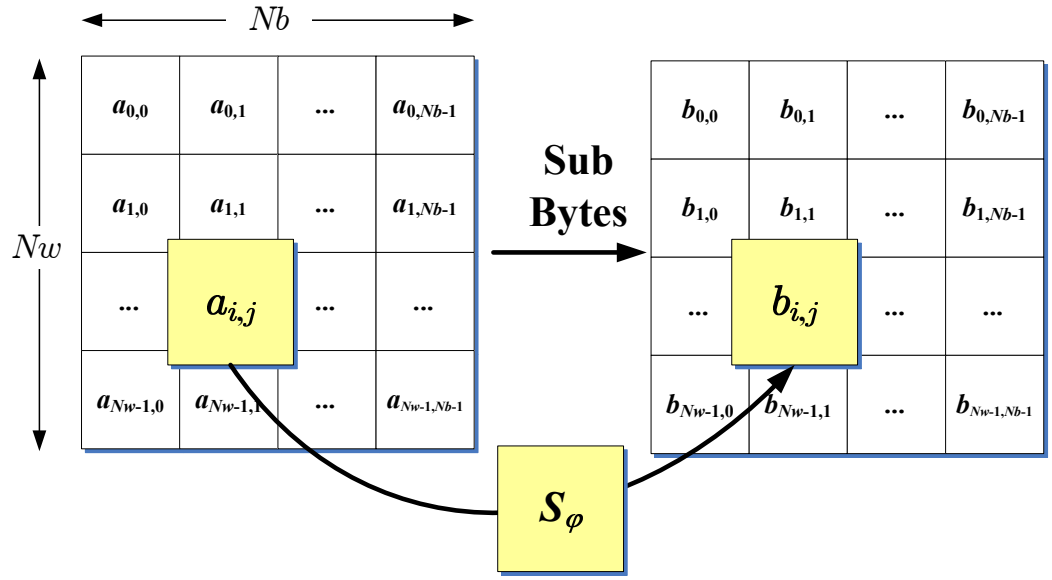
$$\text{XAES}[k] = \sigma[k^{Nr}] \circ \pi \circ \varphi \circ \zeta[k^{Nr-1}] \circ \zeta[k^{Nr-2}] \circ \ldots \circ \zeta[k^2] \circ \zeta[k^1] \circ \sigma[k^0] \qquad (2.4)$$



Hình 2.2. Một chu kỳ mã hóa thường của XAES

2.2 Các thành phần trong quy trình mã hóa của XAES

2.2.1 Biến đổi SubBytes trong XAES



Hình 2.3. Phép biến đổi SubBytes trong XAES.

Chú thích: Trong Rijndael, mỗi khối gồm $Nw = 4$ dòng và $Nb = 4$, 6 hay 8 cột, mỗi phần tử (gồm 8-bit) được xử lý như phần tử trên trường GF(2^8) và S_φ là 8×8 S-box. Trong XAES, mỗi phần tử (gồm m-bit) được xem là phần tử trên trường GF(2^m) và S_φ là $m \times m$ S-box.

Biến đổi SubBytes, ký hiệu là φ, có 2 tham số xử lý là 2 ánh xạ affine trên GF$(2)^m$ được dùng làm bước tiền xử lý và hậu xử lý trong bảng thay thế S-box:

- Ánh xạ affine $\mathcal{A}_\varphi^{(0)}$ trên GF$(2)^m$ được dùng làm bước tiền xử lý. Ánh xạ này là sự kết hợp giữa phép nhân với ma trận nhị phân $\mathcal{M}_\varphi^{(0)}$ kích thước $m \times m$ và cộng với vector nhị phân $c_\varphi^{(0)} = \left(c_0^{(0)}, c_1^{(0)}, \ldots, c_{m-1}^{(0)}\right) \in GF(2)^m \setminus \vec{0}$
- Ánh xạ affine $\mathcal{A}_\varphi^{(1)}$ trên GF$(2)^m$ được dùng làm bước hậu xử lý. Ánh xạ này là sự kết hợp giữa phép nhân với ma trận nhị phân $\mathcal{M}_\varphi^{(1)}$ kích thước $m \times m$ và cộng với vector nhị phân $c_\varphi^{(1)} = \left(c_0^{(1)}, c_1^{(1)}, \ldots, c_{m-1}^{(1)}\right) \in GF(2)^m \setminus \vec{0}$

Hình 2.3 minh họa phép biến đổi SubBytes. Đây là phép thay thế phi tuyến từng phần tử (m-bit) trong trạng thái hiện hành.

$$\varphi : b = \varphi(a) \Leftrightarrow b_{i,j} = S_\varphi\left(a_{i,j}\right) \text{ với } 0 \le i < Nw,\ 0 \le j < Nb \qquad (2.5)$$

với S_φ là bảng thay thế (S-box) m-bit khả nghịch.

❑ **Quá trình thay thế 1 phần tử x gồm m-bit dựa vào S-box bao gồm 3 bước:**

- ***Bước 1:*** Thực hiện biến đổi affine $\mathcal{A}_{\varphi}^{(0)}$ trên biểu diễn nhị phân của $x=(x_0,x_1,...,x_{m-1})$. Kết quả $y=(y_0,y_1,...,y_{m-1})$ của bước 1 được xác định như sau:

$$y^{\mathrm{T}} = \mathcal{M}_{\varphi}^{(0)}.x^{\mathrm{T}} \oplus \left(c_{\varphi}^{(0)}\right)^{\mathrm{T}} \tag{2.6}$$

- ***Bước 2***: Xác định phần tử nghịch đảo $z = y^{-1} \in \mathrm{GF}(2^m)$. Quy ước $0^{-1} = 0$.
- ***Bước 3***: Thực hiện biến đổi affine $\mathcal{A}_{\varphi}^{(1)}$ trên biểu diễn nhị phân của $z=(z_0,z_1,...,z_{m-1})$. Kết quả $t=(t_0,t_1,...,t_{m-1})$ được xác định như sau:

$$t^{\mathrm{T}} = \mathcal{M}_{\varphi}^{(1)}.\ z^{\mathrm{T}} \oplus \left(c_{\varphi}^{(1)}\right)^{\mathrm{T}} \tag{2.7}$$

❑ Ví dụ: Xét thuật toán Rijndael:

- Ở bước 1, sử dụng hằng số $c_{\varphi}^{(0)} = 0$ và $\mathcal{M}_{\varphi}^{(0)}$ là ma trận đơn vị I_8. Như vậy, bước 1 không làm thay đổi giá trị x, xem như quá trình xử lý của S-box chỉ gồm bước 2 và bước 3.
- Ở bước 3, sử dụng hằng số $c_{\varphi}^{(1)} = \{63\} \in \mathrm{GF}(2^8)$ và ma trận $\mathcal{M}_{\varphi}^{(1)}$ như sau:

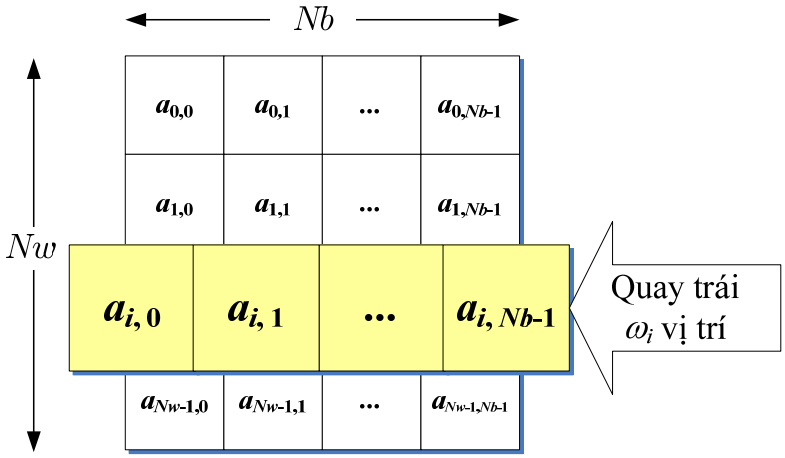
$$\mathcal{M}_{\varphi}^{(1)} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \tag{2.8}$$

❖ **Nhận xét:** Trong thuật toán Rijndael chỉ sử dụng bước 2 và bước 3 của S-box trong XAES. Chính vì vậy, biểu diễn đại số của S-box trong Rijndael chỉ gồm 9 đơn thức có hệ số khác 0 (phần phân tích sẽ được trình bày trong phần 6.2 - Biểu diễn đại số của S-box trong). Trong XAES, chúng tôi đề xuất kiến trúc của XAES ở mức tổng quát đủ để mô tả trường hợp S-box của Rijndael, đồng thời cung cấp khả năng để tạo ra S-box có độ phức tạp đại số cao hơn (xét ở

khía cạnh số lượng đơn thức có hệ số khác 0 trong biểu diễn đại số của S-box). Trong bước 1 của S-box trong XAES, chỉ cần sử dụng biến đổi affine là ánh xạ đồng nhất, ta sẽ có được S-box theo kiến trúc hoàn toàn giống với S-box trong Rijndael.

Phép biến đổi ngược (InvSubBytes), ký hiệu là φ^{-1}, sử dụng bảng thay thế nghịch đảo S_{φ}^{-1} để thay thế từng phần tử (m bit) trong trạng thái.

2.2.2 Biến đổi ShiftRows trong XAES



Hình 2.4. Phép biến đổi ShiftRows

Chú thích: Trong Rijndael, mỗi khối gồm $Nw = 4$ dòng và $Nb = 4$, 6 hay 8 cột, các phần tử trên dòng thứ i (i=0, 1, 2, 3) được quay vòng sang trái i vị trí với (riêng trường hợp $Nb = 8$, dòng 2 và dòng 3 lần lượt quay vòng sang trái 3 và 4 vị trí). Trong XAES, các phần tử trên dòng thứ i (i=0, 1, 2, …, $Nw - 1$) được quay vòng sang trái ω_i vị trí với $\omega_i \neq \omega_i$ khi $i \neq j$.

Biến đổi ShiftRows, ký hiệu là π, có các tham số xử lý là danh sách ω_π gồm giá trị độ dời cho mỗi dòng của khối:

$$\omega_\pi = (\omega_0, \omega_1, \ldots, \omega_{Nw-1}) \in \{0,1,2,\ldots,Nb-1\}^{Nw} \text{ thỏa } \omega_i \neq \omega_j \text{ với } i \neq j \qquad (2.9)$$

Như vậy, ω_π là vector gồm Nw phần tử có giá trị phân biệt, mỗi phần tử nhận giá trị trong đoạn [0, $Nb - 1$]. ω_i là độ dời khi quay dòng thứ i của trạng thái sang trái ($0 \le i < Nw$).

- **Nhận xét:** nhờ điều kiện $Nw \le Nb$ nên đảm bảo có thể chọn danh sách ω_π gồm Nw phần tử có giá trị phân biệt trong đoạn [0, $Nb - 1$].

Hình 2.4 minh họa thao tác ShiftRows: mỗi dòng thứ *i* của trạng thái hiện hành được dịch chuyển quay vòng sang trái với độ dời ω_i khác nhau ($0 \le i < Nw$).

Phần tử $a_{i,j}$ tại dòng *i* cột *j* sẽ dịch chuyển đến cột $(j - \omega_i) \bmod Nb$

$$\pi:\ b = \pi(a) \Leftrightarrow a_{i,j} = b_{i,(j-\omega_i) \bmod Nb} \text{ với } 0 \le i < Nw \text{ và } 0 \le j < Nb \qquad (2.10)$$

Phép biến đổi ngược (InvShiftRows), ký hiệu là π^{-1}, thực hiện việc dịch chuyển quay vòng sang phải các phần tử trên dòng thứ *i* của trạng thái hiện hành với độ dời ω_i ($0 \le i < Nw$).

$$\pi^{-1}:\ = \pi^{-1}(b) \Leftrightarrow b_{i,j} = a_{i,(j+\omega_i) \bmod Nb} \text{ với } 0 \le i < Nw,\ 0 \le j < Nb \qquad (2.11)$$

❑ Ví dụ:

- Trong thuật toán Rijndael, với $Nb = 4$, danh sách $\omega_\pi = (0,1,2,3)$
- Thông thường, có thể chọn $\omega_\pi = (0,1,2,\ldots,Nw-1)$.

2.2.3 Biến đổi MixColumns trong XAES

Mọi ánh xạ tuyến tính $\mathcal{F}:(\mathrm{GF}(2^m))^{Nw} \to (\mathrm{GF}(2^m))^{Nw}$ đều có thể được biểu diễn bằng ma trận $\mathcal{M}^{\mathcal{F}}$ gồm *Nw* dòng và *Nw* cột với các phần tử thuộc $\mathrm{GF}(2^m)$. Nếu $\mathcal{F}$ có biểu diễn bằng ma trận luân hoàn thì $\mathcal{F}$ có thể được xác định dựa vào *Nw* hằng số, tương ứng với 1 dòng trong ma trận $\mathcal{M}^{\mathcal{F}}$, thay vì phải cần đến toàn bộ Nw^2 hằng số trong $\mathcal{M}^{\mathcal{F}}$.

Trong biến đổi MixColumns của XAES sử dụng các ánh xạ tuyến tính dạng này. Đặt $\Phi_{m,Nw}$ là tập hợp các ánh xạ tuyến tính $\mathcal{F}:(\mathrm{GF}(2^m))^{Nw} \to (\mathrm{GF}(2^m))^{Nw}$ có biểu diễn bằng ma trận luân hoàn.

Xác định ảnh qua ánh xạ tuyến tính có biểu diễn bằng ma trận luân hoàn:

Cho biến đổi tuyến tính $\mathcal{F} \in \Phi_{m,Nw}$ và $a = (a_0, a_1, \ldots, a_{Nw-1}) \in (GF(2^m))^{Nw}$. Gọi $b = (b_0, b_1, \ldots, b_{Nw-1}) \in (GF(2^m))^{Nw}$ là ảnh của a qua $\mathcal{F}$. Ta có :

$$b^{\mathrm{T}} = \mathcal{M}^{\mathcal{F}} . a^{\mathrm{T}} \qquad (2.12)$$

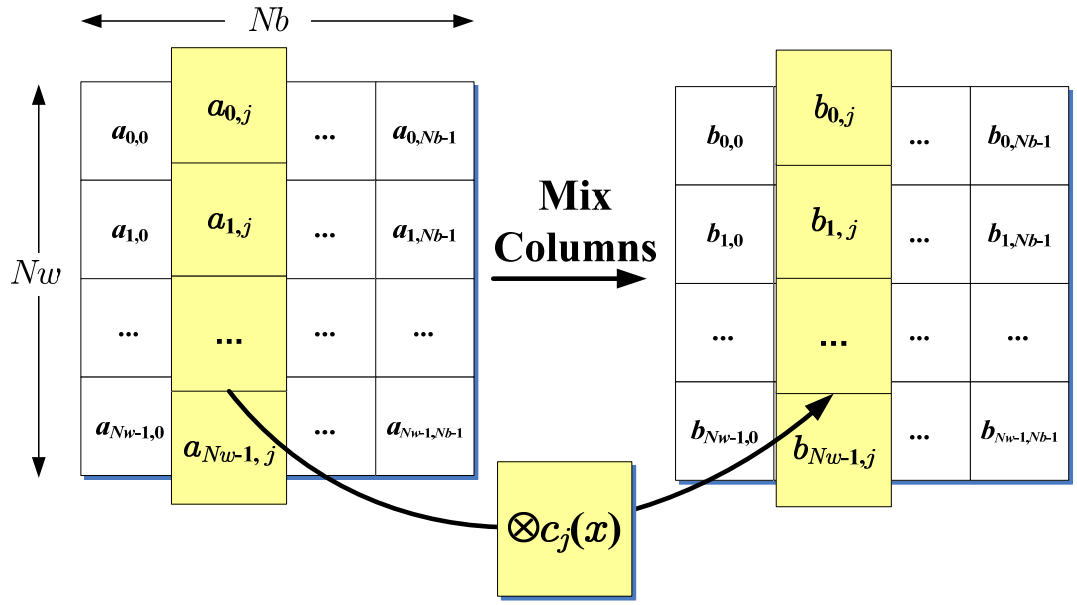
Do $\mathcal{F}$ có thể biểu diễn bằng ma trận luân hoàn, ta có thể thực hiện việc xác định ảnh của vector a qua $\mathcal{F}$ theo cách sau:

- Đặt $a(x) = \sum_{i=0}^{Nw-1} a_i x^i$ và $b(x) = \sum_{i=0}^{Nw-1} b_i x^i$
- Đặt $c(x) = \sum_{i=0}^{Nw-1} c_i x^i$ với c_i là giá trị cột $Nb–1–i$ trên dòng cuối ($Nw–$ 1) của $\mathcal{M}^{\mathcal{F}}$

  ❑ Ví dụ: $\mathcal{M}^{\mathcal{F}} = \begin{pmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{pmatrix}$ tương ứng với $c(x) = 3x^3 + x^2 + x + 2$

- Khi đó, ta có:

$$b(x) = c(x) \otimes a_j(x) = c(x) \times a_j(x) \bmod \left(1 \oplus x^{Nw}\right) \qquad (2.13)$$



Hình 2.5. Phép biến đổi MixColumns

Chú thích: Trong Rijndael, mỗi khối gồm $Nw = 4$ dòng và $Nb = 4$, 6 hay 8 cột; mỗi cột được biểu diễn dưới dạng vector trên $GF(2^8)^4$ và sử dụng chung một đa thức cố định $c(x) = 3x^3 + x^2 + x + 2$ để biến đổi từng cột. Trong XAES, mỗi cột được biểu diễn dưới dạng vector trên $GF(2^m)^{Nw}$ và sử dụng riêng từng đa thức $c_j(x)$ có bậc $Nw - 1$ có các hệ số trên $GF(2^m)$ để biến đổi từng cột j.

Biến đổi MixColumns, ký hiệu là θ, có tham số là danh sách Θ_θ các ánh xạ tuyến tính $\theta_j \in \Phi_{m,Nw}$ được sử dụng để biến đổi từng cột j của trạng thái:

$$\Theta_\theta = (\theta_0, \theta_1, ..., \theta_{Nb-1}) \text{ với } \theta_j \in \Phi_{m,Nw} \text{ thỏa } \mathcal{B}(\theta_j) \geq Nw,\ 0 \leq j < Nb \qquad (2.14)$$

Như vậy, MixColumns sử dụng các biến đổi θ_j có branch number là Nw hay $Nw + 1$. Điều kiện này đảm bảo cho việc chứng minh tính an toàn của XAES đối với phương pháp phân tích mã sai phân [6] và phân tích mã tuyến tính [62] (xin xem 3.2.3-Tỷ lệ truyền của vết sai phân trong XAES). Một số giải thuật để tăng cường tính hiệu quả khi xây dựng các ánh xạ θ_j sẽ được trình bày trong Chương 5.

Hình 2.5 minh họa phép biến đổi MixColumns. Trong biến đổi MixColumns, mỗi cột a_j và b_j của trạng thái trước và sau xử lý được xem là vector gồm Nw phần tử m-bit và b_j chính là ảnh của a_j qua ánh xạ tuyến tính θ_j.

Gọi $c_j(x)$ là đa thức bậc tối đa là $Nw - 1$ có các hệ số trên GF(2^m) tương ứng với ánh xạ tuyến tính θ_j.

Đặt $a_j(x) = \sum_{i=0}^{Nw-1} a_{i,j} x^i$. Khi đó, phép biến đổi MixColumns có thể được biểu diễn lại dưới dạng sau:

$\theta: b = \theta(a)$

$$\Leftrightarrow b_j(x) = c_j(x) \otimes a_j(x) = c_j(x) \times a_j(x) \bmod (1 \oplus x^{Nw}) \text{ với } 0 \leq j < Nb \qquad (2.15)$$

Biến đổi ngược (InvMixColumns), ký hiệu là θ^{-1} , sử dụng danh sách ánh xạ tuyến tính $\Theta_\theta^{-1} = (\theta_0^{-1}, \theta_1^{-1}, ..., \theta_{Nb-1}^{-1})$. Gọi $d_j(x)$ là đa thức có bậc tối đa là $Nw - 1$ có các hệ số trên GF(2^m) tương ứng với ánh xạ θ_j^{-1}. Ta có:

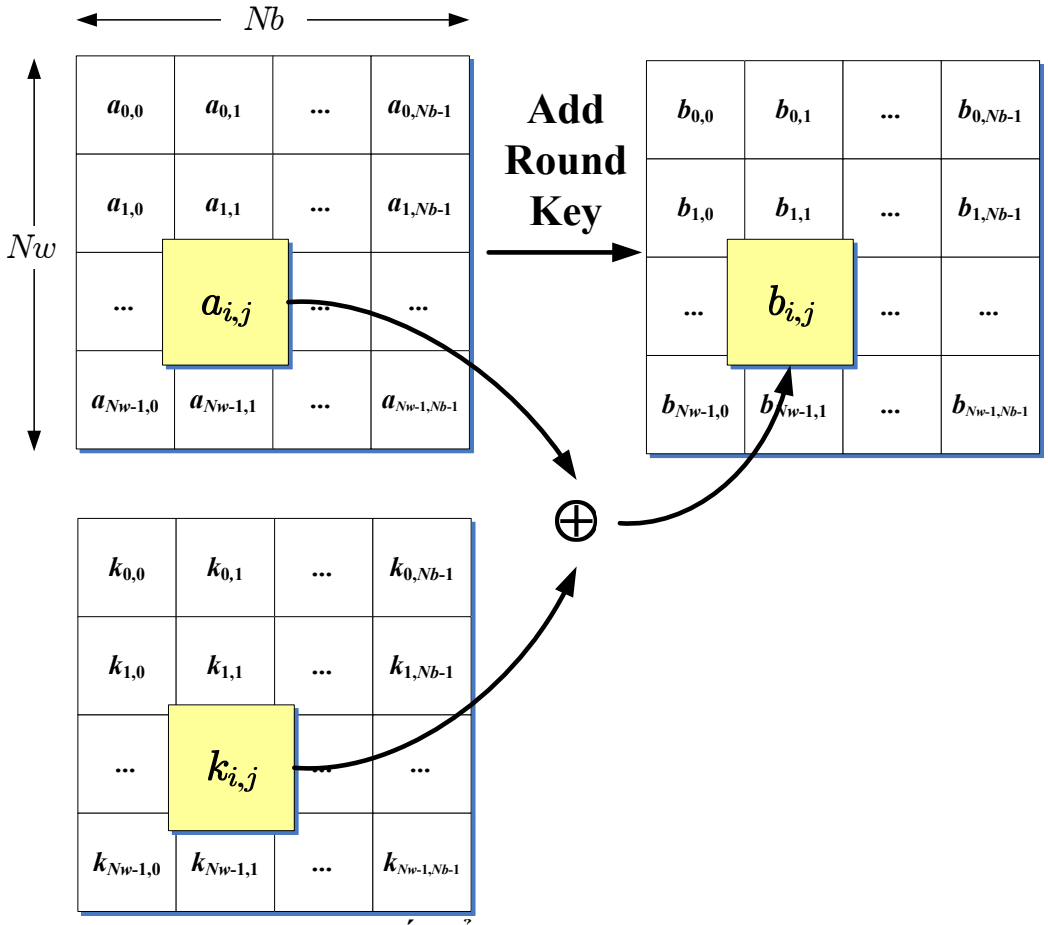
$$d_j(x) \times c_j(x) = 1 \mod (1 \oplus x^{Nw}) \text{ với } 0 \leq j < Nb \qquad (2.16)$$

Tương tự MixColumns, trong biến đổi InvMixColumns, mỗi cột b_j và a_j của trạng thái trước và sau xử lý cũng được xem là vector gồm Nw phần tử m-bit và a_j chính là ảnh của b_j qua ánh xạ tuyến tính θ_j^{-1}.

$\theta^{-1}: a = \theta^{-1}(b)$

$$\Leftrightarrow a_j(x) = d_j(x) \otimes b_j(x) = d_j(x) \times b_j(x) \bmod (1 \oplus x^{Nw}) \text{ với } 0 \leq j < Nb \qquad (2.17)$$

2.2.4 Biến đổi AddRoundKey trong XAES



Hình 2.6. Phép biến đổi AddRoundKey trong XAES

Chú thích: Trong Rijndael, mỗi khối (cũng như khóa của chu kỳ) gồm $Nw = 4$ dòng và $Nb = 4$, 6 hay 8 cột. Trong XAES, mỗi khối (cũng như khóa của chu kỳ) gồm Nw dòng và Nb cột ($Nw \le Nb \le 2Nw$).

Trong biến đổi AddRoundKey, ký hiệu là σ, không có tham số xử lý. Sử dụng khóa chính do người dùng cung cấp, XAES phát sinh ra dãy các khóa $\{k^r\}$ với k^r là khóa dùng trong chu kỳ r ($0 \le r \le Nr$). Khác với khóa chính được biểu diễn bằng ma trận gồm gồm Nw dòng và Nk cột, mỗi khóa k^r có kích thước giống với khối dữ liệu (được biểu diễn bằng ma trận gồm Nw dòng và Nb cột).

Trong chu kỳ r ($0 \le r \le Nr$), mỗi cột của trạng thái hiện hành được XOR với cột tương ứng của khóa k^r.

$$\sigma[k^r]: b = \sigma[k^r](a) \Leftrightarrow b_{i,j} = a_{i,j} \oplus k^r_{i,j} \text{ với } 0 \le i < Nw \text{ và } 0 \le j < Nb \qquad (2.18)$$

❖ **Nhận xét:** Thao tác biến đổi ngược của AddRoundKey cũng chính là AddRoundKey.

2.2.5 Hàm phát sinh khóa trong XAES

Hàm sinh khóa trong XAES được xây dựng bằng cách tổng quát hóa hàm sinh khóa trong AES để phù hợp với các tham số cấu trúc. Chính vì vậy, điểm khác biệt giữa hàm sinh khóa trong XAES với AES như sau:

- Mỗi phần tử được xử lý trong XAES gồm m-bit. Với AES, $m = 8$.
- Mỗi từ trong XAES gồm Nw phần tử m-bit. Với AES, $Nw = 4$.
- Thao tác SubWord trong XAES bao gồm Nw thao tác thay thế phần tử m-bit sử dụng bảng thay thế $m \times m$ S-box được xây dựng với 2 tham số xử lý $\mathcal{A}_{\varphi}^{(0)}$ và $\mathcal{A}_{\varphi}^{(1)}$ và ánh xạ nghịch đảo trên GF(2^m) (xem phần 2.2.1-Biến đổi SubBytes trong XAES). Trong khi đó, thao tác SubWord của AES gồm $Nw = 4$ thao tác thay thế các byte (8-bit) sử dụng bảng thay thế 8×8 S-box được đề xuất trong AES.

Nói cách khác, khi thay thế giá trị $Nw = 4$, $m = 8$ và sử dụng bảng thay thế 8×8 S-box trong AES, hàm sinh khóa trong XAES sẽ trở thành hàm sinh khóa được đề xuất trong AES.

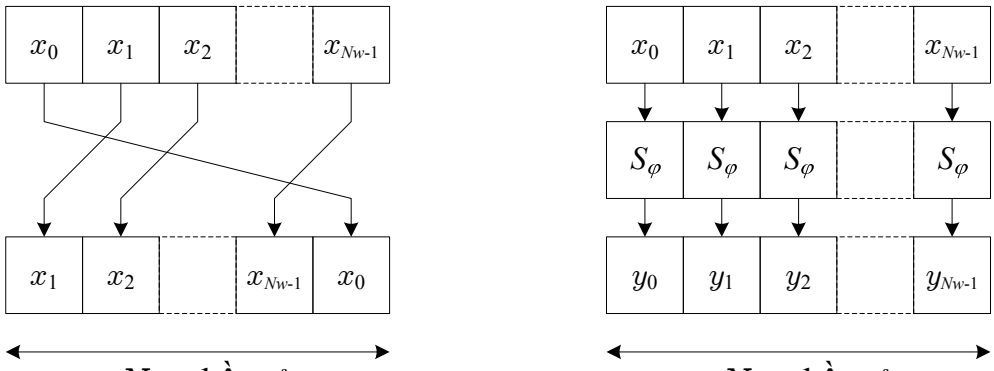
Quy trình phát sinh khóa cho mỗi chu kỳ gồm 2 giai đoạn:

1. Mở rộng khóa chính thành bảng khóa mở rộng,
2. Chọn khóa cho mỗi chu kỳ từ bảng khóa mở rộng.

❖ **Xây dựng bảng khóa mở rộng:**

Trong XAES cần sử dụng $Nr+1$ khóa, mỗi khóa k^r trong chu kỳ r gồm Nb từ, mỗi từ gồm Nw phần tử m-bit. Vậy, cần tạo ra bảng mã khóa mở rộng gồm $(Nr + 1) \times Nb$ từ với chất liệu ban đầu do người sử dụng cung cấp là khóa chính, ký hiệu là k, gồm Nk từ.

Sử dụng k, XAES lần lượt tạo ra $\lceil (Nr+1) \times Nb/Nk \rceil$ vector $H^{(i)}$, mỗi vector gồm Nk từ, để đưa vào bảng khóa mở rộng. Như vậy, trong bảng khóa mở rộng có tối thiểu $(Nr + 1) \times Nb$ từ. Gọi $H_j^{(i)}$ là từ thứ j trong vector $H^{(i)}$.



(a) Hàm RotWord (b) Hàm SubWord

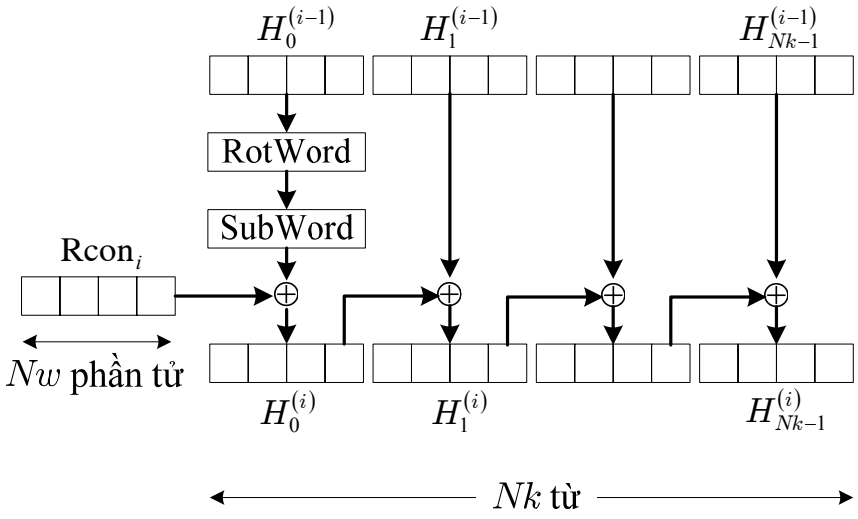
Hình 2.7. Hàm RotWord và SubWord

Quá trình xây dựng bảng khóa mở rộng:

- Đưa khóa chính k vào bảng khóa mở rộng: $H^{(0)} = k$. Lúc này, bảng khóa mở rộng có Nk từ.
- Lặp lại cho đến khi bảng khóa mở rộng có đủ $(Nr + 1)\times Nb$ từ
  - Tạo từ đầu tiên $H_0^{(i)}$ của $H^{(i)}$:
    - Dịch chuyển xoay vòng (RotWord) từ cuối cùng $H_{Nk-1}^{(i-1)}$ của $H^{(i-1)}$ (xem Hình 2.7a)
    - Thực hiện thay thế SubWord: sử dụng S-box để thay thế từng thành phần (m-bit) trong từ nhận được (xem Hình 2.7b)
    - XOR từ nhận được với hằng số $\text{Rcon}_i = (2^{i-1}, 0, 0, \ldots, 0) \in (\text{GF}(2^m))^{Nk}$
  - Tạo các từ $H_j^{(i)}$ còn lại trong $H^{(i)}$:
    - Thực hiện XOR từ ở vị trí j trong $H^{(i-1)}$ với từ thứ j–1 vừa xác định được trong $H^{(i)}$
    - Bổ sung $H^{(i)}$ vào bảng khóa mở rộng. Bảng khóa mở rộng có thêm Nk từ

Hình 2.8 minh họa quá trình phát sinh thêm vector (gồm Nk phần tử) cho bảng khóa mở rộng.

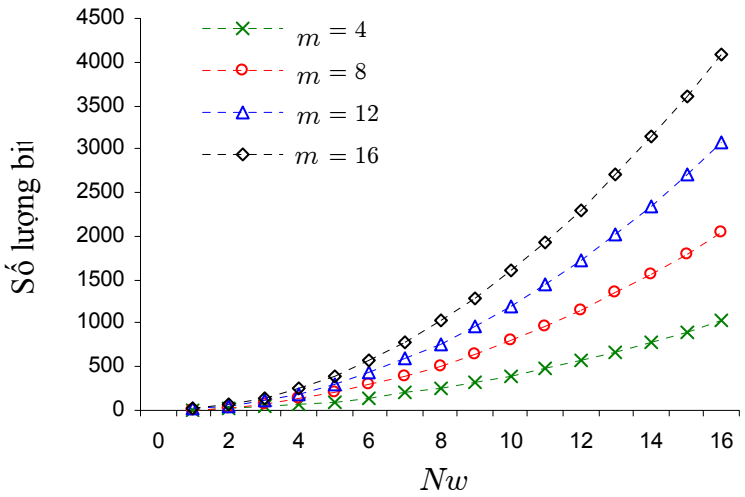
❖ **Xác định khóa của chu kỳ:** Khóa k^r của chu kỳ thứ r được xác định gồm Nb từ có chỉ số từ $Nb\times i$ đến $Nb \times (i+1)$–1 của bảng khóa mở rộng.



Hình 2.8. Quá trình phát sinh thêm vector *Nk* phần tử cho bảng khóa mở rộng

Chú thích: Trong Rijndael, mỗi phần tử gồm m=8 bit, mỗi từ gồm Nw=4 phần tử (8-bit), hàm SubWord sử dụng S-box được đề xuất trong thuật toán Rijndael (sử dụng ánh xạ nghịch đảo trên GF(2^8) và 1 ánh xạ affine trên GF(2)^8 làm bước hậu xử lý). Trong XAES, mỗi phần tử gồm m-bit, mỗi từ gồm Nw phần tử (m-bit) và hàm SubWord sử dụng S-box được tạo bằng sự kết hợp ánh xạ nghịch đảo trên GF(2^m) và 2 ánh xạ affine trên GF(2)^m làm bước tiền và hậu xử lý.

2.3 Kết quả thử nghiệm

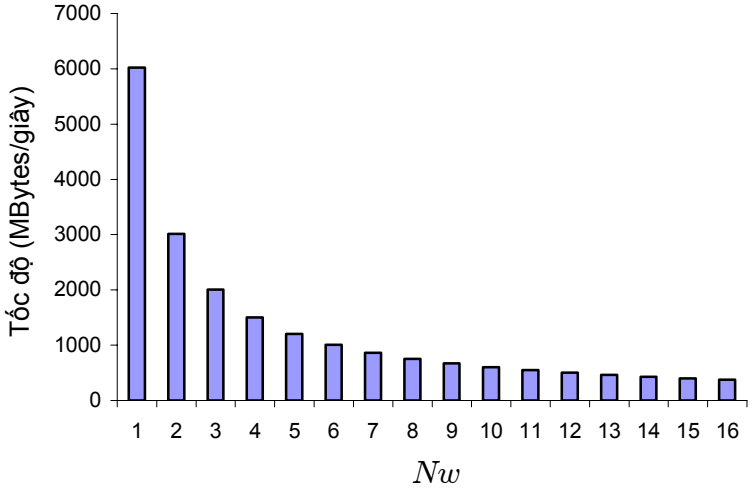


Hình 2.9. Biến thiên của kích thước khóa (tính bằng bit) theo giá trị tham số *Nw* trong trường hợp khóa chính được biểu diễn bằng ma trận vuông (*Nk*=*Nw*).

Trong XAES, kích thước khối là $m \times Nb \times Nw$ bit và kích thước khối là $m \times Nk \times Nw$ bit. Do $Nw \le Nb, Nk \le 2Nw$ nên kích thước khối (cũng như kích thước khóa) có giá trị tối thiểu là mNw^2 và giá trị tối đa là $2mNw^2$.

Với mỗi hệ thống xử lý, giá trị tham số cấu trúc m là cố định. Vì vậy, chúng tôi khảo sát mối quan hệ giữa kích thước khóa (tính bằng bit) theo tham số Nw lần lượt với các giá trị khác nhau của tham số m (xem Hình 2.9). Trong khảo sát này, chúng tôi xét trường hợp khóa chính được biểu diễn bằng ma trận vuông (Nk=Nw) nên kích thước khóa là mNw^2. Do kích thước khóa là hàm bậc 2 theo Nw nên việc tăng giá trị Nw sẽ giúp tăng đáng kể kích thước khóa. Ví dụ khi giá trị Nw tăng 3 lần, kích thước khóa sẽ tăng 9 lần và độ an toàn của XAES tăng lũy thừa 9 lần.

Tính chất này còn có ý nghĩa thực tế trong ngữ cảnh cần tạo ra một thuật toán mã hóa mới có độ an toàn đạt mức yêu cầu mới. Ví dụ muốn tạo ra thuật toán có khóa dài gấp 4 lần độ dài khóa đang sử dụng hiện tại, chỉ cần tạo ra thể hiện mới của XAES tương ứng với giá trị Nw lớn gấp đôi giá trị hiện tại. Giá trị Nw càng nhỏ thì càng tiết kiệm chi phí cho việc xác định các đa thức dùng trong biến đổi MixColumns.



Hình 2.10. Khảo sát tốc độ xử lý của XAES theo tham số Nw trong trường hợp m = 8, khối và khóa đều được biểu diễn dạng ma trận vuông (Nb = Nk = Nw).

Để khảo sát tốc độ xử lý của XAES với các giá trị khác nhau của Nw, gọi E_1 và E_2 lần lượt là 2 thể hiện của XAES với cùng giá trị tham số m nhưng tương ứng với giá trị Nw_1 và Nw_2 của tham số Nw. Không mất tính tổng quát, giả sử $Nw_1 < Nw_2$. Qua thực nghiệm, chúng tôi nhận thấy tốc độ xử lý của E_2 xấp xỉ Nw_1 / Nw_2 lần tốc độ xử lý của E_1. Hình 2.10 thể hiện việc thử nghiệm tốc độ xử lý của XAES trên hệ thống QuadCore 3.2 GHz (giá trị m=8). Kết quả thực nghiệm cho thấy ngay cả trong trường hợp kích thước khối và khóa đều là 2048 bit (tương ứng với Nw = 16), tốc độ xử lý của XAES đạt được mức 376 MB/giây, đảm bảo việc xử lý dữ liệu với kích thước lớn, ví dụ trong các hệ thống streaming dữ liệu multimedia hoặc mã hóa cơ sở dữ liệu…

2.4 Kết luận

Trong chương 3, chúng tôi đã trình bày thuật toán mã hóa được tham số hóa XAES. Điểm đặc trưng của XAES là thuật toán không được đặc tả "cứng" thông qua các giá trị hằng số cụ thể trong mỗi thành phần mã hóa mà được đặc tả với các tham số và quy tắc xây dựng các thành phần mã hóa. Điều này cho phép dễ dàng tạo ra các biến thể của thuật toán với các bộ hằng số khác nhau cho mỗi biến đổi thuật toán. Ngoài ra, XAES cho phép mở rộng không giới hạn kích thước khóa và kích thước khối.

Với mỗi giá trị tham số cấu trúc m và Nw, có thể sử dụng quy tắc xây dựng các tham số xử lý cho từng biến đổi trong XAES để tự tạo ra một bộ giá trị cho tham số xử lý, tạo ra một ***thể hiện cụ thể của XAES***. Mỗi thể hiện của XAES là một thuật toán mã hóa khối có kích thước khối, kích thước khóa xác định và một bộ hằng số xác định cho các biến đổi. Tập hợp các thể hiện của XAES là một tập con (vô hạn) của tập hợp các thuật toán mã hóa khối xây dựng theo chiến lược vết rộng.

Phương pháp sai phân và tuyến tính là hai phương pháp nền tảng để kiểm tra tính an toàn của thuật toán mã hóa khối. Chính vì vậy, trong Chương 3 và Chương 4, chúng tôi sẽ trình bày kết quả chứng minh tính an toàn của XAES đối với hai phương pháp phân tích mã này.

Trong Chương 3, chúng tôi sử dụng hướng tiếp cận truyền thống được J. Daemen đề xuất trong [19] với vết sai phân đơn và vết tuyến tính đơn. Đây là hướng tiếp cận được dùng phổ biến trong việc chứng minh tính an toàn của giải thuật mã hóa khối, ví dụ như Shark, Square, Rijndael, Anubis, Khazad…

Trong Chương 4, chúng tôi sử dụng hướng tiếp cận dựa trên tập vết sai phân và bao tuyến tính. Đây là hướng tiếp cận được quan tâm nhiều từ năm 2000 đến nay [12][38][44][45][46][47][71][72] và tiếp tục được áp dụng để khảo sát và chứng minh tính an toàn đối với các thuật toán mã hóa khối đã được công bố, kể cả Rijndael.

Dựa trên quy tắc xây dựng các thành phần mã hóa trong XAES, chúng tôi chứng minh tính an toàn của XAES đối với các phương pháp phân tích mã hiện nay một cách tổng quát, độc lập với giá trị cụ thể của các tham số cấu trúc và tham số xử lý. Nhờ đó, với mỗi thể hiện của XAES, không cần chứng minh lại tính an toàn đối với phương pháp sai phân và tuyến tính.

Các vấn đề mở:

- Trong XAES, thông qua phần chứng minh sẽ được trình bày trong 2 chương tiếp theo, chúng tôi chứng minh được rằng các thể hiện của XAES đảm bảo tính an toàn đối với các phương pháp phân tích mã hiện tại, cụ thể là với phương pháp tuyến tính và phương pháp sai phân. Câu hỏi đặt ra là liệu có bộ giá trị tham số nào của XAES có tính chất đặc biệt và có thể được khai thác để tạo ra kỹ thuật phân tích mã đặc thù nhằm tấn công vào các thể hiện đặc biệt đó của XAES hay không.
- Khi sử dụng XAES vào một hệ thống cụ thể hay kiến trúc xử lý cụ thể, cần chọn lựa các bộ giá trị tham số, đặc biệt là tham số xử lý theo các các tiêu chí về tính hiệu quả (trong thiết kế, xử lý…).

Chương 3

Khảo sát tính an toàn của XAES dựa trên lan truyền của vết sai phân đơn và vết tuyến tính đơn

Tóm tắt chương:

✍ *Nội dung của chương 3 trình bày kết quả chứng minh tính an toàn của XAES đối với phương pháp sai phân và phương pháp tuyến tính để phân tích mã sử dụng lan truyền của vết sai phân đơn và vết tuyến tính đơn, gồm:*

- *Trình bày tóm tắt về phương pháp sai phân và phương pháp tuyến tính trong phân tích mã.*
- *Trình bày kết quả chứng minh tính an toàn của XAES đối với phương pháp sai phân và phương pháp tuyến tính trong phân tích mã:*
  - *Khảo sát sự lan truyền mẫu hoạt động qua các phép biến đổi trong một chu kỳ (phần 3.2.1).*
  - *Xác định tổng số lượng S-box tối thiểu trong vết lan truyền qua các chu kỳ với kết quả trọng tâm là **Định lý 3.3** và **Định lý 3.4**.*
  - *Xác định chặn trên tổng quát cho tỷ lệ truyền của vết sai phân và độ tương quan của vết tuyến tính dựa vào tổng số lượng S-box tối thiểu trong vết lan truyền. Kết quả trọng tâm được trình bày trong **Định lý 3.5** và **Định lý 3.6**.*

3.1 Phân tích mã sai phân và phân tích mã tuyến tính

Nội dung của phần này sẽ trình bày tóm lược những nội dung chính về phân tích mã sai phân (Differential Cryptanalysis [6]) và phân tích mã tuyến tính (Linear Cryptanalysis [62]), hai phương pháp phân tích mã hiện được xem là hiệu quả nhất đối với các thuật toán mã hóa theo khối [50].

Ý tưởng cơ bản của phương pháp phân tích mã sai phân được Eli Biham và Adi Shamir đề xuất trong [6] là khảo sát sự lan truyền sai phân ở đầu vào qua các chu kỳ mã hóa và khai thác những giá trị *xác suất sai phân* đủ lớn qua T chu kỳ mã hóa. (thường xét $T = Nr - 1$ hay $T = Nr - 2$).

Phương pháp phân tích mã tuyến tính được Mitsuru Matsui đề xuất [62] khảo sát mỗi tương quan giữa đầu ra và đầu vào của các chu kỳ biến đổi và khai thác những giá trị *xác suất tuyến tính* đủ lớn qua T chu kỳ mã hóa (thường xét $T = Nr - 1$ hay $T = Nr - 2$).

3.1.1 Sự lan truyền sai phân và vết sai phân

Cho vector $a \in \{0,1\}^n$ và $a^* \in \{0,1\}^n$. Đặt $\Delta a = a \oplus a^*$ thể hiện sự khác biệt giữa các bit tương ứng của a và a^*. Gọi b và b^* lần lượt là ảnh của a và a^* qua ánh xạ $h : \{0,1\}^n \to \{0,1\}^n$. Đặt $\Delta b = b \oplus b^*$. Ta nói rằng ***vector sai phân*** Δa ở đầu vào của ánh xạ h đã ***lan truyền*** thành ***vector sai phân*** Δb ở đầu ra của h, ký hiệu là $\left(\Delta a \xrightarrow{h} \Delta b\right)$ [19]. Thông thường, với một ánh xạ h cụ thể, Δb không chỉ phụ thuộc vào Δa mà còn phụ thuộc vào vector a [19].

Định nghĩa 3.1. [19] *Tỷ lệ truyền của một lan truyền sai phân* $\left(\Delta a \xrightarrow{h} \Delta b\right)$ *qua ánh xạ* $h : \{0,1\}^n \to \{0,1\}^n$*, ký hiệu là* $\mathrm{R}_P\left(\Delta a \xrightarrow{h} \Delta b\right)$*, được xác định như sau:*

$$\mathrm{R}_P\left(\Delta a \xrightarrow{h} \Delta b\right) = \frac{\sum_a \delta\left(\Delta b \oplus h(a \oplus \Delta a) \oplus h(a)\right)}{2^n} \tag{3.1}$$

với δ *là hàm delta Dirac.*

Định nghĩa 3.2. [19] *Trọng số của một lan truyền sai phân* $\left(\Delta a \xrightarrow{h} \Delta b\right)$ *được xác định bằng số đối của logarithm cơ số* 2 *của tỷ lệ truyền:*

$$\mathrm{w}_r\left(\Delta a \xrightarrow{h} \Delta b\right) = -\log_2 \mathrm{R}_p\left(\Delta a \xrightarrow{h} \Delta b\right) \tag{3.2}$$

Định nghĩa 3.3. [19] *Vết sai phân* Ω *lan truyền qua t chu kỳ* $\rho_1, \rho_2, ..., \rho_t$ *gồm t lan truyền sai phân liên tiếp nhau* $\left(\Delta_{i-1} \xrightarrow{\rho_i} \Delta_i\right)$

$$\Omega = \left(\Delta_0 \xrightarrow{\rho_1} \Delta_1 \xrightarrow{\rho_2} \ldots \xrightarrow{\rho_{t-1}} \Delta_{t-1} \xrightarrow{\rho_t} \Delta_t\right) \tag{3.3}$$

Định nghĩa 3.4. [19] *Trọng số của vết sai phân* Ω *được xác định như sau*:

$$\mathrm{w}_r(\Omega) = \prod_i \mathrm{w}_r\left(\Delta_{i-1} \xrightarrow{\rho_i} \Delta_i\right) \tag{3.4}$$

Định nghĩa 3.5. [19] *Tỷ lệ truyền của vết sai phân* Ω*, ký hiệu là* $\mathrm{R}_P(\Omega)$*, là tỷ lệ các giá trị của vector đầu vào của chu kỳ đầu tiên cho phép tạo ra vết sai phân* Ω*, được tính xấp xỉ là* : $\mathrm{R}_P(\Omega) \approx 2^{-\mathrm{w}_r(\Omega)}$.

3.1.2 Sự tương quan và vết tuyến tính

Cho $a = (a_0, a_1, ..., a_{m-1}) \in \{0,1\}^n$ *và* $b = (b_0, b_1, ..., b_{m-1}) \in \{0,1\}^n$. Phép toán $\bullet$ trên $\{0,1\}^n$ được định nghĩa như sau:

$$a \bullet b = a_0 b_0 \oplus a_1 b_1 \oplus \ldots \oplus a_{n-1} b_{n-1} \tag{3.5}$$

Bản chất của phép toán $\bullet$ là tích vô hướng của 2 vector a và b.

Cho 2 vector $\Gamma b \in \{0,1\}^n$ và $\Gamma a \in \{0,1\}^n$. Gọi b và a lần lượt là ảnh và tiền ảnh tương ứng qua ánh xạ $h : \{0,1\}^n \to \{0,1\}^n$. Nếu $\Gamma a \bullet a = \Gamma b \bullet b$ thì ta nói có rằng có ***sự tương quan*** giữa mặt nạ Γb ở đầu ra và mặt nạ Γa ở đầu vào của ánh xạ h [48], ký hiệu là $\left(\Gamma a \xleftarrow{h} \Gamma b\right)$. Thông thường, vector Γb và Γa được gọi là ***mặt nạ*** [38][48]. Trong [19], J. Daemen gọi vector mặt nạ là ***vector chọn***.

Định nghĩa 3.6. [19] *Cho* f *và* g *là* 2 *ánh xạ Bool được xác định trên* $\{0,1\}^n$. *Hệ số tương quan* $C(f,g)$ *giữa* f *và* g *được định nghĩa như sau*:

$$C(f,g) = 2 \cdot \mathrm{Prob}(f(a) = g(a)) - 1 \tag{3.6}$$

Như vậy, $C(f,g) = C(g,f)$.

Định nghĩa 3.7. [19] *Hệ số tương quan ứng với mặt nạ* Γb *ở đầu ra và mặt nạ* Γa *ở đầu vào của ánh xạ h, ký hiệu là* $\mathrm{C}^{h}_{\Gamma b,\Gamma a}$ *, được xác định như sau:*

$$\mathrm{C}^{h}_{\Gamma b,\Gamma a} = C\left(\Gamma b \bullet h(a), \Gamma a \bullet a\right) \tag{3.7}$$

Định nghĩa 3.8. [19] *Vết tuyến tính* Ξ *lan truyền qua t chu kỳ mã hóa* $\rho_1, \rho_2,\ldots,\rho_t$ *gồm t lan truyền tuyến tính liên tiếp* $\left(\Gamma_{i-1} \xleftarrow{\rho_i} \Gamma_i\right)$ *với* Γ_i *và* Γ_{i-1} *lần lượt là mặt nạ ở đầu ra và đầu vào của chu kỳ* ρ_i

$$\Xi = \left(\Gamma_0 \xleftarrow{\rho_1} \Gamma_1 \xleftarrow{\rho_2} \ldots \xleftarrow{\rho_{t-1}} \Gamma_{t-1} \xleftarrow{\rho_t} \Gamma_t\right) \tag{3.8}$$

Định nghĩa 3.9. [19] *Hệ số tương quan của vết tuyến tính* Ξ*, ký hiệu là* $\mathrm{C}_P(\Xi)$ *được xác định như sau:*

$$\mathrm{C}_P(\Xi) = \prod_i \mathrm{C}^{\rho_i}_{\Gamma_i,\Gamma_{i-1}} \tag{3.9}$$

Định nghĩa 3.10. [19] *Trọng số tương quan* w_c *của vết tuyến tính* Ξ *được xác định như sau*:

$$\mathrm{w}_c(\Xi) = -\log_2\left|\mathrm{C}_P(\Xi)\right| \tag{3.10}$$

3.1.3 Hướng tiếp cận sử dụng vết sai phân/tuyến tính đơn

Trong [16] và [19], J. Daemen và V. Rijmen áp dụng chiến lược chứng minh tính an toàn của thuật toán mã hóa theo khối đối với phương pháp sai phân và phương pháp tuyến tính dựa trên việc khảo sát vết sai phân đơn hay tuyến tính đơn:

- Phương pháp sai phân chỉ có thể được áp dụng nếu có thể dự đoán được sự lan truyền sai phân trong các mẫu đầu vào qua hầu hết các chu kỳ biến đổi (thường xét *Nr*–2 hay *Nr*–1 chu kỳ [16][49]) với tỷ lệ truyền lớn hơn đáng kể so với giá trị 2^{1-n} (với n là độ dài khối, tính bằng bit). Theo hướng tiếp cận này, để đảm bảo an toàn đối với phương pháp sai phân, cần chứng minh là không tồn tại vết sai phân đơn lan truyền qua T chu kỳ (với $T = Nr - 2$ hay $T = Nr - 1$) có tỷ lệ truyền lớn hơn đáng kể so với giá trị 2^{1-n}.
- Phương pháp tuyến tính chỉ có thể được áp dụng nếu hệ số tương quan giữa đầu ra với đầu vào của thuật toán qua hầu hết các chu kỳ (thường xét *Nr*–2 hay $Nr - 1$ chu kỳ [16][49]) có giá trị rất lớn so với $2^{-n/2}$ (với n là độ dài khối, tính bằng bit). Như vậy, để đảm bảo an toàn cho một phương pháp mã hóa, điều kiện cần thiết là không tồn tại vết tuyến tính lan truyền qua T chu kỳ (với $T = Nr - 2$ hay $T = Nr - 1$) có hệ số tượng quan lớn hơn đáng kể so với giá trị $2^{-n/2}$.

Trong [16] và [17], J. Daemen và V. Rijmen đã chứng minh tính an toàn cụ thể cho riêng trường hợp thuật toán Rijndael đối với phương pháp sai phân và phương pháp tuyến tính trong phân tích mã dựa trên hướng tiếp cận sử dụng một vết sai phân đơn hay một vết tuyến tính đơn. Theo hướng tiếp cận này, chúng tôi đã chứng minh tổng quát tính an toàn của XAES đối với phương pháp sai phân và phương pháp tuyến tính độc lập với các giá trị cụ thể của các tham số về cấu trúc và tham số về xử lý của XAES.

3.2 Tỷ lệ truyền của vết sai phân đơn và độ tương quan của vết tuyến tính đơn trong XAES

3.2.1 Sự lan truyền mẫu

Định nghĩa 3.11. [72] *Trong phương pháp phân tích mã sai phân, S-box được gọi là hoạt động nếu có lượng khác biệt ở đầu vào khác* 0. *Trong phương pháp phân tích mã tuyến tính, S-box được gọi là hoạt động nếu có giá trị mặt nạ ở đầu ra khác* 0.

Do các S-box được sử dụng trong XAES đều là song ánh nên S-box hoạt động sẽ có lượng khác biệt đầu ra khác 0 (khi xét trong phân tích mã sai phân) hoặc sẽ có mặt nạ ở đầu vào khác 0 (khi xét trong phân tích mã tuyến tính).

Trong phương pháp sai phân:

- ***số lượng S-box hoạt động*** được xác định bằng số lượng phần tử khác 0 trong ***vector sai phân*** ở đầu vào của chu kỳ,
- ***Mẫu (sai phân) hoạt động*** [16] là mẫu xác định vị trí các S-box hoạt động.

Trong phương pháp tuyến tính:

- ***số lượng S-box hoạt động*** được xác định bằng số lượng phần tử khác 0 trong ***vector chọn*** [19] ở đầu vào của chu kỳ,
- ***Mẫu (tương quan) hoạt động*** [16] là mẫu xác định vị trí các S-box hoạt động.

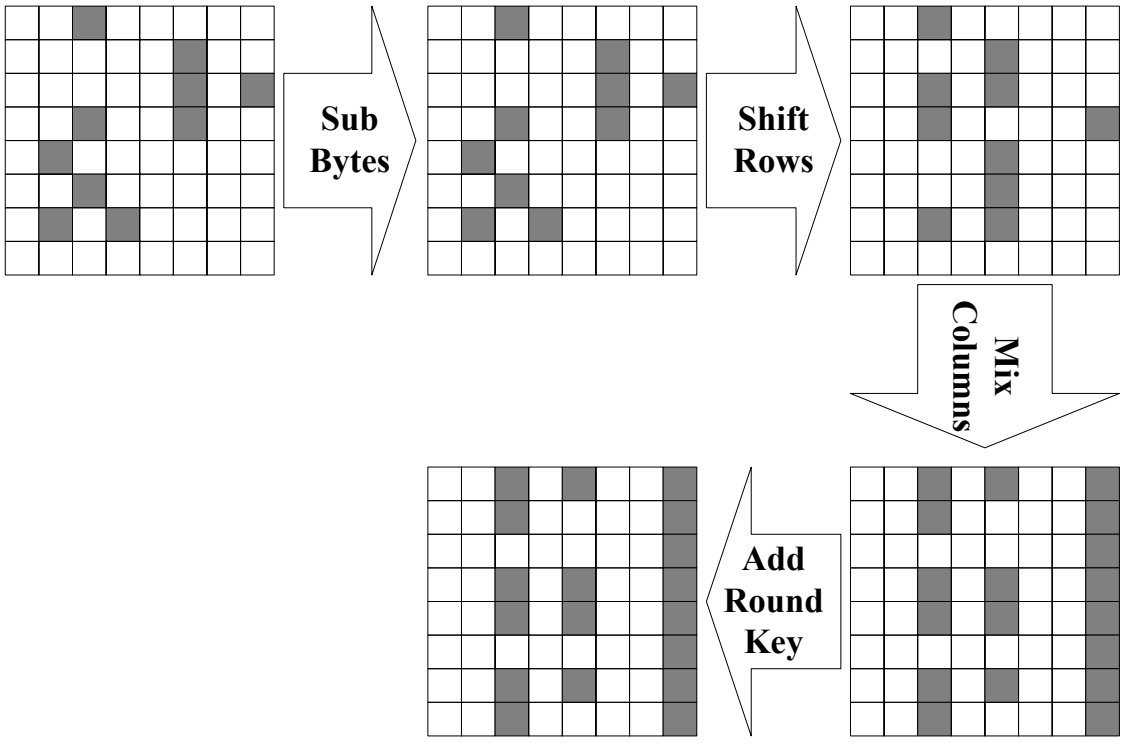
Gọi ***trọng số phần tử*** là số lượng phần tử khác 0 trong mẫu hoạt động (sai phân hoặc tuyến tính)

Mỗi cột trong trạng thái có ít nhất một phần tử (m-bit) hoạt động được gọi ***cột hoạt động***. ***Trọng số cột*** của trạng thái a, ký hiệu là $wt_C(a)$, được định nghĩa là số lượng cột hoạt động trong mẫu. ***Trọng số phần tử*** của cột j trong trạng thái a, ký hiệu là $wt(a)\big|_j$, được định nghĩa là số lượng phần tử (m-bit) hoạt động trong cột này.

Trọng số của một vết lan truyền qua các chu kỳ được tính bằng tổng tất cả các trọng số của các mẫu hoạt động ở đầu vào của mỗi chu kỳ thành phần.

Trong các hình minh họa dưới đây, cột hoạt động được tô màu xám nhạt còn các phần tử hoạt động được tô màu xám đậm.

Hình 3.1 minh họa sự lan truyền các mẫu hoạt động (bao gồm cả mẫu sai phân và mẫu tuyến tính) qua từng phép biến đổi trong một chu kỳ mã hóa của XAES với Nw= 8 và Nb = 8 và $\omega_\pi = \{0,1,2,\ldots,7\}$



Hình 3.1. Ví dụ về sự lan truyền mẫu hoạt động qua từng phép biến đổi trong một chu kỳ của XAES với Nw = 8 và Nb = 8 và ω_π = {0, 1, 2,…, 7}

Mỗi phép biến đổi thành phần trong XAES có tác động khác nhau đối với các mẫu hoạt động và các trọng số:

1. SubBytes và AddRoundKey không làm thay đổi các mẫu hoạt động cũng như giá trị trọng số cột và trọng số phần tử của mẫu.

2. ShiftRows làm thay đổi mẫu hoạt động và trọng số cột. Do phép biến đổi ShiftRows tác động lên từng phần tử (m-bit) của trạng thái một cách độc lập, không có sự tương tác giữa các phần tử (m-bit) trong trạng thái đang xét nên không làm thay đổi trọng số phần tử.

3. MixColumns làm thay đổi mẫu hoạt động và trọng số phần tử. Do phép biến đổi MixColumns tác động lên từng cột của trạng thái một cách độc lập, không có sự tương tác giữa các cột thành phần trong trạng thái đang xét nên không làm thay đổi trọng số cột.

Bảng 3.1 tóm tắt ảnh hưởng của các phép biến đổi lên mẫu hoạt động.

STT	Phép biến đổi	Sự ảnh hưởng		
		Mẫu hoạt động	Trọng số cột	Trọng số phần tử
1	SubBytes	Không	Không	Không
2	ShiftRows	**Có**	**Có**	Không
3	MixColumns	**Có**	Không	**Có**
4	AddRoundKey	Không	Không	Không

Bảng 3.1. Ảnh hưởng của các phép biến đổi lên mẫu hoạt động

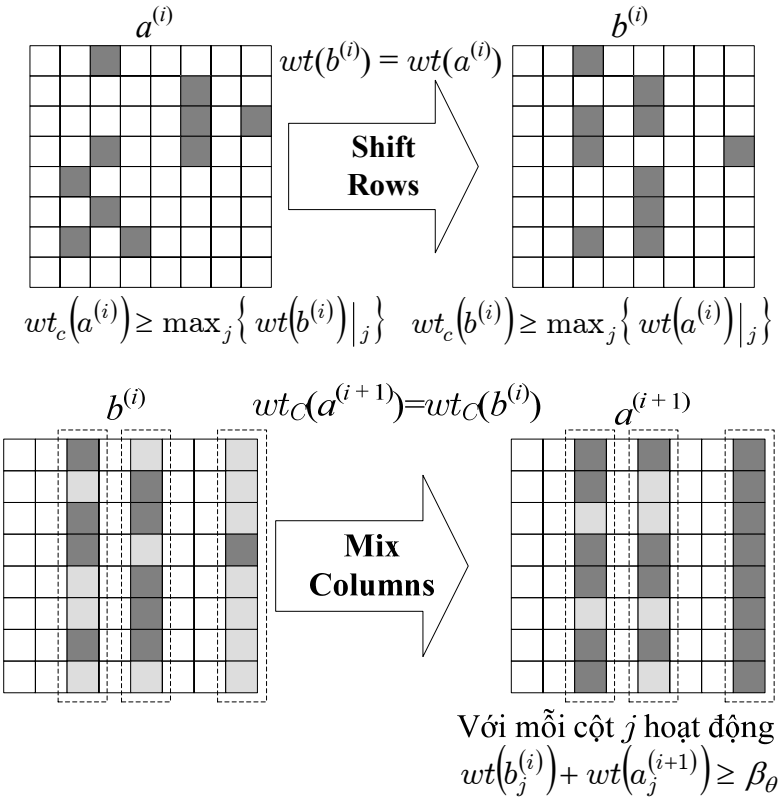
Như vậy, phép biến đổi SubBytes và AddRoundKey không ảnh hưởng đến sự lan truyền các mẫu hoạt động trong vết nên chúng ta có thể bỏ qua các phép biến đổi này trong quá trình khảo sát các vết sai phân và vết tuyến tính dưới đây.

Trong biến đổi MixColumns sử dụng tập Θ_θ gồm Nb biến đổi tuyến tính, mỗi biến đổi tuyến tính có thể có giá trị branch number riêng. Để chứng minh các tính chất về vết sai phân và vết tuyến tính trong XAES, dựa trên định nghĩa về Branch Number của biến đổi tuyến tính (xem **Định nghĩa 1.1**), chúng tôi đề xuất bổ sung định nghĩa về Branch Number cho biến đổi MixColumns như sau:

Định nghĩa 3.12. *Branch Number của biến đổi MixColumns, ký hiệu là β_θ, có giá trị bằng với branch number tối thiểu của các biến đổi tuyến tính θ_i sử dụng trong MixColumns.*

$$\beta_\theta = \min_{i=0,\ldots,Nb-1} \{\mathcal{B}(\theta_i)\} \tag{3.11}$$

Như vậy, với mỗi cột hoạt động trong mẫu đầu vào (hoặc mẫu đầu ra) của một chu kỳ, tổng trọng số phần tử của cột này trong mẫu đầu vào và đầu ra bị chặn dưới bởi β_θ.



Hình 3.2. Sự lan truyền mẫu hoạt động

trong trường hợp $Nw = 8$, $Nb = 8$ và ω_π={0,1,2,…,Nw–1}

Do ShiftRows thực hiện việc dịch chuyển tất cả các byte thành phần trong một cột của mẫu đến tất cả các cột khác nhau nên phép biến đổi ShiftRows có các tính chất đặc biệt sau:

1. Trọng số cột của mẫu đầu ra bị chặn dưới bởi giá trị tối đa của trọng số phần tử của mỗi cột trong mẫu đầu vào.
2. Trọng số cột của mẫu đầu vào bị chặn dưới bởi giá trị tối đa của trọng số phần tử của mỗi cột trong mẫu đầu ra.

Hình 3.2 minh họa sự lan truyền mẫu trong một chu kỳ của XAES (với $Nw = Nb = 8$) và sử dụng tập giá trị dịch chuyển $\omega_\pi = \{0, 1, 2, \ldots, Nw - 1\}$. Mẫu hoạt động ở đầu vào của chu kỳ mã hóa thứ i được ký hiệu là $a^{(i)}$, mẫu hoạt động kết quả sau khi thực hiện phép biến đổi ShiftRows được ký hiệu là $b^{(i)}$. Các chu kỳ biến đổi được đánh số tăng dần bắt đầu từ 1. Như vậy, $a^{(1)}$ là mẫu hoạt động ở đầu vào của chu kỳ mã hóa đầu tiên. Dễ dàng nhận thấy rằng mẫu $a^{(i)}$ và $b^{(i)}$ có cùng trọng số phần tử, mẫu $b^{(j-1)}$ và $a^{(j)}$ có cùng trọng số cột. Trọng số của một vết lan truyền qua m chu kỳ được xác định bằng tổng trọng số của các mẫu $a^{(1)}$, $a^{(2)}$, ..., $a^{(m)}$.

3.2.2 Số lượng S-box hoạt động trong vết lan truyền

Định lý 3.1: *Trọng số của vết lan truyền qua 2 chu kỳ có Q cột hoạt động ở đầu vào của chu kỳ 2 bị chặn dưới bởi $\beta_\theta \times Q$.*

$$wt_c\left(a^{(2)}\right) = Q \Rightarrow wt\left(a^{(1)}\right) + wt\left(a^{(2)}\right) \geq \beta_\theta \times Q \tag{3.12}$$

✍***Chứng minh***: Tổng trọng số phần tử của mỗi cột tương ứng hoạt động trong mẫu $b^{(1)}$ và $a^{(2)}$ bị chặn dưới bởi β_θ. Nếu trọng số cột của $a^{(2)}$ là Q thì tổng trọng số phần tử của $b^{(1)}$ và $a^{(2)}$ bị chặn dưới bởi $\beta_\theta \times Q$.

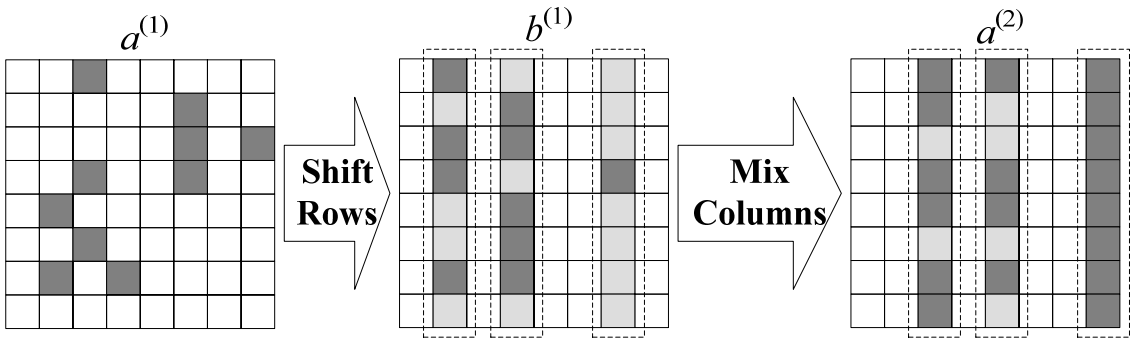
$$wt\left(b^{(1)}\right) + wt\left(a^{(2)}\right) \geq \beta_\theta \times Q \tag{3.13}$$

Do $a^{(1)}$ và $b^{(1)}$ có cùng trọng số phần tử nên tổng trọng số phần tử của $a^{(1)}$ và $a^{(2)}$ bị chặn dưới bởi $\beta_\theta \times Q$.

$$wt\left(a^{(1)}\right) + wt\left(a^{(2)}\right) \geq \beta_\theta \times Q \tag{3.14}$$

Vậy, mọi vết lan truyền qua 2 chu kỳ đều có ít nhất $\beta_\theta \times Q$ phần tử hoạt động. ❑

Hình 3.3 minh họa **Định lý 3.1** trong trường hợp $Nw = 8$, $Nb = 8$, $Q = 3$ và sử dụng tập giá trị dịch chuyển $\omega_\pi = \{0, 1, 2, \ldots, Nw - 1\}$.



Hình 3.3. Minh họa Định lý 3.1 với Q=3

(trường hợp Nw = 8, Nb = 8 và ω_π={0,1,2,…,7})

Định lý 3.2: *Với mỗi vết lan truyền qua 2 chu kỳ, tổng số cột hoạt động trong mẫu đầu vào và mẫu đầu ra tối thiểu là β_θ.*

$$wt_c\left(a^{(1)}\right) + wt_c\left(a^{(3)}\right) \geq \beta_\theta \tag{3.15}$$

✍***Chứng minh***:

Do tập giá trị dịch chuyển ω_π gồm các giá trị phân biệt và $Nw \leq Nb$ nên biến đổi ShiftRows di chuyển tất cả các phần tử trên một cột của $a^{(1)}$ sang các cột hoàn toàn khác nhau của $b^{(1)}$. Vì vậy, trọng số cột của $a^{(1)}$ bị chặn dưới bởi trọng số phần tử của mỗi cột trong $b^{(1)}$:

$$wt_C\left(a^{(1)}\right) \geq \max_j wt\left(b^{(1)}\right)_j \tag{3.16}$$

Tương tự, trọng số cột của $b^{(2)}$ bị chặn dưới bởi trọng số phần tử của mỗi cột trong $a^{(2)}$:

$$wt_C\left(b^{(2)}\right) \geq \max_j wt\left(a^{(2)}\right)_j \tag{3.17}$$

Trong một vết bất kỳ, tồn tại ít nhất một cột hoạt động trong mẫu $b^{(1)}$ (hoặc $a^{(2)}$). Gọi cột hoạt động này là g. Do Branch Number của MixColumns là β_θ nên tổng trọng số phần tử của cột g trong mẫu $b^{(2)}$ và mẫu $a^{(1)}$ bị chặn dưới bởi β_θ.

$$wt\left(b^{(1)}\right)_g + wt\left(a^{(2)}\right)_g \geq \beta_\theta \tag{3.18}$$

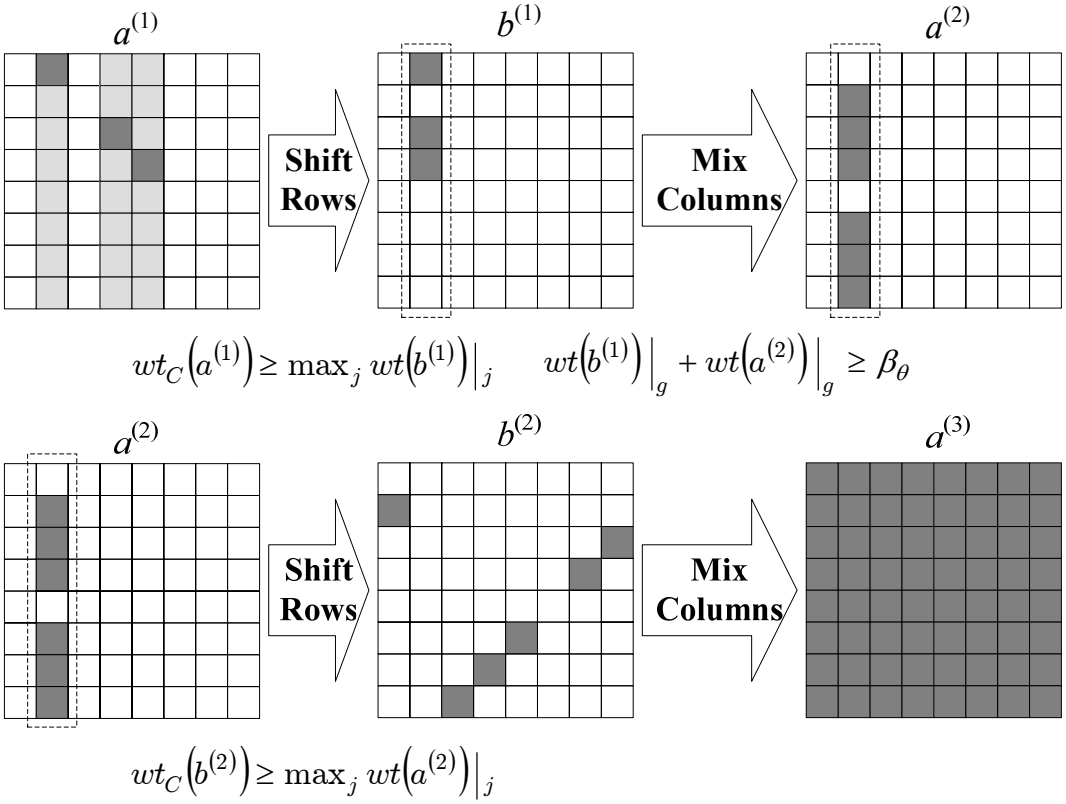
Vậy, ta có:
$$wt_C\left(a^{(1)}\right) + wt\left(b^{(2)}\right) \geq \beta_\theta \tag{3.19}$$

Do $b^{(2)}$ và $a^{(3)}$ có cùng trọng số cột, ta kết luận:

$$wt_c\left(a^{(1)}\right) + wt_c\left(a^{(3)}\right) \geq \beta_\theta \tag{3.20}$$

❑

Hình 3.4 minh họa Định lý 3.2 trong trường hợp $Nw = Nb = 8$ và sử dụng tập giá trị dịch chuyển $\omega_\pi = \{0, 1, 2, \ldots, Nw - 1\}$.



Hình 3.4. Minh họa Định lý 3.2
(trường hợp $Nw = Nb = 8$ và $\omega_\pi = \{0,1,2,\ldots,7\}$)

Định lý 3.3: *Mọi vết lan truyền qua 4 chu kỳ của XAES đều có tối thiểu β_θ^2 phần tử hoạt động.*

✍***Chứng minh***: Áp dụng **Định lý 3.1** cho hai chu kỳ đầu (chu kỳ 1 và 2) và hai chu kỳ sau (chu kỳ 3 và 4), ta có:

$$\begin{cases} wt\left(a^{(0)}\right) + wt\left(a^{(1)}\right) \geq \beta_\theta \times wt_C\left(a^{(1)}\right) \\ wt\left(a^{(2)}\right) + wt\left(a^{(3)}\right) \geq \beta_\theta \times wt_C\left(a^{(3)}\right) \end{cases} \tag{3.21}$$

$$\Rightarrow \sum_{i=1}^{4} wt\left(a^{(i)}\right) \geq \beta_\theta \times \left(wt_C\left(a^{(2)}\right) + wt_C\left(a^{(4)}\right)\right) \tag{3.22}$$

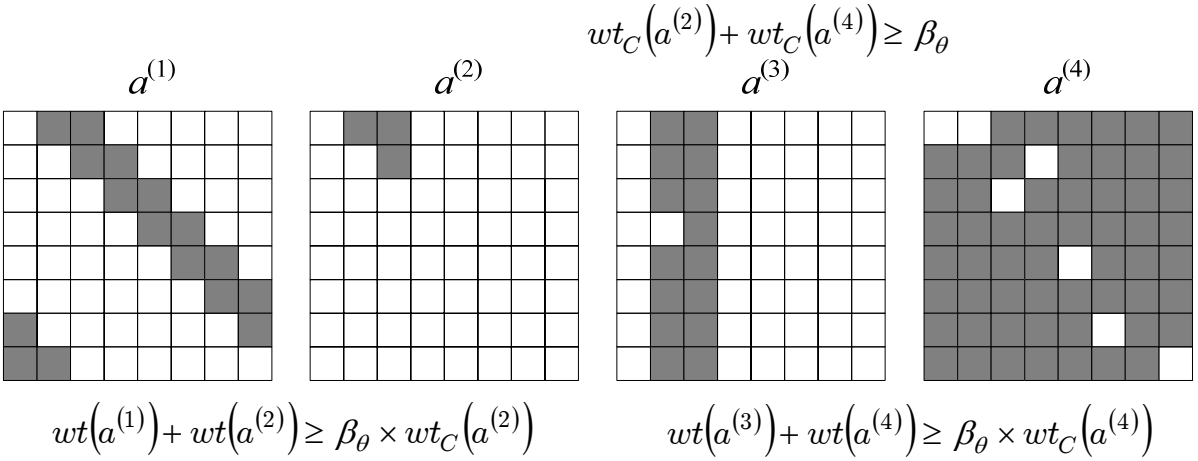
Như vậy, trọng số phần tử của vết bị chặn dưới bởi $\beta_\theta \times \left(wt_C\left(a^{(2)}\right) + wt_C\left(a^{(4)}\right)\right)$

Theo **Định lý 3.2**, tổng trọng số cột của $a^{(2)}$ và $a^{(4)}$ bị chặn dưới bởi β_θ.

$$wt_C\left(a^{(2)}\right) + wt_C\left(a^{(4)}\right) \geq \beta_\theta \tag{3.23}$$

Vậy, trọng số phần tử của vết lan truyền qua 4 chu kỳ bị chặn bởi β_θ^2 hay vết lan truyền qua 4 chu kỳ có ít nhất β_θ^2 phần tử hoạt động. ❑

Hình 3.5 minh họa Định lý 3.3 trong trường hợp $Nw = 8$, $Nb = 8$ và sử dụng tập giá trị dịch chuyển $\omega_\pi = \{0, 1, 2, \ldots, Nw - 1\}$.



Hình 3.5. Minh họa Định lý 3.3

(trường hợp $Nw = 8$, $Nb = 8$ và $\omega_\pi = \{0, 1, 2, \ldots, 7\}$)

Định lý 3.4. *Mọi vết lan truyền qua* $4r$ *chu kỳ* ($r > 0$) *của XAES đều có tối thiểu* $r \times \beta_\theta^2$ *phần tử hoạt động.*

✍***Chứng minh***:

Để tính tổng số phần tử hoạt động của vết lan truyền qua $4r$ chu kỳ, ta tính tổng số phần tử hoạt động trong từng nhóm gồm 4 chu kỳ liên tiếp nhau:

$$\sum_{i=1}^{4r} wt\left(a^{(i)}\right) = \sum_{i=0}^{r-1} \sum_{j=1}^{4} wt\left(a^{(4i+j)}\right) \tag{3.24}$$

Áp dụng **Định lý 3.3** cho từng nhóm gồm 4 chu kỳ liên tiếp, ta có:

$$\sum_{j=1}^{4} wt\left(a^{(4i+j)}\right) \geq \beta_\theta^2 \tag{3.25}$$

Vậy, ta có:

$$\sum_{i=1}^{4r} wt\left(a^{(i)}\right) \geq= \sum_{i=0}^{r-1} \beta_\theta^2 = r \times \beta_\theta^2 \tag{3.26}$$

❑

3.2.3 Tỷ lệ truyền của vết sai phân trong XAES

Bổ đề 3.1. [69] *Cho $\mathcal{F}$ là ánh xạ nghịch đảo trên* GF(2^m). *Giá trị lớn nhất của tỷ lệ truyền của $\mathcal{F}$ là $2^{\varepsilon-m}$ với $\varepsilon = 2$ nếu m là số chẵn, $\varepsilon = 1$ nếu m là số lẻ. Giá trị lớn nhất của độ tương quan của $\mathcal{F}$ là $2^{(\varepsilon-m)/2}$ với $\varepsilon = 2$ nếu m là số chẵn, $\varepsilon = 1$ nếu m là số lẻ.*

Bổ đề 3.2. *Mọi S-box S_φ được sử dụng trong XAES có giá trị lớn nhất của tỷ lệ truyền là $2^{\varepsilon-m}$ và giá trị lớn nhất của độ tương quan là $2^{(\varepsilon-m)/2}$ với $\varepsilon = 2$ nếu m là số chẵn, $\varepsilon = 1$ nếu m là số lẻ.*

✍***Chứng minh***: Trong S-box S_φ gồm 3 bước biến đổi:

- biến đổi affine $\mathcal{A}_\varphi^{(0)}$,
- biến đổi nghịch đảo trên GF(2^m), ký hiệu là $\mathcal{F}$
- biến đổi affine $\mathcal{A}_\varphi^{(1)}$

Vậy, $S_\varphi = \mathcal{A}_\varphi^{(1)} \circ \mathcal{F} \circ \mathcal{A}_\varphi^{(0)}$. Do $\mathcal{A}_\varphi^{(0)}$ và $\mathcal{A}_\varphi^{(1)}$ là đẳng cấu nhóm nên S_φ và $\mathcal{F}$ có cùng giá trị lớn nhất của tỷ lệ truyền và cùng giá trị lớn nhất của độ tương quan [17]. ❑

Định lý 3.5. *Mọi vết sai phân lan truyền qua* 4r *chu kỳ của XAES có tỷ lệ truyền tối đa bị chặn trên là $2^{-r(m-\varepsilon)\beta_\theta^2}$ với $\varepsilon = 2$ nếu m chẵn, $\varepsilon = 1$ nếu m lẻ.*

✍***Chứng minh***: Trong [19], J. Daemen đã chứng minh rằng tỷ lệ truyền của vết sai phân có thể xấp xỉ bằng tích số tỷ lệ truyền của các S-box hoạt động. Theo **Định lý 3.4**, số lượng S-box hoạt động của vết sai phân lan truyền qua 4r chu kỳ tối thiểu là $r \times \beta_\theta^2$. Kết hợp với **Bổ đề 3.2**, ta có được kết luận cần chứng minh. ❑

Định lý 3.5 là cơ sở để chứng minh tổng quát độ an toàn của XAES đối với phương pháp sai phân và phương pháp tuyến tính để phân tích mã. Gọi n là độ dài (tính bằng bit) của khối dữ liệu, theo J. Daemen và V. Rijmen, phương pháp sai phân chỉ hiệu quả nếu tồn tại vết sai phân có tỷ lệ truyền qua hầu hết các chu kỳ lớn hơn

đáng kể so với 2^{-n}, còn phương pháp tuyến tính chỉ hiệu quả nếu tồn tại vết tuyến tính có độ tương quan qua hầu hết các chu kỳ lớn hơn đáng kể so với $2^{-n/2}$ ([16],[17]). Thông thường, trong việc chứng minh, vết (sai phân/tuyến tính) được khảo sát lan truyền qua $T = Nr - 1$ hay $T = Nr - 2$ chu kỳ [50]; một hoặc hai chu kỳ còn lại được dùng để tăng cường giới hạn biên an toàn cho thuật toán.

Trong thuật toán XAES, kích thước của khối dữ liệu là $m \times Nw \times Nb$ bit. Ta cần chứng minh là mọi vết sai phân/tuyến tính lan truyền qua $T \le Nr - 2$ chu kỳ của XAES có tỷ lệ truyền/độ tương quan tối đa không vượt quá $2^{-m \times Nw \times Nb}$ và $2^{-(m \times Nw \times Nb)/2}$.

Để chứng minh điều này, trước tiên, ta sẽ chứng minh tính chất tổng quát sau:

Định lý 3.6. *Trong thuật toán XAES với* $r \ge 1$ *và* $m \ge \varepsilon r$ *(với* $\varepsilon = 2$ *nếu* m *chẵn,* $\varepsilon = 1$ *nếu* m *lẻ), mọi vết sai phân truyền qua* $T = 4r$ *chu kỳ đều có tỷ lệ truyền bị chặn trên bởi* $2^{-(r-1) \times m \times Nw^2}$.

$$\forall r \ge 1, \forall Nw \ge 1, \forall m \ge \varepsilon r, R_{4r}^{P}(a \to b) \le 2^{-(r-1) \times m \times Nw^2} \tag{3.27}$$

✍***Chứng minh***:

Theo **Định lý 3.5**, mọi vết sai phân lan truyền qua $4r$ chu kỳ có tỷ lệ truyền tối đa không vượt quá $2^{-r \times (m-\varepsilon) \times \beta_\theta^2}$. Ta chứng minh:

$$\forall r \ge 1, \forall Nw \ge 1, \forall m \ge \varepsilon r, \quad 2^{-r(m-\varepsilon)\beta_\theta^2} \le 2^{-(r-1)mNw^2}$$

Ta có:

$$m \ge \varepsilon r \quad \Rightarrow rm - r\varepsilon \ge rm - m$$
$$\Rightarrow r(m - \varepsilon) \ge (r - 1)m$$

Do $\beta_\theta \ge Nw > 0$ nên suy ra:

$$r(m - \varepsilon)\beta_\theta^2 \ge (r - 1)mNw^2 \ge 0$$

$$\Rightarrow \quad 2^{-r(m-\varepsilon)\beta_\theta^2} \le 2^{-(r-1)mNw^2}$$

Từ đó suy ra điều phải chứng minh ❑

Kết luận:

- Xét trường hợp r =2, do $m \geq 4 \geq \varepsilon r$ nên có thể áp dụng **Định lý 3.6**: Mọi vết sai phân truyền qua T = 8 chu kỳ đều có tỷ lệ truyền bị chặn trên bởi $2^{-m \times Nw^2}$. Nếu sử dụng khối dữ liệu có $Nb = Nw$, giới hạn này đảm bảo tính an toàn đối với phương pháp tấn công sai phân. Điều này giải thích lý do vì sao trong trường hợp $Nb = Nw$, XAES sử dụng $T + 2 = 10$ chu kỳ biến đổi.
- Xét trường hợp r =3: Nếu $m \geq 6$ thì $m \geq \varepsilon r$. Nếu $m = 5$ thì $\varepsilon = 1$ và $m \geq \varepsilon r$. Vậy, khi $m \geq 5$, có thể áp dụng **Định lý 3.6** với r =3 để kết luận: Mọi vết sai phân truyền qua T = 12 chu kỳ đều có tỷ lệ truyền bị chặn trên bởi 2^{-2mNw^2} . Nếu sử dụng khối dữ liệu có $Nb = 2Nw$, giới hạn này đảm bảo tính an toàn đối với phương pháp tấn công sai phân. Điều này giải thích lý do vì sao trong trường hợp $1.5 < Nb/Nw \leq 2$, XAES sử dụng $T + 2 = 14$ chu kỳ biến đổi.
- Xét trường hợp $r = 4$ và $m = 4$: khi đó, theo **Định lý 3.5**, mọi vết sai phân truyền qua T = 16 chu kỳ đều có tỷ lệ truyền bị chặn trên bởi $2^{-2mNw^2} = 2^{-8Nw^2}$, đảm bảo tính an toàn đối với phương pháp tấn công sai phân cho trường hợp khối dữ liệu có $Nb = 2Nw$.

3.2.4 Độ tương quan của vết tuyến tính trong XAES

Định lý 3.7. *Mọi vết tuyến tính lan truyền qua* $4r$ *chu kỳ của XAES có độ tương quan tối đa bị chặn trên là* $2^{-r \times (m-\varepsilon) \times \beta_\theta^2 / 2}$ *với* $\varepsilon = 2$ *nếu* m *chẵn,* $\varepsilon = 1$ *nếu* m *lẻ.*

✍***Chứng minh***: Trong [19], J. Daemen đã chứng minh rằng độ tương quan của vết tuyến tính có thể xấp xỉ bằng tích số của độ tương quan giữa đầu ra-đầu vào của các S-box hoạt động. Theo **Định lý 3.4**, số lượng S-box hoạt động của vết tuyến tính lan truyền qua $4r$ chu kỳ tối thiểu là $r\beta_\theta^2$. Kết hợp với **Bổ đề 3.2**, ta có được kết luận cần chứng minh.

❑

Định lý 3.8. *Trong thuật toán XAES với* $r \geq 1$ *và* $m \geq \varepsilon r$ *(với* $\varepsilon = 2$ *nếu* m *chẵn,* $\varepsilon = 1$ *nếu* m *lẻ), mọi vết tuyến tính truyền qua* $T = 4r$ *chu kỳ đều có độ tương quan bị chặn trên bởi* $2^{-(r-1)mNw^2/2}$.

$$\forall r \geq 1, \forall Nw \geq 1, \forall m \geq \varepsilon r, C_{4r}^{P}(a,b) \leq 2^{-(r-1)mNw^2/2} \quad (3.28)$$

✍***Chứng minh***:

Theo **Định lý 3.7**, mọi vết tuyến tính lan truyền qua $4r$ chu kỳ có độ tương quan tối đa không vượt quá $2^{-r\times(m-\varepsilon)\times\beta_\theta^2/2}$. Ta chứng minh:

$$\forall r \geq 1, \forall Nw \geq 1, \forall m \geq \varepsilon r, \quad 2^{-r(m-\varepsilon)\beta_\theta^2/2} \leq 2^{-(r-1)mNw^2/2}$$

Ta có: $m \geq \varepsilon r \Rightarrow rm - r\varepsilon \geq rm - m$

$$\Rightarrow r(m-\varepsilon) \geq (r-1)m$$

Do $\beta_\theta \geq Nw > 0$ nên suy ra:

$$r(m-\varepsilon)\beta_\theta^2 \geq (r-1)mNw^2 \geq 0$$

$$\Rightarrow 2^{-r(m-\varepsilon)\beta_\theta^2/2} \leq 2^{-(r-1)mNw^2/2}$$

Từ đó suy ra điều phải chứng minh ❑

Kết luận:

- Xét trường hợp r =2, do $m \geq 4 \geq \varepsilon r$ nên có thể áp dụng **Định lý 3.8**: Mọi vết tuyến tính truyền qua T = 8 chu kỳ đều có độ tương quan bị chặn trên bởi $2^{-m\times Nw^2/2}$. Nếu sử dụng khối dữ liệu có $Nb = Nw$, giới hạn này đảm bảo tính an toàn đối với phương pháp tấn công tuyến tính. Điều này giải thích lý do vì sao trong trường hợp $Nb = Nw$, XAES sử dụng $T + 2 = 10$ chu kỳ biến đổi.
- Xét trường hợp r =3: Nếu $m \geq 6$ thì $m \geq \varepsilon r$. Nếu $m = 5$ thì $\varepsilon = 1$ và $m \geq \varepsilon r$. Vậy, khi $m \geq 5$, có thể áp dụng **Định lý 3.8** với r =3 để kết luận: Mọi vết tuyến tính truyền qua T = 12 chu kỳ đều có độ tương quan bị chặn trên bởi 2^{-mNw^2}. Nếu sử dụng khối dữ liệu có $Nb = 2Nw$, giới hạn này đảm bảo tính an toàn đối với phương pháp tấn công tuyến tính. Điều này giải thích lý do vì sao trong trường hợp $1.5 < Nb/Nw \leq 2$, XAES sử dụng $T + 2 = 14$ chu kỳ biến đổi.
- Xét trường hợp $r = 4$ và $m = 4$: khi đó, theo **Định lý 3.7**, mọi vết tuyến tính truyền qua T = 16 chu kỳ đều có độ tương quan bị chặn trên bởi $2^{-mNw^2} = 2^{-8Nw^2/2}$, đảm bảo tính an toàn đối với phương pháp tấn công tuyến tính cho trường hợp khối dữ liệu có $Nb = 2Nw$. Điều này giải thích lý do vì sao trong trường hợp m=4 và $1.5Nw < \max\{Nb, Nk\} \leq 2Nw$, XAES sử dụng $T + 2 = 18$ chu kỳ biến đổi.

3.3 Kết luận

Sử dụng hướng tiếp cận truyền thống với vết sai phân đơn và vết tuyến tính đơn trong việc chứng minh tính an toàn của thuật toán theo chiến lược vết rộng đối với phương pháp phân tích mã sai phân và phương pháp phân tích mã tuyến tính, chúng tôi đã chứng minh tổng quát các công thức xác định chặn trên của tỷ lệ truyền của vết sai phân đơn và chặn trên của độ tương quan của vết tuyến tính đơn lan truyền qua $T = 4r$ chu kỳ của XAES. Áp dụng các kết quả tổng quát này cho phép kết luận mỗi thể hiện của XAES (với các giá trị cụ thể của tham số cấu trúc và tham số xử lý) đều an toàn đối với phương pháp sai phân và phương pháp tuyến tính, đồng thời giải thích cơ sở việc chọn số chu kỳ mã hóa cho mỗi thể hiện của XAES.

Một số vấn đề mở:

- Việc chứng minh công thức tổng quát xác định chặn trên của tỷ lệ truyền của vết sai phân (**Định lý 3.5**) và độ tương quan của vết tuyến tính (**Định lý 3.7**) hoàn toàn độc lập với giá trị của branch number β_θ của MixColumns. Khi áp dụng vào giải thuật XAES với giá trị β_θ là Nw hay $Nw + 1$, kết quả nhận được gồm **Định lý 3.6** và **Định lý 3.8**. Trong trường hợp các ánh xạ tuyến tính được chọn dùng trong MixColumns không đảm bảo điều kiện $\beta_\theta \geq Nw$, có thể áp dụng **Định lý 3.5** và **Định lý 3.7** để xác định số lượng chu kỳ mã hóa tối thiểu để thuật toán an toàn đối với tấn công sai phân và tấn công tuyến tính. Điều này mở ra khả năng xây dựng các thuật toán mã hóa khối với tầng khuếch tán (có vai trò tương tự như MixColumns trong XAES) có hệ số branch number bất kỳ.
- Bên cạnh hướng tiếp cận sử dụng vết lan truyền đơn được áp dụng thành công trong chứng minh tính an toàn của các thuật toán như Shark[90], Square[18], Rijndael[16], GrandCru[8], Anubis[3], Khazad[91]…, hướng tiếp cận sử dụng tập vết lan truyền được đề xuất và nhiều công trình đã tập trung theo hướng này để khảo sát lại các thuật toán mã hóa khối đã được đề xuất [12][38][44][46][47][71][72]. Chính vì vậy, ngoài việc chứng minh tính an toàn của XAES với vết lan truyền đơn, chúng tôi đã chứng minh tính an toàn của XAES theo hướng tiếp cận mới sử dụng tập vết lan truyền với kết quả được trình bày trong Chương 4.

Chương 4

Tính an toàn của XAES dựa trên xác suất sai phân của tập vết sai phân và xác suất tuyến tính của bao tuyến tính

Tóm tắt chương:

✍ *Nội dung của chương 4 trình bày hướng tiếp cận xác định chặn trên của xác suất sai phân của tập vết sai phân và chặn trên của xác suất tuyến tính của bao vết tuyến tính:*

- *Trình bày tóm tắt về hướng tiếp cận và các công trình liên quan chứng minh tính an toàn của giải thuật sử bằng cách xác định chặn trên của xác suất sai phân của tập vết sai phân và chặn trên của xác suất tuyến tính của bao tuyến tính*
- *Trình bày kết quả xác định chặn trên tổng quát của xác suất sai phân của tập vết sai phân và chặn trên tổng quát của xác suất tuyến tính của bao tuyến tính đối với XAES:*
  - *Khảo sát 2 chu kỳ của XAES để xác định chặn trên của xác suất (sai phân/tuyến tính) của tập vết sai phân/tuyến tính với kết quả trọng tâm là **Định lý 4.2** và **Định lý 4.5**.*
  - *Xác định chặn trên tổng quát cho xác suất (sai phân/tuyến tính) của tập vết sai phân/tuyến tính qua 4 chu kỳ của XAES với kết quả trọng tâm được trình bày trong **Định lý 4.3** và **Định lý 4.6**.*
  - *Xác định chặn trên tổng quát cho xác suất (sai phân/tuyến tính) của tập vết sai phân/tuyến tính qua $r \geq 4$ chu kỳ của XAES với kết quả trọng tâm được trình bày trong **Định lý 4.4** và **Định lý 4.7**.*

4.1 Hướng tiếp cận sử dụng tập vết sai phân và bao tuyến tính

4.1.1 Giới thiệu về hướng tiếp cận sử dụng tập vết sai phân và bao tuyến tính

Ý tưởng sử dụng một vết sai phân hay vết tuyến tính đơn để phân tích mã được áp dụng thành công trong trường hợp thuật toán DES [31]. Chính vì vậy đã dẫn đến phương pháp chứng minh độ an toàn của thuật toán mã hóa theo khối bằng cách chứng minh không tồn tại vết sai phân (hoặc tuyến tính) đơn có tỷ lệ truyền (hoặc độ tương quan) đủ lớn.

Bên cạnh hướng tiếp cận sử dụng vết lan truyền đơn, hướng tiếp cận sử dụng tập hợp các vết lan truyền có cùng chung một số tính chất cũng đã được đề xuất:

Định nghĩa 4.1 [56]**.** *Cho vector sai phân* Δa *và* Δb*.* ***Tập vết sai phân*** *tương ứng với* Δa *và* Δb*, ký hiệu là* DIFF(Δa, Δb)*, là tập hợp tất cả các vết sai phân* $\Omega = \left(\Delta_0 \xrightarrow{\rho_1} \Delta_1 \xrightarrow{\rho_2} \ldots \xrightarrow{\rho_{T-1}} \Delta_{T-1} \xrightarrow{\rho_T} \Delta_T\right)$ *lan truyền qua* $T \geq 1$ *chu kỳ có giá trị sai phân ở đầu vào chu kỳ* 1 *là* $\Delta_0 = \Delta a$ *và giá trị sai phân ở đầu ra chu kỳ* T *là* $\Delta_T = \Delta b$*.*

Định nghĩa 4.2 [70]**.** *Cho vector mặt nạ* Γa *và* Γb*.* ***Bao tuyến tính*** *tương ứng với* Γa *và* Γb*, ký hiệu là* ALH(Γa, Γb)*, là tập hợp tất cả các vết tuyến tính* $\Xi = \left(\Gamma_0 \xleftarrow{\rho_1} \Gamma_1 \xleftarrow{\rho_2} \ldots \xleftarrow{\rho_{T-1}} \Gamma_{T-1} \xleftarrow{\rho_T} \Gamma_T\right)$ *lan truyền qua* $T \geq 1$ *chu kỳ có mặt nạ ở đầu vào chu kỳ* 1 *là* $\Gamma_0 = \Gamma a$ *và mặt nạ ở đầu ra chu kỳ* T *là* $\Gamma_T = \Gamma b$*.*

Để chứng minh tính an toàn của thuật toán mã hóa với phương pháp sai phân hoặc tuyến tính sử dụng tập vết sai phân hoặc bao tuyến tính, cần chứng minh là mọi tập vết sai phân/bao tuyến tính lan truyền qua T chu kỳ (thường $T = Nr - 2$ hay $T = Nr - 1$) đều có xác suất sai phân/tuyến tính đủ nhỏ.

Tuy nhiên, cho đến hiện tại, việc xác định chính xác giá trị tối đa của xác suất sai phân và xác suất tuyến tính của tập hợp vết lan truyền qua các chu kỳ vẫn chưa thể thực hiện trên thực tế được [49]. Vì vậy, nhiều công trình đã tập trung theo hướng xác định chặn trên của giá trị tối đa của xác suất sai phân cũng như xác suất tuyến tính ([12][38][44][46][47][71][72]).

Theo hướng tiếp cận này, chúng tôi đã xác định và chứng minh công thức tổng quát giá trị chặn trên của giá trị tối đa của xác suất sai phân và xác suất tuyến tính, từ đó chứng minh tổng quát tính an toàn của XAES đối với phương pháp sai phân và phương pháp tuyến tính trong phân tích mã độc lập với các giá trị cụ thể của các tham số về cấu trúc và tham số về xử lý của XAES.

4.1.2 Một số khái niệm và tính chất cơ bản

Định nghĩa 4.3. ([38]) *Cho* $\Delta x, \Delta y \in \{0,1\}^n$. *Xác suất sai phân (DP) của ánh xạ* $S : \{0,1\}^n \to \{0,1\}^n$ *được định nghĩa là:*

$$\begin{aligned} DP^S(\Delta x, \Delta y) &= \Pr_x\{S(x) \oplus S(x \oplus \Delta x) = \Delta y\} \\ &= \frac{\#\{x \in Z_2^n : S(x) \oplus S(x \oplus \Delta x) = \Delta y\}}{2^n} \end{aligned} \tag{4.1}$$

Từ định nghĩa của xác suất sai phân, suy ra:

$$\sum_{\Delta u \in \{0,1\}^n} DP^S(\Delta x, \Delta u) = \sum_{\Delta u \in \{0,1\}^n} DP^S(\Delta u, \Delta y) = 1 \tag{4.2}$$

Định nghĩa 4.4. ([38]) *Cho* $\Gamma x, \Gamma y \in \{0,1\}^n$. *Xác suất tuyến tính (LP) của ánh xạ* $S : \{0,1\}^n \to \{0,1\}^n$ *được định nghĩa là:*

$$\begin{aligned} LP^S(\Gamma x, \Gamma y) &= \left(2 \cdot \Pr_x\{\Gamma x \bullet x = \Gamma y \bullet S(x)\} - 1\right)^2 \\ &= \left(\frac{\#\{x \in Z_2^n : \Gamma x \bullet x = \Gamma y \bullet S(x)\}}{2^{n-1}} - 1\right)^2 \end{aligned} \tag{4.3}$$

Từ định lý Parseval [63], suy ra

$$\sum_{\Gamma u \in \{0,1\}^n} LP^S(\Gamma x, \Gamma u) = \sum_{\Gamma u \in \{0,1\}^n} LP^S(\Gamma u, \Gamma y) = 1 \tag{4.4}$$

Định nghĩa 4.5. ([38]) *Xác suất sai phân tối đa p và xác suất tuyến tính tối đa q của S-box* S_φ *được xác định như sau:*

$$p = \max_{\Delta x \neq 0, \Delta y} DP^{S_\varphi}(\Delta x, \Delta y) \tag{4.5}$$

$$q = \max_{\Gamma y \neq 0, \Gamma x} LP^{S_\varphi}(\Gamma x, \Gamma y) \tag{4.6}$$

Định nghĩa 4.6. [38] *Cho ánh xạ tuyến tính* $\mathcal{L}: \mathrm{GF}(2^m)^{Nw} \to \mathrm{GF}(2^m)^{Nw}$. *Đặt* $\Delta x, \Delta y \in GF(2^m)^{Nw}$ *lần lượt là vector sai phân ở đầu vào và đầu ra của* $\mathcal{L}$, $\Gamma x, \Gamma y \in GF(2^m)^{Nw}$ *lần lượt là mặt nạ ở đầu vào và đầu ra của* $\mathcal{L}$.

Branch Number (sai phân) của $\mathcal{L}$ *được định nghĩa như sau:*

$$\mathcal{B}_d(\mathcal{L}) = \min_{\Delta x \neq 0} \{wt(\Delta x) + wt(\Delta y)\} \tag{4.7}$$

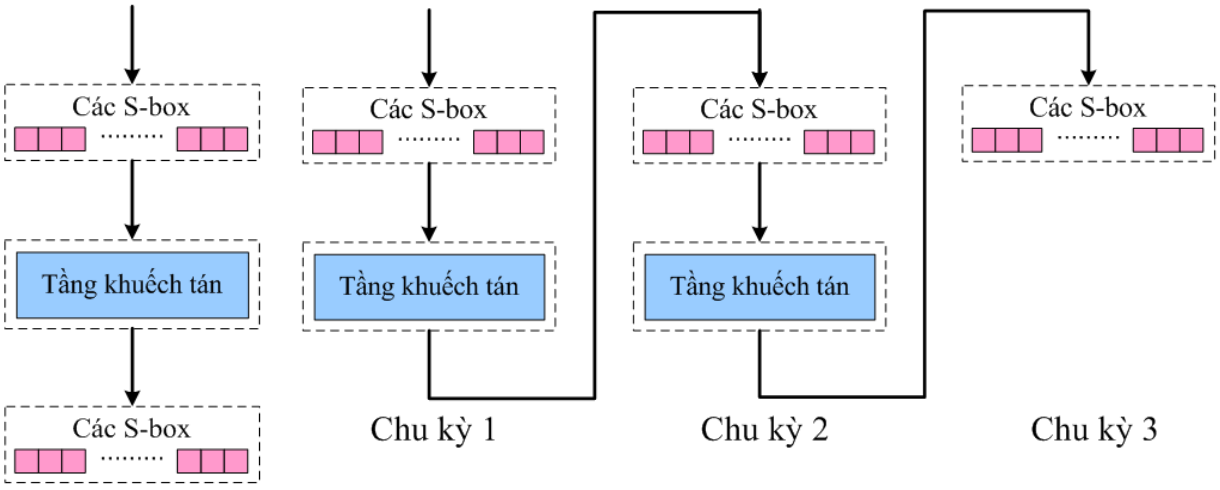
Branch Number (tuyến tính) của $\mathcal{L}$ *được định nghĩa như sau:*

$$\mathcal{B}_l(\mathcal{L}) = \min_{\Gamma y \neq 0} \{wt(\Gamma x) + wt(\Gamma y)\} \tag{4.8}$$

Định nghĩa 4.7. [38] *Cho vector* $x = (x_1, \ldots, x_n) \in Z^n$. *Mẫu các thành phần khác 0 của* x, *ký hiệu là* χ_x, *được định nghĩa là* $\chi_x = (\chi_1, \ldots \chi_n) \in Z_2^n$ *với* $\chi_i = 0$ *nếu* $x_i = 0$ *và* $\chi_i = 1$ *nếu* $x_i \neq 0$.

❑Ví dụ: Với $x = (x_1, x_2, x_3, x_4)$ với $x_1 \neq 0$, $x_2 \neq 0$, $x_3 = x_4 = 0$ thì $\chi_x = (1,1,0,0)$.

4.2 Các công trình liên quan



a) SDS (1 chu kỳ) b) SDS (3 chu kỳ)

Hình 4.1. Một số ví dụ về hàm SDS

Thao tác cộng khóa vào mỗi chu kỳ mã hóa không có ảnh hưởng trong việc khảo sát sự lan truyền của tập vết sai phân cũng như bao tuyến tính [38][44]. Chính vì vậy, trong [38][44], S. Hong đã đề xuất việc khảo sát xác suất của tập vết sai phân và xác

suất của bao tuyến tính lan truyền trong hàm SDS (Substitution – Diffusion – Substitution). Về mặt bản chất, kiến trúc SDS chính là kiến trúc SPN nhưng bỏ qua thao tác cộng khóa trong mỗi chu kỳ.

Hàm SDS (1 chu kỳ) bao gồm tầng thay thế, tầng khuếch tán và tầng thay thế (xem Hình 4.1a). Trong hàm SDS (gồm $r > 1$ chu kỳ), $r - 1$ chu kỳ đầu tiên gồm tầng thay thế và tầng khuếch tán, riêng chu kỳ cuối cùng chỉ gồm tầng thay thế. Hình 4.1b minh họa hàm SDS gồm 3 chu kỳ.

Trong hàm SDS, tầng thay thế gồm n S-box (không nhất thiết phải giống nhau). Gọi p và q lần lượt là xác suất sai phân và xác suất tuyến tính tối đa của S-box trong tầng thay thế. Kết quả quan trọng của công trình [38] và được phân tích chi tiết trong [44] là chứng minh ***xác suất của tập vết sai phân*** và ***xác suất của bao tuyến tính*** lan truyền qua hàm SDS (gồm ***2 chu kỳ***) lần lượt được chặn trên bởi p^n và q^n nếu tầng khuếch tán có branch number (sai phân/tuyến tính) là $n + 1$ (có khả năng khuếch tán tối đa), và được chặn trên bởi p^{n-1} và q^{n-1} nếu tầng khuếch tán có branch number (sai phân/khuếch tán) là n (có khả năng khuếch tán gần tối đa).

Dựa trên các kết quả đối với hàm SDS, trong [71], S. Park đã khảo sát thuật toán Rijndael (trong trường hợp khối dữ liệu được biểu diễn bằng ma trận 4 × 4) và một số thuật toán có cấu trúc tương tự (Crypton [59], Square[18]) để xác định chặn trên cho xác suất của tập vết sai phân và chặn trên cho xác suất của bao tuyến tính. S. Park giới thiệu 1 tập các thuật toán tựa-Rijndael được tạo ra bằng cách sửa giải thuật Rijndael với một hay một số cách thay đổi sau:

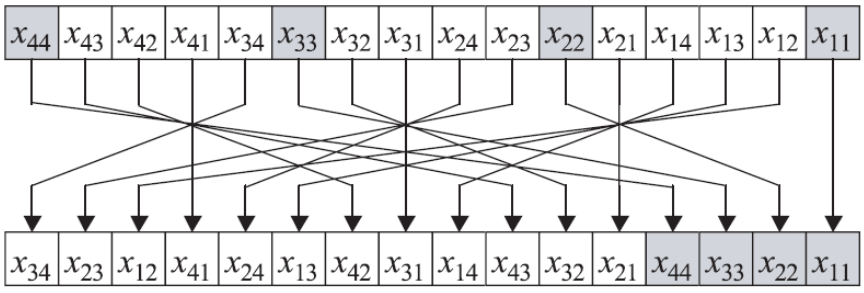
- Thay thế S-box (8 × 8) trong Rijndael bằng S-box (8 × 8) bất kỳ có xác suất sai phân tối đa $p \le 2^{-3}$ và xác suất tuyến tính tối đa $q \le 2^{-3}$.
- Thay thế ánh xạ ShiftRows bằng một ánh xạ π bất kỳ sao cho mỗi cột sau khi biến đổi nhận được byte từ cả 4 cột trước khi biến đổi.
- Có thể chọn ánh xạ θ_i trong tầng khuếch tán (tương ứng MixColumns trong Rijndael) là một ánh xạ bất kỳ trên $GF(2^8)^4$ có branch number là 5 (khuếch tán tối đa).

Điểm khác biệt chính giữa kiến trúc SDS và cấu trúc của các thuật toán tựa-Rijndael được khảo sát trong [71] như sau:

- Trong thuật toán tựa-Rijndael có thêm biến đổi tuyến tính π (tương ứng với biến đổi ShiftRows trong Rijndael) trước tầng khuếch tán (xem Hình 4.2).
- Trong SDS, tầng khuếch tán là một ánh xạ tuyến tính duy nhất, nhận đầu vào là kết quả của các S-box trong tầng thay thế liền trước. Trong khi đó, đối với thuật toán tựa-Rijndael, tầng khuếch tán θ (tương ứng biến đổi MixColumns trong Rijndael) gồm 4 ánh xạ tuyến tính để xử lý riêng θ_1, θ_2, θ_3, θ_4 từng cột trong khối dữ liệu cần mã hóa[1] (xem Hình 4.3).

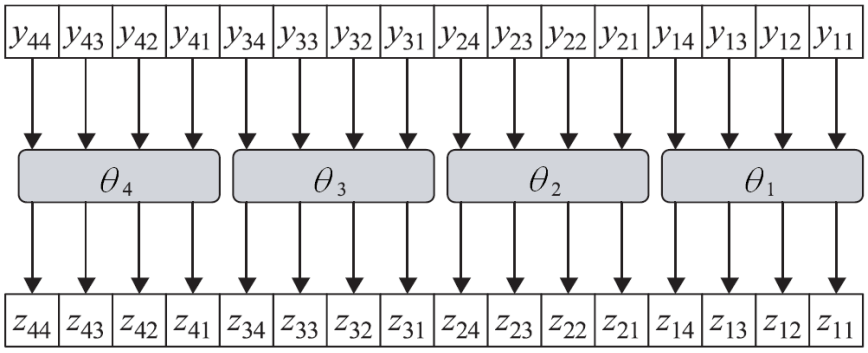
Trong [71], S. Park đã chứng minh rằng, đối với thuật toán tựa-Rijndael thuộc tập được xét, xác suất của tập vết sai phân lan truyền qua 4 chu kỳ là $4p^{19} + 6p^{18} + 4p^{17} + p^{16}$ nếu xác suất sai phân tối đa của S-box trong tầng thay thế đủ nhỏ ($p \le 2^{-3}$) và xác suất của bao tuyến tính lan truyền qua 4 chu kỳ là $4q^{19} + 6q^{18} + 4q^{17} + q^{16}$ nếu xác suất tuyến tính tối đa của S-box trong tầng thay thế đủ nhỏ ($q \le 2^{-3}$). Như vậy, cơ sở của kết quả trong [71] là xác suất sai phân tối đa và xác suất tuyến tính tối đa của S-box.

Kết quả mà S. Park trình bày trong [71] có ưu điểm là được phát biểu tổng quát với tham số (p hoặc q) nên có khả năng áp dụng cho nhiều thuật toán thuộc tập thuật toán tựa-Rijndael được xét trong [71]. Tuy nhiên, kết quả này chưa được tổng quát theo khả năng mở rộng về kích thước của thuật toán tựa-Rijndael.



Hình 4.2. Biến đổi π trong Rijndael (trường hợp khối 128 bit)

[1] Trong Rijndael sử dụng cùng một ánh xạ tuyến tính để xử lý các cột trong biến đổi MixColumns ($\theta_1 = \theta_2 = \theta_3 = \theta_4$)



Hình 4.3. Biến đổi θ với 4 biến đổi tuyến tính θ_1, θ_2, θ_3, θ_4 trong cấu trúc tựa-Rijndael được S. Park trình bày trong [71] (trường hợp khối 128 bit)

Trong [72], S. Park xác định chặn trên chặt hơn cho xác suất của tập vết sai phân và xác suất của bao tuyến tính đối với Rijndael so với kết quả trong [71]. Thay vì dựa trên xác suất (sai phân/tuyến tính) tối đa của S-box, S. Park khảo sát mối quan hệ giữa chặn trên của xác suất sai phân/tuyến tính lan truyền qua 2 chu kỳ của cấu trúc SPN (sử dụng duy nhất một ánh xạ tuyến tính θ_i)[1] với chặn trên của xác suất của tập vết sai phân/bao tuyến tính. Như vậy, trong cách tiếp cận này, thay vì sử dụng ***giá trị tối đa*** của xác suất sai phân/tuyến tính của S-box, S. Park đã dùng ***giá trị*** và ***phân bố*** của xác suất sai phân/tuyến tính của S-box. Nhờ đó, chặn trên của xác suất của tập vết sai phân/bao tuyến tính được xác định chặt hơn.

Dựa trên ý tưởng của cách tiếp cận trong [72], chúng tôi đã áp dụng và chứng minh tổng quát công thức cho XAES để xác định chặn trên của xác suất của tập vết sai phân (phần 4.3) và chặn trên của xác suất của bao tuyến tính (phần 4.3.4).

Để nội dung trình bày được cô đọng, trong những phần tiếp theo của Chương 4, chúng tôi sử dụng các ký hiệu đã được đề xuất trong phần 2.1.2 để thay thế cho tên gọi của các biến đổi trong XAES, cụ thể như sau: φ (biến đổi SubBytes), π (biến đổi ShiftRows), θ (biến đổi MixColumns), σ (biến đổi AddRoundKey).

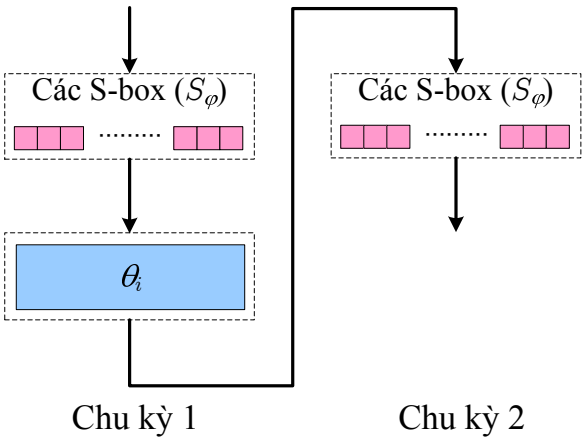
[1] cần lưu ý là tầng khuếch tán trong Rijndael sử dụng Nb = 4, 6, hay 8 ánh xạ tuyến tính giống nhau, trong XAES sử dụng Nb ánh xạ tuyến tính không nhất thiết giống nhau.

4.3 Giá trị chặn trên của xác suất sai phân của tập vết sai phân

4.3.1 Xác suất sai phân của tập vết sai phân lan truyền qua 2 chu kỳ của hàm SDS được xây dựng từ XAES

Đặt β_d là giá trị nhỏ nhất của branch number (sai phân) của các biến đổi tuyến tính được sử dụng trong tầng khuếch tán θ của XAES.

$$\beta_d = \min_{i=0,1,\ldots,Nb-1} \{ \mathcal{B}_d(\theta_i) \} \tag{4.9}$$



Hình 4.4. Hàm SDS gồm 2 chu kỳ với tầng thay thế là các S-box giống nhau (S_φ) và tầng khuếch tán gồm 1 ánh xạ tuyến tính θ_i

Định lý 4.1. ([72-Định lý 1]) *Xét hàm SDS (gồm 2 chu kỳ) với tầng thay thế sử dụng các S-box giống nhau (S_φ) và tầng khuếch tán là ánh xạ tuyến tính θ_i (xem Hình 4.4). Xác suất sai phân tối đa của tập vết sai phân qua hàm SDS này được chặn trên bởi:*

$$DP_2^{\theta_i}(a,b) \le$$

$$\max\left\{ \max_{1\le i\le n} \max_{1\le u\le 2^m-1} \sum_{j=1}^{2^m-1} \left\{ DP^{S_\varphi}(u,j) \right\}^{\mathcal{B}_d(\theta_i)}, \max_{1\le i\le n} \max_{1\le u\le 2^m-1} \sum_{j=1}^{2^m-1} \left\{ DP^{S_\varphi}(j,u) \right\}^{\mathcal{B}_d(\theta_i)} \right\} \tag{4.10}$$

Đặt $\Omega_d = \max\left\{ \max_{1\le i\le n} \max_{1\le u\le 2^m-1} \sum_{j=1}^{2^m-1} \left\{ DP^{S_\varphi}(u,j) \right\}^{\beta_d}, \max_{1\le i\le n} \max_{1\le u\le 2^m-1} \sum_{j=1}^{2^m-1} \left\{ DP^{S_\varphi}(j,u) \right\}^{\beta_d} \right\}$.

Từ **Định lý 4.1**, suy ra:

Bổ đề 4.1. *Xác suất sai phân tối đa của tập vết sai phân qua hàm SDS (gồm 2 chu kỳ) với tầng thay thế sử dụng các S-box giống nhau* (S_φ) *và tầng khuếch tán là ánh xạ tuyến tính* θ_i *được chặn trên bởi:*

$$DP_2^{\theta_i}(a,b) \le \Omega_d \tag{4.11}$$

✍***Chứng minh*:** Do $\beta_d \le \mathcal{B}_d(\theta_i)$ và $DP^{S_\varphi}(x,y) \le 1$ nên **Bổ đề 4.1** có thể được suy ra từ **Định lý 4.1**. ❑

4.3.2 Xác suất sai phân của tập vết sai phân lan truyền qua 2 chu kỳ của XAES

Để khảo sát xác suất sai phân của tập vết sai phân lan truyền qua $r \ge 2$ chu kỳ của XAES, ta quy ước:

- $\Delta a = (\Delta a_1, \ldots, \Delta a_{Nb})$: vector sai phân ở đầu vào, trong đó $\Delta a_j = (\Delta a_{j1}, \Delta a_{j2}, \ldots, \Delta a_{jNw})$ với $1 \le j \le Nb$.
- $\Delta b = (\Delta b_1, \ldots, \Delta b_{Nb})$: vector sai phân ở đầu ra, trong đó $\Delta b_j = (\Delta b_{j1}, \Delta b_{j2}, \ldots, \Delta b_{jNw})$ với $1 \le j \le Nb$.
- $DP_r(\Delta a, \Delta b)$: xác suất sai phân qua r chu kỳ của XAES với vector sai phân ở đầu vào và đầu ra lần lượt là Δa và Δb.

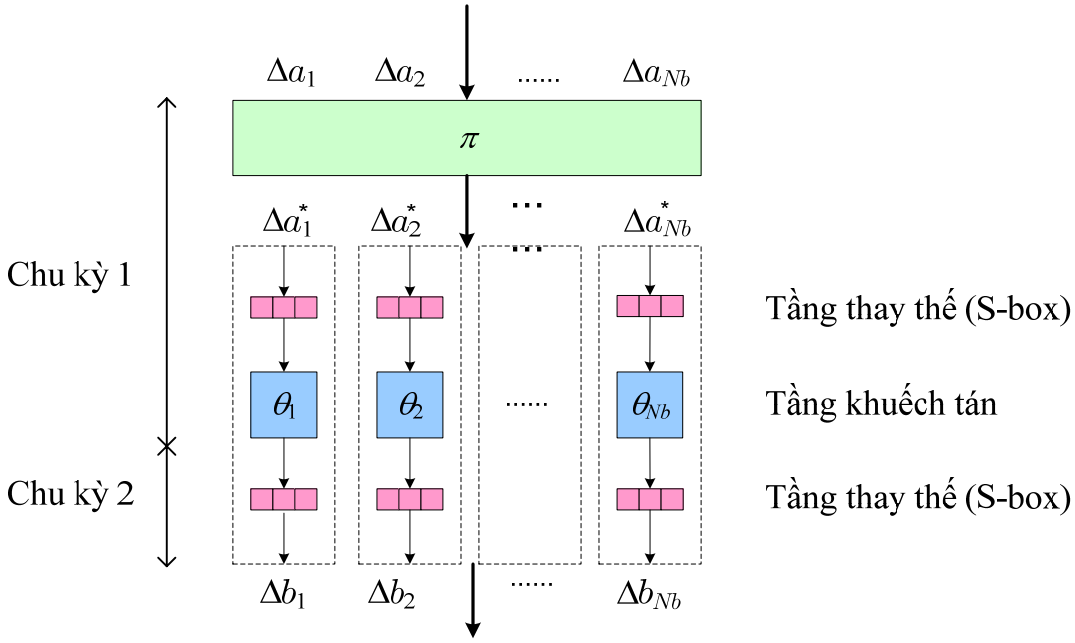
Tương tự như đối với hàm SDS, chúng tôi bỏ qua vai trò của biến đổi cộng khóa (σ) trong XAES vì không làm ảnh hưởng đến việc khảo sát sự lan truyền của tập vết sai phân và bao tuyến tính. Để có sự tương đồng khi áp dụng các kết quả khảo sát đối với hàm SDS vào XAES, khi khảo sát sự lan truyền của tập sai phân qua $r \ge 2$ chu kỳ của XAES, mỗi chu kỳ trong số $r - 1$ chu kỳ đầu gồm 3 biến đổi φ, π, θ, riêng chu kỳ r chi gồm biến đổi φ.

Định lý 4.2.

Nếu $\chi_{\pi(\Delta a)} = \chi_{\Delta b}$ *thì xác suất sai phân tối đa của tập vết sai phân qua 2 chu kỳ của XAES được chặn trên bởi:* $DP_2(\Delta a, \Delta b) \le (\Omega_d)^{wt(\pi(\Delta a))}$.

Nếu $\chi_{\pi(\Delta a)} \neq \chi_{\Delta b}$ *thì* $DP_2(\Delta a, \Delta b) = 0$.

✍***Chứng minh*:**



Hình 4.5. Minh họa Định lý 4.1

Trong 2 chu kỳ mã hóa của XAES, việc hoán vị ánh xạ φ và π trong chu kỳ 1 không làm thay đổi kết quả mã hóa. Hình 4.5 minh họa 2 chu kỳ mã hóa của XAES sau khi đảo thứ tự ánh xạ φ và π trong chu kỳ 1. Khi đó, có thể xem như 2 chu kỳ mã hóa của XAES gồm:

- Thực hiện bước tiền xử lý bằng ánh xạ π.
- Thực hiện Nb hàm SDS song song: hàm SDS thứ i ($1 \le i \le Nb$) gồm 2 chu kỳ và có tầng khuếch tán là biến đổi tuyến tính θ_i.

Đặt $\pi(\Delta a) = \left(\Delta a_1^*, \Delta a_2^*, \ldots, \Delta a_{Nb}^*\right)$. Ta có: $DP_2(\Delta a, \Delta b) = \prod_{i=1}^{Nb} DP_2^{\theta_i}\left(\Delta a_i^*, \Delta b_i\right)$ với $DP_2^{\theta_i}$ là xác suất sai phân qua hàm SDS (gồm 2 chu kỳ) với tầng khuếch tán là biến đổi tuyến tính θ_i.

Từ **Bổ đề 4.1**, suy ra giá trị chặn trên của $DP_2^{\theta_i}\left(\Delta a_i^*, \Delta b_i\right)$ như sau:

$$DP_2^{\theta_i}\left(\Delta a_i^*, \Delta b_i\right) \le \begin{cases} \Omega_d, & \text{nếu } \Delta a_i^* \neq 0, \Delta b_i \neq 0 \\ 1, & \text{nếu } \Delta a_i^* = 0, \Delta b_i = 0 \\ 0, & \text{trong trường hợp còn lại} \end{cases} \quad (4.12)$$

Nếu $\chi_{\pi(\Delta a)} = \chi_{\Delta b}$, tức là Δa_i^* và Δb_i cùng tính chất khác 0 hay cùng bằng 0, thì:

$$\begin{aligned} DP_2(\Delta a, \Delta b) &= \prod_{i \in [1, Nb] \wedge \Delta a_i \neq 0} DP_2^{\theta_i}\left(\Delta a_i^*, \Delta b_i\right) \\ &\le \prod_{i \in [1, Nb] \wedge \Delta a_i \neq 0} \Omega_d = \left(\Omega_d\right)^{wt(\pi(\Delta a))} \end{aligned} \quad (4.13)$$

Ngược lại, nếu tồn tại chỉ số $i \in [1, Nb]$ sao cho Δa_i^* và Δb_i không cùng tính chất khác 0 hay bằng 0 thì $DP_2(\Delta a \to \Delta b) = 0$. ❑

Nhận xét: Do $\Omega_d \le 1$ và $wt(\pi(\Delta a)) \ge 1$ với $\Delta a \neq 0$ nên từ **Định lý 4.2** suy ra xác suất sai phân của tập vết sai phân qua 2 chu kỳ của XAES bị chặn trên bởi Ω_d:

$$DP_2(\Delta a, \Delta b) \le \Omega_d \quad (4.14)$$

Bổ đề 4.2. *Xác suất sai phân của tập vết sai phân lan truyền qua 2 chu kỳ của XAES được chặn trên bởi* $\left(\Omega_d\right)^{wt(\Delta b)}$ *với* Δb *là vector sai phân ở đầu ra.*

$$DP_2(\Delta a, \Delta b) \le \left(\Omega_d\right)^{wt(\Delta b)} \quad (4.15)$$

✍***Chứng minh*:**

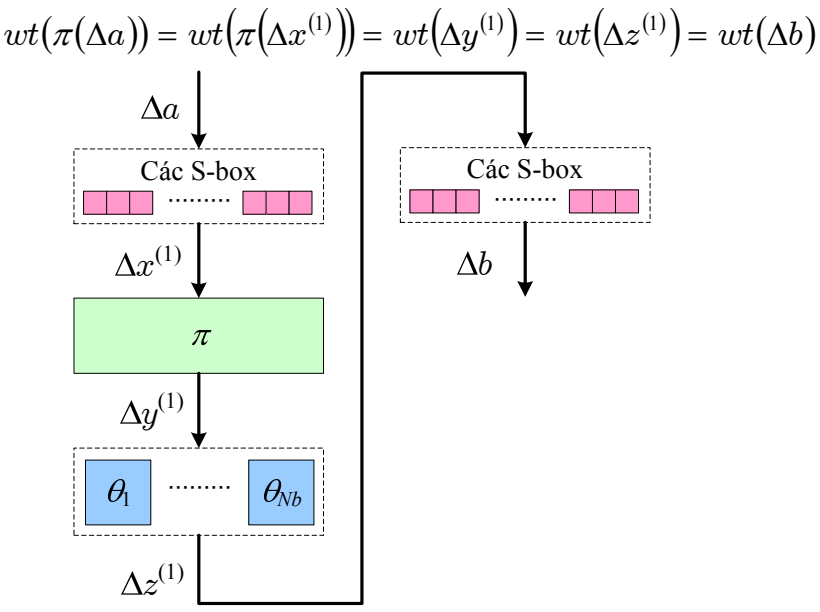
❑ ***Xét chu kỳ 2***:

- Sau khi qua các S-box : $\chi_{\Delta b_i} = \chi_{\Delta z_i^{(1)}}$, suy ra $wt\left(\Delta z^{(1)}\right) = wt(\Delta b)$, tức là $\Delta z^{(1)}$ có đúng $wt(\Delta b)$ cột khác 0.

❑ ***Xét chu kỳ 1***:

- Qua các phép biến đổi θ_i, số lượng và vị trí các cột khác 0 được giữ nguyên, suy ra $wt\left(\Delta y^{(1)}\right) = wt\left(\Delta z^{(1)}\right) = wt(\Delta b)$
- Trong phép biến đổi π: $wt\left(\pi\left(\Delta x^{(1)}\right)\right) = wt\left(\Delta y^{(1)}\right) = wt(\Delta b)$.
- Khi qua các S-box : $\chi_{\Delta x_i^{(1)}} = \chi_{\Delta a}$, suy ra: $wt(\pi(\Delta a)) = wt\left(\pi\left(\Delta x^{(1)}\right)\right) = wt(\Delta b)$

Theo **Định lý 4.2**, $DP_2(\Delta a, \Delta b) \le (\Omega_d)^{wt(\pi(\Delta a))} = (\Omega_d)^{wt(\Delta b)}$ ❑



Hình 4.6. Minh họa Bổ đề 4.2

Bổ đề 4.3. *Đặt* $k = wt\left(\chi_{\pi(\Delta a)}\right)$. *Khi đó:*

$$\sum_{\Delta x^{(2)}} DP_2\left(\Delta a, \Delta x^{(2)}\right) \le (\Omega_d)^{k-1} \tag{4.16}$$

✍***Chứng minh*:**

❑ ***Xét chu kỳ 1***:

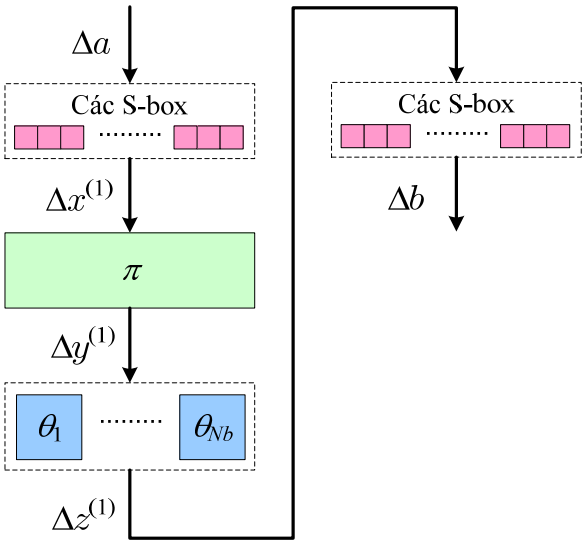
- $wt(\pi(\Delta a)) = wt\left(\chi_{\pi(\Delta a)}\right) = k$
- Sau khi qua các S-box : $\chi_{\Delta x_i^{(1)}} = \chi_{\Delta a_i}$, suy ra $wt\left(\pi\left(\Delta x^{(1)}\right)\right) = wt(\pi(\Delta a)) = k$

- Sau khi qua biến đổi $\pi : \Delta y^{(1)} = \pi\left(\Delta x^{(1)}\right)$, suy ra $wt\left(\Delta y^{(1)}\right) == wt\left(\pi\left(\Delta x^{(1)}\right)\right) = k$
- Các biến đổi θ_i không làm thay đổi số lượng và vị trí các cột khác 0 trong $\Delta y^{(1)}$ nên $wt\left(\Delta z^{(1)}\right) = wt\left(\Delta y^{(1)}\right) = k$.

❑ ***Xét chu kỳ 2***:

- Sau khi qua các S-box : $\chi_{\Delta x_i^{(2)}} = \chi_{\Delta z_i^{(1)}}$, suy ra $wt\left(\Delta x^{(2)}\right) = wt\left(\Delta z^{(1)}\right) = k$, tức là $\Delta x^{(2)}$ có đúng k cột khác 0.

$$wt\left(\chi_{\pi(\Delta a)}\right) = wt(\pi(\Delta a)) = wt\left(\pi\left(\Delta x^{(1)}\right)\right) = wt\left(\Delta y^{(1)}\right) = wt\left(\Delta z^{(1)}\right) = wt\left(\Delta x^{(2)}\right) = k$$



Hình 4.7. Minh họa Bổ đề 4.2

Như vậy, sau khi qua biến đổi π ở chu kỳ 1, $\Delta y^{(1)}, \Delta z^{(1)}$ và $\Delta x^{(2)}$ đều có đúng k cột khác 0 và có vị trí các cột này giống nhau. Đặt $\upsilon = \{v_1, v_2, ...v_k\} \subset \{1,2,...,Nb\}$ là tập chỉ số các cột khác 0 trong $\Delta x^{(2)}$:

$$\Delta x_{v_1}^{(2)} \neq 0, ..., \Delta x_{v_k}^{(2)} \neq 0$$

Ta có:

$$\sum_{\Delta x^{(2)}} DP_2\left(\Delta a, \Delta x^{(2)}\right)$$

$$= \sum_{\Delta x^{(2)}}\left(\prod_{i=1}^{k} DP_2^{\theta_{v_i}}\left(\Delta a_{v_i}^{*}, \Delta x_{v_i}^{(2)}\right)\right)$$

$$= \sum_{\Delta x^{(2)}}\left(DP_2^{\theta_{v_1}}\left(\Delta a_{v_1}^{*}, \Delta x_{v_1}^{(2)}\right)\prod_{i=2}^{k} DP_2^{\theta_{v_i}}\left(\Delta a_{v_i}^{*}, \Delta x_{v_i}^{(2)}\right)\right)$$

Theo **Bổ đề 4.1**, $\forall v_i \in \upsilon, DP_2^{\theta_{v_i}}\left(\Delta a_{v_i}^{*}, \Delta x_{v_i}^{(2)}\right) \le \Omega_d$. Vì vậy, ta có :

$$\sum_{\Delta x^{(2)}} DP_2\left(\Delta a, \Delta x^{(2)}\right) \le \left(\Omega_d\right)^{k-1} \sum_{\Delta x_{v_1}^{(2)}} DP_2^{\theta_{v_1}}\left(\Delta a_{v_1}^{*}, \Delta x_{v_1}^{(2)}\right)$$

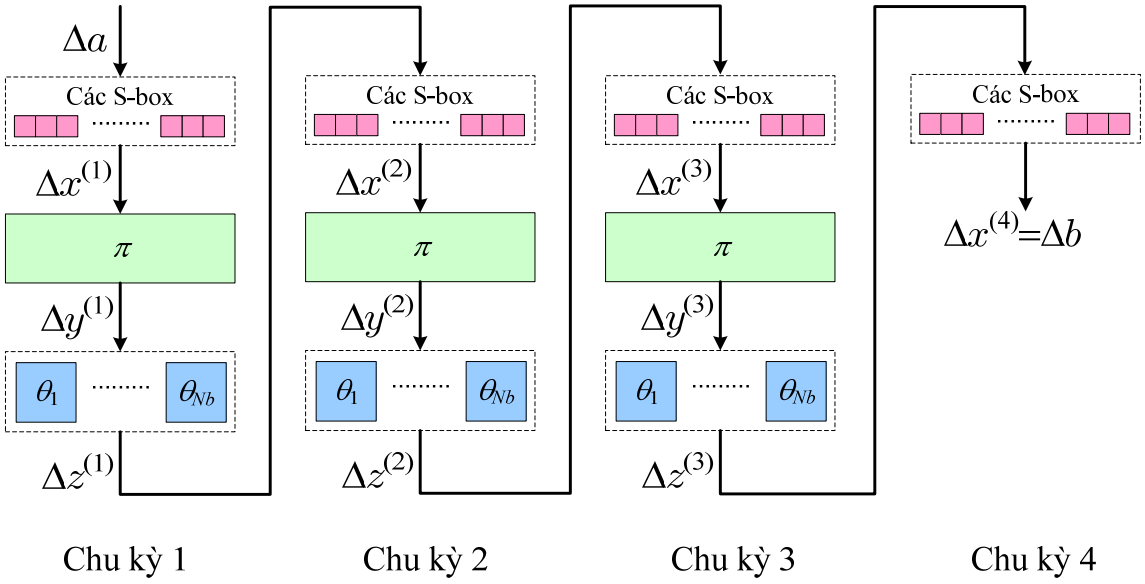
Do $\sum_{\Delta x_{v_1}} DP_2^{\theta_{v_1}}\left(\Delta a_{v_1}^{*}, \Delta x_{v_1}^{(2)}\right) = 1$ nên suy ra:

$$\sum_{\Delta x^{(2)}} DP_2\left(\Delta a, \Delta x^{(2)}\right) \le \left(\Omega_d\right)^{k-1}$$ ❑

4.3.3 Xác suất sai phân của tập vết sai phân lan truyền qua $r \ge 4$ chu kỳ của XAES

Hình 4.8 minh họa sự lan truyền sai phân qua 4 chu kỳ trong XAES. Trong Hình 4.8:

- $\Delta x^{(i)} = \left(\Delta x_1^{(i)}, \ldots, \Delta x_{Nb}^{(i)}\right)$: vector sai phân đầu vào của biến đổi π trong chu kỳ i, trong đó $\Delta x_j^{(i)} = \left(\Delta x_{j1}^{(i)}, \Delta x_{j2}^{(i)}, \ldots, \Delta x_{jNw}^{(i)}\right)$ với $1 \le j \le Nb$.
- $\Delta y^{(i)} = \left(\Delta y_1^{(i)}, \ldots, \Delta y_{Nb}^{(i)}\right)$: vector sai phân đầu ra của biến đổi π trong chu kỳ i và cũng chính là vector sai phân đầu vào của biến đổi θ trong chu kỳ i, trong đó $\Delta y_j^{(i)} = \left(\Delta y_{j1}^{(i)}, \Delta y_{j2}^{(i)}, \ldots, \Delta y_{jNw}^{(i)}\right)$ với $1 \le j \le Nb$.
- $\Delta z^{(i)} = \left(\Delta z_1^{(i)}, \ldots, \Delta z_{Nb}^{(i)}\right)$: vector sai phân đầu ra của biến đổi θ trong chu kỳ i, trong đó $\Delta z_j^{(i)} = \left(\Delta z_{j1}^{(i)}, \Delta z_{j2}^{(i)}, \ldots, \Delta z_{jNw}^{(i)}\right)$ với $1 \le j \le Nb$.



Hình 4.8. Khảo sát sự lan truyền sai phân qua 4 chu kỳ trong XAES

Định lý 4.3. *Xác suất sai phân của tập vết sai phân lan truyền qua* 4 *chu kỳ của XAES được chặn trên bởi* $(\Omega_d)^{\beta_d-1}$.

✍***Chứng minh*:**

Nếu $wt(\chi_{\pi(\Delta a)})+wt(\Delta b)<\beta_d$ thì $DP_4(\Delta a,\Delta b)=0$. Vậy, chỉ xét giá trị chặn trên của $DP_4(\Delta a,\Delta b)$ khi $wt(\chi_{\pi(\Delta a)})+wt(\Delta b)\geq\beta_d$.

- ***Trường hợp 1***: $wt(\chi_{\pi(\Delta a)})\geq\beta_d$.

$$DP_4(\Delta a,\Delta b) = \sum_{\Delta x^{(2)}} DP_2(\Delta a,\Delta x^{(2)})DP_2(\Delta z^{(2)},\Delta b)$$

$$\leq \sum_{\Delta x^{(2)}} DP_2(\Delta a,\Delta x^{(2)}) \leq \max_{\Delta x^{(2)}} DP_2(\Delta a,\Delta x^{(2)})$$

Theo **Định lý 4.2**, ta có: $\max_{x^{(2)}} DP_2(a,x^{(2)}) \leq (\Omega_d)^{wt(\pi(a))}$

Do $\Omega_d\leq 1$ và $wt(\pi(\Delta a))=wt(\chi_{\pi(\Delta a)})\geq\beta_d$ nên $\max_{\Delta x^{(2)}} DP_2(\Delta a,\Delta x^{(2)})\leq(\Omega_d)^{\beta_d}$.

Suy ra $DP_4(\Delta a,\Delta b)\leq(\Omega_d)^{\beta_d}$

- ***Trường hợp 2***: $wt(\Delta b) \geq \beta_d$

$$DP_4(\Delta a, \Delta b) = \sum_{\Delta x^{(2)}} DP_2\left(\Delta a, \Delta x^{(2)}\right) DP_2\left(\Delta z^{(2)}, \Delta b\right)$$

$$\leq \sum_{\Delta z^{(2)}} DP_2\left(\Delta z^{(2)}, \Delta b\right) \leq \max_{\Delta z^{(2)}} DP_2\left(\Delta z^{(2)}, \Delta b\right)$$

Theo **Bổ đề 4.2**, ta có: $\max_{\Delta z^{(2)}} DP_2\left(\Delta z^{(2)}, \Delta b\right) \leq (\Omega_d)^{wt(\Delta b)}$

Do $\Omega_d \leq 1$ và $wt(\Delta b) \geq \beta_d$ nên $\max_{\Delta z^{(2)}} DP_2\left(\Delta z^{(2)}, \Delta b\right) \leq (\Omega_d)^{\beta_d}$

Suy ra $DP_4(\Delta a, \Delta b) \leq (\Omega_d)^{\beta_d}$

- ***Trường hợp 3***: $1 \leq wt\left(\chi_{\pi(\Delta a)}\right) < \beta_d$ và $1 \leq wt(\Delta b) < \beta_d$.

Đặt $k = wt\left(\chi_{\pi(\Delta a)}\right)$ và $l = wt(\Delta b)$. Ta có $1 \leq k, l < \beta_d$ và $k + l \geq \beta_d$.

Ta có :

$$DP_4(\Delta a, \Delta b) = \sum_{\Delta x^{(2)}} DP_2\left(\Delta a, \Delta x^{(2)}\right) DP_2\left(\Delta z^{(2)}, \Delta b\right) \quad (4.17)$$

Áp dụng **Bổ đề 4.2** cho chu kỳ 3 và 4, ta có:

$$DP_2\left(\Delta z^{(2)}, \Delta b\right) \leq (\Omega_d)^{wt(\Delta b)} = (\Omega_d)^l \quad (4.18)$$

Từ (5.17) và (5.18), suy ra :

$$DP_4(\Delta a, \Delta b) \leq (\Omega_d)^l \sum_{\Delta x^{(2)}} DP_2\left(\Delta a, \Delta x^{(2)}\right) \quad (4.19)$$

Áp dụng **Bổ đề 4.3** cho chu kỳ 1 và 2, ta có :

$$\sum_{\Delta x^{(2)}} DP_2\left(\Delta a, \Delta x^{(2)}\right) \leq (\Omega_d)^{wt\left(\gamma_{\pi(\Delta a)}\right)-1} = (\Omega_d)^{k-1} \quad (4.20)$$

Từ (5.19) và (5.20), suy ra :

$$DP_4(\Delta a, \Delta b) \leq (\Omega_d)^{l+k-1}$$

Do $\Omega_d \leq 1$ và $l + k \geq \beta_d$ nên ta có thể kết luận:

$$DP_4(\Delta a, \Delta b) \leq (\Omega_d)^{l+k-1} \leq (\Omega_d)^{\beta_{d-1}}$$

❑

Định lý 4.4. *Đặt* $k = wt(\pi(\Delta a)) \geq 1$.

Xác suất sai phân của tập vết sai phân qua $r \geq 4$ *chu kỳ của XAES được chặn trên bởi* $(\Omega_d)^{\Phi_d(r)}$ *với* $\Phi_d(r) = \beta_d - 1 + \left\lfloor \frac{r-4}{2} \right\rfloor (k-1)$.

✍***Chứng minh*:**

Định lý 4.4 được chứng minh theo phương pháp quy nạp.

Đặt $k = wt(\pi(\Delta a)) \geq 2$.

Với $r = 4,\ \Phi_d(4) = \beta_d - 1$. Theo **Định lý 4.3** :

$$DP_4(\Delta a, \Delta b) \leq (\Omega_d)^{\beta_d - 1} = (\Omega_d)^{\Phi_d(4)}$$

Với $r = 5,\ \Phi_d(5) = \beta_d$. Ta có :

$$DP_5(\Delta a, \Delta b) = \sum_{\Delta x^{(4)}} DP_4\left(\Delta a, \Delta x^{(4)}\right) DP_1\left(\Delta z^{(4)}, \Delta b\right)$$

$$\leq \sum_{\Delta x^{(4)}} DP_4\left(\Delta a, \Delta x^{(4)}\right)$$

Theo **Định lý 4.3**, $DP_4\left(\Delta a, \Delta x^{(4)}\right) \leq (\Omega_d)^{\beta_d - 1}$. Suy ra:

$$DP_5(\Delta a, \Delta b) \leq (\Omega_d)^{\beta_d - 1} = (\Omega_d)^{\Phi_d(5)}$$

Giả sử $\forall r \in [4, \ldots, r_0], DP_r(\Delta a, \Delta b) \leq (\Omega_d)^{\Phi_d(r)}$. Ta sẽ chứng minh :

$$DP_{r+2}(\Delta a, \Delta b) \leq \Omega^{\Phi_d(r+2)}$$

Ta có:

$$DP_{r+2}(\Delta a, \Delta b) = \sum_{\Delta x^{(2)}} DP_2\left(\Delta a, \Delta x^{(2)}\right) DP_r\left(\Delta z^{(2)}, \Delta b\right)$$

Theo giả thiết quy nạp, ta có :

$$DP_r\left(\Delta z^{(2)}, \Delta b\right) \leq (\Omega_d)^{\Phi_d(r)}$$

Suy ra:

$$DP_{r+2}(\Delta a, \Delta b) \leq (\Omega_d)^{\Phi_d(r)} \sum_{\Delta x^{(2)}} DP_2\left(\Delta a, \Delta x^{(2)}\right) \tag{4.21}$$

Theo **Bổ đề 4.3**, áp dụng vào chu kỳ 1 và 2, ta có:

$$\sum_{\Delta x^{(2)}} DP_2\left(\Delta a, \Delta x^{(2)}\right) \leq (\Omega_d)^{k-1} \tag{4.22}$$

Từ (5.21) và (5.22), suy ra:

$$DP_{r+2}(\Delta a, \Delta b) \le (\Omega_d)^{\Phi_d(r)+k-1} = (\Omega_d)^{\Phi_d(r+2)}$$

Kết luận:

$$\forall r \ge 4, DP_r(\Delta a, \Delta b) \le (\Omega_d)^{\Phi_d(r)} \text{ với } \Phi_d(r) = \beta_d - 1 + \left\lfloor \frac{r-4}{2} \right\rfloor (k-1). \qquad \square$$

4.3.4 Áp dụng với một số thể hiện cụ thể của XAES

Nội dung phần này minh họa việc áp dụng các kết quả tổng quát đã chứng minh trong phần 4.3 vào một số thể hiện cụ thể của XAES.

Trong các kết quả đã được trình bày trong phần 4.3, tham số chính Ω_d phụ thuộc vào S-box S_φ và branch number (sai phân) β_d của tầng khuếch tán θ.

$$\Omega_d = \max\left\{ \max_{1\le i\le n} \max_{1\le u\le 2^m-1} \sum_{j=1}^{2^m-1} \left\{DP^{S_\varphi}(u,j)\right\}^{\beta_d}, \max_{1\le i\le n} \max_{1\le u\le 2^m-1} \sum_{j=1}^{2^m-1} \left\{DP^{S_\varphi}(j,u)\right\}^{\beta_d} \right\} \quad (4.23)$$

Trong XAES, S-box được xây dựng với ánh xạ nghịch đảo trên GF(2^m), phụ thuộc vào tham số cấu trúc m. Tầng khuếch tán θ sử dụng Nb ánh xạ tuyến tính θ_i trên GF(2^m)^{Nw} có branch number (sai phân) nhỏ nhất là $\beta_d = Nw$ hay $Nw + 1$. Vì vậy, nội dung của phần này trình bày việc áp dụng các kết quả về chặn trên của xác suất sai phân của tập vết sai phân đối với một số thể hiện cụ thể của XAES với các giá trị khác nhau của tham số cấu trúc (m và Nw).

Xét trường hợp $m = 8$. Chọn $x_0 = 1 \in \mathrm{GF}(2^8)$ và tính bảng phân bố của xác suất sai phân $DP^{S_\varphi}(x_0, y)$ (xem Bảng 4.1). Điều đáng lưu ý là nếu chọn giá trị x_0 khác, chúng ta cũng nhận được bảng phân bố xác suất sai phân tương tự. Theo Bảng 4.1, có ϕ_i lần xác suất sai phân $DP^{S_\varphi}(x, y)$ nhận giá p_i (với i = 1, 2, 3). Từ đó suy ra:

$$\Omega_d = \sum_{y=0}^{2^8-1} \left(DP^{S_\varphi}(x_0, y)\right)^{\beta_d} = \sum_{i=1}^{3} \phi_i p_i^{\beta_d} \quad (4.24)$$

Áp dụng **Định lý 4.3**, ta có chặn trên của xác suất sai phân của tập vết sai phân qua 4 chu kỳ của XAES như sau:

$$DP_4(x,y) \le (\Omega_d)^{\beta_d - 1} = \left(\sum_{i=1}^{3} \phi_i p_i^{\beta_d} \right)^{\beta_d - 1} \tag{4.25}$$

i	1	2	3
p_i	2^{-6}	2^{-7}	0
ϕ_i	1	126	129

Bảng 4.1. Phân bố xác suất sai phân qua S-box trong XAES với m = 8

Với mỗi giá trị của tham số cấu trúc Nw, branch number (sai phân) β_d của tầng khuếch tán θ có thể nhận giá trị là Nw hay $Nw + 1$. Bảng 4.2 thể hiện giá trị chặn trên của xác suất sai phân của tập vết sai phân qua 4 chu kỳ của XAES ($(\Omega_d)^{\beta_d - 1}$) theo giá trị branch number (sai phân) β_d. Trong trường hợp Rijndael (tương ứng với m = 8 và β_d = 5), xác suất sai phân của tập vết sai phân qua 4 chu kỳ mã hóa có chặn trên là 1.16080×2^{-111}. Kết quả này phù hợp với kết quả do S. Park khảo sát riêng cho trường hợp Rijndael được công bố trong [72].

β_d	Ω_d	$(\Omega_d)^{\beta_d - 1}$
1	1	1
2	1.01563×2^{-7}	1.01563×2^{-7}
3	1.04688×2^{-14}	1.09595×2^{-28}
4	1.10938×2^{-21}	1.36532×2^{-63}
5	1.23438×2^{-28}	1.16080×2^{-111}
6	1.48438×2^{-35}	1.80160×2^{-173}
7	1.98438×2^{-42}	1.90806×2^{-247}
8	1.49219×2^{-48}	1.02954×2^{-332}
9	1.24609×2^{-54}	1.45327×2^{-430}
10	1.12305×2^{-60}	1.42089×2^{-539}
11	1.06152×2^{-66}	1.81669×2^{-660}
12	1.03076×2^{-72}	1.39551×2^{-792}
13	1.01538×2^{-78}	1.20100×2^{-936}
14	1.00769×2^{-84}	1.10472×2^{-1092}
15	1.00385×2^{-90}	1.05527×2^{-1260}
16	1.00192×2^{-96}	1.02919×2^{-1440}
17	1.00096×2^{-102}	1.01547×2^{-1632}
18	1.00048×2^{-108}	1.00819×2^{-1836}
19	1.00024×2^{-114}	1.00433×2^{-2052}
20	1.00012×2^{-120}	1.00228×2^{-2280}

Bảng 4.2. Chặn trên của xác suất sai phân của tập vết sai phân qua 2 chu kỳ của hàm SDS và qua 4 chu kỳ của XAES với m = 8

4.4 Giá trị chặn trên của xác suất tuyến tính của bao tuyến tính

4.4.1 Các kết quả chính

Đặt β_l là giá trị nhỏ nhất của branch number (tuyến tính) của các biến đổi tuyến tính được sử dụng trong tầng khuếch tán θ của XAES.

$$\beta_l = \min_{i=0,1,\ldots,Nb-1} \{ \mathcal{B}_l(\theta_i) \} \tag{4.26}$$

Đặt $\Omega_l = \max\left\{ \max_{1\le i\le n} \max_{1\le u\le 2^m-1} \sum_{j=1}^{2^m-1} \left\{LP^{S_\varphi}(u,j)\right\}^{\beta_l}, \max_{1\le i\le n} \max_{1\le u\le 2^m-1} \sum_{j=1}^{2^m-1} \left\{LP^{S_\varphi}(j,u)\right\}^{\beta_l} \right\}$

Việc khảo sát xác suất tuyến tính của bao tuyến tính tương tự như quá trình khảo sát xác suất sai phân của tập vết sai phân, ngoại trừ một số thay đổi sau trong các kết quả và chứng minh:

- thay thế vai trò của giá trị sai phân Δx bằng mặt nạ Γx tương ứng,
- thay thế xác suất sai phân (DP) bằng xác suất tuyến tính (LP) tương ứng,
- thay thế branch number (sai phân) β_d bằng branch number (tuyến tính) β_l
- thay thế Ω_d bằng Ω_l

Tương tự với các kết quả đối với xác suất sai phân của tập vết sai phân, chúng tôi cũng chứng minh được các kết quả tương tự cho xác suất tuyến tính của bao tuyến tính. Dưới đây là các kết quả chính:

Bổ đề 4.4. *Xác suất tuyến tính tối đa của bao tuyến tính qua hàm SDS (gồm 2 chu kỳ) với tầng thay thế sử dụng các S-box giống nhau (S_φ) và tầng khuếch tán là ánh xạ tuyến tính θ_i được chặn trên bởi:*

$$LP_2^{\theta_i}(a,b) \le \Omega_l \tag{4.27}$$

Định lý 4.5.

Nếu $\chi_{\pi(\Gamma a)} = \chi_{\Gamma b}$ thì xác suất tuyến tính tối đa của bao tuyến tính qua 2 chu kỳ của XAES được chặn trên bởi: $LP_2(\Gamma a, \Gamma b) \le (\Omega_l)^{wt(\pi(\Gamma a))}$.

Nếu $\chi_{\pi(\Gamma a)} \ne \chi_{\Gamma b}$ thì $LP_2(\Gamma a, \Gamma b) = 0$.

Bổ đề 4.5. *Xác suất tuyến tính của bao tuyến tính qua* 2 *chu kỳ của XAES được chặn trên bởi* $(\Omega_l)^{wt(\Gamma b)}$ *với* Γb *là vector mặt nạ ở đầu ra.*

$$LP_2(\Gamma a, \Gamma b) \le (\Omega_l)^{wt(\Gamma b)} \tag{4.28}$$

Bổ đề 4.6. *Đặt* $k = wt(\chi_{\pi(\Gamma a)})$. *Khi đó:*

$$\sum_{\Delta x^{(2)}} LP_2(\Gamma a, \Gamma x^{(2)}) \le (\Omega_l)^{k-1} \tag{4.29}$$

Định lý 4.6. *Xác suất tuyến tính của bao tuyến tính qua* 4 *chu kỳ của XAES được chặn trên bởi* $(\Omega_l)^{\beta_l - 1}$.

Định lý 4.7. *Đặt* $k = wt(\pi(\Delta a)) \ge 1$. *Xác suất tuyến tính của bao tuyến tính qua* $r \ge 4$ *chu kỳ của XAES được chặn trên bởi* $(\Omega_l)^{\Phi_l(r)}$ *với* $\Phi_l(r) = \beta_l - 1 + \left\lfloor \frac{r-4}{2} \right\rfloor (k-1)$.

4.4.2 Áp dụng với một số thể hiện cụ thể của XAES

Nội dung phần này minh họa việc áp dụng các kết quả tổng quát đã chứng minh trong phần 4.3 vào một số thể hiện cụ thể của XAES.

Xét trường hợp m = 8. Chọn $y_0 = 1 \in \mathrm{GF}(2^8)$ và tính bảng phân bố của xác suất tuyến tính $LP^{S_\varphi}(x, y_0)$ (xem Bảng 4.3). Khi chọn giá trị y_0 khác, chúng ta cũng nhận được bảng phân bố xác suất tuyến tính tương tự. Theo Bảng 4.3, có ϕ_i lần xác suất tuyến tính $LP^{S_\varphi}(x, y)$ nhận giá p_i (với i = 1, 2,…, 8). Từ đó suy ra:

$$\Omega_l = \sum_{y=0}^{2^8-1} \left(LP^{S_\varphi}(x_0, y)\right)^{\beta_l} = \sum_{i=1}^{9} \phi_i p_i^{\beta_l} \tag{4.30}$$

Áp dụng **Định lý 4.6**, ta có chặn trên của xác suất tuyến tính của bao tuyến tính qua 4 chu kỳ của XAES như sau:

$$LP_4(x, y) \le (\Omega_l)^{\beta_l - 1} = \left(\sum_{i=1}^{9} \phi_i p_i^{\beta_l} \right)^{\beta_l - 1} \tag{4.31}$$

i	1	2	3	4	5	6	7	8	9
p_i	$\left(\frac{8}{64}\right)^2$	$\left(\frac{7}{64}\right)^2$	$\left(\frac{6}{64}\right)^2$	$\left(\frac{5}{64}\right)^2$	$\left(\frac{4}{64}\right)^2$	$\left(\frac{3}{64}\right)^2$	$\left(\frac{2}{64}\right)^2$	$\left(\frac{1}{64}\right)^2$	0
ϕ_i	5	16	36	24	34	40	36	48	17

Bảng 4.3. Phân bố xác suất tuyến tính qua S-box trong XAES với $m = 8$

Với mỗi giá trị của tham số cấu trúc Nw, branch number (tuyến tính) β_l của tầng khuếch tán θ có thể nhận giá trị là Nw hay $Nw + 1$. Bảng 4.4 thể hiện giá trị chặn trên của xác suất tuyến tính của bao tuyến tính qua 4 chu kỳ của XAES ($(\Omega_l)^{\beta_l - 1}$) theo giá trị branch number (tuyến tính) β_l. Trong trường hợp Rijndael (tương ứng với $m = 8$ và $\beta_l = 5$), xác suất tuyến tính của bao tuyến tính qua 4 chu kỳ mã hóa có chặn trên là 1.06388×2^{-106}. Kết quả này phù hợp với kết quả do S. Park khảo sát riêng cho trường hợp Rijndael và được công bố trong [72].

β_l	Ω_l	$(\Omega_l)^{\beta_l - 1}$	β_l	Ω_l	$(\Omega_l)^{\beta_l - 1}$
1	1	1	11	1.47820×2^{-64}	1.55661×2^{-635}
2	1.01563×2^{-7}	1.01563×2^{-7}	12	1.42138×2^{-70}	1.49507×2^{-765}
3	1.29187×2^{-14}	1.66893×2^{-28}	13	1.37935×2^{-76}	1.48232×2^{-907}
4	1.85120×2^{-21}	1.58599×2^{-61}	14	1.34799×2^{-82}	1.51627×2^{-1061}
5	1.43628×2^{-27}	1.06388×2^{-106}	15	1.32444×2^{-88}	1.59698×2^{-1227}
6	1.18211×2^{-33}	1.15416×2^{-164}	16	1.30667×2^{-94}	1.72718×2^{-1405}
7	1.01803×2^{-39}	1.11317×2^{-234}	17	1.2932×2^{-100}	1.91205×2^{-1595}
8	1.81586×2^{-46}	1.01716×2^{-316}	18	1.28297×2^{-106}	1.08019×2^{-1796}
9	1.66362×2^{-52}	1.83354×2^{-411}	19	1.27519×2^{-112}	1.24213×2^{-2010}
10	1.55588×2^{-58}	1.66968×2^{-517}	20	1.26925×2^{-118}	1.44949×2^{-2236}

Bảng 4.4. Chặn trên của xác suất tuyến tính của bao tuyến tính qua 2 chu kỳ của hàm SDS và qua 4 chu kỳ của XAES với $m = 8$

4.5 Kết luận

Chúng tôi đã xác định các công thức tổng quát của giá trị chặn trên của xác suất sai phân của tập vết sai phân và giá trị chặn trên của xác suất tuyến tính của bao tuyến tính lan truyền qua $r \geq 4$ chu kỳ của XAES. Các công thức được chúng tôi đề xuất và chứng minh có tính tổng quan hơn các kết quả đã được công bố [38][44][71][72].

Dựa trên các công thức trong **Định lý 4.3** và **Định lý 4.6** , chúng tôi đã áp dụng để khảo sát cụ thể với các thể hiện của XAES trong trường hợp $m = 8$ (trên trường

Galois của Rijndael), giá trị Nw biến thiên từ 1 đến 19 và giá trị branch number (β_d cũng như β_l) là Nw hay $Nw + 1$. Theo [44], chi phí để có thể áp dụng phương pháp sai phân hay phương pháp tuyến tính được xem là tỷ lệ với nghịch đảo giá trị chặn trên của xác suất sai phân hay xác suất tuyến tính.

Một số vấn đề mở:

- Trong công thức xác định chặn trên của xác suất sai phân và xác suất tuyến tính mà chúng tôi xây dựng sử dụng giá trị và phân bố của xác suất sai phân/tuyến tính của S-box. Việc chọn lựa một giá trị đặc trưng khác của S-box có thể giúp các giá trị chặn trên được xác định chặt hơn.
- Có thể đề xuất cách làm trội khác trong quá trình xây dựng công thức chặn trên của xác suất sai phân và xác suất tuyến tính qua 4 chu kỳ mã hóa để kết quả đạt được chặt hơn.

Chương 5

Phát sinh bộ hệ số cho ánh xạ tuyến tính trong MixColumns

Tóm tắt chương:

✍ *Nội dung của Chương 5 trình bày các tính chất của bộ hệ số được dùng trong ánh xạ tuyến tính của biến đổi MixColumns của thuật toán XAES. Dựa trên các tính chất này, chúng tôi đề xuất các giải thuật nhằm kiểm tra nhanh và hiệu quả một bộ hệ số bất kỳ có đạt được yêu cầu về mặt an toàn khi sử dụng trong giải thuật XAES.*

Các giải thuật được thử nghiệm thực tế với trường hợp $m = 8$ (trên trường Galois của Rijndael) và giá trị Nw từ 4 đến 8. Kết quả thực nghiệm cho thấy:

- trên 95% các bộ hệ số phát sinh ngẫu nhiên là ứng cử viên cho bộ hệ số mạnh,

- trên 99% các ứng cử viên thật sự là bộ hệ số mạnh.

Ngoài ra, chúng tôi còn xác định tất cả các bộ hệ số tối ưu (bộ hệ số mạnh với giá trị hệ số lớn nhất là nhỏ nhất) cho trường hợp m=8, Nw = 2,5,..., .8) trên trường Galois của Rijndael.

5.1 Mở đầu

Trong Chương 2, chúng tôi đã trình bày giải thuật mã hóa được tham số hóa XAES. Với mỗi biến đổi trong XAES, chúng tôi đề nghị tham số hóa các hằng số được sử dụng trong biến đổi thành tham số xử lý, nhờ đó cho phép tạo ra các biến thể khác nhau của XAES với các bộ giá trị tham số xử lý cụ thể do người dùng chọn.

Trong Chương 3 và Chương 4, chúng tôi đã ***khảo sát về mặt lý thuyết*** tính an toàn tổng quát của XAES đối với phương pháp sai phân và phương pháp tuyến tính trong việc phân tích mã. Thông qua các kết quả đã trình bày trong 2 chương này, vai trò trọng yếu của biến đổi MixColumns và SubBytes đã được phân tích rõ: đây chính là tầng khuếch tán và tầng thay thế trong kiến trúc SPN.

Trong Chương 5 và Chương 6, chúng tôi khảo sát XAES từ ***góc độ áp dụng vào thực tế***: tạo ra các bộ giá trị cho tham số xử lý mà trọng tâm là đối với biến đổi MixColumns và SubBytes. Các đề xuất, khảo sát và thực nghiệm được chúng tôi tiến hành trên các thể hiện cụ thể của XAES nhằm minh họa các kỹ thuật để tạo ra các bộ giá trị cho tham số xử lý trong biến đổi MixColumns và SubBytes trong XAES.

5.2 Bộ hệ số cho ánh xạ tuyến tính trong MixColumns

5.2.1 Bộ hệ số mạnh và bộ hệ số mạnh ngưỡng T

Trong biến đổi MixColumns của XAES sử dụng các ánh xạ tuyến tính trên $GF(2^m)^{Nw}$ có biểu diễn bằng ma trận luân hoàn. Tham số xử lý Θ_θ của biến đổi MixColumns là danh sách các ánh xạ tuyến tính $\theta_j \in \Phi_{m,Nw}$ được sử dụng để biến đổi từng cột j của trạng thái:

$$\Theta_\theta = (\theta_0, \theta_1, ..., \theta_{Nb-1}) \text{ với } \theta_j \in \Phi_{m,Nw} \text{ thỏa } \mathcal{B}(\theta_j) \geq Nw,\ 0 \leq j < Nb \qquad (5.1)$$

với $\Phi_{m,Nw}$ là tập các ánh xạ tuyến tính $\mathcal{F} : (GF(2^m))^{Nw} \to (GF(2^m))^{Nw}$ có biểu diễn bằng ma trận luân hoàn.

Cho $c = (c_0, c_1, ..., c_{Nw-1}) \in GF(2^m)^{Nw}$. Đặt $\theta[c] \in \Phi_{m,Nw}$ là ánh xạ tuyến tính được sử dụng tương ứng với đa thức $c(x) = c_0 + c_1 x + ... + c_{Nw-1} x^{Nw-1}$.

Xuất phát từ khái niệm ***tầng khuếch tán tối đa*** của V. Rijmen [18], chúng tôi đề nghị các định nghĩa sau:

Định nghĩa 5.1: *Bộ hệ số* $c = (c_0, c_1, ..., c_{Nw-1}) \in GF(2^m)^{Nw}$ *được gọi là* ***bộ hệ số khuếch tán tối đa*** *nếu ánh xạ tuyến tính* $\theta[c] \in \Phi_{m,Nw}$ *có branch number đạt giá trị* $Nw+1$.

Định nghĩa 5.2: *Bộ hệ số* $c = (c_0, c_1, ..., c_{Nw-1}) \in GF(2^m)^{Nw}$ *được gọi là* ***bộ hệ số khuếch tán gần tối đa*** *nếu ánh xạ tuyến tính* $\theta[c] \in \Phi_{m,Nw}$ *có branch number đạt giá trị* Nw.

Ràng buộc được chúng tôi đề nghị đối với tham số xử lý trong MixColumns của XAES là mỗi biến đổi θ_j có branch number đạt giá trị Nw hay $Nw + 1$. Đây là cơ sở

được dùng cho việc chứng minh tổng quát tính an toàn của XAES đối với phương pháp sai phân và phương pháp tuyến tính trong phân tích mã (xem Chương 3 và Chương 4).

Định nghĩa 5.3: *Bộ hệ số khuếch tán tối đa hay khuếch tán gần tối đa được gọi là* ***bộ hệ số mạnh***.

Vậy, đối với bộ hệ số mạnh c, với mọi vector dữ liệu $t \in \mathrm{GF}(2^m)^{Nw} \setminus \{\vec{0}\}$, ta luôn có

$$wt(t) + wt(\theta[c](t)) \geq \beta \tag{5.2}$$

với $\beta = Nw$ hay $Nw + 1$. Điều này giúp đảm bảo tính khuếch tán thông tin cho ánh xạ $\theta[c]$: ngay cả trong trường hợp yếu nhất cũng có β phần tử khác 0 trong vector dữ liệu đầu vào và vector dữ liệu đầu ra (với vector đầu vào khác $\vec{0}$). Chính vì vậy, các bộ hệ số mạnh này thích hợp trong các ứng dụng mang tính dài hạn, ví dụ như trong mã hóa cơ sở dữ liệu, mã hóa tài liệu điện tử trong lưu trữ, mã hóa thông tin khóa riêng (private key) hay các thông tin credential lưu trữ trên thẻ thông minh.

Định nghĩa 5.4: *Bộ hệ số c được gọi là bộ hệ số khuếch tán tối đa mạnh ngưỡng T nếu*

$$\mathrm{Prob}(wt(t) + wt(\theta[c](t)) \geq \beta) \geq T \tag{5.3}$$

với $t \in \mathrm{GF}(2^m)^{Nw} \setminus \{\vec{0}\}$ *và* $\beta = Nw + 1$.

Định nghĩa 5.5: *Bộ hệ số c được gọi là bộ hệ số mạnh ngưỡng T nếu*

$$\mathrm{Prob}(wt(t) + wt(\theta[c](t)) \geq \beta) \geq T \tag{5.4}$$

với $t \in \mathrm{GF}(2^m)^{Nw} \setminus \{\vec{0}\}$ *và* $\beta = Nw$.

Với bộ hệ số mạnh ngưỡng T, xác suất vector dữ liệu đầu vào $t \neq \vec{0}$ làm cho tổng số phần tử khác 0 trong vector đầu vào và vector đầu ra dưới mức β chỉ là $1 - T$. Như vậy, bộ hệ số mạnh ngưỡng T có thể được dùng trong các ứng dụng có chu kỳ sống tương đối ngắn và có nhu cầu thay đổi theo thời gian, ví dụ như sử dụng để mã hóa thông tin trong 1 phiên làm việc. Trên thực tế, số lượng bộ hệ số mạnh ngưỡng T

nhiều hơn số lượng bộ hệ số mạnh thật sự, cho phép dễ dàng chọn lựa và thay đổi bộ hệ số khi sử dụng trên thực tế , tránh việc dùng lại bộ hệ số cũ.

Vấn đề đặt ra là làm thế nào để kiểm tra tương đối hiệu quả một bộ hệ số có phải là bộ hệ số mạnh hay bộ hệ số mạnh ngưỡng *T*. Trong phần 5.3 và 5.4, chúng tôi đề xuất một số kỹ thuật tương đối hiệu quả nhằm giải quyết vấn đề này.

5.2.2 Một số nhận xét về các bộ hệ số

Dưới đây là một số nhận xét phục vụ việc phát sinh và kiểm tra bộ hệ số mạnh được trình bày trong phần 5.3 và 5.4.

Mỗi phép biến đổi MixColumns tương ứng với một đa thức bậc Nw–1 trong GF(2^m). Ta có thể biểu diễn các hệ số của đa thức này dưới dạng một từ (gồm Nw phần tử m-bit), hoặc vector gồm Nw phần từ m-bit. Nói cách khác, ta có thể biểu diễn một từ (gồm Nw phần từ m-bit) dưới dạng một vector gồm Nw phần tử m-bit hoặc một đa thức có bậc nhỏ hơn Nw.

Nhận xét 5.1: *Phép nhân với x (hay các lũy thừa của x) sẽ tương ứng với phép dịch chuyển xoay vòng các byte thành phần trong một từ.*

✍***Chứng minh*:**

Ta có $x \otimes x^{Nw-1} = x^{Nw} \bmod (x^{Nw}+1) = 1$.

Xét đa thức $a(x) = a_{Nw-1}x^{Nw-1} + a_{Nw-2}x^{Nw-2} + \ldots + a_1x + a_0 = \sum_{i=0}^{Nw-1} a_i x^i$

Kết quả phép nhân $b(x) = a(x) \otimes x = a(x) \times x \bmod(x^{Nw}+1)$ được xác định bằng:

$$b(x) = b_{Nw-2}x^{Nw-1} + a_{Nw-3}x^{Nw-2} + \ldots + a_0x + a_{Nw-1} = \sum_{i=0}^{Nw-1} a_{(i-1) \bmod Nw} x^i$$

❑

Từ **Nhận xét 5.1**, suy ra :

Nhận xét 5.2: B*iến đổi RotWord thực hiện phép dịch chuyển xoay vòng vector $(a_0,\ldots,a_{Nw-1})$ thành vector $(a_1,\ldots,a_{Nw-1},a_0)$ tương ứng phép nhân với x^{Nw-1}. Biến đổi ngược (RotWord^{-1}) thực hiện phép dịch chuyển xoay vòng vector $(a_0,\ldots,a_{Nw-1})$ thành vector $(a_{Nw-1},a_0,\ldots,a_{Nw-2})$ tương ứng phép nhân với x.*

Nhận xét 5.3: *Nếu* $c = (c_0, c_1, ..., c_{Nw-1})$ *là bộ hệ số khuếch tán tối đa thì tất cả các hệ số* c_i *đều khác* 0 *(*$0 \le i < Nw$*).*

✍***Chứng minh*:**

Giả sử hệ số $c_i = 0$ ($0 \le i < Nw$). Đặt $\theta[c] \in \Phi_{m,Nw}$ là ánh xạ tuyến tính được sử dụng tương ứng với đa thức $c(x) = c_0 + c_1 x + ... + c_{Nw-1} x^{Nw-1}$.

Chọn vector thử nghiệm $t = (t_0, t_1, ..., t_{Nw-1})$ với $t_i = 1$ và $t_j = 0$, $\forall\ j \ne i$. Vậy, $wt(t) = 1$. Rõ ràng phần tử thứ i của vector kết quả sẽ bằng 0. Suy ra:

$$wt(\theta[c](t)) \le Nw - 1.$$

Vậy, $\mathcal{B}(\theta[c]) \le wt(t) + wt(\theta[c](t)) \le Nw < Nw + 1$, tức là c không phải bộ hệ số khuếch tán tối đa. ❑

Nhận xét 5.4: *Branch number của ánh xạ tuyến tính tương ứng với bộ hệ số* $(c_0, c_1, ..., c_{Nw-1})$ *bằng với branch number của ánh xạ tuyến tính tương ứng với bộ hệ số* $(c_1, ..., c_{Nw-1}, c_0)$.

$$\mathcal{B}(\theta[c_0, c_1, ..., c_{Nw-1}]) = \mathcal{B}(\theta[c_1, c_2, ..., c_{Nw-1}, c_0]) \qquad (5.5)$$

✍***Chứng minh*:**

Đặt $\theta_1 = \theta[c_0, c_1, ..., c_{Nw-1}]$, $\theta_2 = \theta[c_1, ..., c_{Nw-1}, c_0]$. Ta chứng minh $\mathcal{B}(\theta_1) = \mathcal{B}(\theta_2)$.

$\forall t = (t_0, t_1, ..., t_{Nw-1}) \in \mathrm{GF}(2^8)^{Nw} \setminus \{\vec{0}\}$. Đặt $t' = \mathrm{RotWord}^{-1}(t) = (t_{Nw-1}, t_0, ..., t_{Nw-2})$.

Do $a(x) \otimes t(x) = a(x) \times t(x) \mod x^{Nw}+1$

$= a(x) \times t(x) \times x^{Nw} \mod x^{Nw}+1$

$= (a(x) \times x^{Nw-1}) \times (t(x) \times x) \mod x^{Nw}+1$

$= (a(x) \otimes x) \otimes (t(x) \otimes x^{Nw-1})$

nên $\theta_1(t) = \theta_2(t')$.

Vậy, $\forall t \in \mathrm{GF}(2^8)^{Nw} \setminus \{\vec{0}\}$, $\exists t' = \mathrm{RotWord}^{-1}(t)$,

$wt(t) = wt(t')$ và $wt(\theta_1(t)) = wt(\theta_2(t'))$ (5.6)

Tương tự, ta cũng chứng minh được

$\forall t \in \mathrm{GF}(2^8)^{Nw} \setminus \{\vec{0}\}, \exists t' = \mathrm{RotWord}(t)$,

$$wt(t') = wt(t) \text{ và } wt(\theta_2(t)) = wt(\theta_1(t')) \tag{5.7}$$

Vậy kết luận: $\mathcal{B}(\theta_1) = \mathcal{B}(\theta_2)$ ❑

Ý nghĩa của Nhận xét 5.4: Nếu tìm được 1 bộ hệ số mạnh (có branch number đạt Nw hay Nw+1), ta có thể phát sinh ra thêm Nw đa thức bằng cách xoay vòng các hệ số. Ngược lại, nếu 1 đa thức có Branch Number nhỏ hơn Nw thì các đa thức có được bằng cách xoay vòng hệ số cũng có Branch Number nhỏ hơn Nw.

Nhận xét 5.5: Cho (b_0, b_1,.., $b_{Nw\text{-}1}$) là một hoán vị của (a_0, a_1,.., $a_{Nw\text{-}1}$). Không thể kết luận rằng $\mathcal{B}(\theta[b_0, b_1, .., b_{Nw\text{-}1}]) = \mathcal{B}(\theta[a_0, a_1, .., a_{Nw\text{-}1}])$

❑Ví dụ: Xét thuật toán Rijndael (Nw=4). Bộ hệ số ({01}, {01}, {02}, {03}) cho kết quả Branch Number là 5, trong khi bộ hệ số ({01}, {02}, {01}, {03}) cho kết quả Branch Number là 4.

Nhận xét 5.6: *Cho đa thức* $c(x) = \sum_{i=0}^{Nw-1} c_i x^i$ *khả nghịch.*

Đặt $d(x) = \sum_{i=0}^{Nw-1} d_i x^i$ *là đa thức nghịch đảo của* $c(x) \pmod{x^{Nw} + 1}$.

Đặt $u(x) = \sum_{i=0}^{Nw-1} a_{(i+1) \bmod Nw} x^i$ *và* $v(x) = \sum_{i=0}^{Nw-1} b_{(i-1) \bmod Nw} x^i$.

Khi đó : $u(x) \otimes v(x) = 1$, *tức là* $v(x) = u^{-1}(x) \pmod{x^{Nw} + 1}$.

✍***Chứng minh*:** ta có:

$$\begin{aligned} 1 &= c(x) \otimes d(x) \\ &= c(x) \times d(x) && \bmod x^{Nw} + 1 \\ &= c(x) \times d(x) \times x^{Nw} && \bmod x^{Nw} + 1 \\ &= (c(x) \times x) \times (d(x) \times x^{Nw-1}) && \bmod x^{Nw} + 1 \\ &= u(x) \otimes v(x) \end{aligned}$$

❑

Ý nghĩa của Nhận xét 5.6: Khi xoay vòng (k vị trí) các hệ số của ánh xạ tuyến tính θ_i, ta vẫn đảm bảo tính khả nghịch của biến đổi này. Ánh xạ ngược θ_i^{-1} tương ứng có bộ hệ số được xác định bằng cách xoay vòng (theo chiều ngược lại k vị trí) bộ hệ số của ánh xạ ngược ban đầu.

Từ **Nhận xét 5.4** và **Nhận xét 5.6**, suy ra:

Nhận xét 5.7: *Khi đã chọn được một bộ hệ số mạnh, ta có thể phát sinh ra thêm* $Nw - 1$ *bộ hệ số mạnh (có cùng giá trị Branch Number) bằng cách dịch chuyển xoay vòng các hệ số này đi* k *vị trí,* $k = 1,2, \ldots, Nw - 1$. *Bộ hệ số trong ánh xạ ngược tương ứng được xác định bằng cách dịch chuyển xoay vòng (theo chiều ngược lại* k *vị trí) bộ hệ số của ánh xạ ngược ban đầu.*

5.3 Kiểm tra sơ bộ với vector nhị phân

5.3.1 Giải thuật kiểm tra sơ bộ

Từ định nghĩa về Branch Number, suy ra:

Nhận xét 5.8: *Nếu* $\theta[c]$ *có* $\mathcal{B}(\theta[c]) \geq Nw$ *thì với mọi vector thử nghiệm nhị phân (khác* $\vec{0}$*)* $t = (t_0, t_1, .., t_{Nw-1})$, $t_i \in \{0,1\}$, *ta luôn có* $wt(t) + wt(\theta[c](t)) \geq Nw$.

Như vậy, chúng ta có thể loại bỏ ngay những bộ hệ số không đạt được BranchNumber bằng Nw hay Nw+1 thông qua việc **kiểm tra sơ bộ** với các vector thử nghiệm nhị phân (khác $\vec{0}$). Khi đó, không gian thử nghiệm được thu hẹp từ $(GF(2^m))^{Nw} \setminus \{\vec{0}\}$ với $2^{m \times Nw} - 1$ phần tử về $(GF(2))^{Nw} \setminus \{\vec{0}\}$ với $2^{Nw} - 1$ phần tử.

Hình 5.1 trình bày giải thuật cải tiến dựa trên kiểm tra sơ bộ nhằm loại bỏ ngay những bộ hệ số không "mạnh" (branch number không đạt giá trị $\beta = Nw$ hay $Nw + 1$)

Hình 5.1. Giải thuật kiểm tra sơ bộ Branch number với ngưỡng $\beta = Nw$ hay $Nw + 1$

$\mathcal{B} \leftarrow +\infty$

Với mỗi vector dữ liệu nhị phân t khác 0, $t = [t_0, t_1, .., t_{Nw-1}] \in (GF(2))^{Nw}$

  $t' \leftarrow \theta[c](t)$

  $\mathcal{B} \leftarrow \min \{\mathcal{B}, wt(t) + wt(t')\}$

  Nếu $\mathcal{B} < \beta$ **thì**

    Bỏ qua việc tính Branch Number cho bộ hệ số này

    Trả về kết quả «THẤT BẠI»

  Cuối nếu

Cuối với mỗi

Trả về kết quả « THÀNH CÔNG »

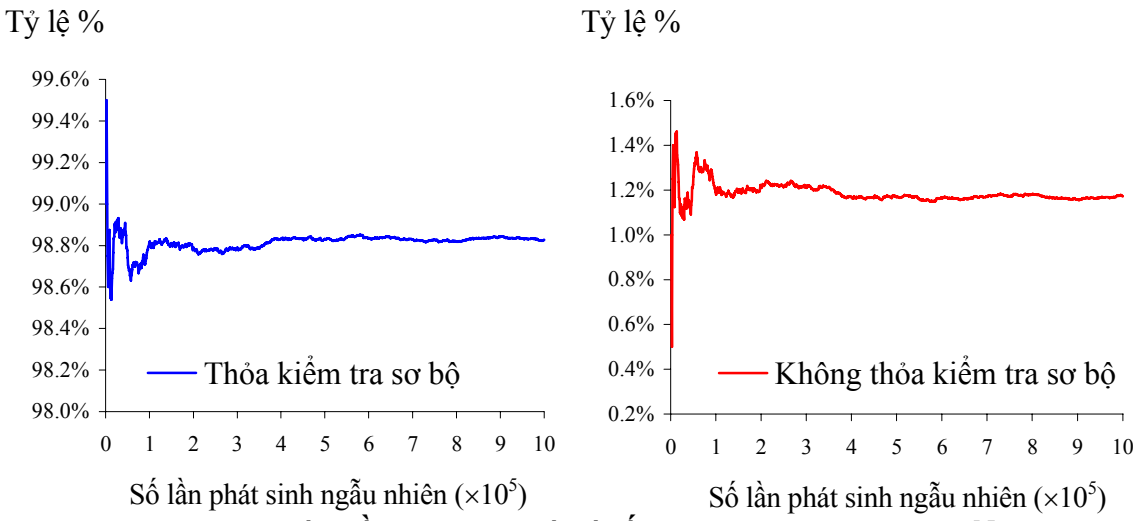
5.3.2 Kết quả thực nghiệm

Việc thử nghiệm kiểm tra sơ bộ được thực hiện với XAES trong trường hợp $m = 8$ trên trường Galois của Rijndael.

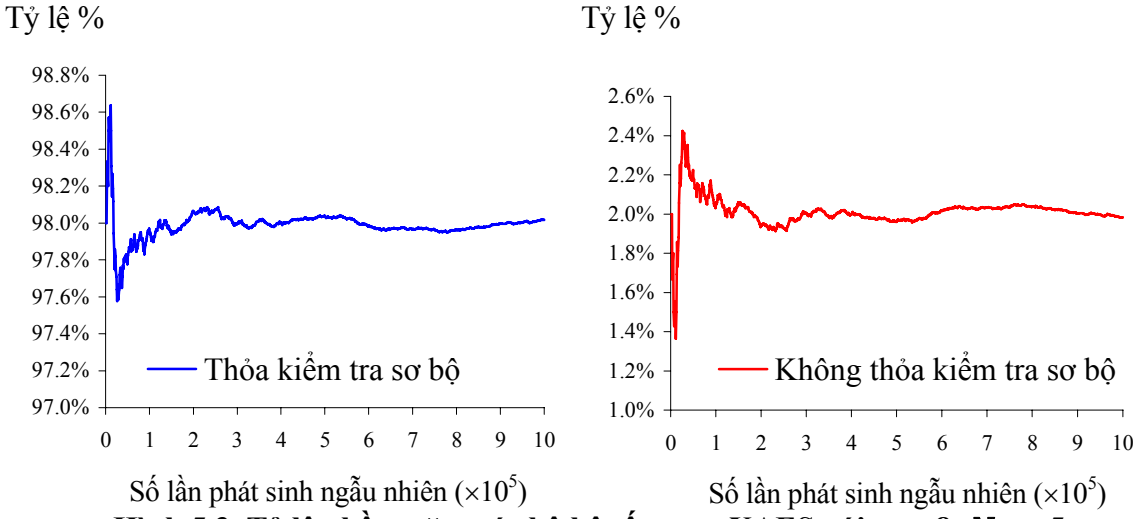
Theo giải thuật trên, trong quá trình phát sinh ngẫu nhiên các bộ hệ số, các bộ hệ số được phát sinh ngẫu nhiên có thể chia ra thành hai nhóm chính:

- Nhóm I: gồm các bộ hệ số tương ứng với các đa thức khả nghịch và vượt qua được quá trình kiểm tra sơ bộ. Đây là các ứng cử viên cho bộ hệ số "mạnh".
- Nhóm II: gồm các bộ hệ số tương ứng với đa thức không khả nghịch (gọi là nhóm các bộ hệ số không khả nghịch) hoặc không vượt qua được quá trình kiểm tra sơ bộ.

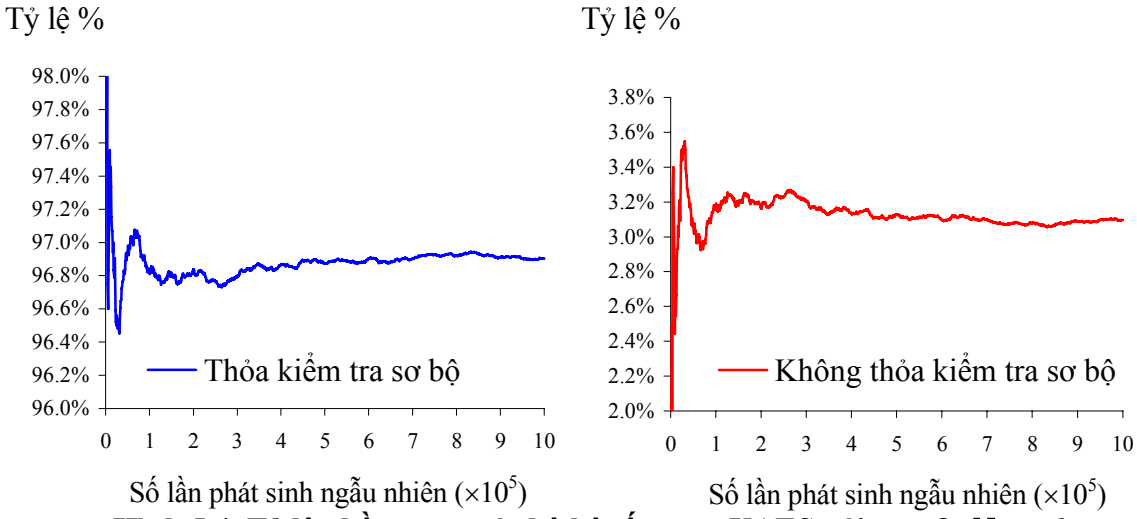
Hình 5.2, Hình 5.3, Hình 5.4, Hình 5.5 và Hình 5.6 lần lượt thể hiện kết quả khảo sát tỷ lệ phần trăm của từng nhóm các bộ hệ số trong n lần phát sinh ngẫu nhiên bộ hệ số với n = 100, 200, 300, …, 10^6 với $\beta = Nw + 1$.Qua kết quả khảo sát, các ứng cử viên cho bộ hệ số mạnh chiếm tỉ lệ khá cao (trên 95%).



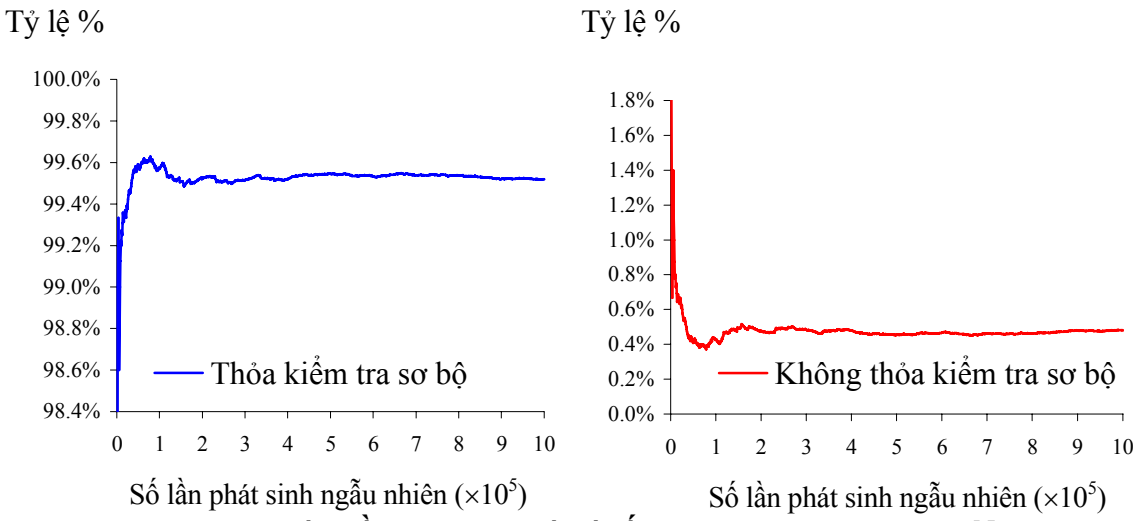
Hình 5.2. Tỷ lệ phần trăm các bộ hệ số trong XAES với $m = 8$, $Nw = 4$



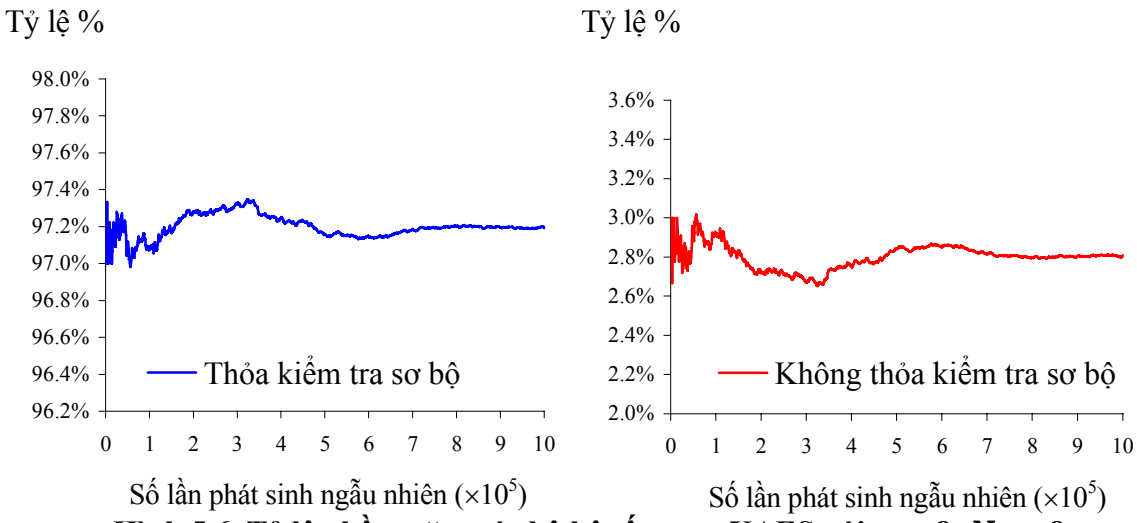
Hình 5.3. Tỷ lệ phần trăm các bộ hệ số trong XAES với $m = 8$, $Nw = 5$



Hình 5.4. Tỷ lệ phần trăm các bộ hệ số trong XAES với $m = 8$, $Nw = 6$

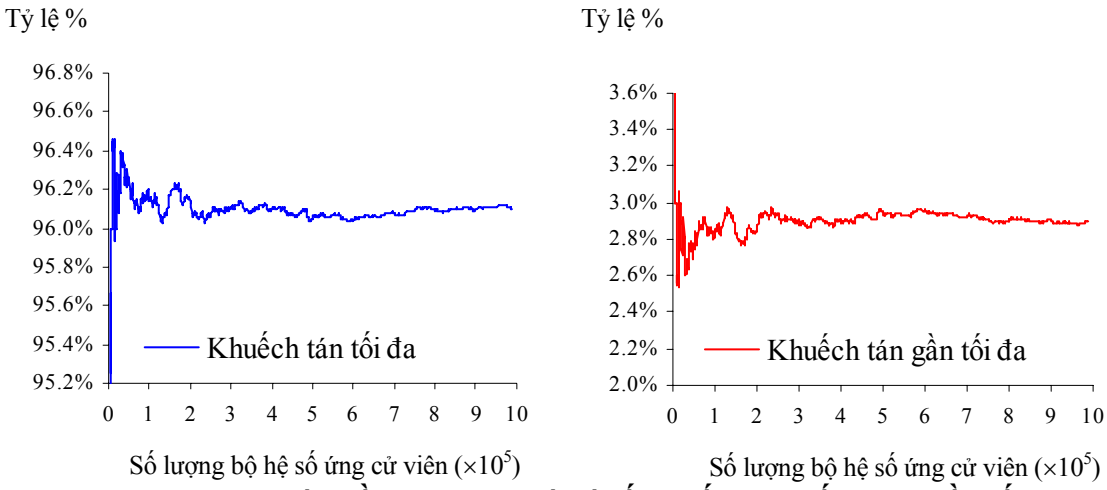


Hình 5.5. Tỷ lệ phần trăm các bộ hệ số trong XAES với m = 8, Nw = 7

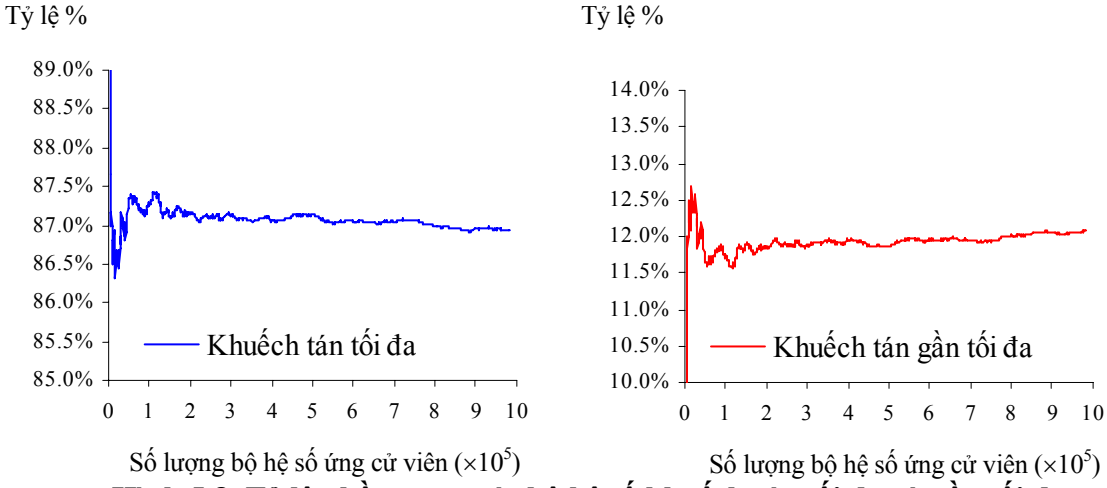


Hình 5.6. Tỷ lệ phần trăm các bộ hệ số trong XAES với m = 8, Nw = 8

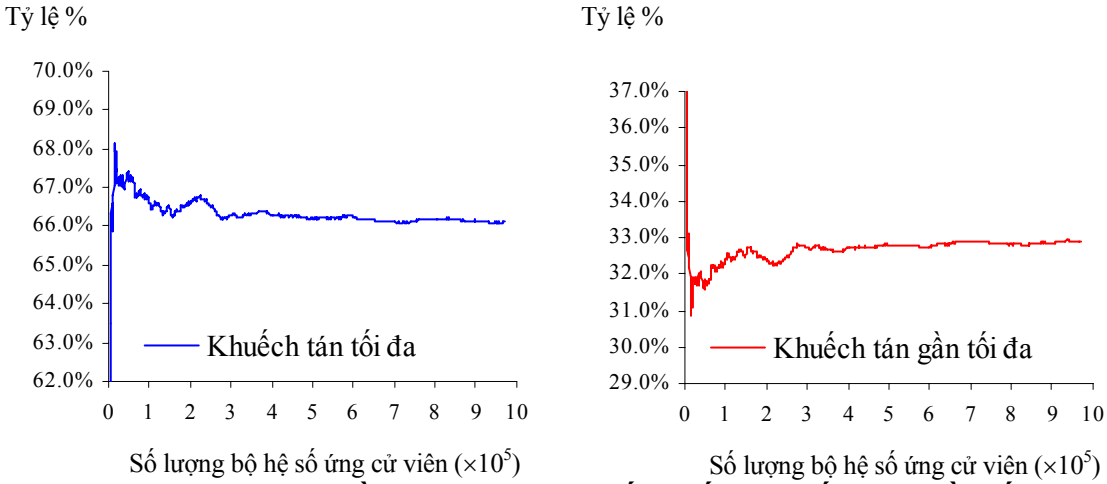
Để kiểm tra tính hiệu quả của việc sử dụng phép kiểm tra sơ bộ, chúng tôi khảo sát tỷ lệ bộ hệ số khuếch tán tối đa và tỷ lệ bộ hệ số khuếch tán gần tối đa trong số n ứng cử viên được xác định thông qua phép kiểm tra sơ bộ (với n = 100, 200, …, 10^6). Kết quả việc khảo sát trong trường hợp m = 8 và Nw có giá trị từ 4 đến 8 lần lượt được thể hiện qua Hình 5.7, Hình 5.8, Hình 5.9, Hình 5.10 và Hình 5.11.



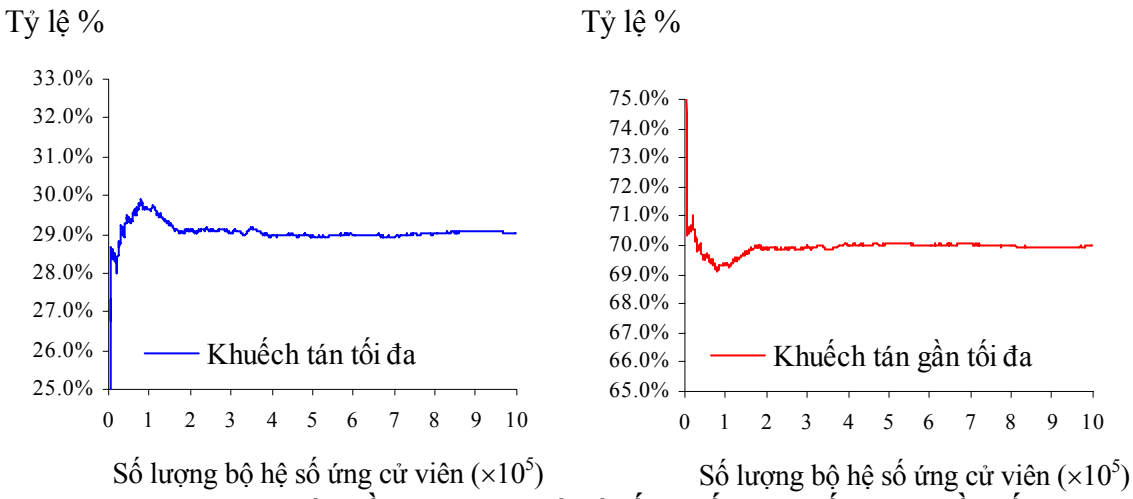
Hình 5.7. Tỷ lệ phần trăm các bộ hệ số khuếch tán tối đa và gần tối đa trong các bộ hệ số ứng cử viên ($m = 8$, $Nw = 4$)



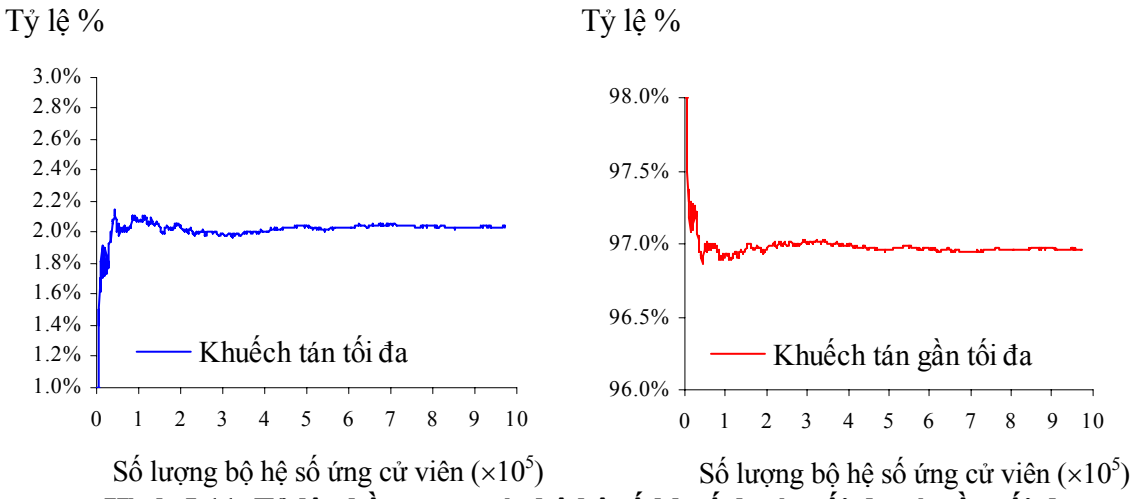
Hình 5.8. Tỷ lệ phần trăm các bộ hệ số khuếch tán tối đa và gần tối đa trong các bộ hệ số ứng cử viên ($m = 8$, $Nw = 5$)



Hình 5.9. Tỷ lệ phần trăm các bộ hệ số khuếch tán tối đa và gần tối đa trong các bộ hệ số ứng cử viên ($m = 8$, $Nw = 6$)



Hình 5.10. Tỷ lệ phần trăm các bộ hệ số khuếch tán tối đa và gần tối đa trong các bộ hệ số ứng cử viên (m = 8, Nw = 7)



Hình 5.11. Tỷ lệ phần trăm các bộ hệ số khuếch tán tối đa và gần tối đa trong các bộ hệ số ứng cử viên (m = 8, Nw = 8)

Qua kết quả khảo sát, có thể rút ra một số nhận xét sau:

- Khi Nw càng lớn thì tỷ lệ bộ hệ số khuếch tán tối đa trong các ứng cử viên càng giảm nhưng tỷ lệ bộ hệ số khuếch tán gần tối đa trong các ứng cử viên lại tăng. Điều này cho thấy khi Nw càng lớn, việc chọn bộ hệ số khuếch tán tối đa sẽ khó hơn việc chọn bộ hệ số khuếch tán gần tối đa. Do cả hai loại bộ hệ số này đều đáp ứng yêu cầu về tính an toàn cho thuật toán XAES đối với phương pháp sai phân và phương pháp tuyến tính trong phân tích mã (xin xem chứng minh trong Chương 3 và Chương 4) nên trong XAES chúng tôi đã đề nghị sử dụng cả hai loại bộ hệ số này.

- Trên 99% các ứng cử viên trong các thử nghiệm thật sự là bộ hệ số mạnh (bao gồm bộ hệ số khuếch tán tối đa và khuếch tán gần tối đa). Hầu hết các bộ hệ số ứng cử viên còn lại, thông qua quá trình kiểm tra ngẫu nhiên (được trình bày trong phần 5.4) được xác định là các bộ hệ số mạnh ngưỡng $T = 0.99$

5.4 Kiểm tra ngẫu nhiên

5.4.1 Giải thuật cải tiến sử dụng kiểm tra ngẫu nhiên

Cho bộ hệ số ứng cử viên $c = (c_0, c_1,.., c_{Nw-1})$. Ta cần kiểm định 2 giả thuyết sau:

H_0: c là bộ hệ số "mạnh"

H_1: c không phải là bộ hệ số "mạnh"

Chọn ngẫu nhiên ν vector thử nghiệm $t \in \mathrm{GF}(2^m)^{Nw} \setminus \{\vec{0}\}$. Nếu bộ hệ số ứng cử viên c không vượt qua được việc kiểm thử với ν vector thử nghiệm này thì giả thiết H_0 bị bác bỏ, tức là bộ hệ số c sẽ bị loại bỏ.

Ngược lại, nếu bộ hệ số c vượt qua việc kiểm thử với toàn bộ ν vector thử nghiệm ngẫu nhiên này thì có thể xem là c là bộ hệ số "mạnh" đối với ν vector thử nghiệm được chọn ngẫu nhiên này. Khi đó, xác suất ξ chọn được vector $t \in \mathrm{GF}(2^m)^{Nw} \setminus \{\vec{0}\}$ sao cho $wt(t) + wt(\theta[c](t)) < \beta$ với $\beta = Nw$ không vượt quá $1/\nu$, hay nói cách khác, có thể xem c là bộ hệ số mạnh ngưỡng $T \approx (\nu - 1)/\nu$.

Hình 5.12. Giải thuật cải tiến kiểm tra Branch Number bằng bộ test ngẫu nhiên

```
// Kiểm tra ngẫu nhiên
B ← +∞
Với mỗi vector trong số ν vector ngẫu nhiên t khác 0
    t' ← θ[c] (t)
    B ← min {B, wt(t) + wt(t') }
    Nếu B < β thì
        Bỏ qua việc tính Branch Number cho bộ hệ số này
        Trả về kết quả «THẤT BẠI»
    Cuối nếu
Cuối với mỗi
Trả về kết quả « THÀNH CÔNG »
```

5.4.2 Kết quả thực nghiệm

Sử dụng phép kiểm thử ngẫu nhiên (với $\beta = Nw$) cho các bộ hệ số ứng cử viên được xác định sau quá trình kiểm tra sơ bộ (trong các trường hợp m=8 và Nw có giá trị từ 4 đến 8), kết quả thực nghiệm cho thấy trên 99% các bộ hệ số ứng cử viên đều là các bộ hệ số mạnh ngưỡng T = 0.999 (sử dụng ν = 1000), tức là xác suất ξ chọn được vector dữ liệu làm cho bộ hệ số có hệ số branch number nhỏ hơn $\beta = Nw$ không vượt quá $1 / \nu = 10^{-3}$.

5.5 Bộ hệ số tối ưu

Theo [16], để tối ưu tốc độ thực hiện trên các thiết bị có năng lực xử lý và lưu trữ hạn chế, chúng ta cần chọn các bộ hệ số có giá trị hệ số càng nhỏ càng tốt. Từ đó, chúng tôi đề xuất định nghĩa về bộ hệ số "tối ưu":

Định nghĩa 5.6: *Bộ hệ số "tối ưu"(với tham số m và Nw trên trường* GF(2^m) *cho trước) là bộ hệ số khuếch tán tối đa và giá trị hệ số lớn nhất trong bộ hệ số là nhỏ nhất.*

$c = (c_0, c_1, ..., c_{Nw-1}) \in \mathrm{GF}(2^m)^{Nw}$ *là bộ hệ số tối ưu*

$$\Leftrightarrow \mathcal{B}(\theta[c]) = Nw + 1 \ \wedge$$

$$\forall c' = (c'_0, c'_1, ..., c'_{Nw-1}) \in \mathrm{GF}(2^m)^{Nw},\ \mathcal{B}(\theta[c']) = Nw + 1 \Rightarrow \max_{i=0..Nw-1}\{c_i\} \le \max_{i=0..Nw-1}\{c'_i\}$$

Bảng 5.1 thể hiện số lượng các bộ hệ số tối ưu và giá trị hệ số lớn nhất trong trường hợp m = 8 trên trường Galois của Rijndael và giá trị Nw từ 2 đến 8.

m	Nw	Số lượng bộ hệ số tối ưu	Giá trị hệ số lớn nhất
8	2	2	2
8	3	6	2
8	4	24	3
8	5	30	3
8	6	12	4
8	7	63	4
8	8	128	7

Bảng 5.1. Bảng thống kê số lượng bộ hệ số tối ưu và giá trị hệ số lớn nhất (trường hợp m = 8, Nw = 2, 3, …, 8 trên trường Galois của Rijndael)

Danh sách các bộ hệ số tối ưu này được trình bày chi tiết trong Phụ lục B .

5.6 Kết luận

Các kỹ thuật kiểm tra sơ bộ và kiểm tra ngẫu nhiên được đề nghị xuất phát từ nhu cầu thực tế cần phát sinh các bộ hệ số mạnh (hoặc bộ hệ số mạnh ngưỡng *T*) khi sử dụng XAES. Chính vì vậy, chúng tôi chọn thử nghiệm các kỹ thuật trong các trường hợp phổ biến là m = 8 (mỗi đơn vị dữ liệu được xử lý là byte) và Nw từ 4 đến 8 (phù hợp kiến trúc 32-bit và 64-bit). Kết quả thử nghiệm cho thấy kỹ thuật kiểm tra sơ bộ và kiểm tra ngẫu nhiên được đề nghị tương đối hiệu quả trong việc kiểm tra một bộ hệ số phát sinh ngẫu nhiên có phải là bộ hệ số mạnh (hoặc bộ hệ số mạnh ngưỡng *T*) hay không.

Trong kiểm tra sơ bộ với vector nhị phân để kiểm tra một bộ hệ số được phát sinh ngẫu nhiên có phải là bộ hệ số mạnh hay không, không gian thử nghiệm được thu hẹp từ $(\mathrm{GF}(2^m))^{Nw} \setminus \{\vec{0}\}$ với $2^{m \times Nw} - 1$ phần tử về $(\mathrm{GF}(2))^{Nw} \setminus \{\vec{0}\}$ với $2^{Nw} - 1$ phần tử. Việc thử nghiệm được chúng tôi thực hiện với m = 8 (trên trường Galois của Rijndael) và giá trị Nw từ 4 đến 8. Kết quả thực nghiệm cho thấy có trên 95% các bộ hệ số phát sinh ngẫu nhiên là ứng cử viên cho bộ hệ số mạnh, trong đó, trên 99% các ứng cử viên thật sự là bộ hệ số mạnh.

Với kỹ thuật kiểm tra ngẫu nhiên với ν vector thử nghiệm được phát sinh ngẫu nhiên cho phép xác định các bộ hệ số mạnh ngưỡng $T \approx (\nu - 1)/\nu$. Áp dụng kiểm tra ngẫu nhiên vào các kết quả thử nghiệm của kiểm tra sơ bộ cho thấy trên 99% các bộ hệ số ứng cử viên đều là các bộ hệ số mạnh ngưỡng T = 0.999 (sử dụng ν = 1000), tức là xác suất ξ chọn được vector dữ liệu làm cho bộ hệ số có hệ số branch number nhỏ hơn $\beta = Nw$ không vượt quá $1 / \nu = 10^{-3}$.

Ngoài ra, chúng tôi đã xác định tất cả các bộ hệ số tối ưu trong trường hợp m = 8 trên trường Galois của Rijndael và giá trị Nw từ 2 đến 8.

Một số vấn đề mở:

- Chứng minh hình thức hiệu quả của các kỹ thuật kiểm tra sơ bộ và kiểm tra ngẫu nhiên trong trường hợp tổng quát.
- Việc đề ra tiêu chí để đánh giá chính xác và hiệu quả một bộ hệ số mạnh hiện còn là vấn đề mở.
- Kỹ thuật tổng quát để ước lượng chặn dưới của giá trị Branch Number của ánh xạ bất kỳ hiện vẫn chưa được giải quyết.

Chương 6

Gray S-box cho AES

Tóm tắt chương:

✍ Nội dung chương 6 trình bày về Gray S-box cho thuật toán AES. Gray S-box được xây dựng bằng cách bổ sung biến đổi từ biểu diễn nhị phân thông thường sang mã Gray nhị phân làm bước tiền xử lý cho S-box trong AES.

Nếu như đa thức biểu diễn của S-box nguyên thủy trong AES chỉ gồm 9 đơn thức, Gray S-box có biểu diễn đại số là đa thức gồm đầy đủ 255 đơn thức có hệ số khác 0. Nhờ đó, Gray S-box có độ an toàn cao hơn đối với tấn công đại số và tấn công nội suy. Do Gray S-box sử dụng trọn vẹn S-box của AES nên có thể tận dụng các thiết kế tối ưu đã được đề xuất cho việc cài đặt S-box của AES trên phần cứng.

Ngoài ra, Gray S-box vẫn đảm bảo các tính chất mật mã quan trọng của S-box nguyên thủy trong AES, gồm tính đồng nhất sai phân và tiêu chí SAC (Strict Avalanche Criterion).

Việc đề xuất Gray S-box cho AES nhằm minh họa cho việc bổ sung ánh xạ tuyến tính làm bước tiền xử lý trong việc xây dựng S-box của XAES có thể giúp tăng độ an toàn của S-box đối với tấn công đại số và tấn công nội suy.

6.1 Mở đầu

Ngoài thành phần S-box trong biến đổi SubBytes, các biến đổi còn lại trong AES đều tuyến tính trên Z_2. Do đó, S-box có vai trò quan trọng trong thuật toán. Nhiều chuyên gia mật mã học đã nghiên cứu cấu trúc của AES. Trong [28], N. Ferguson đã trình bày vấn đề về tính đơn giản trong biểu diễn đại số của AES, đặc biệt là thành phần S-box của AES. Các vấn đề về cấu trúc đại số trong AES được phân tích cụ thể hơn trong [68], và biểu diễn đa thức của AES được đề xuất trong [74]. Trong các công trình này đều đặt ra vấn đề là S-box trong AES tuy có bậc cao (bậc 254) nhưng là đa thức thưa với 9 đơn thức nên S-box có thể trở thành mục tiêu tấn công của

phương pháp nội suy [41] và phương pháp đại số [14], đặc biệt với sự phát triển của các phương pháp đại số [57], [97]. Mặc dù hiện tại chưa có công trình nào công bố việc tấn công thành công vào S-box của AES nói riêng hay toàn bộ thuật toán AES nói chung, việc nghiên cứu nhằm tăng cường khả năng an toàn của S-box bằng cách nâng độ phức tạp trong biểu diễn đại số của S-box đã được nhiều người quan tâm.

Trong [60], Y.Cao đề xuất một phương án để cải tiến S-box trong AES bằng cách đảo ngược thứ tự ánh xạ nghịch đảo trên $GF(2^8)$ với ánh xạ affine trong S-box của AES. Như vậy, ánh xạ affine trong S-box của AES trở thành bước tiền xử lý cho ánh xạ nghịch đảo. Kết quả nhận được là S-box có biểu diễn đại số gồm đầy đủ 255 đơn thức khác 0. Tuy nhiên, với giải pháp này chưa tận dụng được các giải pháp tối ưu khi cài đặt S-box của AES trên phần cứng theo nhiều tiêu chí tối ưu khác nhau, ví dụ như tốc độ xử lý [37][42], tiết kiệm năng lượng [61][67], kích thước nhỏ [10][65][73][77].

J. Liu đề xuất việc tái sử dụng toàn bộ cấu trúc của S-box trong AES, đồng thời dùng chính ánh xạ affine trong AES làm bước tiền xử lý cho S-box[15]. Như vậy, S-box cải tiến có 2 ánh xạ affine giống nhau (chính là ánh xạ đã được dùng trong S-box của AES). Cách tiếp cận này cho phép tận dụng các ưu điểm của các thiết kế tối ưu được đề xuất cho S-box trong AES. Tuy nhiên, S-box sau khi cải tiến có biểu diễn đại số gồm 253 đơn thức khác 0, chưa đạt được số lượng đơn thức tối đa.

Trong chương này, trước tiên chúng tôi trình bày cách xây dựng biểu diễn đại số của S-box trong AES cũng như XAES, từ đó, rút ra nhận xét là kiến trúc S-box được sử dụng trong XAES có thể tạo ra các S-box có số đơn thức tối đa trong biểu diễn đại số nhiều hơn so với kiến trúc được dùng trong AES. Tiếp theo, chúng tôi đề xuất một cải tiến cho S-box trong AES bằng cách dùng biến đổi từ biểu diễn nhị phân sang mã Gray nhị phân để tạo ra Gray S-box có biểu diễn đại số là đa thức gồm đầy đủ 255 đơn thức, đồng thời đảm bảo các tiêu chí an toàn của S-box ban đầu trong thuật toán AES. Gray S-box là một minh họa cụ thể tính an toàn của kiến trúc S-box được sử dụng trong XAES.

6.2 Biểu diễn đại số của S-box trong XAES và AES

Do S-box trong XAES có cấu trúc tổng quát hơn so với trong AES, chúng tôi trình bày cách xác định biểu diễn đại số của S-box tổng quát trong XAES, từ đó áp dụng vào trường hợp S-box trong AES.

6.2.1 Xác định biểu diễn đại số của S-box trong XAES

Trong thao tác biến đổi SubBytes, mỗi phần tử (m-bit) được xem là một phần tử của trường GF(2^m).

Đặt $\mathcal{F}(x)$ là hàm nghịch đảo trên GF(2^m). Quy ước $\mathcal{F}(0) = 0$

$$\mathcal{F}(x) = \begin{cases} x^{-1}, x \neq 0 \\ 0, \quad x = 0 \end{cases} \tag{6.1}$$

Trên GF(2^m), $x^{-1} = x^{2^m-2}$ với $x \neq 0$. Do đó, $\mathcal{F}(x)$ có thể được biểu diễn lại như sau:

$$\mathcal{F}(x) = x^{2^m-2} \tag{6.2}$$

Đặt $\mathcal{L}_\varphi^{(0)}$ và $\mathcal{L}_\varphi^{(1)}$ lần lượt là ánh xạ tuyến tính trên GF$(2)^m$ tương ứng với ma trận nhị phân $\mathcal{M}_\varphi^{(0)}$ và $\mathcal{M}_\varphi^{(1)}$ được sử dụng trong bước tiền xử lý và hậu xử lý S-box của XAES. Do $\mathcal{L}_\varphi^{(0)}$ và $\mathcal{L}_\varphi^{(1)}$ đều là ánh xạ tuyến tính trên Z_2 nên có thể được biểu diễn dưới dạng đa thức tuyến tính hóa [58] trên GF(2^m):

$$\mathcal{L}_\varphi^{(0)}(x) = \sum_{j=0}^{m-1} \lambda_j^{(0)} x^{2^j} \text{ và } \mathcal{L}_\varphi^{(1)}(x) = \sum_{j=0}^{m-1} \lambda_j^{(1)} x^{2^j} \tag{6.3}$$

Thực chất, ánh xạ affine $\mathcal{A}_\varphi^{(i)}$ là sự kết hợp của ánh xạ tuyến tính $\mathcal{L}_\varphi^{(i)}$ và phép cộng với hằng số $c_\varphi^{(i)}$ (với i = 0, 1). Vì vậy, ánh xạ affine $\mathcal{A}_\varphi^{(0)}$ và $\mathcal{A}_\varphi^{(1)}$ có thể biểu diễn như sau trên GF(2^m):

$$\mathcal{A}_\varphi^{(0)}(x) = \sum_{j=0}^{m-1} \lambda_j^{(0)} x^{2^j} + c_\varphi^{(0)} \text{ và } \mathcal{A}_\varphi^{(1)}(x) = \sum_{j=0}^{m-1} \lambda_j^{(1)} x^{2^j} + c_\varphi^{(0)} \tag{6.4}$$

S-box S_φ trong XAES là sự kết hợp của ánh xạ affine $\mathcal{A}_\varphi^{(0)}$, ánh xạ nghịch đảo $\mathcal{F}(x)$ và ánh xạ affine $\mathcal{A}_\varphi^{(1)}$:

$$S_\varphi = \mathcal{A}_\varphi^{(1)} \circ \mathcal{F} \circ \mathcal{A}_\varphi^{(0)} \tag{6.5}$$

❑ **Nhận xét**: mọi biến đổi affine trên GF(2^m) luôn có biểu diễn đa thức tuyến tính hóa gồm tối đa $m + 1$ đơn thức có bậc 0, 1, 2, …, 2^{m-1}. Nếu bỏ qua bước tiền xử lý bằng ánh xạ affine $\mathcal{A}_{\varphi}^{(0)}$, hay nói cách khác, xem $\mathcal{A}_{\varphi}^{(0)}$ là ánh xạ đồng nhất thì S-box nhận được sẽ có biểu diễn đại số trên GF(2^m) gồm tối đa $m + 1$ đơn thức. Đây chính là vấn đề xảy ra đối với S-box trong AES.

6.2.2 Áp dụng để xác định biểu diễn đại số của S-box trong AES

Với AES, $m = 8$ nên ta có :

$$\mathcal{F}(x) = x^{254} \tag{6.6}$$

Trong AES chỉ sử dụng một ánh xạ affine làm bước hậu xử lý. Do đó, có thể xem S-box trong AES có ánh xạ affine tiền xử lý là ánh xạ đồng nhất trên GF(2^8) :

$$\mathcal{A}_{\varphi}^{(0)}(x) = x \tag{6.7}$$

Ánh xạ affine $\mathcal{A}_{\varphi}^{(1)}$ là sự kết hợp của ánh xạ tuyến tính $\mathcal{L}_{\varphi}^{(1)}$ trên GF(2)^8 tương ứng với ma trận $\mathcal{M}_{\varphi}^{(1)}$ và phép cộng trên GF(2^8) với hằng số $c_{\varphi}^{(1)} = \{63\}$, trong đó :

$$\mathcal{M}_{\varphi}^{(1)} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \tag{6.8}$$

$\mathcal{L}_{\varphi}^{(1)}$ có biểu diễn dạng đa thức tuyến tính hóa trên GF(2^8) như sau:

$$\begin{aligned} \mathcal{L}_{\varphi}^{(1)}(x) &= \{8\mathrm{f}\}x^{\{80\}} + \{\mathrm{b}5\}x^{\{40\}} + \{01\}x^{\{20\}} + \{\mathrm{f}4\}x^{\{10\}} + \\ &\quad \{25\}x^{\{08\}} + \{\mathrm{f}9\}x^{\{04\}} + \{09\}x^{\{02\}} + \{05\}x \end{aligned} \tag{6.9}$$

Suy ra :

$$\begin{aligned} \mathcal{A}_{\varphi}^{(1)}(x) &= \{8\mathrm{f}\}x^{\{80\}} + \{\mathrm{b}5\}x^{\{40\}} + \{01\}x^{\{20\}} + \{\mathrm{f}4\}x^{\{10\}} + \\ &\quad \{25\}x^{\{08\}} + \{\mathrm{f}9\}x^{\{04\}} + \{09\}x^{\{02\}} + \{05\}x + \{63\} \end{aligned} \tag{6.10}$$

Vậy, biểu diễn đại số trên GF(2^8) của S-box trong AES như sau :

$$\begin{aligned} S_{\varphi}(x) &= \{8\mathrm{f}\}x^{\{7\mathrm{f}\}} + \{\mathrm{b}5\}x^{\{\mathrm{bf}\}} + \{01\}x^{\{\mathrm{df}\}} + \{\mathrm{f}4\}x^{\{\mathrm{ef}\}} + \\ &\quad \{25\}x^{\{\mathrm{f}7\}} + \{\mathrm{f}9\}x^{\{7\mathrm{b}\}} + \{09\}x^{\{7\mathrm{d}\}} + \{05\}x^{\{7\mathrm{e}\}} + \{63\} \end{aligned} \tag{6.11}$$

❑ **Nhận xét**: mọi biến đổi tuyến tính trên GF(2^8) luôn có biểu diễn đa thức tuyến tính hóa gồm tối đa 8 đơn thức. Do đó, nếu thay biến đổi tuyến tính $\mathcal{L}_{\varphi}^{(1)}$ được chọn trong thuật toán AES bằng một hàm tuyến tính khác, hoặc chọn hằng số $c_{\varphi}^{(1)}$ khác thì S-box cũng chỉ gồm tối đa 9 đơn thức trong biểu diễn đại số.

6.3 Gray S-box cho AES

6.3.1 Mã Gray nhị phân

Mã Gray được Frank Gray tại Bell Labs đề nghị năm 1947 (với tên gọi ban đầu là "reflected binary code") [34]. Đây là hệ thống số nhị phân mà trong đó hai giá trị liên tiếp nhau sai khác duy nhất 1 ký số [33]. Hiện nay, mã Gray thường được sử dụng trong cơ chế sửa lỗi của các hệ thống truyền thông kỹ thuật số, ví dụ như truyền hình cáp hay truyền hình kỹ thuật số mặt đất. Tuy nhiên, việc ứng dụng mã Gray trong lĩnh vực mã hóa thông tin còn là vấn đề khá mới mẻ.

Định nghĩa 6.1 [33]**:** Mã nhị phân Gray thứ $n \geq 1$ là một danh sách của tất cả các phần tử $(a_{n-1}, \ldots, a_1, a_0) \in \{0, 1\}^n$ sao cho mỗi lần ta ***di chuyển*** theo ***thứ tự*** danh sách thì ***chỉ có một thành tố nhị phân*** (bit) ***được thay đổi***, và được thay đổi một cách đơn giản và đều đặn.

❑ Ví dụ: Với $n = 3$; danh sách có 8 phần tử

{(0, 0, 0) , (0, 0, 1) , (0, 1, 1) , (0, 1, 0) , (1, 1, 0) , (1, 1, 1) , (1, 0, 1) , (1, 0, 0)}

Hình 6.1. Thuật toán chuyển biểu diễn nhị phân sang mã Gray nhị phân

Cho $x[0..n]$ là mảng bit biểu diễn nhị phân của giá trị x, $x[0]$ là bit LSB
Cho $y[0..n]$ là mảng bit dùng để lưu biểu diễn mã Gray tương ứng
$y[n] = x[n]$
for $i = n - 1$ downto 0
    $y[i] = x[i+1] \oplus x[i]$

Hình 6.1 thể hiện thuật toán chuyển biểu diễn nhị phân sang mã Gray nhị phân. Thao tác này tương ứng với biến đổi tuyến tính $\mathcal{G}$ trên GF(2)^8

$$\begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{pmatrix} \tag{6.12}$$

với x_i là bit thứ i của byte x (x_0 là bit LSB) và y_i là bit thứ i của byte y.

Đa thức tuyến tính hóa (trên trường Galois của Rijndael) tương ứng với ánh xạ $\mathcal{G}$ như sau:

$$\begin{aligned} \mathcal{G}(x) \;=\; & \{98\}\, x^{\{80\}} + \{e5\}\, x^{\{40\}} + \{4e\}\, x^{\{20\}} + \{3c\}\, x^{\{10\}} + \\ & \{13\}\, x^{\{08\}} + \{93\}\, x^{\{04\}} + \{9b\}\, x^{\{02\}} + \{15\}\, x \end{aligned} \tag{6.13}$$

6.3.2 Gray S-box cho AES

Ý tưởng chính trong phương án được đề xuất nhằm tăng độ phức tạp đại số của S-box của AES là thay thế x trong S-box của AES bằng 1 đa thức theo x (gồm nhiều đơn thức). Mã Gray nhị phân có ưu điểm dễ cài đặt, có thể nhanh chóng chuyển đổi từ biểu diễn nhị phân thông thường, ngoài ra, còn có biểu diễn đại số tương đối phức tạp (đa thức bậc 128 gồm 8 đơn thức) nên có thể được dùng để tích hợp làm bước tiền xử lý cho S-box mà vẫn đảm bảo tính hiệu quả của thuật toán.

Gray S-box S_{Gray} được xây dựng bằng cách dùng ánh xạ $\mathcal{G}$ để chuyển từ biểu diễn nhị phân sang mã Gray nhị phân làm bước tiền xử lý cho S-box trong AES. Nói cách khác, ánh xạ affine tiền xử lý $\mathcal{A}_{\varphi}^{(0)}$ được chọn dùng là $\mathcal{G}$.

$$S_{\text{Gray}} = \mathcal{A}_{\varphi}^{(1)} \circ \mathcal{F} \circ \mathcal{G} \tag{6.14}$$

Biểu diễn đại số của Gray S-box (trên trường Galois của Rijndael) như sau :

$$
\begin{aligned}
S_{\text{Gray}}(x) = {} & \{63\}\,x^{\{00\}} + \{78\}\,x^{\{01\}} + \{5b\}\,x^{\{02\}} + \{3c\}\,x^{\{03\}} + \{dd\}\,x^{\{04\}} + \{de\}\,x^{\{05\}} + \\
& \{52\}\,x^{\{06\}} + \{1f\}\,x^{\{07\}} + \{b1\}\,x^{\{08\}} + \{b3\}\,x^{\{09\}} + \{08\}\,x^{\{0a\}} + \{d8\}\,x^{\{0b\}} + \\
& \{52\}\,x^{\{0c\}} + \{a3\}\,x^{\{0d\}} + \{20\}\,x^{\{0e\}} + \{c8\}\,x^{\{0f\}} + \{76\}\,x^{\{10\}} + \{05\}\,x^{\{11\}} + \\
& \{22\}\,x^{\{12\}} + \{1b\}\,x^{\{13\}} + \{2e\}\,x^{\{14\}} + \{49\}\,x^{\{15\}} + \{99\}\,x^{\{16\}} + \{5c\}\,x^{\{17\}} + \\
& \{ee\}\,x^{\{18\}} + \{7d\}\,x^{\{19\}} + \{99\}\,x^{\{1a\}} + \{1e\}\,x^{\{1b\}} + \{b6\}\,x^{\{1c\}} + \{2d\}\,x^{\{1d\}} + \\
& \{de\}\,x^{\{1e\}} + \{75\}\,x^{\{1f\}} + \{22\}\,x^{\{20\}} + \{c6\}\,x^{\{21\}} + \{11\}\,x^{\{22\}} + \{90\}\,x^{\{23\}} + \\
& \{eb\}\,x^{\{24\}} + \{f2\}\,x^{\{25\}} + \{05\}\,x^{\{26\}} + \{72\}\,x^{\{27\}} + \{a0\}\,x^{\{28\}} + \{92\}\,x^{\{29\}} + \\
& \{a2\}\,x^{\{2a\}} + \{11\}\,x^{\{2b\}} + \{52\}\,x^{\{2c\}} + \{50\}\,x^{\{2d\}} + \{20\}\,x^{\{2e\}} + \{ce\}\,x^{\{2f\}} + \\
& \{82\}\,x^{\{30\}} + \{c8\}\,x^{\{31\}} + \{4b\}\,x^{\{32\}} + \{7d\}\,x^{\{33\}} + \{20\}\,x^{\{34\}} + \{fe\}\,x^{\{35\}} + \\
& \{d7\}\,x^{\{36\}} + \{2c\}\,x^{\{37\}} + \{f4\}\,x^{\{38\}} + \{41\}\,x^{\{39\}} + \{8b\}\,x^{\{3a\}} + \{44\}\,x^{\{3b\}} + \\
& \{6b\}\,x^{\{3c\}} + \{77\}\,x^{\{3d\}} + \{95\}\,x^{\{3e\}} + \{26\}\,x^{\{3f\}} + \{57\}\,x^{\{40\}} + \{92\}\,x^{\{41\}} + \\
& \{6f\}\,x^{\{42\}} + \{b4\}\,x^{\{43\}} + \{ce\}\,x^{\{44\}} + \{97\}\,x^{\{45\}} + \{1a\}\,x^{\{46\}} + \{7b\}\,x^{\{47\}} + \\
& \{6e\}\,x^{\{48\}} + \{e0\}\,x^{\{49\}} + \{b1\}\,x^{\{4a\}} + \{ba\}\,x^{\{4b\}} + \{df\}\,x^{\{4c\}} + \{38\}\,x^{\{4d\}} + \\
& \{e1\}\,x^{\{4e\}} + \{2b\}\,x^{\{4f\}} + \{62\}\,x^{\{50\}} + \{25\}\,x^{\{51\}} + \{3d\}\,x^{\{52\}} + \{7f\}\,x^{\{53\}} + \\
& \{dd\}\,x^{\{54\}} + \{92\}\,x^{\{55\}} + \{a5\}\,x^{\{56\}} + \{61\}\,x^{\{57\}} + \{63\}\,x^{\{58\}} + \{b6\}\,x^{\{59\}} + \\
& \{e5\}\,x^{\{5a\}} + \{32\}\,x^{\{5b\}} + \{26\}\,x^{\{5c\}} + \{fe\}\,x^{\{5d\}} + \{c9\}\,x^{\{5e\}} + \{26\}\,x^{\{5f\}} + \\
& \{b5\}\,x^{\{60\}} + \{7a\}\,x^{\{61\}} + \{b5\}\,x^{\{62\}} + \{98\}\,x^{\{63\}} + \{13\}\,x^{\{64\}} + \{49\}\,x^{\{65\}} + \\
& \{15\}\,x^{\{66\}} + \{d4\}\,x^{\{67\}} + \{a5\}\,x^{\{68\}} + \{92\}\,x^{\{69\}} + \{df\}\,x^{\{6a\}} + \{a3\}\,x^{\{6b\}} + \\
& \{46\}\,x^{\{6c\}} + \{7e\}\,x^{\{6d\}} + \{7b\}\,x^{\{6e\}} + \{6b\}\,x^{\{6f\}} + \{13\}\,x^{\{70\}} + \{a4\}\,x^{\{71\}} + \\
& \{91\}\,x^{\{72\}} + \{ac\}\,x^{\{73\}} + \{88\}\,x^{\{74\}} + \{92\}\,x^{\{75\}} + \{c4\}\,x^{\{76\}} + \{13\}\,x^{\{77\}} + \\
& \{3d\}\,x^{\{78\}} + \{53\}\,x^{\{79\}} + \{f3\}\,x^{\{7a\}} + \{66\}\,x^{\{7b\}} + \{e6\}\,x^{\{7c\}} + \{5c\}\,x^{\{7d\}} + \\
& \{be\}\,x^{\{7e\}} + \{7c\}\,x^{\{7f\}} + \{1c\}\,x^{\{80\}} + \{68\}\,x^{\{81\}} + \{d0\}\,x^{\{82\}} + \{f2\}\,x^{\{83\}} + \\
& \{5b\}\,x^{\{84\}} + \{e1\}\,x^{\{85\}} + \{bd\}\,x^{\{86\}} + \{f2\}\,x^{\{87\}} + \{2c\}\,x^{\{88\}} + \{af\}\,x^{\{89\}} + \\
& \{9e\}\,x^{\{8a\}} + \{a4\}\,x^{\{8b\}} + \{5b\}\,x^{\{8c\}} + \{55\}\,x^{\{8d\}} + \{22\}\,x^{\{8e\}} + \{19\}\,x^{\{8f\}} + \\
& \{e6\}\,x^{\{90\}} + \{b1\}\,x^{\{91\}} + \{1a\}\,x^{\{92\}} + \{37\}\,x^{\{93\}} + \{8d\}\,x^{\{94\}} + \{25\}\,x^{\{95\}} + \\
& \{03\}\,x^{\{96\}} + \{97\}\,x^{\{97\}} + \{a0\}\,x^{\{98\}} + \{2d\}\,x^{\{99\}} + \{a8\}\,x^{\{9a\}} + \{92\}\,x^{\{9b\}} + \\
& \{45\}\,x^{\{9c\}} + \{c7\}\,x^{\{9d\}} + \{5d\}\,x^{\{9e\}} + \{99\}\,x^{\{9f\}} + \{94\}\,x^{\{a0\}} + \{71\}\,x^{\{a1\}} + \\
& \{c1\}\,x^{\{a2\}} + \{4e\}\,x^{\{a3\}} + \{33\}\,x^{\{a4\}} + \{85\}\,x^{\{a5\}} + \{02\}\,x^{\{a6\}} + \{6b\}\,x^{\{a7\}} + \\
& \{86\}\,x^{\{a8\}} + \{b1\}\,x^{\{a9\}} + \{79\}\,x^{\{aa\}} + \{6c\}\,x^{\{ab\}} + \{11\}\,x^{\{ac\}} + \{fd\}\,x^{\{ad\}} + \\
& \{54\}\,x^{\{ae\}} + \{e9\}\,x^{\{af\}} + \{73\}\,x^{\{b0\}} + \{5f\}\,x^{\{b1\}} + \{4d\}\,x^{\{b2\}} + \{89\}\,x^{\{b3\}} + \\
& \{44\}\,x^{\{b4\}} + \{35\}\,x^{\{b5\}} + \{55\}\,x^{\{b6\}} + \{36\}\,x^{\{b7\}} + \{8b\}\,x^{\{b8\}} + \{93\}\,x^{\{b9\}} + \\
& \{37\}\,x^{\{ba\}} + \{b4\}\,x^{\{bb\}} + \{be\}\,x^{\{bc\}} + \{b0\}\,x^{\{bd\}} + \{2f\}\,x^{\{be\}} + \{78\}\,x^{\{bf\}} + \\
& \{b5\}\,x^{\{c0\}} + \{82\}\,x^{\{c1\}} + \{fb\}\,x^{\{c2\}} + \{88\}\,x^{\{c3\}} + \{76\}\,x^{\{c4\}} + \{e7\}\,x^{\{c5\}} + \\
& \{42\}\,x^{\{c6\}} + \{3c\}\,x^{\{c7\}} + \{74\}\,x^{\{c8\}} + \{23\}\,x^{\{c9\}} + \{27\}\,x^{\{ca\}} + \{f4\}\,x^{\{cb\}} + \\
& \{2e\}\,x^{\{cc\}} + \{dc\}\,x^{\{cd\}} + \{73\}\,x^{\{ce\}} + \{f3\}\,x^{\{cf\}} + \{8d\}\,x^{\{d0\}} + \{9b\}\,x^{\{d1\}} + \\
& \{13\}\,x^{\{d2\}} + \{83\}\,x^{\{d3\}} + \{88\}\,x^{\{d4\}} + \{cd\}\,x^{\{d5\}} + \{f4\}\,x^{\{d6\}} + \{24\}\,x^{\{d7\}} + \\
& \{f4\}\,x^{\{d8\}} + \{89\}\,x^{\{d9\}} + \{14\}\,x^{\{da\}} + \{19\}\,x^{\{db\}} + \{af\}\,x^{\{dc\}} + \{bc\}\,x^{\{dd\}} + \\
& \{76\}\,x^{\{de\}} + \{d9\}\,x^{\{df\}} + \{94\}\,x^{\{e0\}} + \{16\}\,x^{\{e1\}} + \{43\}\,x^{\{e2\}} + \{9a\}\,x^{\{e3\}} + \\
& \{eb\}\,x^{\{e4\}} + \{1b\}\,x^{\{e5\}} + \{25\}\,x^{\{e6\}} + \{42\}\,x^{\{e7\}} + \{db\}\,x^{\{e8\}} + \{35\}\,x^{\{e9\}} + \\
& \{eb\}\,x^{\{ea\}} + \{0b\}\,x^{\{eb\}} + \{2e\}\,x^{\{ec\}} + \{06\}\,x^{\{ed\}} + \{da\}\,x^{\{ee\}} + \{f7\}\,x^{\{ef\}} + \\
& \{b7\}\,x^{\{f0\}} + \{d7\}\,x^{\{f1\}} + \{73\}\,x^{\{f2\}} + \{64\}\,x^{\{f3\}} + \{54\}\,x^{\{f4\}} + \{98\}\,x^{\{f5\}} + \\
& \{7d\}\,x^{\{f6\}} + \{fe\}\,x^{\{f7\}} + \{ff\}\,x^{\{f8\}} + \{84\}\,x^{\{f9\}} + \{0d\}\,x^{\{fa\}} + \{84\}\,x^{\{fb\}} + \\
& \{f6\}\,x^{\{fc\}} + \{ab\}\,x^{\{fd\}} + \{fc\}\,x^{\{fe\}}
\end{aligned}
\tag{6.15}
$$

Như vậy, việc sử dụng biến đổi từ dạng nhị phân thông thường sang biểu diễn bằng mã Gray nhị phân làm bước tiền xử lý cho S-box đã giúp tăng số lượng đơn thức trong biểu diễn đại số của S-box trong AES từ 9 đơn thức lên 255 đơn thức (số lượng đơn thức tối đa của đa thức trên GF (2^8)). Điều này giúp khắc phục vấn đề tính thưa trong biểu diễn đại số (trên trường Galois của Rijndael) của S-box trong AES đối với tấn công nội suy [41] và tấn công tuyến tính [14].

Ánh xạ ngược của Gray S-box, ký hiệu S^{-1}_{Gray}, có biểu diễn đại số (trên trường Galois của Rijndael) như sau:

$$
\begin{aligned}
S^{-1}_{\text{Gray}}(x) = \\
&\{63\}\,x^{\{00\}} + \{f2\}\,x^{\{01\}} + \{9f\}\,x^{\{02\}} + \{c2\}\,x^{\{03\}} + \{7c\}\,x^{\{04\}} + \{47\}\,x^{\{05\}} + \\
&\{83\}\,x^{\{06\}} + \{f3\}\,x^{\{07\}} + \{89\}\,x^{\{08\}} + \{f7\}\,x^{\{09\}} + \{14\}\,x^{\{0a\}} + \{c5\}\,x^{\{0b\}} + \\
&\{36\}\,x^{\{0c\}} + \{08\}\,x^{\{0d\}} + \{7f\}\,x^{\{0e\}} + \{23\}\,x^{\{0f\}} + \{36\}\,x^{\{10\}} + \{ab\}\,x^{\{11\}} + \\
&\{cd\}\,x^{\{12\}} + \{64\}\,x^{\{13\}} + \{e0\}\,x^{\{14\}} + \{b5\}\,x^{\{15\}} + \{5f\}\,x^{\{16\}} + \{da\}\,x^{\{17\}} + \\
&\{c7\}\,x^{\{18\}} + \{27\}\,x^{\{19\}} + \{22\}\,x^{\{1a\}} + \{1e\}\,x^{\{1b\}} + \{a4\}\,x^{\{1c\}} + \{4f\}\,x^{\{1d\}} + \\
&\{97\}\,x^{\{1e\}} + \{39\}\,x^{\{1f\}} + \{9b\}\,x^{\{20\}} + \{ae\}\,x^{\{21\}} + \{a1\}\,x^{\{22\}} + \{eb\}\,x^{\{23\}} + \\
&\{c1\}\,x^{\{24\}} + \{4f\}\,x^{\{25\}} + \{e4\}\,x^{\{26\}} + \{24\}\,x^{\{27\}} + \{26\}\,x^{\{29\}} + \{c5\}\,x^{\{2a\}} + \\
&\{24\}\,x^{\{2b\}} + \{4a\}\,x^{\{2c\}} + \{8d\}\,x^{\{2d\}} + \{75\}\,x^{\{2e\}} + \{b6\}\,x^{\{2f\}} + \{3e\}\,x^{\{30\}} + \\
&\{f0\}\,x^{\{31\}} + \{d8\}\,x^{\{32\}} + \{f6\}\,x^{\{33\}} + \{62\}\,x^{\{34\}} + \{4e\}\,x^{\{35\}} + \{53\}\,x^{\{36\}} + \\
&\{37\}\,x^{\{37\}} + \{d5\}\,x^{\{38\}} + \{95\}\,x^{\{39\}} + \{d8\}\,x^{\{3a\}} + \{46\}\,x^{\{3b\}} + \{7f\}\,x^{\{3c\}} + \\
&\{31\}\,x^{\{3d\}} + \{38\}\,x^{\{3e\}} + \{de\}\,x^{\{3f\}} + \{e7\}\,x^{\{40\}} + \{19\}\,x^{\{41\}} + \{d3\}\,x^{\{42\}} + \\
&\{4a\}\,x^{\{43\}} + \{06\}\,x^{\{44\}} + \{c1\}\,x^{\{45\}} + \{11\}\,x^{\{46\}} + \{ea\}\,x^{\{47\}} + \{1b\}\,x^{\{48\}} + \\
&\{d3\}\,x^{\{49\}} + \{1b\}\,x^{\{4a\}} + \{43\}\,x^{\{4b\}} + \{9b\}\,x^{\{4c\}} + \{dc\}\,x^{\{4d\}} + \{43\}\,x^{\{4e\}} + \\
&\{b7\}\,x^{\{4f\}} + \{19\}\,x^{\{50\}} + \{ab\}\,x^{\{51\}} + \{80\}\,x^{\{52\}} + \{f9\}\,x^{\{53\}} + \{94\}\,x^{\{54\}} + \\
&\{9d\}\,x^{\{55\}} + \{05\}\,x^{\{56\}} + \{4f\}\,x^{\{57\}} + \{e4\}\,x^{\{58\}} + \{02\}\,x^{\{59\}} + \{e0\}\,x^{\{5a\}} + \\
&\{f1\}\,x^{\{5b\}} + \{b3\}\,x^{\{5c\}} + \{ff\}\,x^{\{5d\}} + \{0f\}\,x^{\{5e\}} + \{cc\}\,x^{\{5f\}} + \{a6\}\,x^{\{60\}} + \\
&\{bd\}\,x^{\{61\}} + \{ab\}\,x^{\{62\}} + \{36\}\,x^{\{63\}} + \{b2\}\,x^{\{64\}} + \{39\}\,x^{\{65\}} + \{90\}\,x^{\{66\}} + \\
&\{f7\}\,x^{\{67\}} + \{8a\}\,x^{\{68\}} + \{bb\}\,x^{\{69\}} + \{dc\}\,x^{\{6a\}} + \{92\}\,x^{\{6b\}} + \{a2\}\,x^{\{6c\}} + \\
&\{51\}\,x^{\{6d\}} + \{e2\}\,x^{\{6e\}} + \{36\}\,x^{\{6f\}} + \{0f\}\,x^{\{70\}} + \{53\}\,x^{\{71\}} + \{77\}\,x^{\{72\}} + \\
&\{97\}\,x^{\{73\}} + \{dd\}\,x^{\{74\}} + \{14\}\,x^{\{75\}} + \{58\}\,x^{\{76\}} + \{07\}\,x^{\{77\}} + \{a4\}\,x^{\{78\}} + \\
&\{c3\}\,x^{\{79\}} + \{9c\}\,x^{\{7a\}} + \{eb\}\,x^{\{7b\}} + \{52\}\,x^{\{7c\}} + \{48\}\,x^{\{7d\}} + \{d7\}\,x^{\{7e\}} + \\
&\{40\}\,x^{\{7f\}} + \{28\}\,x^{\{80\}} + \{c8\}\,x^{\{81\}} + \{ce\}\,x^{\{82\}} + \{75\}\,x^{\{83\}} + \{5b\}\,x^{\{84\}} + \\
&\{40\}\,x^{\{85\}} + \{3d\}\,x^{\{86\}} + \{85\}\,x^{\{87\}} + \{38\}\,x^{\{88\}} + \{49\}\,x^{\{89\}} + \{9b\}\,x^{\{8a\}} + \\
&\{62\}\,x^{\{8b\}} + \{32\}\,x^{\{8c\}} + \{db\}\,x^{\{8d\}} + \{15\}\,x^{\{8e\}} + \{a4\}\,x^{\{8f\}} + \{b8\}\,x^{\{90\}} + \\
&\{e4\}\,x^{\{91\}} + \{b8\}\,x^{\{92\}} + \{0b\}\,x^{\{93\}} + \{63\}\,x^{\{94\}} + \{f6\}\,x^{\{95\}} + \{08\}\,x^{\{96\}} + \\
&\{3a\}\,x^{\{97\}} + \{d4\}\,x^{\{98\}} + \{82\}\,x^{\{99\}} + \{47\}\,x^{\{9a\}} + \{a8\}\,x^{\{9b\}} + \{2a\}\,x^{\{9c\}} + \\
&\{25\}\,x^{\{9d\}} + \{47\}\,x^{\{9e\}} + \{ca\}\,x^{\{9f\}} + \{fd\}\,x^{\{a0\}} + \{d3\}\,x^{\{a1\}} + \{19\}\,x^{\{a2\}} + \\
&\{a5\}\,x^{\{a3\}} + \{7b\}\,x^{\{a4\}} + \{aa\}\,x^{\{a5\}} + \{25\}\,x^{\{a6\}} + \{3f\}\,x^{\{a7\}} + \{99\}\,x^{\{a8\}} + \\
&\{0a\}\,x^{\{a9\}} + \{bd\}\,x^{\{aa\}} + \{80\}\,x^{\{ab\}} + \{fa\}\,x^{\{ac\}} + \{19\}\,x^{\{ad\}} + \{6a\}\,x^{\{ae\}} +
\end{aligned}
$$

$$\{cb\}\,x^{\{af\}} + \{e0\}\,x^{\{b0\}} + \{dc\}\,x^{\{b1\}} + \{32\}\,x^{\{b2\}} + \{6f\}\,x^{\{b3\}} + \{1e\}\,x^{\{b4\}} +$$
$$\{da\}\,x^{\{b5\}} + \{3f\}\,x^{\{b6\}} + \{81\}\,x^{\{b7\}} + \{36\}\,x^{\{b8\}} + \{a9\}\,x^{\{b9\}} + \{d0\}\,x^{\{ba\}} +$$
$$\{ec\}\,x^{\{bb\}} + \{d6\}\,x^{\{bc\}} + \{78\}\,x^{\{bd\}} + \{d2\}\,x^{\{be\}} + \{6e\}\,x^{\{bf\}} + \{5b\}\,x^{\{c0\}} +$$
$$\{47\}\,x^{\{c1\}} + \{46\}\,x^{\{c2\}} + \{a9\}\,x^{\{c3\}} + \{ff\}\,x^{\{c4\}} + \{14\}\,x^{\{c5\}} + \{6e\}\,x^{\{c6\}} +$$
$$\{c2\}\,x^{\{c7\}} + \{d6\}\,x^{\{c8\}} + \{50\}\,x^{\{c9\}} + \{27\}\,x^{\{ca\}} + \{ed\}\,x^{\{cb\}} + \{d4\}\,x^{\{cc\}} +$$
$$\{ab\}\,x^{\{cd\}} + \{fb\}\,x^{\{ce\}} + \{ea\}\,x^{\{cf\}} + \{c1\}\,x^{\{d0\}} + \{fc\}\,x^{\{d1\}} + \{32\}\,x^{\{d2\}} +$$
$$\{54\}\,x^{\{d3\}} + \{fa\}\,x^{\{d4\}} + \{5a\}\,x^{\{d5\}} + \{41\}\,x^{\{d6\}} + \{ac\}\,x^{\{d7\}} + \{3a\}\,x^{\{d8\}} +$$
$$\{61\}\,x^{\{d9\}} + \{64\}\,x^{\{da\}} + \{d9\}\,x^{\{db\}} + \{ed\}\,x^{\{dc\}} + \{85\}\,x^{\{dd\}} + \{69\}\,x^{\{de\}} +$$
$$13\}\,x^{\{df\}} + \{be\}\,x^{\{e0\}} + \{2a\}\,x^{\{e1\}} + \{23\}\,x^{\{e2\}} + \{c0\}\,x^{\{e3\}} + \{64\}\,x^{\{e4\}} +$$
$$\{21\}\,x^{\{e5\}} + \{56\}\,x^{\{e6\}} + \{10\}\,x^{\{e7\}} + \{95\}\,x^{\{e8\}} + \{27\}\,x^{\{e9\}} + \{cd\}\,x^{\{ea\}} +$$
$$\{b7\}\,x^{\{eb\}} + \{df\}\,x^{\{ec\}} + \{54\}\,x^{\{ed\}} + \{c9\}\,x^{\{ee\}} + \{16\}\,x^{\{ef\}} + \{92\}\,x^{\{f0\}} +$$
$$\{d0\}\,x^{\{f1\}} + \{a1\}\,x^{\{f2\}} + \{0a\}\,x^{\{f3\}} + \{e5\}\,x^{\{f4\}} + \{da\}\,x^{\{f5\}} + \{41\}\,x^{\{f6\}} +$$
$$\{9e\}\,x^{\{f7\}} + \{14\}\,x^{\{f8\}} + \{2b\}\,x^{\{f9\}} + \{e9\}\,x^{\{fa\}} + \{d1\}\,x^{\{fb\}} + \{be\}\,x^{\{fc\}} +$$
$$\{8c\}\,x^{\{fd\}} + \{fc\}\,x^{\{fe\}} \qquad (6.16)$$

Biểu diễn đại số (trên trường Galois của Rijndael) của ánh xạ ngược S^{-1}_{Gray} có bậc tối đa (bậc 254) với 254/255 đơn thức khác 0. Điều này hạn chế khả năng vận dụng thành công cách tiếp cận phân tích mã bằng phương pháp đại số hay nội suy đối với việc khai thác ánh xạ ngược S^{-1}_{Gray}.

Bảng 6.1 thể hiện quy tắc thay thế bằng Gray S-box: mỗi giá trị $\{xy\}$ được thay thế bằng giá trị tại dòng x cột y của bảng.

		y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	63	7c	7b	77	6f	c5	6b	f2	fe	d7	76	ab	67	2b	01	30
	1	ad	d4	af	a2	72	c0	a4	9c	fa	59	f0	47	c9	7d	82	ca
	2	04	c7	c3	23	05	9a	96	18	eb	27	75	b2	80	e2	12	07
	3	34	a5	f1	e5	31	15	d8	71	36	3f	cc	f7	93	26	fd	b7
	4	d0	ef	fb	aa	33	85	4d	43	50	3c	a8	9f	02	7f	f9	45
	5	bc	b6	21	da	f3	d2	ff	10	92	9d	f5	38	40	8f	a3	51
	6	53	d1	ed	00	b1	5b	fc	20	4a	4c	cf	58	be	39	cb	6a
	7	52	3b	b3	d6	2f	84	e3	29	1b	6e	a0	5a	2c	1a	83	09
	8	ba	78	2e	25	b4	c6	a6	1c	4b	bd	8a	8b	74	1f	dd	e8
	9	61	35	b9	57	1d	9e	c1	86	48	03	0e	f6	b5	66	3e	70
	a	8c	a1	0d	89	42	68	e6	bf	b0	54	16	bb	2d	0f	99	41
	b	9b	1e	e9	87	28	df	55	ce	69	d9	94	8e	98	11	f8	e1
	c	e0	32	0a	3a	24	5c	06	49	91	95	79	e4	ac	62	d3	c2
	d	6c	56	ea	f4	ae	08	7a	65	8d	d5	a9	4e	37	6d	c8	e7
	e	60	81	dc	4f	90	88	2a	22	de	5e	db	0b	b8	14	ee	46
	f	c4	a7	3d	7e	19	73	5d	64	5f	97	17	44	13	ec	0c	cd

Bảng 6.1. Bảng thay thế Gray S-box

6.4 Một số tính chất của Gray S-box

6.4.1 Tính đồng nhất sai phân

Định nghĩa 6.1. Tính đồng nhất sai phân (differential uniformity)[69]

Cho G_1 và G_2 là nhóm Abel hữu hạn. Ánh xạ $f : G_1 \to G_2$ được gọi là đồng nhất sai phân mức δ nếu:

$$\forall \alpha \in G_1, \alpha \neq 0, \forall \beta \in G_2, \left|\{z \in G_1 | f(z+\alpha) - f(z) = \beta\}\right| \leq \delta \qquad (6.17)$$

δ được gọi là mức đồng nhất sai phân của f.

Giá trị δ càng nhỏ thì ánh xạ f càng an toàn đối với tấn công mật mã sai phân [6] và tấn công mật mã tuyến tính [62].

Trong [26], D. Feng và W. Wu đã chứng minh với $n \times m$ S-box, mức đồng nhất sai phân bị chặn dưới là $\delta_{\min} = 2^{n-m+1}$. S-box đạt được mức đồng nhất $\delta_{\min}$ được gọi là Almost Perfect Nonlinear (APN) [5]. Tuy nhiên, không tồn tại APN S-box có n bit đầu vào và n bit đầu ra với n chẵn [69]. Vì vậy, trong thuật toán AES sử dụng 8×8 S-box , mức đồng nhất sai phân tối thiểu (lý tưởng) là $\delta = 2^2 = 4$.

Ma trận phân bố sai phân của Gray S-box được xác định như sau:

$$E_{ij} = \left|\{x | S_{Gray}(x) \oplus S_{Gray}(i \oplus x) = j\}\right| \qquad (6.18)$$

Trong ma trận này, các phần tử nhận 1 trong 3 giá trị 0, 2, hoặc 4 (trừ phần tử E_{00}). Do đó, Gray S-box đạt được mức đồng nhất vi phân tối thiểu (lý tưởng) $\delta = 4$.

6.4.2 Strict Avalanche Criterion

Định nghĩa 6.2 Strict Avalanche Criterion (SAC) [98]:

Hàm $f(x) : \mathrm{GF}(p)^n \to \mathrm{GF}(p)$ thỏa tiêu chuẩn Strict Avalanche Criterion (SAC) khi và chỉ khi

$$prob(f(x+a) = f(x) + a) = \frac{1}{p} \ \forall a \in \mathrm{GF}(p)^n \text{ thỏa } wt(a)=1 \qquad (6.19)$$

Xét $m \times n$ S-box gồm m bit đầu vào và n bit đầu ra. Có thể xem S-box này gồm n hàm $f_0, f_1, \ldots, f_0, f_1, \ldots, f_{n-1}$, trong đó, hàm $f_j : \mathrm{GF}(2)^m \rightarrow \mathrm{GF}(2)$ xác định bit thứ j trong kết quả của S-box ($0 \leq j < n$).

Một trong những tiêu chí đánh giá độ an toàn của S-box là từng hàm f_j phải đạt hay "gần đạt" tiêu chuẩn SAC, tức là, nếu 1 bit đầu vào của S-box bị thay đổi thì mỗi bit đầu ra sẽ bị thay đổi với xác suất xấp xỉ 1/2.

Trong Bảng 6.2 và Bảng 6.3, phần tử tại dòng i cột j là số trường hợp giá trị của hàm f_j bị thay đổi khi bit đầu vào thứ i bị thay đổi đối với Gray S-box và S-box nguyên thủy trong thuật toán AES. Mặc dù tất cả các hàm f_j thì Gray S-box và S-box nguyên thủy đều không thỏa tiêu chí SAC nhưng số trường hợp kết quả của f_j bị thay đổi khi bit đầu vào thứ i bị thay đổi xấp xỉ 128. Như vậy, khi 1 bit đầu vào bị thay đổi, mỗi bit đầu ra sẽ thay đổi với xác suất xấp xỉ 1/2.

	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7
bit 0	132	132	116	144	116	124	116	128
bit 1	120	128	136	120	132	120	136	136
bit 2	136	120	120	128	140	136	136	112
bit 3	132	136	128	124	132	136	112	132
bit 4	120	132	124	124	116	112	132	132
bit 5	120	128	124	120	140	132	132	120
bit 6	120	136	120	136	136	132	120	132
bit 7	128	140	136	132	144	120	132	120

Bảng 6.2. Khảo sát sự thay đổi của các hàm nhị phân thành phần f_j khi bit đầu vào thứ i bị thay đổi đối với Gray S-box

	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7
bit 0	132	132	116	144	116	124	116	128
bit 1	120	124	144	128	124	116	128	136
bit 2	132	132	128	120	144	128	136	128
bit 3	136	136	120	116	128	136	128	140
bit 4	116	128	116	132	128	128	140	136
bit 5	116	132	132	120	120	140	136	136
bit 6	136	136	120	132	120	136	136	124
bit 7	132	144	132	136	124	136	124	132

Bảng 6.3. Khảo sát sự thay đổi của các hàm nhị phân thành phần f_j khi bit đầu vào thứ i bị thay đổi đối với S-box trong AES

6.5 So sánh giữa Gray S-box với các S-box cải tiến khác

Bảng 6.4 so sánh tính chất của S-box trong AES với S-box cải tiến được L. Cui đề nghị trong [15], S-box cải tiến do Y. Cao đề nghị trong [60] và Gray S-box mà chúng tôi đề nghị. Kết quả cho thấy cả 3 S-box cải tiến được đề nghị đề đảm bảo tính đồng nhất sai phân và tiêu chuẩn SAC tương đương với S-box trong AES. Trong bảng này, chúng tôi xét độ phức tạp đại số của từng S-box dựa trên số lượng đơn thức khác 0 trong biểu diễn đại số của từng S-box trên trường Galois của Rijndael.

So với S-box được Y.Cao đề xuất trong [60], Gray S-box có cùng số lượng (tối đa) các đơn thức khác 0 trong biểu diễn đại số. Tuy nhiên, trong S-box mà J. Liu đề xuất không tái sử dụng toàn bộ S-box của AES mà chỉ dùng từng ánh xạ thành phần trong S-box của AES. Gray S-box bổ sung bước tiền xử lý vào S-box của AES nên có thể tái sử dụng toàn bộ S-box của AES, đặc biệt là có thể kế thừa toàn bộ các thiết kế tối ưu khi cài đặt S-box của AES trên phần cứng.

S-box được J. Liu đề xuất trong [15] cũng tái sử dụng được toàn bộ S-box của AES, nhưng ánh xạ tuyến tính được Y. Cao chọn làm bước tiền xử lý cho S-box chưa giúp tạo ra S-box có biểu diễn đại số gồm đầy đủ 255 đơn thức khác 0. Đối với Gray S-box, sử dụng ánh xạ tuyến tính là biến đổi từ dạng nhị phân sang mã Gray nhị phân, chúng tôi đã tạo ra được S-box cải tiến có biểu diễn đại số gồm đầy đủ 255 đơn thức.

S-box	SAC	Mức đồng nhất sai phân	Số đơn thức trong biểu diễn đại số	Tái sử dụng cấu trúc S-box trong AES
S-box trong AES	~ 1/2	4	9 đơn thức	
S-box cải tiến [15]	~ 1/2	4	253 đơn thức	Toàn bộ
S-box cải tiến [60]	~ 1/2	4	255 đơn thức	Từng phần
Gray S-box	~ 1/2	4	255 đơn thức	Toàn bộ
Giá trị tối ưu	1/2	4	255 đơn thức	

Bảng 6.4. So sánh các tính chất của S-box trong AES với các S-box cải tiến

Để tăng tốc độ xử lý, việc cài đặt S-box được thực hiện bằng kỹ thuật bảng tra. Khi cải tiến S-box thành Gray S-box, nếu áp dụng bảng tra được xây dựng sẵn thì chi phí xử lý không thay đổi (so với S-box nguyên thủy). Đối với các thiết bị hạn chế về khả năng lưu trữ (ví dụ như smart card), nếu không lưu trữ sẵn bảng tra cho S-box,

thao tác chuyển đổi từ biểu diễn nhị phân thông thường sang biểu diễn mã Gray cần $m - 1$ phép toán XOR giữa 2 bit với m là số bit trong biểu diễn nhị phân của giá trị cần biến đổi. Cụ thể với S-box trên byte (8 bit), thao tác biến đổi này cần 7 phép XOR giữa 2 bit.

Giải pháp tích hợp bước tiền xử lý chuyển đổi dữ liệu từ dạng nhị phân thông thường sang mã Gray nhị phân trước khi thực hiện thao tác biến đổi S-box của AES là một trong những giải pháp khả thi và hiệu quả trong việc nâng cao độ an toàn của S-box trong AES. Mã Gray nhị phân có ưu điểm dễ cài đặt, hiệu quả trong việc xử lý, và được sử dụng khá phổ biến (mã sửa lỗi) trong các hệ thống kỹ thuật số.

6.6 Kết luận

Đối với hầu hết các bài toán đại số, việc thay đổi cách biểu diễn thường ảnh hưởng đến độ phức tạp của bài toán[92]. Trong [92], V. Rijmen đã nêu ra một ví dụ minh họa kinh điển về việc thay đổi cách biểu diễn một đường cong elliptic bất kỳ từ dạng $Ay^2 + Byx + Cy = Dx^3 + Ex^2 + Fx + G$ sang dạng $y^2 = x^3 + ax + b$ đơn giản hơn. Riêng đối với thuật toán Rijndael, việc thay đổi cách biểu diễn cũng đã được áp dụng trong việc đơn giản hóa hoặc tối ưu hóa cài đặt trên thiết bị phần cứng [77][93][100]. Ngoài ra, một số cách biểu diễn đơn giản hoặc có tính chất đặc biệt nhằm phục vụ việc phân tích mã đối với AES cũng được đề xuất [28][29]. Hiện tại chưa có công trình phân tích mã nào theo hướng thay đổi cách biểu diễn hoặc sử dụng biểu diễn tương đương đạt được kết quả phân tích thành công hay có ảnh hưởng quan trọng đối với việc phân tích AES. Tuy nhiên, khả năng áp dụng kỹ thuật thay đổi cách biểu diễn với hi vọng tìm ra cách tấn công đặc biệt đối với AES nói riêng và các hệ mã nói chung vẫn là hướng mở.

Đối với S-box trong AES, các nghiên cứu phân tích mã khai thác tính thưa của S-box [14][28][29] chủ yếu được xây dựng dựa trên cách biểu diễn chuẩn trên trường Galois của Rijndael GF(2)[x]/$\mu(x)$ với $\mu(x) = x^8 + x^4 + x^3 + x + 1$. Chính vì vậy, các S-box cải tiến được đề xuất với tiêu chí tăng cường độ phức tạp trong biểu diễn đại số của S-box trên trường GF(2)[x]/$\mu(x)$. Khi biểu diễn trên trường này, Gray S-

box cũng như các S-box cải tiến của J. Liu [15] và Y.Cao [60] đều nâng độ phức tạp trong biểu diễn đại số lên đáng kể so với S-box nguyên thủy trong AES (với 9 đơn thức trong biểu diễn đại số). Chính nhờ điều này giúp hạn chế khả năng áp dụng các kỹ thuật tấn công khai thác tính thưa của S-box trong AES vốn đang dựa vào cách biểu diễn chuẩn trên trường Galois của Rijndael.

Một số vấn đề mở:

- Việc đề xuất cải tiến S-box nhằm khắc phục tính thưa trong biểu diễn S-box trên trường GF(2)[x]/$\mu(x)$ có thể thúc đẩy các nghiên cứu phân tích mã theo hướng kết hợp việc thay đổi cách biểu diễn nhằm tạo ra biểu thức đơn giản và các kỹ thuật để phân tích mã với biểu thức đơn giản được tạo ra.
- Khả năng tìm ra không gian phù hợp và cách biểu diễn phù hợp để chuyển dạng biểu diễn của S-box trong AES nói riêng và cả thuật toán AES nói chung sang dạng biểu diễn đơn giản hơn hay có tính chất đặc biệt hiện vẫn là vấn đề mở. Ngoài ra, việc khai thác được các đặc tính của dạng biểu diễn mới này nhằm phân tích mã đối với AES cũng cần có những công trình nghiên cứu sâu hơn.

Kết luận

Các kết quả đạt được

Ý tưởng chủ đạo trong luận án là xây dựng thuật toán mã hóa khối được tham số hóa làm bước chuyển tiếp giữa kiến trúc thuật toán ở mức trừu tượng với các thuật toán mã hóa cụ thể. Mỗi thuật toán mã hóa khối được tham số hóa xác định một lớp các thuật toán mã hóa khối có cùng kiến trúc và chiến lược xây dựng các thành phần mã hóa. Giải thuật XAES được chúng tôi xây dựng nhằm cụ thể hóa ý tưởng về việc xây dựng thuật toán mã hóa được tham số hóa.

Do các thành phần mã hóa trong Rijndael đã được nghiên cứu kỹ trong những năm gần đây, các tính chất mật mã quan trọng của Rijndael đã được khảo sát chi tiết, chúng tôi quyết định chọn phương án tổng quát hóa các thành phần trong Rijndael để xây dựng các thành phần trong XAES.

XAES được tham số hóa với 2 nhóm tham số: tham số cấu trúc và tham số xử lý. Với hai tham số cấu trúc được đề nghị, gồm số lượng bit trong mỗi nhóm dữ liệu được xử lý (ký hiệu là m) và số lượng nhóm (m bit) trong mỗi từ (ký hiệu là Nw), XAES có khả năng mở rộng linh hoạt và không giới hạn về kích thước khối và kích thước khóa, đồng thời thích nghi với nhiều kiến trúc xử lý khác nhau. Tham số m cho phép XAES phù hợp với các hệ thống mà đơn vị dữ liệu cơ bản được xử lý không phải là byte, ví dụ như trong các thiết bị ubiquitous hoặc thiết bị cảm ứng. Tham số Nw cho phép XAES tận dụng khả năng xử lý của các bộ xử lý 64 (với Nw = 8 và m = 8).

Điểm đặc trưng của XAES là thuật toán không được đặc tả "cứng" thông qua các giá trị hằng số cụ thể trong mỗi thành phần mã hóa mà được đặc tả với các tham số xử lý và quy tắc xây dựng các thành phần mã hóa. Điều này cho phép dễ dàng tạo ra các biến thể của thuật toán với các bộ hằng số khác nhau cho mỗi biến đổi thuật toán.

Sử dụng hướng tiếp cận truyền thống với vết sai phân đơn và vết tuyến tính đơn, chúng tôi đã chứng minh tổng quát các công thức xác định chặn trên của tỷ lệ truyền

của vết sai phân đơn và chặn trên của độ tương quan của vết tuyến tính đơn lan truyền qua $T = 4r$ chu kỳ của XAES, từ đó kết luận về tính an toàn đối với phương pháp sai phân và phương pháp tuyến tính cho tất cả các thể hiện của XAES (với các giá trị cụ thể của tham số cấu trúc và tham số xử lý).

Với cách tiếp cận sử dụng tập vết sai phân và bao tuyến tính, chúng tôi đã xác định các công thức tổng quát của giá trị chặn trên của xác suất sai phân của tập vết sai phân và giá trị chặn trên của xác suất tuyến tính của bao tuyến tính lan truyền qua $r \geq 4$ chu kỳ của XAES.

Từ góc độ ứng dụng, chúng tôi đã khảo sát việc tạo ra các bộ giá trị cho tham số xử lý mà trọng tâm là đối với biến đổi MixColumns và SubBytes. Nhằm đáp ứng nhu cầu tạo ra các bộ hệ số có khả năng khuếch tán cao cho ánh xạ tuyến tính trong biến đổi MixColumns của XAES, chúng tôi đã đề xuất kỹ thuật kiểm tra sơ bộ và kiểm tra ngẫu nhiên. Việc thử nghiệm tính hiệu quả của các kỹ thuật này được chúng tôi tiến hành trong các trường hợp phổ biến là $m = 8$ (mỗi đơn vị dữ liệu được xử lý là byte) và Nw từ 4 đến 8 (phù hợp kiến trúc 32-bit và 64-bit). Kết quả thử nghiệm cho thấy kỹ thuật kiểm tra sơ bộ và kiểm tra ngẫu nhiên được đề nghị tương đối hiệu quả trong việc kiểm tra một bộ hệ số phát sinh ngẫu nhiên có phải là bộ hệ số mạnh (hoặc bộ hệ số mạnh ngưỡng T) hay không. Ngoài ra, chúng tôi đã xác định tất cả các bộ hệ số tối ưu trong trường hợp $m = 8$ trên trường Galois của Rijndael và giá trị Nw từ 2 đến 8.

Nhằm minh họa ưu điểm của việc sử dụng hai ánh xạ tuyến tính làm bước tiền xử lý và hậu xử lý cho S-box trong XAES để khắc phục tính thưa của S-box với kiến trúc được dùng trong AES, chúng tôi đã chọn ánh xạ tuyến tính là biến đổi từ dạng biểu diễn nhị phân thông thường sang mã Gray làm bước tiền xử lý cho S-box trong AES. Kết quả nhận được là S-box có biểu diễn đại số gồm đầy đủ 255 đơn thức có hệ số khác 0 trên trường Galois của Rijndael. Chính nhờ điều này giúp hạn chế khả năng áp dụng các kỹ thuật tấn công khai thác tính thưa của S-box trong AES vốn đang dựa vào cách biểu diễn chuẩn trên trường Galois của Rijndael.

Dựa trên các kết quả nghiên cứu về mã hóa và các thuật toán mã hóa khối được chúng tôi đề nghị, chúng tôi đã xây dựng và thử nghiệm một số ứng dụng bảo vệ thông tin. Một số ứng dụng trong multimedia được chúng tôi giới thiệu tóm tắt trong Phụ lục A .

Hướng phát triển

Các vấn đề mở đã được chúng tôi lần lượt trình bày trong phần cuối của mỗi chương. Dưới đây, chúng tôi tóm tắt lại các vấn đề chính trong hướng phát triển của luận án:

- Khảo sát khả năng xác định các bộ giá trị tham số của XAES có tính chất đặc biệt để tạo ra kỹ thuật phân tích mã đặc thù nhằm tấn công vào các thể hiện đặc biệt đó của XAES.
- Nghiên cứu và đề xuất các tiêu chí cho việc chọn các bộ giá trị tham số, đặc biệt là tham số xử lý để tối ưu hóa việc sử dụng XAES vào một hệ thống cụ thể hay kiến trúc xử lý cụ thể
- Ứng dụng các kết quả chứng minh công thức tổng quát xác định chặn trên của tỷ lệ truyền của vết sai phân và độ tương quan của vết tuyến tính vào việc xây dựng các hệ mã theo khối với tầng khuếch tán (có vai trò tương tự như MixColumns trong XAES) có hệ số branch number bất kỳ.
- Nghiên cứu việc chọn lựa một giá trị đặc trưng khác của S-box cũng như cách làm trội khác trong quá trình xây dựng công thức chặn trên của xác suất sai phân và xác suất tuyến tính qua các chu kỳ mã hóa để xác định các giá trị chặn trên chặt hơn của xác suất sai phân/xác suất tuyến tính.
- Chứng minh hình thức hiệu quả của các kỹ thuật kiểm tra sơ bộ và kiểm tra ngẫu nhiên trong trường hợp tổng quát; đề ra tiêu chí để đánh giá chính xác và hiệu quả một bộ hệ số mạnh cũng như kỹ thuật tổng quát để ước lượng chặn dưới của giá trị Branch Number của ánh xạ bất kỳ.
- Khả năng tìm ra không gian phù hợp và cách biểu diễn phù hợp để chuyển dạng biểu diễn của S-box trong AES nói riêng và cả thuật toán AES nói chung sang dạng biểu diễn đơn giản hơn hay có tính chất đặc biệt để có thể dùng trong phân tích mã.

Tài liệu tham khảo

Tiếng Anh:

[1] C.M. Adams (1997), "Constructing Symmetric Ciphers Using the CAST Design Procedure", *Designs, Codes, and Cryptography*, 12(3), tr. 283–316.

[2] R. Anderson, E. Biham, L. Knudsen (1998), "Serpent: A Proposal for the Advanced Encryption Standard", *First Advanced Encryption Standard (AES) Conference*.

[3] P.S.L.M. Baretto, V. Rijmen (2000), "The Anubis block cipher", *Submission to the NESSIE Project.*

[4] E. Barkan, E. Biham (2002), "In how many ways can you write Rijndael?", *ASIACRYPT '02*, LNCS, vol.2501, Springer, tr.160–175.

[5] T. Beth, C. Ding (1994), "On almost perfect nonlinear permutations", *EUROCRYPT '93*, LNCS, vol. 765, Springer, tr. 65-76.

[6] E. Biham, A. Shamir (1991), "Differential cryptanalysis of DES-like cryptosystems", *Journal of Cryptology*, 4 (1), tr. 3-72.

[7] E. Biham (1993), "New types of cryptanalytic attacks using related keys", *EUROCRYPT '93*, LNCS, vol. 765, Springer, tr. 398-409.

[8] J. Borst (2000), "The block cipher: GrandCru", *Submission to the NESSIE Project.*

[9] C. Burwick, D. Coppersmith, D. D'Avignon, R. Gennaro, S. Halevi, C. Jutla, S. M. Matyas Jr., L. O'Connor, M. Peyravian, D. Safford, N. Zunic (1999), "MARS – a candidate cipher for AES", *AES Algorithm Submission*.

[10] D. Canright (2005), "A Very Compact S-Box for AES", *Cryptographic Hardware and Embedded Systems – CHES 2005*, tr. 441—455.

[11] A. Canteaut, M. Videau (2002), "Degree of composition of highly nonlinear functions and applications to higher order differential cryptanalysis", *EUROCRYPT '02*, LNCS, Vol. 2332, tr.518 – 533.

[12] K. Chun, S. Kim, S. Lee, S.H. Sung, S. Yoon (2003), "Differential and linear cryptanalysis for 2-round SPNs", *Information Processing Letters*, Vol. 87, tr. 277–282.

[13] C. Cid, S. Murphy, M.J. Robshaw (2005), "Small scale variants of the AES", *Fast Software Encryption FSE '05*, LNCS, Vol. 3557, Springer, tr.145–162.

[14] N. Courtois, J. Pieprzyk (2002), "Cryptanalysis of Block Ciphers with Overdefined Systems of Equations", *ASIACRYPT '02*, LNCS, Vol. 2501, tr. 267–287.

[15] L. Cui, Y. Cao (2007), "A new S-box structure named Affine-Power-Affine", International Journal of Innovative Computing, Information and Control, Vol. 3, No. 3, tr. 751-759.

[16] J. Daemen, V. Rijmen (1999), "AES Proposal: Rijndael", *AES Algorithm Submission*.

[17] J. Daemen, V. Rijmen (2002), *The Design of Rijndael: AES - The Advanced Encryption Standard*, Springer-Verlag, ISBN 3-540-42580-2.

[18] J. Daemen, L.R. Knudsen, V. Rijmen (1997), "The block cipher Square", *Fast Software Encryption FSE '97*, LNCS, Vol. 1267, Springer, tr. 149-165.

[19] J. Daemen (1995), *Cipher and hash function design strategies based on linear and differential cryptanalysis*, Doctoral Dissertation, K.U.Leuven.

[20] J. Daemen, V. Rijmen (2001), "The wide trail design strategy", *IMA Int. Conf.*, tr.222–238.

[21] J. Daemen, V. Rijmen (2002), "Security of a wide trail design", *INDOCRYPT '02*, tr.1–11.

[22] P. J. Davis (1994), *Circulant Matrices, 2nd ed,* Chelsea.

[23] A. M. Eskicioglu, J. Town, E. J. Delp (2003), "Security of digital entertainment content from creation to comsumption", *Signal Processing: Image Communication, Special Issue on Image Security*, Vol 18, tr. 237-262.

[24] H. Feistel (1973), "Cryptography and computer privacy", *Scientific American*, Vol. 228, No. 5, tr. 15-23.

[25] H. Feistel, W.A. Notz, J.L. Smith (1975), "Some cryptographic techniques for machine to machine data communications", *Proceedings of the IEEE*, Vol. 63, No. 11, tr. 1545-1554.

[26] D. Feng, W. Wu (2000), *Design and Analysis of Block Ciphers*, Tsinghua University Press.

[27] N. Ferguson, R. Schroeppel, D. Whiting (2004), "The inverse s-box, non-linear polynomial relations and cryptanalysis of block ciphers", *AES-2004*, LNCS, Vol. 3373, tr.170–188.

[28] N. Ferguson, R. Schroeppel, D. Whiting (2001), "A simple algebraic representation of Rijndael", Selected Areas in Cryptography SAC'01, LNCS, Vol. 2259, tr.103–111.

[29] J. Fuller, W. Millan (2003), "On linear redundancy in S-boxes", *Fast Software Encryption '03*, LNCS 2887, Springer-Verlag, 2003, tr. 74–86.

[30] National Institute of Standards and Technology (2001), *FIPS 197, Announcing the Advanced Encryption Standard (AES)*.

[31] National Institute of Standards and Technology (1977), *FIPS 46, Data Encryption Standard (DES)*.

[32] National Institute of Standards and Technology (2000), *FIPS Announcing the Digital Signature Standard (DSS)*.

[33] M. Gardner (1986), "*The binary Gray code*", *Knotted Doughnuts and Other Mathematical Entertainments*, New York.

[34] F. Gray (1947), *Pulse code communication*. U.S. Patent 2,632,058.

[35] S. Guth (2003), "Rights Expression Languages", *Digital Rights Management – Technological, Economic, Legal and Political Aspects*, LNCS 2770, SpringerLink, tr. 101-112

[36] J. Herre (2003), "Content Based Identification (Fingerprinting)", *Digital Rights Management – Technological, Economic, Legal and Political Aspects*, SpringerLink, pp 93-100.

[37] R. Hobson, S. Wakelin (2005), "An Area-Efficient High-Speed AES S-Box Method", *Fifth International Workshop on System-on-Chip for Real-Time Applications (IWSOC'05)*, tr. 376-379.

[38] S. Hong, S. Lee, J. Lim, J. Sung, D. Cheon, I. Cho (2000), "Provable security against differential and linear cryptanalysis for the SPN structure", *Fast Soft Encryption FSE'00*, LNCS, Vol. 1978, Springer, Berlin, tr. 273–283.

[39] *IEEE - Copy Protection for DVD Video*, http://www.dvd-copy.com/reference/IEEE-doc-copyproc.pdf

[40] IPR Systems Pty Ltd (2002), *Open Digital Rights Language (ODRL) version 1.1*, http://odrl.net/1.1/ODRL-11.pdf

[41] T. Jakobsen, L.R. Knudsen (1997), "The interpolation attack on block ciphers", *Fast Software Encryption FSE'97*, LNCS, Vol. 1267, Springer, tr. 28-40.

[42] K. U. Järvinen, M. Tommiska, J, Skyttä (2003), "A fully pipelined memoryless 17.8 Gbps AES-128 encryptor", FPGA 2003, tr. 207-215.

[43] J. B. Kam, G. I. Davida (1979), "Structured design of substitution-permutation encryption networks", *IEEE Transactions on Computers*, vol.C-28, no.10, tr.747–753.

[44] J. Kang, S. Hong, S. Lee, O. Yi, C. Park, J. Lim (2001), "Practical and provable security against differential and linear cryptanalysis for substitution-permutation networks", *ETRI Journal*, 23(4): tr. 158–167.

[45] L. Keliher, H. Meijer, S. Tavares (1999), "Modeling Linear Characteristics of Substitution-Permutation Networks", *Selected Areas in Cryptography 1999*, LNCS, Vol. 1758, Springer, tr. 78-91.

[46] L. Keliher, H. Meijer, S. Tavares (2001), "New method for upper bounding the maximum average linear hull probability for SPNs", *EUROCRYPT 2001*, LNCS, Vol. 2045, Springer, tr. 420–436

[47] L. Keliher, H. Meijer, S. Tavares (2001), "Improving the upper bound on the maximum average linear hull probability for Rijndael", S*elected Areas in Cryptography - SAC 2001*, LNCS, Vol. 2259, Springer, tr. 112–128.

[48] L. Keliher (2003), *Linear Cryptanalysis of Substitution-Permutation Networks*, PhD. Thesis, Queen's University, Kingston, Ontario, Canada.

[49] L. Keliher (2004*),* "Refined Analysis of Bounds Related to Linear and Differential Cryptanalysis for the AES", *Advanced Encryption Standard – AES 2004*, LNCS, Vol. 3373, tr. 42-57.

[50] L. Keliher, J. Sui (2007)*, "*Exact maximum expected differential and linear cryptanalysis for two-round Advanced Encryption Standard*", IET Information Security*, Vol. 1, No. 2, tr. 53-57.

[51] J. Kelsey, B. Schneier, D. Wagner (1996), "Key-schedule cryptanalysis of IDEA, GDES, GOST, SAFER, and Triple-DES", *Advances in Cryptology*, tr. 237-252.

[52] L.R. Knudsen (1995), "Truncated and higher order differentials," *Fast Software Encryption FSE'95*, LNCS, Vol.1008, Springer-Verlag , tr.196–211.

[53] L. R. Knudsen, V. Rijmen, R. L. Rivest, M. J. B. Robshaw (1998), "On the Design and Security of RC2", *Fast Software Encryption 1998*, tr. 206–221.

[54] D.Kwon et. al. (2003), "New block cipher: ARIA", *CISC 2003*, LNCS vol. 2971, Springer-Verlag, tr. 432–445.

[55] D.Kwon et. al. (2000), "Camellia: A 128-bit block cipher suitable for multiple platforms - design and analysis," *Selected Areas in Cryptography – SAC 2000*, Springer, tr.39–56.

[56] X. Lai, J. L. Massey, S. Murphy (1991), "Markov Ciphers and Differential Cryptanalysis", *EUROCRYPT '91*, LNCS vol. 547, Springer-Verlag, tr. 17-38.

[57] L. Li , Y. Kobayashi (2006), "A Block Recursive Algorithm for the Linear Complementarity Problem with an M-matrix", *International Journal of Innovative Computing, Information and Control*, vol. 2, No. 6, tr. 1327-1335.

[58] R. Lidl, H. Niederreiter (1994), *Introduction to Finite Fields and their Applications*, Cambridge University Press; 2nd edition.

[59] C. H. Lim (1999), "A revised version of CRYPTON: CRYPTON V1.0", *Fast Software Encryption - FSE'99*, LNCS vol. 1636, tr. 31-45.

[60] J. Liu, B. Wei, X. Cheng, X. Wang (2005), "An AES S-box to increase complexity and cryptographic analysis", *19th International Conference on Advanced Information Networking and Applications - AINA'05*, tr. 724- 728.

[61] Liu Zhenglin, Zeng Yonghong, Zou Xuecheng, Han Yu, Chen Yicheng (2007), "A High-Security and Low-Power AES S-Box Full-Custom Design for Wireless Sensor Network", *International Conference on Wireless Communications, Networking and Mobile Computing – WiCom 2007*, tr. 2499 – 2502

[62] M. Matsui (1994), "Linear cryptanalysis method for DES cipher", *EUROCRYPT '93*, LNCS vol. 765, Springer-Verlag, tr. 386-397.

[63] W. Meier, O. Staffelbach (1990), "Nonlinearity Criteria for Cryptographic Functions", *EUROCRYPT '89*, LNCS vol. 434, Springer-Verlag, tr. 549-562.

[64] A. J. Menezes, P.C. van Oorschot, S.A. Vanstone (1997), *Handbook of Applied Cryptography*, CRC Press.

[65] N. Mentens, L. Batina, B. Preneel, I. Verbauwhede (2005), "A systematic evaluation of compact hardware implementations for the Rijndael S-box", *CT-RSA*, LNCS 3376, tr. 323-333

[66] J. Monnerat, S. Vaudenay (2004), "On some weak extensions of AES and BES," *Information and Communications Security*, LNCS vol.3269, Springer, tr.414–426.

[67] S. Morioka, A. Satoh (2002), "An Optimized S-Box Circuit Architecture for Low Power AES Design", *Proceedings of Workshop on Cryptographic Hardware and Embedded Systems*, LNCS 2523, tr. 172 - 186 .

[68] S. Murphy, M.J. Robshaw (2002), "Essential algebraic structure within the AES", *Crypto'02*, LNCS, vol.2442, Springer Berlin / Heidelberg, , tr.1–16.

[69] K. Nyberg (1993), “Differentially uniform mappings for cryptography”, *EUROCRYPT '93*, LNCS vol. 765, Springer-Verlag, tr 55-64.

[70] K. Nyberg (1995), “Linear approximation of block ciphers”, *EUROCRYPT '94*, LNCS vol. 950, Springer-Verlag, tr. 439-444.

[71] S. Park, S. H. Sung, S. Chee, E-J.Yoon, J. Lim (2002), “On the security of Rijndael-like structures against differential and linear cryptanalysis”, *ASIACRYPT 2002*, LNCS vol. 2501, Springer-Verlag, tr. 176–191.

[72] S. Park, S. H. Sung, S. Lee, J. Lim (2003), “Improving the Upper Bound on the Maximum Differential and the Maximum Linear Hull Probability for SPN Structures and AES”, *Fast Software Encryption FSE 2003*, LNCS vol. 2887, Springer-Verlag, tr. 247-260.

[73] C. Pawed , K. Gaj (2003), “Very compact FPGA implementation of the AES algorithm”, *Proceedings of 5th International Workshop on Cryptographic Hardware and Embedded Systems (CHES)*, LNCS 2779, tr. 319-333

[74] J. Rosenthal (2003), “A Polynomial Description of the Rijndael Advanced Encryption Standard”, *Journal of Algebra and its Applications* 2(2), tr. 223-236.

[75] R. Rivest, A. Shamir, L. Adleman (1978). "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems", *Communications of the ACM,* **21** (2), tr.120-126.

[76] R. L.Rivest, M.J.B. Robshaw, R. Sidney, Y. L. Yin (1998), “The RC6 Block Cipher: A simple fast secure AES proposal”, *AES Algorithm Submission*.

[77] A. Satoh, S. Morioka, K. Takano, S. Munetoh (2001), “A compact Rijndael hardware architecture with S-box optimization”, *ASIACRYPT 2001*, LNCS 2248, tr. 239–254

[78] C.E. Shannon (1949), “Communication theory of secrecy systems”, *Bell System Technical Journal*, vol. 28, no. 4, tr. 656-715.

[79] N. Shankaran, X. D. Koutsoukos, D. C. Schmidt, Y. Xue, C. Lu (2008), "Hierarchical control of multiple resources in distributed real-time and embedded systems", *Real-Time Syst* Vol. 39, tr. 237–282.

[80] B. Schneier, J. Kelsey, D. Whiting, D. Wagner, C. Hall, N. Ferguson (1998), "Twofish: A 128-Bit Block Cipher", *AES Algorithm Submission*.

[81] B. Schneier, "Description of a New Variable-Length Key, 64-bit Block Cipher (Blowfish)", *Fast Software Encryption 1993*, tr. 191-204

[82] A. Shimizu, S. Miyaguchi (1988), "Fast data encipherment algorithm FEAL", *Eurocrypt '87*, Springer-Verlag, tr. 267–280.

[83] R. E. Smith (1997), *Internet Cryptography*, Addison-Wesley.

[84] G. Spenger (2003), "Authentication, Identification Techniques, and Secure Containers - Baseline Technologies", *Digital Rights Management - Technological, Economic, Legal and Political Aspects*, LNCS 2770, SpringerLink, tr. 62-88.

[85] W. Stallings (2003), *Cryptography and Network Security: Principles and Practice, Third Edition,* Prentice Hall.

[86] D. R. Stison (1995), *Cryptography – Theory and Practice*, CRC Press.

[87] H. Raddum (2004), "More dual Rijndaels", *AES-2004*, LNCS vol.3373, tr.142–147.

[88] B. Rosenblatt, G. Dykstra (2003), *Integrating Content Management with Digital Rights Management*, White Paper, http://www.xrml.org/reference/CM-DRMwhitepaper.pdf

[89] Rightscom, *The MPEG-21 Rights Expression Language. White paper*, http://www.xrml.org/reference/MPEG21_REL_whitepaper_Rightscom.pdf

[90] V. Rijmen, J. Daemen, B. Preneel, A. Bosselaers, E. D. Win (1996). "The Cipher SHARK", *Fast Software Encryption FSE '96*, LNCS vol.1039, Springer-Verlag, tr.99–111.

[91] V. Rijmen, P.S.L.M. Barreto (2000), "The Khazad legacy-level block cipher", *Submission to the NESSIE Project* .

[92] V. Rijmen, E. Oswald (2004), “Representations and Rijndael descriptions”, *AES-2004*, LNCS vol.3373, tr.148–158.

[93] A. Rudra, P. K. Dubey, C. S. Jutla, V. Kumar, J. R. Rao, P. Rohatgi (2001), “Efficient Rijndael encryption implementation with composite field arithmetic”, *CHES 2001*, LNCS 2162, Springer-Verlag, tr. 171–184.

[94] N. Rump (2003), “Definition, Aspects, and Overview”, *Digital Rights Management – Technological, Economic, Legal and Political Aspects*, LNCS 2770, SpringerLink, pp 3-15

[95] Toshiba Corporation (2000), *Specification of Hierocrypt-3*.

[96] B. Y. Yang, J. M. Chen (2004), “Theoretical Analysis of XL over Small Fields”, *ACISP 2004*, Lecture Notes in Computer Science vol. 3108, tr.277-288.

[97] S. Vidalis, M. Pilgermann, E. Morakis, A. Blyth (2005), “Security in Heterogeneous Large Scale Environments”. *International Journal of Innovative Computing*, Vol 1, No4, tr. 715-725.

[98] A. F. Webster, E. Tavares (1985), “On the design of S-boxes”, Crypto '85, LNCS 219, tr. 523-534.

[99] D. J. Wheeler, R. M. Needham (1994), "TEA, a tiny encryption algorithm". *Fast Software Encryption 1994*, *LNCS 1008*, tr. 363–366.

[100] Shee-You Wu, Shih-Chuan Lu, Chi Sung Laih (2004), “Design of AES Based on Dual Cipher and Composite Field,” *Topics in Cryptology — CT-RSA 2004*, LNCS 2964, Springer-Verlag, 2004, tr. 25–38

Các công trình đã công bố

Một số bài báo khoa học công bố tại các hội nghị và tạp chí khoa học quốc tế:

1. Tran Minh Triet, Bui Doan Khanh, Duong Anh Duc (2008), "Gray S-box for Advanced Encryption Standard", *Proceedings of 2008 IEEE International Conference on Computational Intelligence and Security (CIS'08)*, Suzhou-SIP, China, Dec. 13-17, 2008, ISBN 978-0-7695-3508-1, tr. 253-258.
2. Minh-Triet Tran, Thanh-Trung Nguyen, Isao Echizen (2008), "Pool-Based APROB Channel to Provide Resistance against Global Active Adversary under Probabilistic Real-Time Condition", *2008 IEEE/IFIP International Symposium on Trust, Security and Privacy for Pervasive Applications (TSP'08)*, *Proceedings of IEEE/IFIP International Conference on Embedded and Ubiquitous Computing (EUC'08),* Vol. 2, Shanghai, China, Dec. 17-20, 2008, ISBN 978-0-7695-3492-3, tr. 257-263.
3. Tran Minh Triet, Duong Anh Duc, Bui Doan Khanh (2007), "Safe Coeffcient Sets of the MixColumns Transformation in Rijndael Block Cipher", *2007 International Conference on Research, Innovation & Vision for the Future (RIVF'07)*, Mar. 05-09, 2007, Ha Noi, Vietnam, ISBN 2-912590-4-0, tr. 125-130.
4. Tran Minh Triet, Duong Anh Duc (2005), "Applying the AES and Its Extended Versions in a General Framework for Hiding Information in Digital Images", *Proceedings of 2005 International Conference on Computational Intelligence and Security (CIS'05)*, Part II, LNAI 3802, Springer-Verlag Berlin Heidelberg, ISBN 978-3-540-30819, tr. 605 – 610.
5. Tran Minh Triet, Duong Anh Duc (2004), "Applying the Robust Psychoacoustic Audio Watermarking Technique in Internet Digital Traditional Music Museum in Vietnam", *Proceedings of 38th IEEE International Carnahan Conference on Security Technology (ICCST2004)*, Albuquerque, New Mexico, USA, Oct 11-14, 2004, ISBN 0-7803-8506-3, tr. 285-288
6. Duong Anh Duc, Tran Minh Triet, Luong Han Co(2002), "The extended Rijndael-like Block Ciphers", *Proceedings of International Conference on Information Technology: Coding and Computing – 2002*, The Orleans, Las Vegas, Nevada, USA, Apr 8-10, 2002, ISBN 0-7695-1506-1, tr. 183-188.

7. Duong Anh Duc, Tran Minh Triet, Luong Han Co (2002), "The Advanced Encryption Standard And Its Application in the examination security in Vietnam", *International Conference on Information Technology: Coding and Computing – 2002*, The Orleans, Las Vegas, Nevada, USA, Apr 8-10, 2002, ISBN 0-7695-1506-1, tr. 171-176.
8. Duong Anh Duc, Tran Minh Triet, Luong Han Co (2001), "The extended version of the Rijndael Block Cipher", *Journal of Institute of Mathematics and Computer Sciences*, India, Vol. 12, No. 2, tr. 201-218.

Một số bài báo khoa học công bố tại các hội nghị và tạp chí khoa học trong nước:

1. Trần Minh Triết, Trần Ngọc Bảo, Đặng Hải Vân (2008), "Về tính dễ mở rộng của các thuật toán mã hóa khối phổ biến" , *Hội thảo Công nghệ Thông tin & Truyền thông lần thứ nhất* (ICTFIT08), Tp. Hồ Chí Minh, 14 tháng 11 năm 2008, Tuyển tập Công trình nghiên cứu Công nghệ thông tin và Truyền thông 2008, NXB Khoa học và Kỹ thuật, tr. 16-24.
2. Trần Minh Triết, Dương Anh Đức, Đậu Ngọc Hà Dương, Châu Thành Đức (2008), "Hệ thống bảo mật nội dung và kiểm soát truy cập triển khai với thiết bị nhúng tích hợp vào dịch vụ multimedia", *kỷ yếu Hội thảo quốc gia "Một số vấn đề chọn lọc của CNTT và truyền thông"*, Đại Lải, 14-15 tháng 09 năm 2007, NXB Khoa học tự nhiên và Công nghệ, tr. 131-141
3. Trần Minh Triết, Dương Anh Đức, Hồ Ngọc Lâm, Thân Võ Chí Nhân (2005), "Tổng quan về watermarking trên Audio", *kỷ yếu Hội nghị quốc gia về Công nghệ Thông tin và Truyền thông*, Thái Nguyên, 23-31 tháng 08 năm 2003, NXB Khoa học Kỹ thuật, tr. 442-446.
4. Dương Anh Đức, Nguyễn Thanh Sơn, Trần Minh Triết (2004), "Bảo mật dữ liệu với kỹ thuật AES-DCT watermarking", *tạp chí Khoa học Công nghệ ĐHQG*, số 4-5, tập 7, tr. 77-82.
5. Dương Anh Đức, Trần Minh Triết, Lương Hán Cơ (2001), "The 256/384/512-bit version of the Rijndael Block Cipher", *Tạp chí Tin học và Điều khiển*, Việt Nam, tập 17, số 4, tháng 12 năm 2001, tr. 45-56.
6. Dương Anh Đức, Trần Minh Triết, Lương Hán Cơ (2001), "Ứng dụng chuẩn mã hóa AES và các phiên bản mở rộng vào Hệ thống Thư điện tử an toàn tại Việt Nam", *Hội nghị khoa học kỷ niệm 25 năm Viện Công Nghệ Thông Tin*, Hà Nội, Việt Nam, 24-25 tháng 12 năm 2001, tr. 46-53.

Phụ lục A
Một số quy trình ứng dụng

✍ *Chúng tôi đã ứng dụng các kết quả nghiên cứu về mã hóa và các thuật toán mã hóa được chúng tôi đề nghị vào một số mô hình và quy trình bảo vệ thông tin.*
Trong phụ lục này, chúng tôi trình bày hai ứng dụng được chúng tôi đề nghị vào lĩnh vực multimedia:

- *Quy trình nhúng thông tin mật trong dữ liệu multimedia.*
- *Hệ thống dịch vụ multimedia tích hợp bảo mật nội dung và kiểm soát truy cập*

A.1 Quy trình nhúng thông tin mật vào dữ liệu multimedia

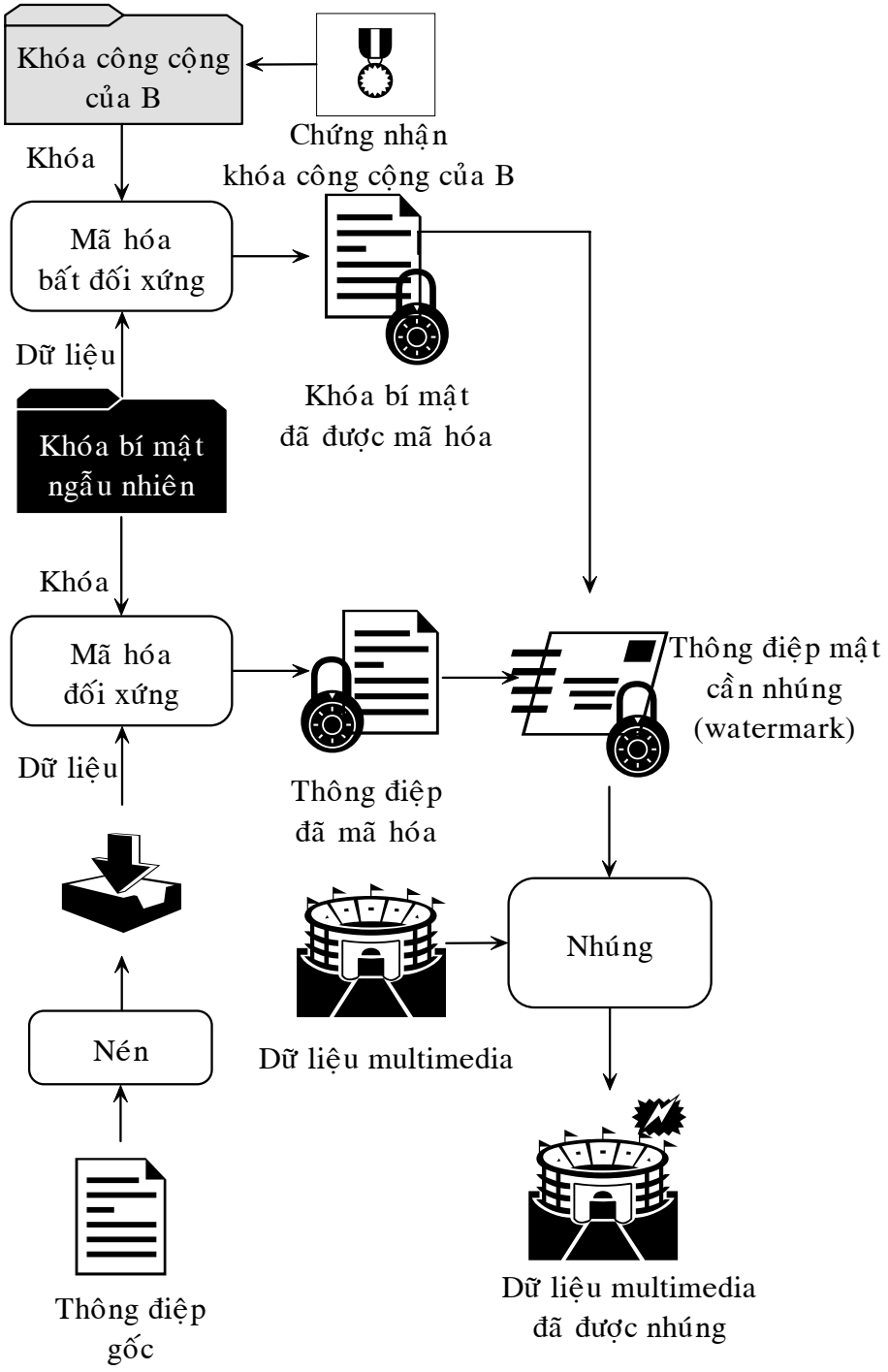
A.1.1 Giới thiệu

Ngày nay, với sự phát triển của Internet và các giao dịch điện tử, khối lượng *thông tin số* được đưa vào truy cập công khai ngày càng tăng lên. Vấn đề phát sinh là làm cách nào có thể *bảo vệ được bản quyền sở hữu trí tuệ* đối với những thông tin, dữ liệu số này một cách hiệu quả trong khi vẫn *đảm bảo khả năng truy cập tự do* của mọi người. Khái niệm *watermarking* được đề cập đến như một giải pháp nhằm giải quyết bài toán bảo vệ quyền sở hữu trí tuệ đối với các thông tin số có giá trị [94].

Một cách tổng quát, *khái niệm watermarking được hiểu là kỹ thuật nhúng thông tin (gọi là vết watermark) vào trong thông tin khác (gọi là thông tin mang)* [94].

A.1.2 Quy trình nhúng thông tin mật vào dữ liệu multimedia

Trong quy trình nhúng tin mật vào dữ liệu multimedia, đầu tiên, dữ liệu cần nhúng sẽ được nén lại để giảm dung lượng thông tin cần nhúng và hạn chế các mẫu dữ liệu đặc biệt trong thông tin ban đầu. Sau đó, dữ liệu đã được nén sẽ được mã hóa bằng một phương pháp mã hóa đối xứng có độ an toàn cao , ví dụ như: Rijndael [16], MARS [9], Serpent [2], RC6 [76], TwoFish [80]…, hoặc các thuật toán cụ thể XAES (với m = 8 và Nw từ 5 đến 8).

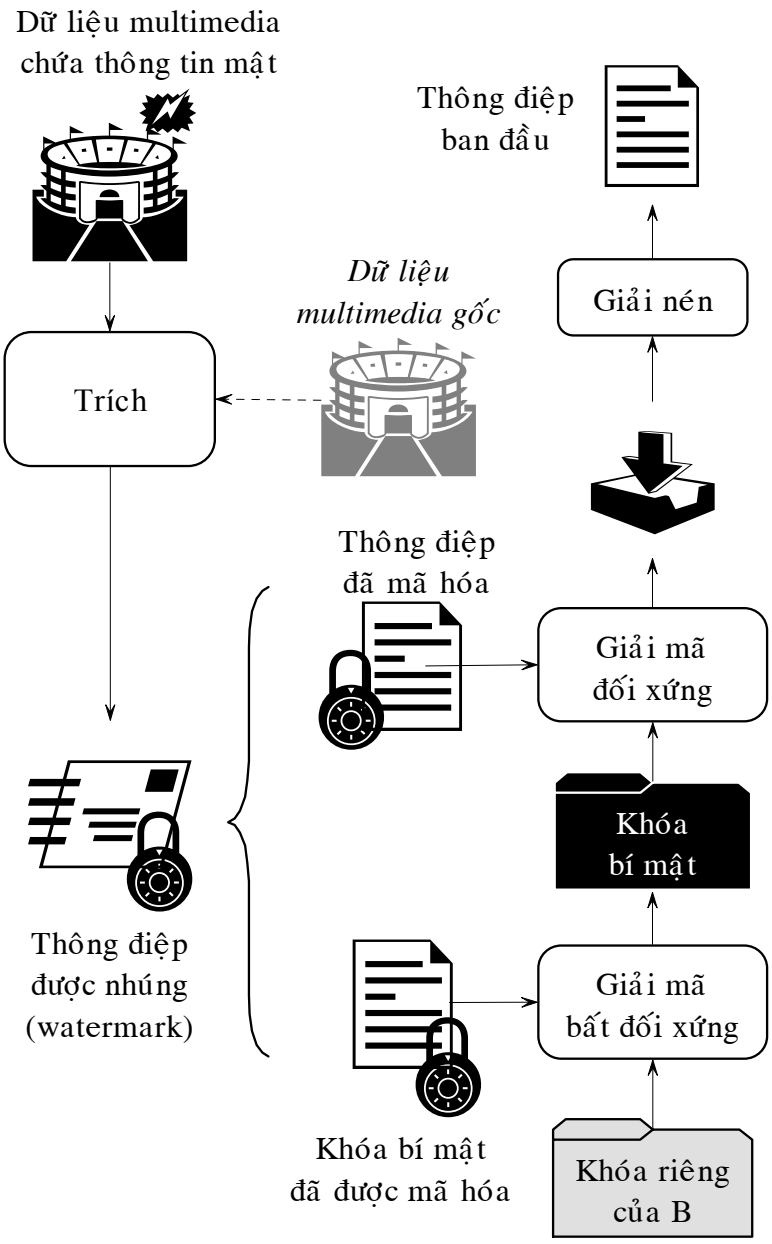


Hình A.1. Quy trình nhúng tin mật vào dữ liệu multimedia

Khóa bí mật được sử dụng trong quá trình mã hóa thông tin mật sẽ được mã hóa bằng phương pháp bất đối xứng sử dụng khóa công cộng của người nhận.

Vết watermark, bao gồm dữ liệu mật sau khi mã hóa cùng với khóa đã được mã hóa, được nhúng vào dữ liệu multimedia và truyền cho người nhận.

A.1.3 Quy trình trích thông tin mật từ dữ liệu multimedia



Hình A.2. Quy trình trích và giải mã thông tin mật trong dữ liệu multimedia

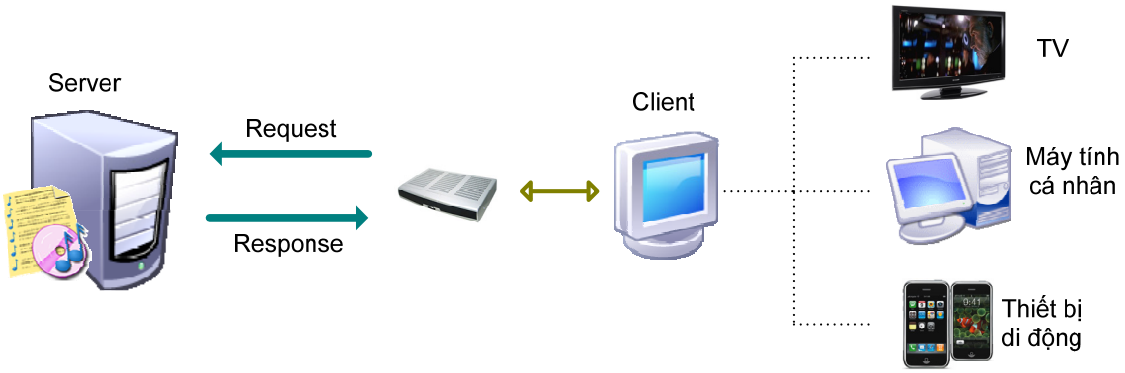
Khi nhận được dữ liệu multimedia có chức thông điệp bí mật, người nhận sẽ trích vết watermark (bao gồm thông điệp mật đã mã hóa và khóa bí mật đã được mã hóa) ra khỏi dữ liệu multimedia. Quá trình trích này có thể cần sử dụng hay không cần sử dụng dữ liệu multimedia gốc (trước khi nhúng), tùy thuộc vào phương pháp watermarking được sử dụng là phương pháp không mù [94] hay phương pháp mù [94].

Vết watermark được phân tích ra thành thông điệp mật đã mã hóa và khóa bí mật đã được mã hóa. Sử dụng khóa riêng của mình, người nhận giải mã được khóa bí mật đã dùng để mã hóa thông điệp, từ đó, giải mã và nhận được thông điệp ban đầu.

A.2 Hệ thống bảo mật nội dung và kiểm soát truy cập triển khai với thiết bị nhúng tích hợp vào dịch vụ multimedia

A.2.1 Giới thiệu

Hiện nay, nhu cầu dịch vụ multimedia trực tuyến băng thông rộng ngày càng phát triển mạnh. Đáp ứng xu thế mới này, rất nhiều dịch vụ multimedia đã ra đời như truyền hình cáp, xem phim theo yêu cầu (Video on Demand – VoD), IP TV… Tuy nhiên, vấn đề đặt ra là làm sao có thể vừa cung cấp rộng rãi các dịch vụ multimedia, vừa bảo vệ quyền sở hữu trí tuệ đối với các tài nguyên multimedia. Vì vậy, việc xây dựng các hệ thống bảo vệ bản quyền hay quản lý quyền số (digital rights management – DRM [94]) để tích hợp vào các dịch vụ multimedia là điều cần thiết.



Hình A.3. Mô hình dịch vụ Multimedia trực tuyến tích hợp hệ thống nhúng

Bên cạnh các yêu cầu về bảo mật nội dung, xác định người sử dụng và kiểm soát việc truy cập, hệ thống quản lý quyền số cần đơn giản trong việc triển khai, tiện dụng và thân thiện đối với khách hàng - người sử dụng dịch vụ multimedia. Do các thiết bị nhúng có lợi thế là dễ mang chuyển và triển khai, tiện dụng, hoạt động ổn định, giá thành rẻ (khi sản xuất đại trà hàng loạt) và khả năng hạn chế tấn công cao (đặc biệt là đối với cách tấn công sửa đổi mã lệnh thực thi) nên xu hướng hiện nay là các nhà cung cấp dịch vụ cố gắng tích hợp các thành phần xử lý lên một hệ thống nhúng như

các set top box, các bộ giải mã cho truyền hình cáp… nhằm giúp người dùng dễ dàng trong việc lắp đặt và sử dụng, đồng thời đảm bảo cho nhà cung cấp một giải pháp bảo vệ và kiểm soát dữ liệu hiệu quả [88][79].

Đối với các mô hình bảo vệ bản quyền tài nguyên multimedia, tùy vào từng quy trình và hệ thống cụ thể, thiết bị nhúng có thể được xây dựng để cung cấp một phần hay tất cả các tính năng sau [23]:

- Mã hóa và giải mã tín hiệu,
- Mã hóa (để bảo mật nội dung) và giải mã nội dung,
- Kiểm soát quyền truy cập tài nguyên,
- Khôi phục tín hiệu đã bị làm nhiễu (de-scramble)
- Nhúng các thông tin bảo mật vào dữ liệu multimedia (ví dụ như thông tin bản quyền nhà cung cấp, hoặc nhúng fingerprint của khách hàng để kiểm soát tài nguyên multimedia đã được cung cấp cho ai…)

Trong phần này, sau khi giới thiệu tổng quan về hệ thống quản lý quyền số - DRM, chúng tôi trình bày một mô hình cụ thể giúp bảo mật nội dung dữ liệu multimedia trong truyền thông và kiểm soát truy cập có thể tích hợp vào dịch vụ multimedia. Trong mô hình này, phân hệ xử lý tại phía khách hàng được đề xuất tích hợp trên thiết bị nhúng để dễ dàng triển khai trên thực tế, tăng tính ổn định và an toàn của hệ thống nhờ hạn chế những lỗi và lỗ hổng so với việc sử dụng giải pháp phần mềm. Ngoài ra, phân hệ này còn có khả năng đóng vai trò làm máy chủ trung gian để lưu lại dữ liệu đã giải mã để truyền cho các máy tính khác cùng truy cập để giảm tải cho máy chủ trung tâm. Chúng tôi đã xây dựng và thử nghiệm hệ thống này, trong đó, phân hệ tại phía khách hàng được cài đặt trên board S3CEB2410 với chip ARM920T. Một số kết quả thử nghiệm được trình bày trong phần 4, cuối cùng là kết luận và hướng phát triển.

A.2.2 Tổng quan về Hệ thống quản lý quyền số - DRM

Nội dung của phần này giới thiệu tóm tắt một số khái niệm chính về Hệ thống quản lý quyền số - DRM, làm cơ sở cho việc trình bày hệ thống cụ thể được chúng tôi đề xuất để bảo mật nội dung và kiểm soát truy cập trong phần 3 của bài viết này.

Quản lý quyền số (Digital Rights Management - DRM) là một thuật ngữ thông dụng được hình thành vào khoảng những năm 1990, khi các nhà cung cấp nội dung và công nghệ phim ảnh bắt đầu đối mặt với hệ quả của mạng máy tính tràn ngập những phân phối trái phép các tài liệu có bản quyền [88].

Tùy ứng dụng cụ thể, một số yêu cầu chính mà một hệ thống quản lý quyền số cụ thể cần giải quyết có thể gồm:

1. *Gói an toàn*: bảo vệ cho nội dung không thể truy cập đối với những người sử dụng chưa được cập quyền truy cập đến các nội dung dữ liệu này[84]. Các gói dữ liệu multimedia thường được bảo mật nội dung bằng cách mã hóa khóa đối xứng (ví dụ như Rijndael [16], MARS [9], Serpent [2], RC6 [76], TwoFish [80]…)
2. *Mô tả quyền*: được sử dụng để mô tả giấy phép cho người sử dụng truy cập đến nội dung dữ liệu được chứa trong gói an toàn. Những mô tả quyền được định dạng hoặc theo cách đơn giản chỉ là mô tả quyền bằng cờ hiệu hoặc theo cách phức tạp bằng ngôn ngữ mô tả quyền [35], ví dụ như ODRL [40] hay MPEG – REL [89]. Ngôn ngữ mô tả quyền thường kèm theo bộ từ điển về dữ liệu quyền [35].
3. *Hệ thống định danh và mô tả nội dung:* được dùng để tạo định danh duy nhất cho mỗi tài nguyên multimedia. Có thể kể đến một vài hệ thống định danh nổi tiếng như hệ thống ISRC (International Standard Recording Codes - www.ifpi.org/isrc/) dành cho việc ghi âm, hệ thống ISAN (International Standard Audio-visual Numbers - www.isan.org/) dành cho các dữ liệu âm thanh và hình ảnh, hệ thống DOI (Digital Object Identifiers - www.doi.org/) dành cho nội dung bất kỳ.
4. *Định danh người sử dụng và việc kiểm tra định danh*: Hệ thống định danh người dùng là một yếu tố quyết định của hệ thống DRM, một yêu cầu để hệ thống có khả năng giới hạn truy cập nội dung đối với những người dùng có quyền truy cập. *Kiểm tra định danh* là vấn đề liên quan mật thiết với vấn đề định danh người sử dụng. Nhiệm vụ của nó là xác minh người sử dụng hay tổ chức muốn

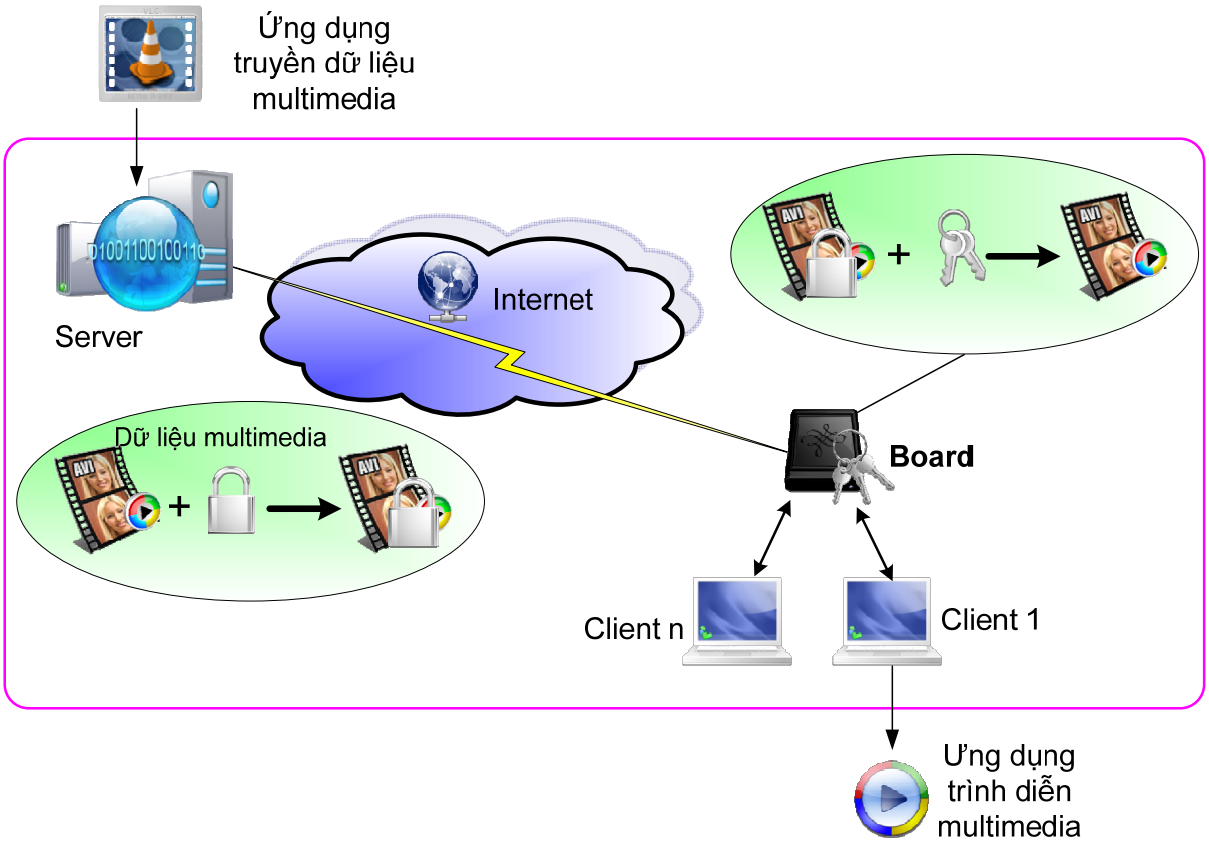
truy cập vào nội dung. Chức năng này sẽ yêu cầu có những thuật toán mã hoá và có thể cần đến một tổ chức thứ ba dùng để phân phát các "passport" hay giấy chứng nhận điện tử.

5. *Dấu vân tay*: Đây công nghệ liên quan mật thiết với việc định danh nội dung. Công nghệ nổi bật nhất trong miền này là watermarking và fingerprinting. Trong hầu hết mọi trường hợp, watermarking và fingerprinting được sử dụng để có thể chứng minh sự vi phạm bản quyền xảy ra. Ví dụ, nó được đưa vào chuẩn CSS (Content Srambling System - [39]) cho DVD. Khi có sự sao chép nội dung hay đổi sang định dạng khác, các dấu ấn này vẫn còn khả năng tồn tại để nhận ra được nội dung này là nội dung nào và sở hữu của ai.
6. *Hệ thống tính tiền*. Tùy vào hệ thống cụ thế, chức năng tính tiền sẽ kết nối với ngân hang hay các hệ thống thanh toán khác.

A.2.3 Mô hình dịch vụ Multimedia tích hợp hệ thống bảo mật nội dung và kiểm soát truy cập sử dụng thiết bị nhúng

Mô hình tổng quát:

Hình A.4 thể hiện mô hình tổng quát được chúng tôi đề nghị. Trong mô hình này, ***Server*** nhận dữ liệu multimedia từ một máy chủ khác lưu trữ multimedia, hoặc từ một ***phần mềm truyền dữ liệu multimedia*** (ví dụ như VLC hoặc QuickStream). Các gói dữ liệu multimedia sẽ được mã hóa và truyền đến ***Board*** tại ***Client***. ***Board*** sẽ giải mã và truyền dữ liệu multimedia đến cho 1 hay nhiều máy tính ***Client*** đang kết nối vào ***Board***.



Hình A.4. Mô hình tổng quát hệ thống

Các bước xử lý chính trong mô hình gồm có:

- ***Client* đăng nhập**: ***Client*** sẽ đăng nhập hệ thống với tên và mật khẩu. Sau khi đang nhập hợp lệ, ***Client*** có thể yêu cầu ***Server*** để nhận dữ liệu mong muốn.
- ***Client* yêu cầu dữ liệu**: ***Client*** gửi yêu cầu dữ liệu cần nhận trên ***Server***. Dữ liệu này là dữ liệu đang được truyền tại ***Server*** và chỉ có một dữ liệu được truyền tại một thời điểm (truyền trực tiếp).
- *Server* gửi dữ liệu về *Client*.

Quy trình đăng nhập hệ thống :

Hình A.5 thể hiện quy trình đăng nhập hệ thống.

Tại phía *Server*:

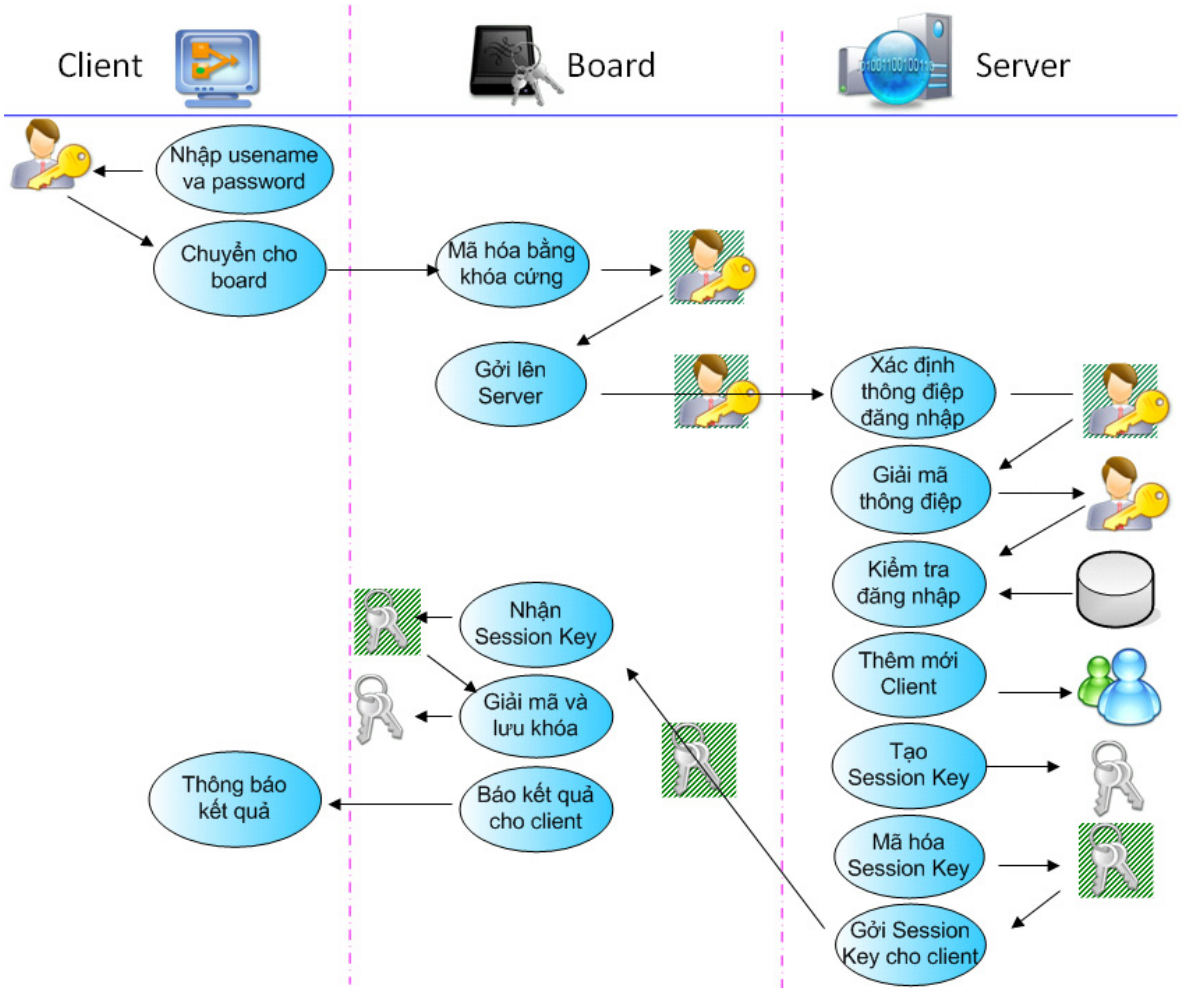
- Bước 1: ***Server*** xác định thông điệp đăng nhập tài khoản. Sau đó sẽ trích lấy các thông tin đăng nhập gồm: tên tài khoản, mật khẩu, số hiệu phiên.

- Bước 2: ***Server*** giải mã các thông tin đăng nhập bằng khóa cứng của ***Client***. Đối với số hiệu phiên, ***Server*** kiểm tra xem có phải đây là một phiên làm việc mới của ***Client*** hay không. Nếu đúng là phiên làm việc mới thì ***Server*** sẽ thực hiện việc thêm một đối tượng quản lý ***Client*** vào danh sách đã có, với trường hợp tài khoản đăng nhập là hợp lệ. Ngược lại ***Server*** chỉ trả thông điệp kết quả về.
- Bước 3: ***Server*** thực hiện việc kiểm tra tài khoản bằng yêu cầu đến cơ sở dữ liệu thông tin người dùng. Nếu tài khoản đăng nhập là hợp lệ thì cơ sở dữ liệu sẽ trả về kết quả đúng, ngược lại trả về kết quả sai.
- Bước 4: Nếu tài khoản đăng nhập là hợp lệ, Server sẽ tạo mới một đối tượng quản lý ***Client*** và thêm vào danh sách các đối tượng quản lý đã có. Việc quản lý các đối tượng này sẽ giúp cho ***Server*** biết được những ***Client*** hiện đang kết nối đến và chỉ thực hiện truyền dữ liệu đối với những ***Client*** nằm trong danh sách này.
- Bước 5: Sau khi thêm mới đối tượng ***Client***, ***Server*** sẽ phát sinh một khóa ngẫu nhiên gọi là khóa phiên. Khóa này được tạo ra bằng cách băm tên tài khoản, mật khẩu, số hiệu phiên và nhãn thời gian làm việc. Khóa này sẽ được xem là khóa chỉ định duy nhất cho mỗi ***Board*** mà ***Server*** làm việc tại một thời điểm.

  Soft_Key = Hash(Username, Password, SessionID, Time).
- Bước 6: Gửi thông điệp trả lời đến ***Client***. Thông điệp thành công nếu tài khoản là hợp lệ, và thông điệp thất bại nếu người dùng đã nhập sai tài khoản. Đồng thời ***Server*** cũng trả về số hiệu phiên mới cho ***Client***.

Phía *Client*:

- Bước 5: ***Board*** sẽ nhận dữ liệu khóa phiên từ ***Server*** và giải mã trực tiếp bằng khóa cứng của mình. Sau đó ***Board*** lưu khóa phiên lại để dùng cho các thao tác mã hóa về sau.
- Bước 6: Nếu đăng nhập thành công ***Board*** sẽ gửi kết quả là thành công về ***Client***. Nếu đăng nhập thất bại ***Board*** sẽ gửi kết quả là thất bại về ***Client***. ***Client*** sẽ thể hiện kết quả ra màn hình.



Hình A.5. Quy trình đăng nhập hệ thống

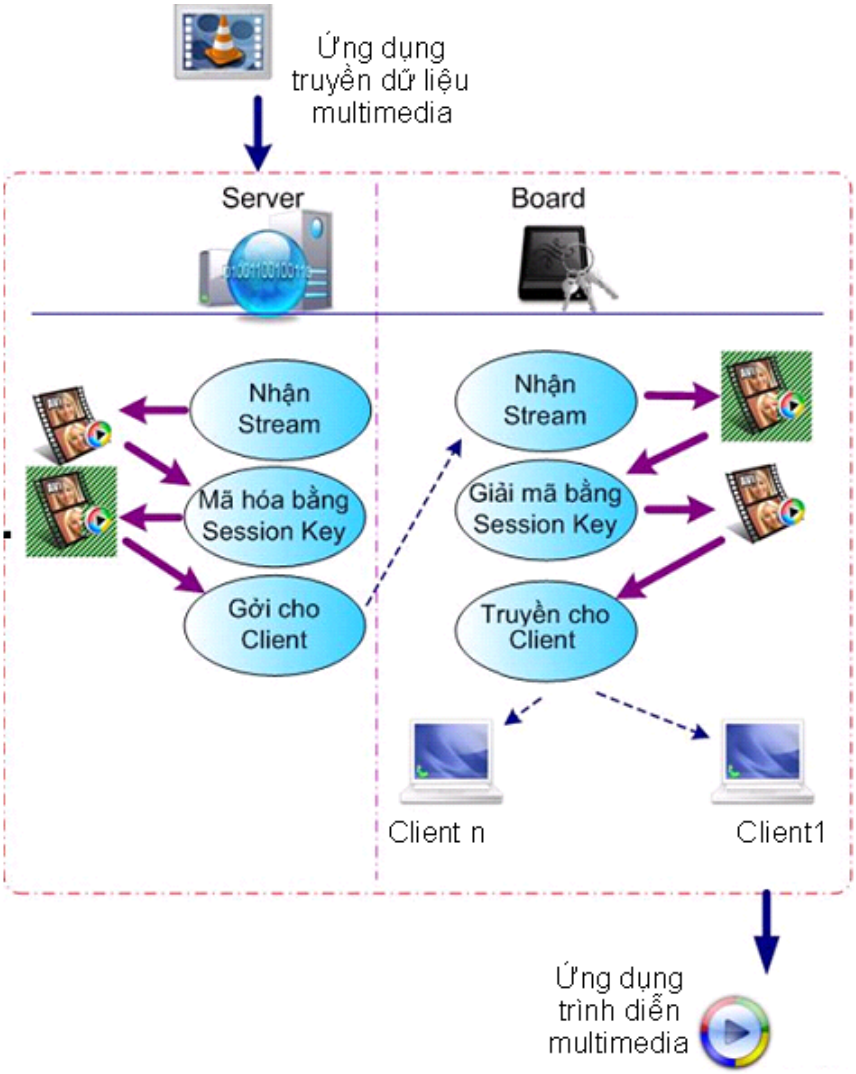
Quy trình truyền dữ liệu giữa Server và Client: Quy trình truyền dữ liệu giữa ***Server*** và ***Client*** được thể hiện trong Hình A.6.

Phía *Server*:

- Bước 1: ***Server*** nhận dữ liệu truyền từ các ***ứng dụng truyền dữ liệu multimedia*** ứng với các giao thức truyền được hỗ trợ (RTP và UDP). Việc nhận dữ liệu này được thực hiện tại một cổng nhất định. Cổng này phải được thiết lập trước khi tiến hành nhận dữ liệu truyền và phải đồng bộ với cổng phát của ***ứng dụng truyền dữ liệu multimedia***.
- Bước 2: ***Server*** thực hiện việc mã hóa bằng các thuật toán có độ an toàn cao, ví dụ như: Rijndael [16], MARS [9], Serpent [2], RC6 [76], TwoFish [80]…, hoặc

các thuật toán cụ thể XAES (với m = 8 và Nw từ 5 đến 8). Khóa mã hóa chính là khóa phiên ứng với mỗi ***Client*** mà ***Server*** quản lý.

- Bước 3: ***Server*** truyền dữ liệu đến tất cả các ***Client*** đang được quản lý.



Hình A.6. Quy trình truyền dữ liệu

Phía *Client*:

- Bước 1: Sau khi xác định được thông điệp từ ***Server***, ***Board*** sẽ trích lấy ra dữ liệu đã mã hóa để tiến hành giải mã.
- Bước 2: ***Board*** sẽ tiến hành giải mã bằng khóa phiên với các thuật toán hỗ trợ. Sau đó ***Board*** sẽ lưu dữ liệu lại.

- Bước 3: ***Board*** truyền dữ liệu đến các máy ***Client*** đã kết nối đến ***Board***. Các máy ***Client*** này nhận dữ liệu đã giải mã từ ***Board*** và truyền đến ứng dụng trình diễn multimedia tại ***Client***.

A.2.4 Nhận xét, đánh giá về mô hình

Mô hình được đề xuất có độ linh hoạt cao, có khả năng tương thích và tích hợp dễ dàng đối với các ***ứng dụng dùng để truyền dữ liệu*** tại ***Server*** và các ***ứng dụng trình diễn dữ liệu multimedia*** tại ***Client***. Các ***ứng dụng truyền dữ liệu*** có thể được tích hợp ngay trên máy server và tùy chọn theo nhu cầu của nhà sản xuất. Các ***ứng dụng trình diễn multimedia*** cũng linh hoạt đối với người sử dụng trên các hệ thống khác nhau cũng như trên các hệ điều hành khác nhau.

Toàn bộ dữ liệu đều được mã hóa ở ***Server*** và giải mã ở ***Client*** với các thuật toán có độ an toàn cao (như Rijndael [16], MARS [9], Serpent [2], RC6 [76], TwoFish [80] …), đảm bảo bí mật nội dung các gói tin multimedia trên đường truyền.

Một đặc điểm quan trọng của hệ thống là tính phi chuẩn trong giao thức: dữ liệu được truyền trên đường mạng không theo chuẩn quy ước. Các ***ứng dụng truyền dữ liệu*** trên ***Server*** sẽ chỉ truyền tín hiệu đến máy ***Server*** (localhost), mà không truyền ra ngoài mạng. Đồng thời ngăn chặn khả năng xâm nhập của các ứng dụng khác đến cổng truyền của ứng dụng truyền dữ liệu bằng Firewall. Chính ứng dụng tại máy ***Server*** sau khi nhận dữ liệu từ ***ứng dụng truyền dữ liệu*** sẽ tiến hành truyền ra ngoài qua một cổng khác. Vì dữ liệu này cũng đã được mã hóa nên khả năng bị phát hiện là khá thấp [23].

Việc sử dụng board ở phía client có các ưu điểm sau:

- **Tương thích với các máy cấu hình thấp:** với các máy ***Client*** sử dụng thiết bị nhúng như một bộ giải mã thì không cần có cấu hình cao. Module giải mã được thực thi trên thiết bị nhúng, và do đó máy ***Client*** chỉ có nhiệm vụ chuyển nhận dữ liệu đến thiết bị nhúng và các máy con khác trong mạng LAN.
- **Khả năng chống tấn công cao:** vì phân hệ giải mã được thực hiện trên thiết bị nhúng nên khó có thể tấn công vào sự thực thi các mã nhị phân này.

- **Chi phí thấp:** nếu các thiết bị nhúng được sản xuất hàng loạt thì giá thành đối với mỗi thiết bị sẽ rất thấp. Do đó chi phí cho toàn hệ thống sẽ có thể chấp nhận đối với người dùng.
- **Dễ sử dụng:** chỉ cần cài đặt thiết bị nhúng vào hệ thống mạng hiện thời, các máy con khác trong mạng đều có thể xem được nội dung multimedia yêu cầu.
- **Kiểm soát dữ liệu:** thiết bị có thể tích hợp dữ liệu thông tin để chứng thực người dùng sau này (fingerprinting).

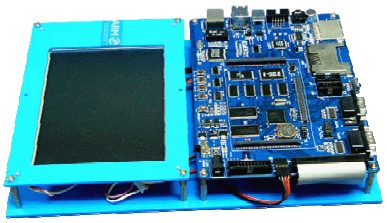
A.2.5 Triển khai thử nghiệm

Bảng A.1. thể hiện tốc độ xử lý mã hóa và giải mã dữ liệu trên board thử nghiệm. Số liệu thử nghiệm cho thấy việc giải mã thông tin trên thiết bị nhúng đủ nhanh để phục vụ việc trình diễn multimedia thời gian thực.

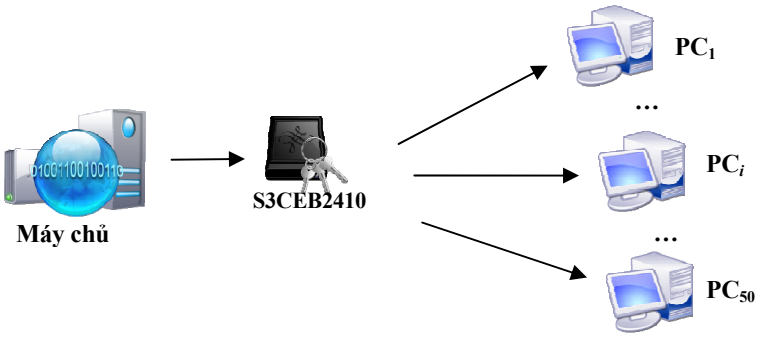
Thuật toán mã hóa	Mã hóa	Giải mã
RC6	4.89 Mbit/giây	4.92 Mbit/giây
MARS	3.25 Mbit/giây	3.33 Mbit/giây
Serpent	4.57 Mbit/giây	4.46 Mbit/giây
Rijndael	5.79 Mbit/giây	5.79 Mbit/giây
Twofish	1.12 Mbit/giây	1.12 Mbit/giây
XAES (m = 8, Nw = 5)	4.21 Mbit/giây	4.20 Mbit/giây
XAES (m=8, Nw = 6)	3.47 Mbit/giây	3.47 Mbit/giây
XAES (m=8, Nw = 7)	2.96 Mbit/giây	2.95 Mbit/giây
XAES (m=8, Nw = 8)	2.48 Mbit/giây	2.53 Mbit/giây

Bảng A.1. Tốc độ xử lý mã hóa và giải mã dữ liệu trên board S3CEB2410

Hệ thống được triển khai thử nghiệm với 50 máy tính (client) cùng kết nối và sử dụng chung board S3CEB2410 (với chip ARM920T 200 MHz, hệ điều hành Embedded Linux Mizi). Server multimedia sử dụng VLC streaming server (trên máy tính 2.4 GHz dùng hệ điều hành Windows Server 2003)



Hình A.7. Board S3CEB2410 (với chip ARM920T 200 MHz, hệ điều hành Embedded Linux Mizi)



Hình A.8. Mô hình thử nghiệm

Trong mô hình thử nghiệm, ***board*** nhận dữ liệu từ ***server*** (thông qua mạng Internet), giải mã và sau đó đóng vai trò là media server chung để phát đến cho các máy tính trong LAN, giúp tăng hiệu quả hệ thống so với việc từng máy tính trong mạng truy cập và nhận dữ liệu từ ***server*** (nhất là khi triển khai qua mạng Internet). Thử nghiệm thực tế cho thấy tốc độ xử lý tại ***board*** (bao gồm nhận dữ liệu từ ***server***, giải mã, phát lại cho các ***client***) dưới mức 40 miligiây cho 1 frame, điều này đảm bảo cho tính chất thời gian thực và chất lượng của hệ thống. Vấn đề duy nhất cần quan tâm là băng thông để ***board*** nhận dữ liệu từ ***server*** bên ngoài (qua mạng Internet).

A.2.6 Kết luận

Chúng tôi đã đề xuất và xây dựng thử nghiệm một mô hình cụ thể để kiểm soát truy cập và bảo mật nội dung có khả năng tích hợp vào các hệ thống dịch vụ multimedia. Hệ thống cung cấp khả năng bảo mật thông tin cao: toàn bộ các gói tin được truyền từ Server đều được mã hóa với các giải thuật mã hóa có độ an toàn cao như Rijndael, MARS, TwoFish, RC6, Serpent, hoặc các thuật toán cụ thể XAES (với $m = 8$ và Nw từ 5 đến 8). Ngoài ra, board xử lý còn được xây dựng bổ sung tính năng lưu lại dữ liệu multimedia để làm server cục bộ và truyền dữ liệu này cho nhiều Client đang cùng truy cập vào board, giúp giảm lưu lượng yêu cần trực tiếp đến máy chủ trung tâm. Kết quả thử nghiệm trên thực tế với board S3CEB2410 sử dụng chip ARM920T cho thấy hệ thống đáp ứng nhu cầu xử lý thời gian thực và có khả năng được áp dụng trên thực tế phục vụ hệ thống đào tạo từ xa, truyền hình internet…

Hiện tại, chúng tôi đang hoàn thiện hệ thống, xây dựng khả năng xử lý nhiều kênh multimedia với cùng 1 board xử lý, tích hợp khả năng nhúng thông tin về người sử dụng (fingerprinting [36]) vào dữ liệu để có thể xác định nguồn gốc dữ liệu bị sao chép và phát hành lại (sau khi đã được giải mã tại client).

Phụ lục B

Các bộ hệ số tối ưu cho biến đổi MixColumns của thuật toán XAES với m = 8 và Nw = 4, 5, …, 8

Dưới đây là các bộ hệ số tối ưu cho biến đổi MixColumns trong thuật toán XAES trong trường hợp m = 8 và Nw = 4, 5, …, 8. Với mỗi trường hợp, chúng tôi chọn trình bày các bộ hệ số độc lập[1]. Với mỗi bộ hệ số được trình bày, chúng ta có thể tạo ra Nw – 1 bộ hệ số tối ưu khác bằng cách dịch chuyển xoay vòng các hệ số đi k vị trí (với k=1, 2, …, Nw – 1). Khi đó, bộ hệ số tương ứng được sử dụng trong biến đổi **InvMixColumns** được xác định bằng cách dịch chuyển xoay vòng (theo chiều ngược lại) k vị trí bộ hệ số của biến đổi **InvMixColumns** ban đầu.

Các bộ hệ số tối ưu (độc lập) với m=8, Nw = 2

Dưới đây là 1 bộ hệ số tối ưu (độc lập) với m=8, Nw = 2.

1 bộ hệ số tối ưu còn lại được tạo ra từ bộ hệ số tối ưu này bằng cách dịch chuyển xoay vòng các hệ số đi k = 1vị trí.

$c(x) = \{02\}\ x^{\{01\}} + \{01\}\ x^{\{00\}}$

$c^{-1}(x) = \{a4\}\ x^{\{01\}} + \{52\}\ x^{\{00\}}$

Các bộ hệ số tối ưu (độc lập) với m=8, Nw = 3

Dưới đây là 2 bộ hệ số tối ưu (độc lập) với m=8, Nw = 3.

4 bộ hệ số tối ưu còn lại được tạo ra từ 2 bộ hệ số tối ưu này bằng cách dịch chuyển xoay vòng các hệ số đi k vị trí (với k=1, 2).

$c(x) = \{02\}\ x^{\{02\}} + \{01\}\ x^{\{01\}} + \{01\}\ x^{\{00\}}$

$c^{-1}(x) = \{7b\}\ x^{\{02\}} + \{8d\}\ x^{\{01\}} + \{7b\}\ x^{\{00\}}$

$c(x) = \{02\}\ x^{\{02\}} + \{02\}\ x^{\{01\}} + \{01\}\ x^{\{00\}}$

$c^{-1}(x) = \{f7\}\ x^{\{02\}} + \{f7\}\ x^{\{01\}} + \{01\}\ x^{\{00\}}$

---

[1] Hai bộ hệ số được xem là độc lập nếu không thể tạo ra bộ hệ số này bằng cách xoay vòng các hệ số của bộ hệ số còn lại.

Các bộ hệ số tối ưu (độc lập) với m=8, Nw = 4

Dưới đây là 6 bộ hệ số tối ưu (độc lập) với m=8, Nw = 4.

18 bộ hệ số tối ưu còn lại được tạo ra từ 9 bộ hệ số tối ưu này bằng cách dịch chuyển xoay vòng các hệ số đi k vị trí (với k=1, 2, …, 3).

$c(x) = \{03\}\, x^{\{03\}} + \{02\}\, x^{\{02\}} + \{01\}\, x^{\{01\}} + \{01\}\, x^{\{00\}}$

$c^{-1}(x) = \{0b\}\, x^{\{03\}} + \{0e\}\, x^{\{02\}} + \{09\}\, x^{\{01\}} + \{0d\}\, x^{\{00\}}$

$c(x) = \{02\}\, x^{\{03\}} + \{03\}\, x^{\{02\}} + \{01\}\, x^{\{01\}} + \{01\}\, x^{\{00\}}$

$c^{-1}(x) = \{0d\}\, x^{\{03\}} + \{09\}\, x^{\{02\}} + \{0e\}\, x^{\{01\}} + \{0b\}\, x^{\{00\}}$

$c(x) = \{03\}\, x^{\{03\}} + \{02\}\, x^{\{02\}} + \{02\}\, x^{\{01\}} + \{01\}\, x^{\{00\}}$

$c^{-1}(x) = \{32\}\, x^{\{03\}} + \{11\}\, x^{\{02\}} + \{f9\}\, x^{\{01\}} + \{57\}\, x^{\{00\}}$

$c(x) = \{03\}\, x^{\{03\}} + \{03\}\, x^{\{02\}} + \{02\}\, x^{\{01\}} + \{01\}\, x^{\{00\}}$

$c^{-1}(x) = \{42\}\, x^{\{03\}} + \{85\}\, x^{\{02\}} + \{10\}\, x^{\{01\}} + \{21\}\, x^{\{00\}}$

$c(x) = \{02\}\, x^{\{03\}} + \{02\}\, x^{\{02\}} + \{03\}\, x^{\{01\}} + \{01\}\, x^{\{00\}}$

$c^{-1}(x) = \{f9\}\, x^{\{03\}} + \{11\}\, x^{\{02\}} + \{32\}\, x^{\{01\}} + \{57\}\, x^{\{00\}}$

$c(x) = \{02\}\, x^{\{03\}} + \{03\}\, x^{\{02\}} + \{03\}\, x^{\{01\}} + \{01\}\, x^{\{00\}}$

$c^{-1}(x) = \{10\}\, x^{\{03\}} + \{85\}\, x^{\{02\}} + \{42\}\, x^{\{01\}} + \{21\}\, x^{\{00\}}$

Các bộ hệ số tối ưu (độc lập) với m=8, Nw = 5

Dưới đây là 6 bộ hệ số tối ưu (độc lập) với m=8, Nw = 5.

24 bộ hệ số tối ưu còn lại được tạo ra từ 6 bộ hệ số tối ưu này bằng cách dịch chuyển xoay vòng các hệ số đi k vị trí (với k=1, 2, …, 4).

$c(x) = \{02\}\, x^{\{04\}} + \{03\}\, x^{\{03\}} + \{02\}\, x^{\{02\}} + \{01\}\, x^{\{01\}} + \{01\}\, x^{\{00\}}$

$c^{-1}(x) = \{27\}\, x^{\{04\}} + \{4f\}\, x^{\{03\}} + \{f6\}\, x^{\{02\}} + \{4f\}\, x^{\{01\}} + \{27\}\, x^{\{00\}}$

$c(x) = \{03\}\, x^{\{04\}} + \{02\}\, x^{\{03\}} + \{03\}\, x^{\{02\}} + \{01\}\, x^{\{01\}} + \{01\}\, x^{\{00\}}$

$c^{-1}(x) = \{5c\}\, x^{\{04\}} + \{e5\}\, x^{\{03\}} + \{8d\}\, x^{\{02\}} + \{e5\}\, x^{\{01\}} + \{5c\}\, x^{\{00\}}$

$c(x) = \{03\}\, x^{\{04\}} + \{03\}\, x^{\{03\}} + \{01\}\, x^{\{02\}} + \{02\}\, x^{\{01\}} + \{01\}\, x^{\{00\}}$

$c^{-1}(x) = \{8d\}\, x^{\{04\}} + \{5c\}\, x^{\{03\}} + \{e5\}\, x^{\{02\}} + \{e5\}\, x^{\{01\}} + \{5c\}\, x^{\{00\}}$

$c(x) = \{03\}\ x^{\{04\}} + \{01\}\ x^{\{03\}} + \{02\}\ x^{\{02\}} + \{02\}\ x^{\{01\}} + \{01\}\ x^{\{00\}}$
$c^{-1}(x) = \{4f\}\ x^{\{04\}} + \{4f\}\ x^{\{03\}} + \{27\}\ x^{\{02\}} + \{f6\}\ x^{\{01\}} + \{27\}\ x^{\{00\}}$

$c(x) = \{02\}\ x^{\{04\}} + \{03\}\ x^{\{03\}} + \{03\}\ x^{\{02\}} + \{02\}\ x^{\{01\}} + \{01\}\ x^{\{00\}}$
$c^{-1}(x) = \{b8\}\ x^{\{04\}} + \{69\}\ x^{\{03\}} + \{69\}\ x^{\{02\}} + \{b8\}\ x^{\{01\}} + \{01\}\ x^{\{00\}}$

$c(x) = \{03\}\ x^{\{04\}} + \{02\}\ x^{\{03\}} + \{02\}\ x^{\{02\}} + \{03\}\ x^{\{01\}} + \{01\}\ x^{\{00\}}$
$c^{-1}(x) = \{69\}\ x^{\{04\}} + \{b8\}\ x^{\{03\}} + \{b8\}\ x^{\{02\}} + \{69\}\ x^{\{01\}} + \{01\}\ x^{\{00\}}$

Các bộ hệ số tối ưu (độc lập) với m=8, Nw = 6

Dưới đây là 2 bộ hệ số tối ưu (độc lập) với m=8, Nw = 6.

10 bộ hệ số tối ưu còn lại được tạo ra từ 2 bộ hệ số tối ưu này bằng cách dịch chuyển xoay vòng các hệ số đi k vị trí (với k=1, 2, …, 5).

$c(x) = \{04\}\ x^{\{05\}} + \{04\}\ x^{\{04\}} + \{03\}\ x^{\{03\}} +$
$\{01\}\ x^{\{02\}} + \{02\}\ x^{\{01\}} + \{01\}\ x^{\{00\}}$
$c^{-1}(x) = \{c2\}\ x^{\{05\}} + \{30\}\ x^{\{04\}} + \{0a\}\ x^{\{03\}} +$
$\{7e\}\ x^{\{02\}} + \{cd\}\ x^{\{01\}} + \{4a\}\ x^{\{00\}}$

$c(x) = \{03\}\ x^{\{05\}} + \{04\}\ x^{\{04\}} + \{04\}\ x^{\{03\}} +$
$\{01\}\ x^{\{02\}} + \{02\}\ x^{\{01\}} + \{01\}\ x^{\{00\}}$
$c^{-1}(x) = \{c2\}\ x^{\{05\}} + \{4a\}\ x^{\{04\}} + \{cd\}\ x^{\{03\}} +$
$\{7e\}\ x^{\{02\}} + \{0a\}\ x^{\{01\}} + \{30\}\ x^{\{00\}}$

Các bộ hệ số tối ưu (độc lập) với m=8, Nw = 7

Dưới đây là 9 bộ hệ số tối ưu (độc lập) với m=8, Nw = 7.

54 bộ hệ số tối ưu còn lại được tạo ra từ 9 bộ hệ số tối ưu này bằng cách dịch chuyển xoay vòng các hệ số đi k vị trí (với k=1, 2, …, 6).

$c(x) = \{03\}\ x^{\{06\}} + \{02\}\ x^{\{05\}} + \{04\}\ x^{\{04\}} +$
$\{02\}\ x^{\{03\}} + \{03\}\ x^{\{02\}} + \{01\}\ x^{\{01\}} + \{01\}\ x^{\{00\}}$
$c^{-1}(x) = \{61\}\ x^{\{06\}} + \{51\}\ x^{\{05\}} + \{ee\}\ x^{\{04\}} +$
$\{cb\}\ x^{\{03\}} + \{ee\}\ x^{\{02\}} + \{51\}\ x^{\{01\}} + \{61\}\ x^{\{00\}}$

$$
\begin{aligned}
c(x) &= \{03\}\,x^{\{06\}} + \{04\}\,x^{\{05\}} + \{02\}\,x^{\{04\}} + \\
&\quad \{04\}\,x^{\{03\}} + \{03\}\,x^{\{02\}} + \{01\}\,x^{\{01\}} + \{01\}\,x^{\{00\}} \\
c^{-1}(x) &= \{fc\}\,x^{\{06\}} + \{d1\}\,x^{\{05\}} + \{59\}\,x^{\{04\}} + \\
&\quad \{8d\}\,x^{\{03\}} + \{59\}\,x^{\{02\}} + \{d1\}\,x^{\{01\}} + \{fc\}\,x^{\{00\}}
\end{aligned}
$$

$$
\begin{aligned}
c(x) &= \{04\}\,x^{\{06\}} + \{03\}\,x^{\{05\}} + \{02\}\,x^{\{04\}} + \\
&\quad \{03\}\,x^{\{03\}} + \{04\}\,x^{\{02\}} + \{01\}\,x^{\{01\}} + \{01\}\,x^{\{00\}} \\
c^{-1}(x) &= \{89\}\,x^{\{06\}} + \{ee\}\,x^{\{05\}} + \{d1\}\,x^{\{04\}} + \\
&\quad \{8d\}\,x^{\{03\}} + \{d1\}\,x^{\{02\}} + \{ee\}\,x^{\{01\}} + \{89\}\,x^{\{00\}}
\end{aligned}
$$

$$
\begin{aligned}
c(x) &= \{03\}\,x^{\{06\}} + \{04\}\,x^{\{05\}} + \{04\}\,x^{\{04\}} + \\
&\quad \{03\}\,x^{\{03\}} + \{01\}\,x^{\{02\}} + \{02\}\,x^{\{01\}} + \{01\}\,x^{\{00\}} \\
c^{-1}(x) &= \{8d\}\,x^{\{06\}} + \{89\}\,x^{\{05\}} + \{d1\}\,x^{\{04\}} + \\
&\quad \{ee\}\,x^{\{03\}} + \{ee\}\,x^{\{02\}} + \{d1\}\,x^{\{01\}} + \{89\}\,x^{\{00\}}
\end{aligned}
$$

$$
\begin{aligned}
c(x) &= \{04\}\,x^{\{06\}} + \{03\}\,x^{\{05\}} + \{03\}\,x^{\{04\}} + \\
&\quad \{04\}\,x^{\{03\}} + \{01\}\,x^{\{02\}} + \{02\}\,x^{\{01\}} + \{01\}\,x^{\{00\}} \\
c^{-1}(x) &= \{8d\}\,x^{\{06\}} + \{fc\}\,x^{\{05\}} + \{59\}\,x^{\{04\}} + \\
&\quad \{d1\}\,x^{\{03\}} + \{d1\}\,x^{\{02\}} + \{59\}\,x^{\{01\}} + \{fc\}\,x^{\{00\}}
\end{aligned}
$$

$$
\begin{aligned}
c(x) &= \{03\}\,x^{\{06\}} + \{04\}\,x^{\{05\}} + \{03\}\,x^{\{04\}} + \\
&\quad \{01\}\,x^{\{03\}} + \{02\}\,x^{\{02\}} + \{02\}\,x^{\{01\}} + \{01\}\,x^{\{00\}} \\
c^{-1}(x) &= \{ee\}\,x^{\{06\}} + \{ee\}\,x^{\{05\}} + \{61\}\,x^{\{04\}} + \\
&\quad \{51\}\,x^{\{03\}} + \{cb\}\,x^{\{02\}} + \{51\}\,x^{\{01\}} + \{61\}\,x^{\{00\}}
\end{aligned}
$$

$$
\begin{aligned}
c(x) &= \{04\}\,x^{\{06\}} + \{01\}\,x^{\{05\}} + \{02\}\,x^{\{04\}} + \\
&\quad \{03\}\,x^{\{03\}} + \{03\}\,x^{\{02\}} + \{02\}\,x^{\{01\}} + \{01\}\,x^{\{00\}} \\
c^{-1}(x) &= \{ee\}\,x^{\{06\}} + \{51\}\,x^{\{05\}} + \{51\}\,x^{\{04\}} + \\
&\quad \{ee\}\,x^{\{03\}} + \{61\}\,x^{\{02\}} + \{cb\}\,x^{\{01\}} + \{61\}\,x^{\{00\}}
\end{aligned}
$$

$$
\begin{aligned}
c(x) &= \{04\}\,x^{\{06\}} + \{04\}\,x^{\{05\}} + \{01\}\,x^{\{04\}} + \\
&\quad \{03\}\,x^{\{03\}} + \{02\}\,x^{\{02\}} + \{03\}\,x^{\{01\}} + \{01\}\,x^{\{00\}} \\
c^{-1}(x) &= \{d1\}\,x^{\{06\}} + \{8d\}\,x^{\{05\}} + \{d1\}\,x^{\{04\}} + \\
&\quad \{fc\}\,x^{\{03\}} + \{59\}\,x^{\{02\}} + \{59\}\,x^{\{01\}} + \{fc\}\,x^{\{00\}}
\end{aligned}
$$

$$
\begin{aligned}
c(x) &= \{04\}\,x^{\{06\}} + \{02\}\,x^{\{05\}} + \{04\}\,x^{\{04\}} + \\
&\quad \{01\}\,x^{\{03\}} + \{03\}\,x^{\{02\}} + \{03\}\,x^{\{01\}} + \{01\}\,x^{\{00\}} \\
c^{-1}(x) &= \{d1\}\,x^{\{06\}} + \{d1\}\,x^{\{05\}} + \{89\}\,x^{\{04\}} + \\
&\quad \{ee\}\,x^{\{03\}} + \{8d\}\,x^{\{02\}} + \{ee\}\,x^{\{01\}} + \{89\}\,x^{\{00\}}
\end{aligned}
$$

Các bộ hệ số tối ưu (độc lập) với m=8, Nw = 8

Dưới đây là 16 bộ hệ số tối ưu (độc lập) với m=8, Nw = 8.

112 bộ hệ số tối ưu còn lại được tạo ra từ 16 bộ hệ số tối ưu này bằng cách dịch chuyển xoay vòng các hệ số đi k vị trí (với k=1, 2, …, 7).

$a(x) = \{04\}\,x^{\{07\}} + \{07\}\,x^{\{06\}} + \{03\}\,x^{\{05\}} + \{05\}\,x^{\{04\}} + \{07\}\,x^{\{03\}} + \{02\}\,x^{\{02\}} + \{02\}\,x^{\{01\}} + \{01\}\,x^{\{00\}}$

$c^{-1}(x) = \{62\}\,x^{\{07\}} + \{ae\}\,x^{\{06\}} + \{bc\}\,x^{\{05\}} + \{7c\}\,x^{\{04\}} + \{43\}\,x^{\{03\}} + \{49\}\,x^{\{02\}} + \{39\}\,x^{\{01\}} + \{c9\}\,x^{\{00\}}$

$c(x) = \{07\}\,x^{\{07\}} + \{07\}\,x^{\{06\}} + \{02\}\,x^{\{05\}} + \{05\}\,x^{\{04\}} + \{04\}\,x^{\{03\}} + \{02\}\,x^{\{02\}} + \{03\}\,x^{\{01\}} + \{01\}\,x^{\{00\}}$

$c^{-1}(x) = \{43\}\,x^{\{07\}} + \{ae\}\,x^{\{06\}} + \{39\}\,x^{\{05\}} + \{7c\}\,x^{\{04\}} + \{62\}\,x^{\{03\}} + \{49\}\,x^{\{02\}} + \{bc\}\,x^{\{01\}} + \{c9\}\,x^{\{00\}}$

$c(x) = \{04\}\,x^{\{07\}} + \{03\}\,x^{\{06\}} + \{07\}\,x^{\{05\}} + \{07\}\,x^{\{04\}} + \{05\}\,x^{\{03\}} + \{06\}\,x^{\{02\}} + \{03\}\,x^{\{01\}} + \{01\}\,x^{\{00\}}$

$c^{-1}(x) = \{90\}\,x^{\{07\}} + \{77\}\,x^{\{06\}} + \{d0\}\,x^{\{05\}} + \{e6\}\,x^{\{04\}} + \{b6\}\,x^{\{03\}} + \{20\}\,x^{\{02\}} + \{3d\}\,x^{\{01\}} + \{01\}\,x^{\{00\}}$

$c(x) = \{03\}\,x^{\{07\}} + \{06\}\,x^{\{06\}} + \{05\}\,x^{\{05\}} + \{07\}\,x^{\{04\}} + \{07\}\,x^{\{03\}} + \{03\}\,x^{\{02\}} + \{04\}\,x^{\{01\}} + \{01\}\,x^{\{00\}}$

$c^{-1}(x) = \{3d\}\,x^{\{07\}} + \{20\}\,x^{\{06\}} + \{b6\}\,x^{\{05\}} + \{e6\}\,x^{\{04\}} + \{d0\}\,x^{\{03\}} + \{77\}\,x^{\{02\}} + \{90\}\,x^{\{01\}} + \{01\}\,x^{\{00\}}$

$c(x) = \{02\}\,x^{\{07\}} + \{02\}\,x^{\{06\}} + \{07\}\,x^{\{05\}} + \{05\}\,x^{\{04\}} + \{03\}\,x^{\{03\}} + \{07\}\,x^{\{02\}} + \{04\}\,x^{\{01\}} + \{01\}\,x^{\{00\}}$

$c^{-1}(x) = \{39\}\,x^{\{07\}} + \{49\}\,x^{\{06\}} + \{43\}\,x^{\{05\}} + \{7c\}\,x^{\{04\}} + \{bc\}\,x^{\{03\}} + \{ae\}\,x^{\{02\}} + \{62\}\,x^{\{01\}} + \{c9\}\,x^{\{00\}}$

$c(x) = \{07\}\,x^{\{07\}} + \{06\}\,x^{\{06\}} + \{04\}\,x^{\{05\}} + \{07\}\,x^{\{04\}} + \{03\}\,x^{\{03\}} + \{03\}\,x^{\{02\}} + \{05\}\,x^{\{01\}} + \{01\}\,x^{\{00\}}$

$c^{-1}(x) = \{d0\}\,x^{\{07\}} + \{20\}\,x^{\{06\}} + \{90\}\,x^{\{05\}} + \{e6\}\,x^{\{04\}} + \{3d\}\,x^{\{03\}} + \{77\}\,x^{\{02\}} + \{b6\}\,x^{\{01\}} + \{01\}\,x^{\{00\}}$

$$
\begin{aligned}
c(x) &= \{05\}\,x^{\{07\}} + \{03\}\,x^{\{06\}} + \{03\}\,x^{\{05\}} + \{07\}\,x^{\{04\}} + \\
&\quad \{04\}\,x^{\{03\}} + \{06\}\,x^{\{02\}} + \{07\}\,x^{\{01\}} + \{01\}\,x^{\{00\}} \\
c^{-1}(x) &= \{b6\}\,x^{\{07\}} + \{77\}\,x^{\{06\}} + \{3d\}\,x^{\{05\}} + \{e6\}\,x^{\{04\}} + \\
&\quad \{90\}\,x^{\{03\}} + \{20\}\,x^{\{02\}} + \{d0\}\,x^{\{01\}} + \{01\}\,x^{\{00\}}
\end{aligned}
$$

$$
\begin{aligned}
c(x) &= \{03\}\,x^{\{07\}} + \{02\}\,x^{\{06\}} + \{04\}\,x^{\{05\}} + \{05\}\,x^{\{04\}} + \\
&\quad \{02\}\,x^{\{03\}} + \{07\}\,x^{\{02\}} + \{07\}\,x^{\{01\}} + \{01\}\,x^{\{00\}} \\
c^{-1}(x) &= \{bc\}\,x^{\{07\}} + \{49\}\,x^{\{06\}} + \{62\}\,x^{\{05\}} + \{7c\}\,x^{\{04\}} + \\
&\quad \{39\}\,x^{\{03\}} + \{ae\}\,x^{\{02\}} + \{43\}\,x^{\{01\}} + \{c9\}\,x^{\{00\}}
\end{aligned}
$$

$$
\begin{aligned}
c(x) &= \{07\}\,x^{\{07\}} + \{05\}\,x^{\{06\}} + \{03\}\,x^{\{05\}} + \{05\}\,x^{\{04\}} + \\
&\quad \{04\}\,x^{\{03\}} + \{03\}\,x^{\{02\}} + \{02\}\,x^{\{01\}} + \{02\}\,x^{\{00\}} \\
c^{-1}(x) &= \{53\}\,x^{\{07\}} + \{45\}\,x^{\{06\}} + \{df\}\,x^{\{05\}} + \{50\}\,x^{\{04\}} + \\
&\quad \{72\}\,x^{\{03\}} + \{54\}\,x^{\{02\}} + \{5a\}\,x^{\{01\}} + \{13\}\,x^{\{00\}}
\end{aligned}
$$

$$
\begin{aligned}
c(x) &= \{03\}\,x^{\{07\}} + \{04\}\,x^{\{06\}} + \{05\}\,x^{\{05\}} + \{03\}\,x^{\{04\}} + \\
&\quad \{05\}\,x^{\{03\}} + \{07\}\,x^{\{02\}} + \{02\}\,x^{\{01\}} + \{02\}\,x^{\{00\}} \\
c^{-1}(x) &= \{13\}\,x^{\{07\}} + \{5a\}\,x^{\{06\}} + \{54\}\,x^{\{05\}} + \{72\}\,x^{\{04\}} + \\
&\quad \{50\}\,x^{\{03\}} + \{df\}\,x^{\{02\}} + \{45\}\,x^{\{01\}} + \{53\}\,x^{\{00\}}
\end{aligned}
$$

$$
\begin{aligned}
c(x) &= \{04\}\,x^{\{07\}} + \{05\}\,x^{\{06\}} + \{02\}\,x^{\{05\}} + \{05\}\,x^{\{04\}} + \\
&\quad \{07\}\,x^{\{03\}} + \{03\}\,x^{\{02\}} + \{03\}\,x^{\{01\}} + \{02\}\,x^{\{00\}} \\
c^{-1}(x) &= \{72\}\,x^{\{07\}} + \{45\}\,x^{\{06\}} + \{5a\}\,x^{\{05\}} + \{50\}\,x^{\{04\}} + \\
&\quad \{53\}\,x^{\{03\}} + \{54\}\,x^{\{02\}} + \{df\}\,x^{\{01\}} + \{13\}\,x^{\{00\}}
\end{aligned}
$$

$$
\begin{aligned}
c(x) &= \{07\}\,x^{\{07\}} + \{05\}\,x^{\{06\}} + \{05\}\,x^{\{05\}} + \{07\}\,x^{\{04\}} + \\
&\quad \{03\}\,x^{\{03\}} + \{04\}\,x^{\{02\}} + \{03\}\,x^{\{01\}} + \{02\}\,x^{\{00\}} \\
c^{-1}(x) &= \{d7\}\,x^{\{07\}} + \{f7\}\,x^{\{06\}} + \{b0\}\,x^{\{05\}} + \{1c\}\,x^{\{04\}} + \\
&\quad \{df\}\,x^{\{03\}} + \{da\}\,x^{\{02\}} + \{91\}\,x^{\{01\}} + \{63\}\,x^{\{00\}}
\end{aligned}
$$

$$
\begin{aligned}
c(x) &= \{05\}\,x^{\{07\}} + \{04\}\,x^{\{06\}} + \{07\}\,x^{\{05\}} + \{07\}\,x^{\{04\}} + \\
&\quad \{03\}\,x^{\{03\}} + \{05\}\,x^{\{02\}} + \{03\}\,x^{\{01\}} + \{02\}\,x^{\{00\}} \\
c^{-1}(x) &= \{b0\}\,x^{\{07\}} + \{da\}\,x^{\{06\}} + \{d7\}\,x^{\{05\}} + \{1c\}\,x^{\{04\}} + \\
&\quad \{91\}\,x^{\{03\}} + \{f7\}\,x^{\{02\}} + \{df\}\,x^{\{01\}} + \{63\}\,x^{\{00\}}
\end{aligned}
$$

$$
\begin{aligned}
c(x) &= \{03\}\,x^{\{07\}} + \{03\}\,x^{\{06\}} + \{07\}\,x^{\{05\}} + \{05\}\,x^{\{04\}} + \\
&\quad \{02\}\,x^{\{03\}} + \{05\}\,x^{\{02\}} + \{04\}\,x^{\{01\}} + \{02\}\,x^{\{00\}} \\
c^{-1}(x) &= \{df\}\,x^{\{07\}} + \{54\}\,x^{\{06\}} + \{53\}\,x^{\{05\}} + \{50\}\,x^{\{04\}} + \\
&\quad \{5a\}\,x^{\{03\}} + \{45\}\,x^{\{02\}} + \{72\}\,x^{\{01\}} + \{13\}\,x^{\{00\}}
\end{aligned}
$$

$$
\begin{aligned}
c(x) &= \{03\}\,x^{\{07\}} + \{05\}\,x^{\{06\}} + \{03\}\,x^{\{05\}} + \{07\}\,x^{\{04\}} + \\
&\quad \{07\}\,x^{\{03\}} + \{04\}\,x^{\{02\}} + \{05\}\,x^{\{01\}} + \{02\}\,x^{\{00\}} \\
c^{-1}(x) &= \{df\}\,x^{\{07\}} + \{f7\}\,x^{\{06\}} + \{91\}\,x^{\{05\}} + \{1c\}\,x^{\{04\}} + \\
&\quad \{d7\}\,x^{\{03\}} + \{da\}\,x^{\{02\}} + \{b0\}\,x^{\{01\}} + \{63\}\,x^{\{00\}}
\end{aligned}
$$

$$
\begin{aligned}
c(x) &= \{03\}\,x^{\{07\}} + \{04\}\,x^{\{06\}} + \{03\}\,x^{\{05\}} + \{07\}\,x^{\{04\}} + \\
&\quad \{05\}\,x^{\{03\}} + \{05\}\,x^{\{02\}} + \{07\}\,x^{\{01\}} + \{02\}\,x^{\{00\}} \\
c^{-1}(x) &= \{91\}\,x^{\{07\}} + \{da\}\,x^{\{06\}} + \{df\}\,x^{\{05\}} + \{1c\}\,x^{\{04\}} + \\
&\quad \{b0\}\,x^{\{03\}} + \{f7\}\,x^{\{02\}} + \{d7\}\,x^{\{01\}} + \{63\}\,x^{\{00\}}
\end{aligned}
$$