VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Trịnh Văn Anh

MỘT SỐ HỆ MÃ HÓA VỚI QUYỀN GIẢI MÃ LINH ĐỘNG

LUẬN ÁN TIẾN SĨ KỸ THUẬT

Hà Nội – Năm 2021

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Trịnh Văn Anh

MỘT SỐ HỆ MÃ HÓA VỚI QUYỀN GIẢI MÃ LINH ĐỘNG

Chuyên ngành : Hệ thống thông tin

Mã số: 9.48.01.04

LUẬN ÁN TIẾN SĨ KỸ THUẬT

NGƯỜI HƯỚNG DẪN KHOA HỌC:

1. GS. TS Nguyễn Bình
2. TS. Hồ Văn Hương

Hà Nội - Năm 2021

LỜI CAM ĐOAN

Tôi cam đoan đây là công trình nghiên cứu của riêng tôi. Các số liệu, kết quả nêu trong luận án là trung thực và chưa từng được ai công bố trong bất kỳ công trình nào khác.

Tác giả luận án

Trịnh Văn Anh

LỜI CẢM ƠN

Trong quá trình học tập, nghiên cứu và thực hiện luận án, Nghiên cứu sinh đã nhận được sự định hướng, giúp đỡ, các ý kiến đóng góp quý báu và những lời động viên khích lệ chân thành của các nhà khoa học, các thầy cô, các tác giả cùng nghiên cứu, đồng nghiệp và gia đình.

Có được kết quả hôm nay, trước hết, nghiên cứu sinh xin bày tỏ lời cảm ơn chân thành tới các thầy hướng dẫn, cùng các nhóm nghiên cứu các công trình nghiên cứu đã công bố. Xin chân thành cảm ơn các thầy, cô ở khoa Đào tạo Sau Đại học và các thầy, cô ở Học viện Công nghệ Bưu chính Viễn thông đã giúp đỡ nghiên cứu sinh trong suốt thời gian thực hiện luận án.

Nghiên cứu sinh chân thành cảm ơn Ban Giám đốc Học viện Công nghệ Bưu chính Viễn thông đã tạo điều kiện thuận lợi để nghiên cứu sinh hoàn thành nhiệm vụ nghiên cứu.

Nghiên cứu sinh xin chân thành cám ơn lãnh đạo và đồng nghiệp Trường Đại học Văn hóa, Thể thao và Du lịch Thanh Hóa đã tạo điều kiện và giúp đỡ về mọi mặt để Nghiên cứu sinh hoàn thành được luận án.

Cuối cùng, nghiên cứu sinh bày tỏ lời cảm ơn tới gia đình, bạn bè đã luôn động viên, chia sẻ, ủng hộ, khuyến khích và giúp đỡ nghiên cứu sinh trong suốt quá trình học tập và nghiên cứu vừa qua.

Hà Nội, ngày 15 tháng 7 năm 2020

Học viên

Trịnh Văn Anh

MỤC LỤC

LỜI CAM ĐOAN ii
DANH MỤC TỪ VIẾT TẮT vii
DANH MỤC CÁC HÌNH VẼ x
DANH MỤC CÁC KÝ TỰ TOÁN HỌC xi
MỞ ĐẦU 1
CHƯƠNG 1. TỔNG QUAN VỀ MÃ HÓA QUẢNG BÁ VÀ MÃ HÓA DỰA TRÊN THUỘC TÍNH 6
1.1. Khái quát chung về mã hóa 6
1.1.1. Định nghĩa và mô hình an toàn của hệ mã hóa quảng bá 7
1.1.2. Định nghĩa 8
1.1.3. Mô hình an toàn 9
1.2. Khái quát về một số hệ mã hóa quảng bá quan trọng và tình hình nghiên cứu hiện nay 12
1.2.1. Hệ mã hóa NNL và các cải tiến 13
1.2.1.1. Hệ mã thứ nhất NNL-1[44] 13
1.2.1.2. Hệ mã thứ hai NNL-2 [44] 16
1.2.2. Hệ mã hóa BGW và các cải tiến 20
1.2.2.1. Công cụ ánh xạ song tuyến 21
1.2.2.2. Một số cải tiến của hệ mã BGW [46, 10, 29] 23
1.2.3. Hệ mã hóa Delerablee và các cải tiến 24
1.3. Tình hình nghiên cứu hiện nay của Mã hóa quảng bá 27
1.4. Mã hóa quảng bá đa kênh và mã hóa dựa trên thuộc tính 30
1.4.1. Tổng quan về mã hóa quảng bá đa kênh 30
1.4.2. Tổng quan về Mã hóa dựa trên thuộc tính 31
1.5. Kết luận chương 1 34

CHƯƠNG 2: MÃ HÓA QUẢNG BÁ ĐA KÊNH 35
2.1. Định nghĩa và mô hình an toàn của hệ mã hóa quảng bá đa kênh 35
2.1.1. Định nghĩa 35
2.1.2. Mô hình an toàn.....38
2.2. Một số hệ mã hóa quảng bá đa kênh quan trọng 39
2.2.1. Hệ mã hóa quảng bá đa kênh - $MCBE_1$ 39
2.2.2. Hệ mã hóa quảng bá đa kênh - $MCBE_2$ 44
2.2.3. Một số cải tiến đối với hệ $MCBE_1$ và $MCBE_2$.....46
2.3. Lược đồ mã hóa quảng bá đa kênh đề xuất 46
2.3.1. Ý tưởng xây dựng.....47
2.3.2. Lược đồ mã hóa đề xuất và so sánh 47
2.3.3. Đánh giá an toàn.....51
2.3.4. Cài đặt và đánh giá hiệu quả 56
2.4. Kết luận chương 2.....58
CHƯƠNG 3: HỆ MÃ HÓA DỰA TRÊN THUỘC TÍNH 60
3.1. Định nghĩa và mô hình an toàn của hệ mã hóa dựa trên thuộc tính 60
3.1.1. Định nghĩa 61
3.1.2. Mô hình an toàn.....63
3.2. Một số hệ mã hóa dựa trên thuộc tính nền tảng quan trọng hiện nay.....64
3.2.1. Hệ mã hóa dựa trên thuộc tính của Rouselakis-Waters năm 2013.....64
3.2.2. Hệ mã hóa dựa trên thuộc tính của Agrawal-Chase17.....66
3.3. Mã hóa dựa trên thuộc tính (CP-ABE-01) đề xuất.....69
3.3.1. Ý tưởng xây dựng.....70
3.3.2. Mã hóa đề xuất và so sánh.....70

3.3.3. Đánh giá an toàn.....74
3.3.4. Cài đặt và đánh giá hiệu quả 78
3.4. Đề xuất thứ hai (CP-ABE-02) về mã hóa dựa trên thuộc tính.....80
3.4.1. Ý tưởng xây dựng và so sánh 80
3.4.2. Lược đồ mã hóa đề xuất thứ 2 dựa trên thuộc tính 82
3.4.3. Đánh giá an toàn dữ liệu.....86
3.4.4. Đánh giá an toàn từ khóa.....93
3.5. Kết luận chương 3.....101
KẾT LUẬN VÀ KIẾN NGHỊ.....103
CÁC CÔNG TRÌNH CÔNG BỐ TRONG LUẬN ÁN 104
TÀI LIỆU THAM KHẢO.....105

DANH MỤC TỪ VIẾT TẮT

Từ viết tắt	Tiếng Anh	Tiếng Việt
ABBE	Attribute-Based Broadcast Encryption	Mã hóa quảng bá dựa trên thuộc tính
ABE	Attribute-Based Encryption	Mã hóa dựa trên thuộc tính
AES	Advanced Encryption Standard	Chuẩn mã hóa khóa đối xứng
BDHE		Bài toán khó BDHE
BE	Broadcast Encryption	Mã hóa quảng bá
BGW	D. Boneh, C. Gentry, and B. Waters	Mã hóa quảng bá BGW
CA	Center Authority	Trung tâm chứng thực số
CCA	Chosen Ciphertext Attack	Tấn công chọn trước bản mã
CDH		Bài toán khó CDH
CNF	Conjuntive Normal Form	Dạng liên kết chuẩn
CPA	Chosen Plantext Attack	Tấn công chọn trước bản rõ
CP-ABE	Ciphertext-Policy Attribute-Based Encryption	Mã hóa dựa trên thuộc tính có chính sách bản mã
DBDHE		Bài toán khó DBDHE
DDH		Bài toán khó DDH
DVD	Digital Versatile Disc	Đĩa lưu trữ dữ liệu
GDDHE		Bài toán khó GDDHE
Hdr		Bản mã của khóa phiên
IBE	Identity-Based Encryption	Mã hóa dựa trên định danh
ID	Identity-Based	Định danh
ISI	Institute for Scientific Information	Viện thông tin khoa học
KP-ABE	Key-Policy Attribute-Based Encryption	Mã hóa dựa trên thuộc tính có chính sách khóa

LSS	Linear Secret Sharing	Chia sẻ bí mật tuyến tính
LSSS	Linear Secret Sharing Scheme	Lược đồ chia sẻ bí mật tuyến tính
LWE	Learning With Errors	Học từ lỗi
MCBE	Multi-Channel Broadcast Encryption	Mã hóa quảng bá đa kênh
NCS		Nghiên cứu sinh
NNL	D. Naor, M. Naor, and J.Lotspiech	Mã hóa quảng bá NNL
PKG	Private Key Generator	Trung tâm tạo khóa bí mật
PKI	Public Key Infrastructure	Cơ sở hạ tầng khóa công khai
ROM	Random Oracle	Bộ tiên tri ngẫu nhiên
RSA	Rivest–Shamir–Adleman	Mã hóa khóa công khai RSA
GWIBE		Lược đồ mã hóa GWIBE
EK		Khóa bí mật dùng để mã hóa

DANH MỤC CÁC BẢNG TRONG LUẬN ÁN

STT	Tên Bảng	Trang
1	Bảng 2.1. So sánh một số hệ mã hóa đa kênh với MCBE đề xuất	50
2	Bảng 2.2. Thực nghiệm cài đặt lược đồ MCBE đề xuất	58
3	Bảng 3.1. So sánh một số hệ mã hóa dựa trên thuộc tính đã có với mã hóa đề xuất	73
4	Bảng 3.2. Kết quả thực nghiệm cài đặt hệ CP-ABE đề xuất	80

DANH MỤC CÁC HÌNH VẼ

STT	Tên hình	Trang
1	Hình 1.1. Hệ NNL-1	15
2	Hình 1.2. Hệ NNL-2	17
3	Hình 3.1. Bài toán khó (P, Q, R, f) – GDDHE	75
4	Hình 3.2. So sánh mô hình hoạt động giữa hệ tìm kiếm đề xuất và các hệ khác	82

DANH MỤC CÁC KÝ TỰ TOÁN HỌC

STT	KÝ HIỆU	Ý NGHĨA
1	λ	Tham số an toàn
2	*id*	Định danh người dùng
3	S	Tập người dùng có khả năng giải mã
4	K	Khóa phiên
5	Param	Khóa công khai của hệ thống
6	$\mathcal{A}$	Kẻ tấn công
7	$\mathcal{C}$	Người thách thức
8	msk	Khóa bí mật của hệ thống
10	Λ_C	Danh sách người dùng đã bị người tấn công biết khóa bí mật
11	Λ_D	Danh sách người dùng đã bị người tấn công biết bản rõ
12	$\mathcal{K}$	Là không gian của khóa phiên
13	b	Bít
14	$\cap$	Phép giao
15	$\mathbf{Succ}^{\mathrm{IND}}$	Kết quả thành công của kẻ tấn công
16	Pr	Xác suất
17	$\mathbf{Adv}^{\mathrm{IND}}$	Lợi thế của kẻ tấn công
18	$\mathcal{N}$	Tập tất cả người dùng trong hệ thống
19	$\mathcal{R}$	Tập người dùng không có khả năng giải mã trong hệ thống
20	*log*	Logarit
21	SK_{id}	Khóa bí mật của người dùng id
22	$\mathbb{G}$	Nhóm Abelian hữu hạn chứa các phần tử g
31	$\mathbb{G}_T$	Nhóm Abelian hữu hạn chứa các phần tử t
32	$\widetilde{\mathbb{G}}$	Nhóm Abelian hữu hạn chứa các phần tử $\tilde{g}$

33	e	Ánh xạ song tuyến
34	p	Số nguyên tố

MỞ ĐẦU

Mật mã đã được phát triển và sử dụng từ hàng ngàn năm nay, với mục tiêu ban đầu là cho phép người gửi gửi thông tin một cách an toàn tới người nhận thông qua một kênh không an toàn. Để thực hiện điều đó, người gửi và người nhận thống nhất trước với nhau một khóa bí mật chung ban đầu. Thông tin trước khi gửi sẽ được biến đổi (gọi là mã hóa) dựa trên khóa bí mật chung này sang một dạng khác không có ý nghĩa, gọi là bản mã. Tiếp theo, bản mã sẽ được gửi tới người nhận thông qua kênh không an toàn. Người nhận cuối cùng dựa trên khóa chung này để chuyển bản mã thành dạng thông tin ban đầu (gọi là giải mã) có ý nghĩa. Các kẻ tấn công có thể dựa trên kênh truyền không an toàn để lấy được bản mã, nhưng do không biết khóa bí mật chung của người gửi và người nhận nên không thể nào giải mã được. Một hệ thống với các bước gửi nhận thông tin như vậy có thể được gọi là một hệ mã hóa.

1. Lý do chọn đề tài

Trong thực tiễn, an toàn thông tin đang là vấn đề cấp bách của xã hội, việc xác định cách bảo mật, cách xây dựng hệ thống an toàn thông tin tránh hiện tượng mất cắp, rò rỉ thông tin đang được các nhà khoa học nghiên cứu, đây cũng là vấn đề đang được nước ta và các quốc gia trên thế giới đặc biệt quan tâm. Việc để những thông tin mật, thông tin quan trọng bị xâm hại trái phép là mối nguy hiểm cho toàn bộ người dùng, cơ quan, tổ chức.

Để giải quyết vấn đề an toàn thông tin cho các hệ thống, kỹ thuật được dùng cơ bản hiện nay là mã hóa. Tuy nhiên, trong các hệ thống thực tế ngày nay, yêu cầu về các dạng mã hóa phải linh động và đa dạng hơn. Ví dụ, với hệ thống truyền hình trả tiền hay radio cho quân đội, trung tâm phát sóng sẽ mã hóa sóng trước khi phát và rất nhiều người dùng với các đầu thu của mình có thể giải mã sóng để xem (hoặc nghe). Như vậy, trong trường hợp này mã hóa không còn ở dạng 1-1 (tức là thông tin chỉ hiểu được hay giải mã được bởi một người nhận duy nhất) mà là 1-n với n > 1 là số người dùng có khả năng giải mã. Dĩ nhiên cách đơn giản để chuyển từ mã hóa 1-1 sang 1-n là cho phép n người dùng cùng biết một khóa bí mật, tuy nhiên vấn đề nảy sinh là hệ thống không thể loại bỏ một đầu thu không cho phép giải mã nữa (ví dụ

đầu thu này hết hạn không nạp tiền thuê bao) mà không ảnh hưởng đến các đầu thu khác, vì các đầu thu cùng chia sẻ chung một khóa bí mật. Để giải quyết vấn đề này, kỹ thuật mã hóa quảng bá viết tắt là BE (Broadcast Encryption) đã được giới thiệu bởi Fiat and Naor [28], trong đó hệ thống cho phép mỗi đầu thu sở hữu một khóa bí mật khác nhau, ở mỗi lần mã hóa trung tâm phát sóng có thể dễ dàng loại bỏ những đầu thu cụ thể khỏi tập các đầu thu có thể giải mã được. Cụ thể, ở mỗi lần mã hóa bản mã *m* trung tâm phát sóng có thể chọn tùy ý một tập người dùng S có khả năng giải mã.

Phan và các tác giả [47] đã giới thiệu mã hóa quảng bá đa kênh viết tắt là MCBE (Multi-Channel Broadcast Encryption) là mở rộng khái niệm của mã hóa quảng bá từ việc gửi một thông tin *m* đến một nhóm người dùng *S*, đến việc cho phép cùng lúc gửi nhiều thông tin $m_1, m_2,..., m_k$ đến các tập người dùng khác nhau tương ứng $S_1, S_2,..., S_k$, và người dùng trong tập nào thì chỉ có thể giải mã được bản mã cho tập đó.

Một loại hệ mã hóa khác là mã hóa dựa trên thuộc tính được viết tắt là ABE (Attribute-Based Encryption), được giới thiệu bởi Sahai và Waters [53], là mở rộng của mã hóa quảng bá, trong đó cho phép điều kiện giải mã linh động hơn so với mã hóa quảng bá. Với mã hóa quảng bá, người lập mã phải biết cụ thể tập người dùng có thể giải mã được tại thời điểm lập mã, tuy nhiên trong thực tế, người lập mã không phải lúc nào cũng biết được điều này. Ví dụ, công ty FPT lưu trữ dữ liệu của họ trên đám mây, họ muốn lưu trữ một văn bản mà cho phép các nhân viên của phòng kỹ thuật và phòng hỗ trợ khách hàng, đồng thời tham gia trong dự án e-Health có thể giải mã được. Với kỹ thuật mã hóa quảng bá, công ty FPT phải biết ngay tại thời điểm mã hóa văn bản là những nhân viên cụ thể nào của hai phòng trên tham gia vào dự án. Trong thực tế, do tính chất công việc, dự án e-Health có thể thêm nhân viên từ các phòng trên. Để giải quyết vấn đề này, công ty FPT phải thực hiện lại quá trình mã hóa văn bản và tải lên trên Cloud, điều này là không hợp lý.

Mã hóa dựa trên thuộc tính có thể giải quyết tốt những vấn đề như vậy. Trong một hệ thống mã hóa thuộc tính, ta có thể định nghĩa một tập các thuộc tính. Ví dụ,

Dự án e-Health (e-H), phòng kỹ thuật (PKT), phòng chăm sóc khách hàng (PCS), nhân viên (NV), trưởng phòng (TP),... là các thuộc tính. Nếu người dùng X thuộc phòng kỹ thuật, là nhân viên và tham gia dự án thì sẽ nhận các thuộc tính là PKT, e-H, NV và nhận khóa bí mật tương ứng với các thuộc tính này. Công ty FPT khi mã hóa văn bản chỉ đơn giản là thực hiện việc mã hóa trong đó quy định cho những nhân viên của hai phòng này và làm trong dự án e-Health có thể giải mã được mà không cần biết cụ thể là nhân viên nào. Điều kiện giải mã có thể được mô tả bằng một biểu thức boolean như sau:

(NV and PKT and e-H) or (NV and PCS and e-H)

Khi một nhân viên mới thuộc một trong hai phòng này tham gia dự án, người này sẽ nhận thêm thuộc tính là Dự án e-Health và nhận khóa bí mật tương ứng, hiển nhiên nhân viên mới này sẽ có khả năng giải mã vì đáp ứng được điều kiện giải mã.

Ngày nay, với sự phát triển của mạng Internet, các thiết bị tham gia hệ thống có thể có năng lực rất yếu, dẫn đến các hệ mã như: Mã hóa quảng bá, mã hóa quảng bá đa kênh và mã hóa dựa trên thuộc tính ngoài yêu cầu đảm bảo về an toàn phải thực sự đảm bảo về hiệu quả, đặc biệt là ở ba tính chất là độ dài bản mã, độ dài khóa bí mật và tốc độ giải mã.

Để giải quyết một số tồn tại trong mã hóa quảng bá, mã hóa quảng bá đa kênh và mã hóa dựa trên thuộc tính, NCS chọn đề tài nghiên cứu: "*Một số hệ mã hóa với quyền giải mã linh động*".

2. Mục tiêu nghiên cứu

Đề tài tập trung nghiên cứu một số loại mã hóa: Mã hóa quảng bá, mã hóa quảng bá đa kênh và mã hóa dựa trên thuộc tính, nhằm đạt được các mục tiêu chính sau đây:

1. Nắm bắt được tổng quan tình hình nghiên cứu hiện nay của một số loại mã hóa như: Mã hóa quảng bá, mã hóa quảng bá đa kênh và mã hóa dựa trên thuộc tính.

2. Xây dựng được lược đồ mã hóa quảng bá đa kênh mới, khắc phục được một số điểm yếu của hệ BE hiện có như: Tốc độ giải mã chậm, chỉ hệ thống mới có khả năng mã hóa.

3. Xây dựng được lược đồ mã hóa ABE mới có các tính chất như: Độ dài bản mã ngắn, độ dài khóa bí mật và tốc độ giải mã không quá dài, quá chậm, so với các hệ khác, hỗ trợ chức năng tìm kiếm trên dữ liệu đã được mã hóa.

3. Đối tượng, phạm vi nghiên cứu, phương pháp và nội dung nghiên cứu

Đối tượng và phạm vi nghiên cứu trong luận án là một số hệ mã hóa: Mã hóa quảng bá, mã hóa quảng bá đa kênh và mã hóa dựa trên thuộc tính. Trong phạm vi đề tài sẽ thực hiện các nội dung nghiên cứu sau đây:

1. Tìm hiểu một số kỹ thuật, đưa ra lược đồ mã hóa cải tiến để xây dựng hoàn thiện hơn cho hệ mã hóa quảng bá, hệ mã hóa quảng bá đa kênh.

2. Nghiên cứu lược đồ mã hóa quảng bá đa kênh mới, dựa trên các kỹ thuật xây dựng một số hệ mã hóa quảng bá khác như hệ mã hóa quảng bá Delerablee [25] và các cải tiến được viết tại các tài liệu [55, 56].

3. Tìm hiểu một số kỹ thuật về mã hóa dựa trên thuộc tính, đưa ra lược đồ mã hóa mới để góp phần xây dựng các hệ mã hóa dựa trên thuộc tính hiện nay được hiệu quả hơn.

4. Nghiên cứu lược đồ mã hóa dựa trên thuộc tính và một số kỹ thuật xây dựng hệ mã hóa quảng bá, mã hóa quảng bá đa kênh, đặc biệt tập trung vào việc xây dựng lược đồ mã hóa dựa trên thuộc tính, có tính chất là độ dài bản mã là hằng số và tìm kiếm trên dữ liệu đã được mã hóa.

5. Nghiên cứu mức an toàn của một số hệ mã hóa dựa trên thuộc tính hiện nay.

4. Bố cục luận án

Luận án bao gồm 3 chương:

CHƯƠNG 1. TỔNG QUAN VỀ MÃ HÓA QUẢNG BÁ VÀ MÃ HÓA DỰA TRÊN THUỘC TÍNH

Nội dung chương sẽ trình bày và giới thiệu về một số hệ mã hóa cơ bản, quan trọng đang được sử dụng hiện nay. Bao gồm ba loại mã: Mã hóa quảng bá, mã hóa quảng bá đa kênh và mã hóa dựa trên thuộc tính. Ba loại mã hóa này hỗ trợ quyền giải mã linh động và đang được ứng dụng trong rất nhiều loại ứng dụng hiện nay như

các ứng dụng truyền hình trả tiền, chia sẻ files, social network (facebooks, twitter,...), lưu trữ an toàn dữ liệu trên đám mây cho những ứng dụng như e-Health, chính phủ điện tử,...

CHƯƠNG 2. MÃ HÓA QUẢNG BÁ ĐA KÊNH

Trong chương này, Nghiên cứu sinh sẽ trình bày tổng quan về mã hóa quảng bá đa kênh, bao gồm định nghĩa về mã hóa quảng bá đa kênh, mô hình an toàn của mã hóa quảng bá đa kênh và một số hạn chế cần phải khắc phục. Bên cạnh đó Nghiên cứu sinh (NCS) sẽ đề xuất và trình bày ý tưởng để khắc phục một số điểm yếu của mã hóa quảng bá đa kênh nhằm khắc phục một số hạn chế còn tồn đọng.

CHƯƠNG 3. MÃ HÓA DỰA TRÊN THUỘC TÍNH

Tác giả trình bày về hệ mã hóa dựa trên thuộc tính bao gồm, định nghĩa chung về mã hóa dựa trên thuộc tính, mô hình an toàn của mã hóa dựa trên thuộc tính, một số hệ mã hóa dựa trên thuộc tính hiện nay. Phần cuối chương sẽ trình bày 02 lược đồ mã hóa dựa trên thuộc tính mới được đề xuất trong luận án.

CHƯƠNG 1. TỔNG QUAN VỀ MÃ HÓA QUẢNG BÁ VÀ MÃ HÓA DỰA TRÊN THUỘC TÍNH

Phần đầu chương, nghiên cứu sinh giới thiệu chung về ba loại mã hóa cụ thể hiện nay: Thứ nhất là mã hóa quảng bá, thứ hai là mã hóa quảng bá đa kênh và thứ ba là mã hóa dựa trên thuộc tính. Trong phần nội dung, tác giả trình bày chi tiết một số mã hóa quảng bá hiện nay mà luận án nghiên cứu, sau đó trình bày sơ lược kết quả nghiên cứu mới và các vấn đề tồn đọng cần khắc phục đối với ba loại mã này.

1.1. Khái quát chung về mã hóa

Một hệ thống bao gồm thuật toán tạo khóa bí mật, thuật toán mã hóa, thuật toán giải mã được gọi chung là một hệ mã hóa. Trong đó, một hệ mã hóa mà khóa dùng để mã hóa và khóa dùng để giải mã là như nhau, được gọi là một hệ mã hóa khóa bí mật. Hệ mã hóa khóa bí mật đang được dùng phổ biến nhất hiện nay là AES với các biến thể cho khóa bí mật là 128, 192 và 256 bits. Ưu điểm của mã hóa khóa bí mật là tốc độ mã hóa và giải mã nhanh. Nhược điểm của các hệ này là giữa người gửi và người nhận phải tiếp xúc trước với nhau để thống nhất một khóa bí mật chung, điều này rất khó thực hiện trong môi trường thực tế hiện nay. Để giải quyết vấn đề trên, hệ mã hóa khóa công khai đã được giới thiệu, trong đó khóa dùng để mã hóa gọi là khóa công khai và khóa dùng để giải mã là khóa bí mật. Khóa công khai của người nhận được công bố trước, mỗi người gửi khi muốn gửi thông tin cho người nhận sẽ dùng khóa công khai này để mã hóa thông tin (không cần tiếp xúc trước với người nhận để thỏa thuận khóa bí mật chung), người nhận sẽ có một khóa bí mật tương ứng với khóa công khai này dùng để giải mã. Như vậy, trong một hệ mã hóa khóa công khai mỗi người dùng (người gửi hoặc người nhận) tham gia hệ thống sẽ có một cặp khóa công khai và bí mật, khóa công khai được công bố công khai trước, trong khi khóa bí mật được giữ bí mật riêng mình. Whitfield Diffie and Martin Hellman có thể được xem là những người đầu tiên đề xuất cụ thể một hệ mã hóa khóa công khai, một số hệ mã hóa khóa công khai hiện được dùng phổ biến hiện nay như hệ RSA hay Elgamal.

Ngày nay, để tận dụng cả hai ưu điểm của mã hóa khóa bí mật và mã hóa khóa công khai, khi gửi thông tin người ta thường dùng hệ mã hóa lai. Với hệ mã hóa lai, trước mỗi lần gửi thông tin người gửi sẽ chọn một giá trị gọi là khóa phiên, tiếp theo họ sẽ dùng khóa công khai của người nhận để mã hóa khóa phiên này, sau đó dùng khóa phiên này như là khóa bí mật trong hệ mã hóa khóa bí mật để mã hóa thông tin. Như vậy, người gửi đã đồng thời dùng cả hai giải thuật mã hóa của hai loại, hệ mã hóa công khai và hệ mã hóa bí mật. Bản mã sẽ bao gồm bản mã của khóa phiên và bản mã của thông tin. Người nhận, trước tiên dùng giải thuật giải mã của hệ mã hóa khóa công khai để giải mã bản mã của khóa phiên thu về giá trị khóa phiên, sau đó dùng giải thuật giải mã của hệ mã hóa khóa bí mật với khóa phiên chính là khóa bí mật đã biết để giải mã, thu về thông tin. Thông thường, thông tin cần gửi thì rất lớn nhưng giá trị khóa phiên chỉ cần rất bé, do đó với mã hóa lai ta tận dụng được ưu thế của cả hai loại hệ mã hóa bí mật và công khai. Người gửi không cần thống nhất khóa bí mật chung trước với người nhận, thông tin vẫn được mã hóa và giải mã dùng giải thuật của hệ mã hóa khóa bí mật. Các hệ mã đã trình bày ở trên như AES, RSA, Elgamal đều là các mã hóa ở dạng 1-1, tức là với mỗi bản mã chỉ có duy nhất một người có khả năng giải mã. Hay là quyền giải mã của người dùng bị giới hạn rằng chỉ giải mã được nếu biết khóa bí mật tương ứng với bản mã. Các ứng dụng hiện đại ngày nay như truyền hình trả tiền, mạng xã hội,…Yêu cầu rằng quyền giải mã phải ở dạng linh động hơn. Cụ thể, một hệ mã hóa có quyền giải mã linh động thì với một bản mã, người lập mã có thể tùy ý quy định một nhóm người khác nhau với các khóa bí mật khác nhau đều có thể giải mã được, mã hóa như vậy phải ở dạng 1-n với n > 1. Một trong những hệ mã hóa 1-n hiện nay là hệ mã hóa quảng bá.

1.1.1. Định nghĩa và mô hình an toàn của hệ mã hóa quảng bá

Mã hóa quảng bá được giới thiệu bởi Fiat and Naor [28] với mục tiêu tạo ra một hệ mã hóa mà ở mỗi lần mã hóa người mã hóa có thể chọn một tập người dùng tùy ý có thể giải mã được. Trong khi đó, cả độ dài bản mã, độ dài khóa bí mật và tốc độ giải mã khắc phục được nhược điểm về độ dài khóa và độ dài bản mã.

“Với hệ mã hóa quảng bá quyền cơ bản nhất của kẻ tấn công là biết bản mã và khóa công khai. Ngoài ra, kẻ tấn công còn có thể có thêm các quyền khác như quyền biết khóa bí mật của các người dùng không có khả năng giải mã (những người dùng này nằm bên ngoài tập S), quyền tùy ý chọn một bản mã và biết bản rõ tương ứng,…”.

1.1.2. Định nghĩa

Mã hóa quảng bá được định nghĩa như sau:

Khởi tạo (λ):

Đầu vào của giải thuật khởi tạo là tham số an toàn λ. Trong đó tham số an toàn λ có nghĩa là để phá được hệ mã này kẻ tấn công cần thực hiện ít nhất 2^{λ} phép toán cơ bản nhị phân của máy tính.. Đầu ra của giải thuật là khóa công khai và khóa bí mật của hệ thống.

Tạo khóa (msk, *id*, list, param):

Đầu vào của giải thuật là khóa bí mật của hệ thống, định danh của người dùng, danh sách các người dùng hiện thời đã được cấp khóa của hệ thống, cuối cùng là khóa công khai của hệ thống. Nếu định danh của người dùng *id* là hợp lệ và $id \notin$ list thì giải thuật sẽ trả về khóa bí mật SK_{id} cho người dùng *id*, sau đó *id* sẽ được đưa vào danh sách các người dùng hiện thời đã được cấp khóa của hệ thống. Ngược lại giải thuật sẽ trả về $\perp$ (giải thuật ngừng và kết quả đầu ra là null).

Mã hóa (*S*, param):

Đầu vào của giải thuật là tập người dùng có khả năng giải mã *S* và khóa công khai của hệ thống. Đầu ra của giải thuật là khóa phiên làm việc *K* và bản mã của nó chứa cả *S*, ký hiệu là Hdr. Lưu ý rằng, trong thực tế thì khóa phiên làm việc *K* sau đó sẽ được dùng như khóa bí mật trong hệ mã hóa khóa bí mật (ví dụ AES) để mã hóa dữ liệu. Như vậy, bản mã đầy đủ trong thực tế gọi là sẽ bao gồm cả Hdr và là bản mã của dữ liệu thực tế được mã hóa dưới khóa bí mật.

Giải mã (Hdr, SK_{id}, param):

Đầu vào của giải thuật là bản mã Hdr của khóa phiên làm việc *K*, khóa bí mật của người dùng SK_{id}, khóa công khai của hệ thống. Đầu ra của giải thuật là khóa phiên làm việc K nếu như $id \in S$, ngược lại đầu ra của giải thuật là $\perp$. Lưu ý rằng, trong

thực tế thì sau khi giải mã tìm được khóa phiên *K*, người dùng sẽ dùng *K* như khóa bí mật để giải mã tìm lại dữ liệu thực tế đã được mã hóa.

Mã hóa được mã hóa với cơ chế như trên được gọi là hệ mã hóa lai. Lý do hệ mã hóa lai được dùng trong thực tế là do nó tận dụng được cả hai ưu thế của hệ mã hóa khóa công khai truyền thống và hệ mã hóa khóa bí mật. Cụ thể, nhược điểm của mã hóa khóa công khai là có tốc độ mã hóa chậm, trong khi ưu điểm là không cần thống nhất khóa bí mật chung giữa người gửi và người nhận. Còn nhược điểm của mã hóa khóa bí mật là phải thống nhất trước khóa bí mật chung giữa người gửi và người nhận, trong khi ưu điểm là tốc độ mã hóa nhanh. Hệ mã hóa lai là tận dụng lợi thế của cả hai hệ mã hóa này, cụ thể khóa phiên làm việc *K* ngắn sẽ được mã hóa bằng hệ mã hóa khóa công khai có tốc độ chậm, còn dữ liệu dài sẽ được mã hóa bằng hệ mã hóa khóa bí mật có tốc độ nhanh dưới khóa phiên *K*. Như vậy, với đối với mã hóa lai giữa người gửi và người nhận không cần thống nhất trước khóa bí mật chung, dữ liệu được mã hóa bằng hệ mã hóa khóa bí mật.

Ở các tài liệu về mã hóa quảng bá [9, 10, 25, 26, 44, 46, 49, 55], để cho đơn giản người ta chỉ xét việc mã hóa và giải mã của khóa phiên làm việc *K*, do việc mã hóa và giải mã dữ liệu thực tế dùng *K* như là khóa bí mật là giống nhau ở tất cả các hệ mã hóa quảng bá.

1.1.3. Mô hình an toàn

Khi ta nói một hệ mã hóa là an toàn là ta nói một cách chung chung. Còn cụ thể, một hệ mã hóa được chứng minh là an toàn nếu ta cho kẻ tấn công có các quyền A, B, C,...Ví dụ: Với hệ mã RSA thì kẻ tấn công có thể có quyền biết các tham số công khai như tích của hai số nguyên tố *p* và *q*, thậm chí biết một số cặp khóa bí mật và công khai,...Và ta chứng minh về mặt toán học rằng để phá được hệ mã thì kẻ tấn công với các quyền A, B, C,... phải giải được một bài toán khó nào đó, ví dụ như bài toán phân tích ra thừa số nguyên tố hay bài toán logarit rời rạc. Với mã hóa quảng bá, quyền cơ bản nhất của kẻ tấn công là biết bản mã và khóa công khai. Ngoài ra, kẻ tấn công còn có thể có thêm các quyền khác như quyền biết khóa bí mật của các người

dùng không có khả năng giải mã (những người dùng nằm bên ngoài tập *S*), quyền tùy ý chọn một bản mã và biết bản rõ tương ứng,…

Khi mô hình hóa quyền của kẻ tấn công, ta gọi đó là mô hình an toàn, còn bài toán khó thì ta gọi là giả thuyết. Như vậy, nói một hệ mã hóa quảng bá an toàn một cách đầy đủ là phải nói nó an toàn dưới mô hình nào và dưới giả thuyết gì. Mô hình an toàn chuẩn cho một hệ mã hóa quảng bá được định nghĩa như sau:

Chúng ta xét một kịch bản giữa kẻ tấn công $\mathcal{A}$ và kẻ thách thức $\mathcal{C}$ (đại diện cho độ an toàn của hệ mã):

Khởi tạo (λ): Đầu tiên $\mathcal{C}$ chạy giải thuật khởi tạo **Khởi tạo** (λ) để tạo ra tham số công khai param, khóa bí mật msk của hệ thống. Tiếp theo, $\mathcal{C}$ công bố param cho $\mathcal{A}$ đồng thời giữ bí mật msk. Ngoài ra, $\mathcal{C}$ cũng khởi tạo ba danh sách rỗng list, Λ_C, Λ_D, trong đó list là danh sách các người dùng hiện thời đã được cấp khóa của hệ thống, Λ_C là danh sách các người dùng đã bị $\mathcal{A}$ biết khóa bí mật, còn Λ_D là danh sách các bản mã Hdr đã bị $\mathcal{A}$ biết bản rõ tương ứng.

Giai đoạn truy vấn 1: Kẻ tấn công $\mathcal{A}$ có thể tùy ý yêu cầu để biết các thông tin sau:

1. $\mathcal{A}$ yêu cầu được biết khóa bí mật của người dùng có định danh là *id*: Kẻ thách thức $\mathcal{C}$ chạy giải thuật **Tạo khóa** (msk, *id*, list, param):

Để tạo ra khóa bí mật SK_{id}, sau đó công bố SK_{id} cho $\mathcal{A}$, đồng thời thêm định danh *id* vào danh sách các người dùng đã bị $\mathcal{A}$ biết khóa bí mật Λ_C.

2. $\mathcal{A}$ yêu cầu được biết bản rõ tương ứng của bản mã Hdr (Hdr chứa cả *S*). $\mathcal{C}$ dùng khóa bí mật của người dùng với định danh $id' \in S$ để giải mã và gửi kết quả là bản rõ *K* lại cho $\mathcal{A}$. Hdr sau đó được thêm vào danh sách các bản mã Hdr đã bị $\mathcal{A}$ biết bản rõ tương ứng Λ_D.

Giai đoạn thách thức: Kẻ tấn công $\mathcal{A}$ cho đầu ra (gửi cho $\mathcal{C}$) là tập các người dùng S^* mà $\mathcal{A}$ sẽ tấn công. $\mathcal{C}$ chạy giải thuật **mã hóa** (S^*, param): Để thu về bản mã và bản rõ (khóa phiên) là (Hdr*, K^*). Tiếp theo $\mathcal{C}$ chọn ngẫu nhiên một bít $b \xleftarrow{\$} \{0,1\}$. Nếu $b = 1$, $\mathcal{C}$ chọn $K^* \xleftarrow{\$} \mathcal{K}$($\mathcal{K}$ là không gian của khóa phiên), còn nếu $b = 0$ thì $\mathcal{C}$ giữ nguyên

khóa K^*. Như vậy khóa K^* sẽ là số ngẫu nhiên không liên quan gì đến bản mã Hdr∗ nếu $b = 1$. Cuối cùng $\mathcal{C}$ công bố (Hdr∗, K^*) cho $\mathcal{A}$.

Giai đoạn truy vấn 2: Kẻ tấn công $\mathcal{A}$ tiếp tục có quyền yêu cầu biết các thông tin như trong giai đoạn truy vấn 1.

Giai đoạn dự đoán kết quả: Kẻ tấn công $\mathcal{A}$ đưa ra dự đoán bít $b' \in \{0,1\}$ cho bít b. Ta nói rằng kẻ tấn công $\mathcal{A}$ thắng trong kịch bản trên nếu như $b' = b$ và $S^* \cap \Lambda_C = \emptyset$ và (Hdr*, S^*) $\in \Lambda_D$. Ta ký hiệu $\mathbf{Succ}^{\mathrm{IND}}(\mathcal{A}) = \Pr[b' = b]$ là xác suất mà $\mathcal{A}$ thắng trong kịch bản trên, và lợi thế của nó là:

$$\begin{aligned}\mathbf{Adv}^{\mathrm{IND}}(\mathcal{A}) &= 2 \times \mathbf{Succ}^{\mathrm{IND}}(\mathcal{A}) - 1 \\ &= \Pr[1 \leftarrow \mathcal{A} \mid b = 1] - \Pr[1 \leftarrow \mathcal{A} \mid b = 0].\end{aligned} \tag{1.1}$$

Khái niệm an toàn ở trên gọi là an toàn không phân biệt được khóa, tức là kẻ tấn công không có khả năng phân biệt giữa một khóa phiên K đúng và một giá trị ngẫu nhiên. Một khái niệm an toàn yếu hơn an toàn không phân biệt được khóa, gọi là an toàn không tính toán được. Tức là kẻ tấn công chỉ không có khả năng tính ra được khóa phiên K, tuy nhiên, nó có thể có khả năng phân biệt giữa một khóa phiên K đúng và một giá trị ngẫu nhiên.

Ngoài ra, ta cũng phân biệt các khả năng của $\mathcal{A}$ như sau:

• Nếu kẻ tấn công $\mathcal{A}$ phải công bố tập các người dùng $\mathcal{A}$ muốn tấn công S^* trước khi giải thuật **khởi tạo** (λ) được chạy, thì kịch bản tấn công trên gọi là bảo mật có chọn lọc.

• Nếu kẻ tấn công $\mathcal{A}$ không cần công bố tập các người dùng $\mathcal{A}$ muốn tấn công S^* trước khi giải thuật **khởi tạo** (λ) được chạy, thì kịch bản tấn công trên gọi là bảo mật thích ứng. Tức là lúc này $\mathcal{A}$ sẽ có quyền lớn hơn.

• Nếu kẻ tấn công $\mathcal{A}$ có thể truy vấn để biết bản rõ tương ứng với một bản mã bất kỳ, thì kịch bản tấn công trên gọi là tấn công chọn trước bản mã. Nếu không thì kịch bản ở trên gọi là tấn công chọn trước bản rõ, tức là kẻ tấn công chọn bản rõ và biết bản mã tương ứng. Đây là điều là hiển nhiên vì kẻ tấn công biết khóa công khai param nên luôn có được khả năng này. Do đó, ta thấy rằng tấn công chọn trước bản mã viết tắt là CCA (Chosen Ciphertext Attack) sẽ mạnh hơn so với tấn công chọn trước bản

rõ viết tắt CPA (Chosen Plantext Attack). Vì quyền của kẻ tấn công trong CCA bao gồm cả quyền của kẻ tấn công trong CPA. Như vậy, kết hợp lại ta có các mô hình an toàn CCA/CPA.

1.2. Khái quát về một số hệ mã hóa quảng bá quan trọng và tình hình nghiên cứu hiện nay

Trong mục này, tác giả sẽ lần lượt giới thiệu một số hệ mã hóa quảng bá nền tảng quan trọng hiện nay. Các mã hóa này cùng các cải tiến của nó hiện đang được triển khai dùng rộng rãi trong thực tế với các ứng dụng như truyền hình trả tiền, chia sẻ files, radio quân đội, social network (facebooks, twitter,...).

Tình hình nghiên cứu trong nước

Mã hóa quảng bá đa kênh và mã hóa dựa trên thuộc tính vẫn là những hướng nghiên cứu mới hiện nay ở Việt Nam. Hiện nay, có nhiều công trình nghiên cứu về vấn đề này, trong đó có công trình viết "giới thiệu mã hóa dựa trên định danh" viết trên công thông tin điện tử https://nacis.gov.vn của tiến sĩ Hồ Văn Hương thuộc Ban Cơ yếu Chính phủ. Bên cạnh đó, Việt Nam cũng đã công bố chuẩn về mã hóa định danh ***TCVN11367-5:2018***.

Bài viết về "mã hóa dựa trên thuộc tính" của tác giả Phạm Quốc Hoàng viết tại tạp chí An toàn thông tin số 2, Năm 2010 và bài viết "mã hóa dựa trên định danh" của tác giả Trần Duy Lai viết trên tạp chí an toàn thông tin năm 2009. Ngoài ra còn có nhiều công trình khác viết về các hướng nghiên cứu này tại Việt Nam.

Tình hình nghiên cứu trên thế giới

Mã hóa quảng bá có thể được xây dựng trực tiếp từ các hệ mã hóa khóa công khai thông thường như RSA hay Elgamal. Có hai phương pháp cơ bản để xây dựng.

• **Phương pháp 1:** Giả sử m là thông tin người gửi muốn gửi đến n người dùng khác nhau. Để mã hóa m thông tin đó đến n người dùng thì khi đó sẽ có n bản mã và n khóa công khai. Nếu dùng hệ RSA với tham số an toàn là 80 thì bản mã để gửi thông tin m = 1024 bits sẽ ít nhất là $n \cdot 1024$ bits. Với băng thông hạn chế hiện nay, độ dài bản mã như vậy là rất lớn. Trên thực tế, các nhà nghiên cứu mong muốn bản mã phải có độ dài ngắn hơn nhiều so với giá trị n.

• **Phương pháp 2:** Gán cho mỗi nhóm người dùng một cặp khóa công khai/ bí mật chung, khi gửi thông tin *m* cho nhóm người dùng nào thì dùng khóa công khai của nhóm người dùng đó để mã hóa *m*. Với giải pháp như vậy, người dùng tham gia bao nhiêu nhóm thì sẽ phải lưu trữ bấy nhiêu khóa bí mật. Số khóa bí mật tối đa mà người dùng có thể sẽ phải lưu trữ lên tới con số rất lớn, điều này là không khả thi trong thực tế.

Hệ mã hóa quảng bá đầu tiên được giới thiệu bởi Fiat and Naor [28] với mục tiêu tạo ra một hệ mã hóa để giải quyết điểm yếu của 2 phương pháp trên về độ dài khóa bí mật và độ dài bản mã.

1.2.1. Hệ mã hóa NNL và các cải tiến

Như đã trình bày ở phần trên, hai phương pháp xây dựng trực tiếp mã hóa quảng bá từ các hệ mã hóa khóa công khai thông thường như RSA hay Elgamal sẽ dẫn đến trường hợp các hệ mã hóa quảng bá hoặc là có độ dài bản mã rất lớn, hoặc là có độ dài khóa bí mật rất lớn.

Để giải quyết vấn đề này, Naor và các tác giả [44] đề xuất hai hệ mã hóa có tốc độ lập mã và giải mã rất nhanh, đồng thời cân bằng được cả hai yếu tố là độ dài bản mã cũng như độ dài khóa bí mật. Điểm yếu của hai hệ mã này là cả hai hệ mã đều ở dạng hệ mã hóa khóa bí mật, tức là để thực hiện mã hóa thì người mã hóa cần phải biết khóa bí mật của người nhận. Như vậy, các hệ mã của Naor và các tác giả phù hợp với một số loại ứng dụng đặc thù như truyền hình trả tiền, radio quân đội, hay bảo vệ bản quyền trên các đĩa DVD,... Khi mà người lập mã là người chủ của toàn bộ hệ thống, do đó họ biết khóa bí mật của từng thành viên của hệ thống.

1.2.1.1. Hệ mã thứ nhất NNL-1[44]

Giả sử $\mathcal{N}$ là tập của tất cả những người dùng trong hệ thống, $|\mathcal{N}| = N$, và $\mathcal{R} \in \mathcal{N}$ là một tập con người dùng của $\mathcal{N}$ không có khả năng giải mã. Vậy tập người dùng $\mathcal{N}/\mathcal{R}$ chính là tập người dùng có quyền giải mã. Giả sử $|\mathcal{R}| = r$. Mục tiêu của hệ là cho phép người lập mã tạo ra bản mã mà chỉ những người dùng thuộc tập $\mathcal{N}/\mathcal{R}$ là có khả năng giải mã, còn tất cả các người dùng thuộc tập $|\mathcal{R}|$ dù có liên kết với nhau cũng không thể giải mã được.

Mã hóa NNL-1 được mô tả như sau:

Khởi tạo (λ):

Hệ thống chọn một hệ mã hóa khóa đối xứng AES để mã hóa. Giả sử $\mathcal{N}$ là tập của tất cả các người dùng trong hệ thống, $|\mathcal{N}| = N$. Tập người dùng này được bố trí trên một cây nhị phân có *N* lá, mỗi lá là một người dùng.

Tiếp theo, gán cho mỗi node trên cây nhị phân một khóa bí mật (gọi là khóa bí mật con). Như vậy, toàn hệ thống có $2N - 1$ khóa bí mật. Khóa bí mật của toàn hệ thống sẽ là tập $2N - 1$ khóa bí mật này.

Tạo khóa (msk, *id*):

Với mỗi người dùng là một lá *id*, khóa bí mật tương ứng cho người dùng đó sẽ là tập hợp các khóa bí mật con tương ứng với các node trên cây thuộc con đường từ lá (người dùng đó) tới node gốc của cây. Như vậy, mỗi người dùng sẽ có $logN+1$ khóa bí mật con do độ sâu của cây nhị phân N lá là $logN + 1$. Tức là SK_{id} sẽ bao gồm $logN + 1$ giá trị.

Mã hóa (S, param, msk):

Đầu vào của giải thuật là tập người dùng có khả năng giải mã *S* và khóa công khai, khóa bí mật của hệ thống. Đặt tập $|\mathcal{R}| = \mathcal{N}/S$, giả sử $|\mathcal{R}| = r$. Để mã hóa, trước tiên ta phân hoạch cây nhị phân *N* lá ở trên ra ω cây nhị phân con riêng biệt không giao nhau $(S_{i_1}, S_{i_2}, \dots, S_{i_\omega})$ sao cho $S_{i_1} \cup S_{i_2} \cup, \dots, \cup S_{i_\omega} = S = \mathcal{N}/\mathcal{R}$. Cách phân hoạch như sau, minh họa trên Hình 1.1:

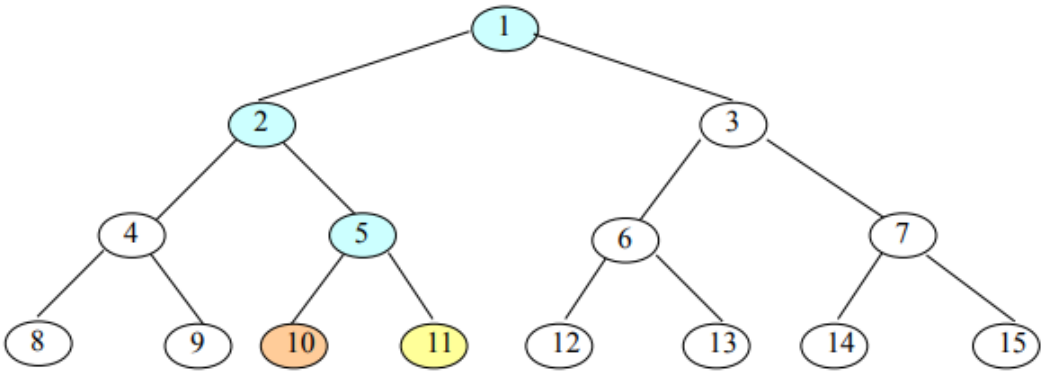
1. Tìm cây khung nhỏ nhất trong cây nhị phân đầy đủ có *N* lá ở trên chỉ chứa các lá có mặt trong tập $\mathcal{R}$ (cây này gọi là cây Steiner Tree ($\mathcal{R}$)).

2. Tập các cây con $(S_{i_1}, S_{i_2}, \dots, S_{i\omega})$ sẽ là các cây con có gốc là nodes anh em (liền kề) với các nodes trên cây khung nhỏ nhất trên. Ký hiệu các nodes gốc của các cây con này là $(S_{i_1}, S_{i_2}, \dots, S_{i\omega})$. Do mỗi nodes $(S_{i_1}, S_{i_2}, \dots, S_{i\omega})$ được gán tương ứng một khóa bí mật con, giả sử lần lượt là $(SK_{i_1}, SK_{i_2}, \dots, SK_{i\omega})$. Để mã hóa khóa phiên làm việc *K*, người lập mã sẽ mã hóa khóa phiên ω lần với hệ mã hóa khóa bí mật AES dưới ω khóa bí mật $(SK_{i_1}, SK_{i_2}, \dots, SK_{i\omega})$. Cụ thể bản mã sẽ là:

$$\text{Hdr} = \left(\left(AES\left(SK_{i_1}, K\right), S_{i_1}\right), \left(AES\left(SK_{i_2}, K\right), S_{i_2}\right), \ldots, \left(AES\left(SK_{i_\omega}, K\right), S_{i_\omega}\right)\right) \quad (1.2)$$

Giải mã (Hdr, SK_{id}, param):

Đầu vào của giải thuật là bản mã Hdr của khóa phiên làm việc K, khóa bí mật của người dùng SK_{id}, khóa công khai của hệ thống. Với mỗi người dùng *id* thuộc tập *S* không thuộc tập $\mathcal{R}$ tức là không thuộc cây Steiner Tree($\mathcal{R}$), thì sẽ thuộc một trong các tập con (S_{i_1}, $S_{i_2}, \ldots, S_{i\omega}$), là lá của một trong các cây con có nodes gốc là (S_{i_1}, $S_{i_2}, \ldots, S_{i\omega}$). Nghĩa là SK_{id} sẽ chứa một trong các khóa bí mật $\left(SK_{i_1}, SK_{i_2}, \ldots, SK_{i\omega}\right)$.



Hình 1.1: Hệ NNL-1

Lúc này, giải thuật trước tiên sẽ tìm node gốc $S_{i_j}, 1 \le j \le \omega$, sao cho *id* là lá trên cây con này, sau đó dùng khóa bí mật (SK_{i_j} để giải bản mã ($AES(SK_{i_j}, K)$ để thu lại khóa phiên *K*.

Lưu ý rằng: Nếu người dùng *id* không thuộc tập *S*, tức là thuộc tập $\mathcal{R}$ hay là thuộc cây ($\mathcal{R}$), thì sẽ không thuộc bất kỳ tập (S_{i_1}, $S_{i_2}, \ldots, S_{i_\omega}$) nào, tức là không có bất kỳ khóa $\left(SK_{i_1}, SK_{i_2}, \ldots, SK_{i_\omega}\right)$ nào, do đó không thể giải được mã.

Đánh giá độ hiệu quả: Người ta đã chứng minh được rằng $\omega = r \log \frac{N}{r}$, do vậy độ dài của khóa bí mật và độ dài của bản mã đều chỉ phụ thuộc vào *logN* chứ không phải tuyến tính với *N* như trong phương pháp thông thường. Ngoài ra, do dùng hệ mã AES trong mã hóa/giải mã nên tốc độ mã hóa/giải mã là hiệu quả.

1.2.1.2. Hệ mã thứ hai NNL-2 [44]

Ở hệ mã thứ nhất, độ dài của bản mã vẫn còn phụ thuộc vào tổng số người dùng trong hệ thống là N. Trong thực tế thì N sẽ khá lớn, do đó Naor và các tác giả [44] tiếp tục đề xuất một hệ mã thứ hai gọi là NNL-2. Với hệ mã này, độ dài bản mã chỉ là $2r - 1$ hoàn toàn không phụ thuộc vào tham số N, tuy nhiên độ dài khóa bí mật lúc này không còn là $logN + 1$ nữa mà sẽ có độ dài tuyến tính với log^2N.

Ý tưởng của tác giả lúc này là đánh số thứ tự các nodes trên cây nhị phân N lá. Quy định tập $S_{i,j}$ bao gồm các nodes lá là con cháu của node i nhưng không phải là con cháu của nodes j. Ví dụ, trên Hình 1.2 tập $S_{2,10}$ chứa các nodes 8, 9, 11 là con cháu của node 2 nhưng không phải là con cháu của node 10. Lúc này gán khóa bí mật của nhóm $S_{i,j}$ cho tất cả các lá là con cháu của node i nhưng không phải là con cháu của node j. Như vậy, có thể dễ dàng thấy rằng số khóa bí mật mà mỗi người dùng phải lưu trữ sẽ tuyến tính với N.

Để giảm độ dài của khóa bí mật của người dùng Naor và các tác giả đã áp dụng hàm tạo số giả ngẫu nhiên G để tạo khóa cho các tập $S_{i,j}$. Cụ thể, gán cho mỗi node i trên cây nhị phân N lá một nhãn L_i . Dùng ba hàm tạo số giả ngẫu nhiên sau:

• Hàm G_L tạo ra nhãn cho node con trái của node i.

• Hàm G_R tạo ra nhãn cho node con phải của node i.

• Hàm G_M tạo ra khóa bí mật từ nhãn tại node đó.

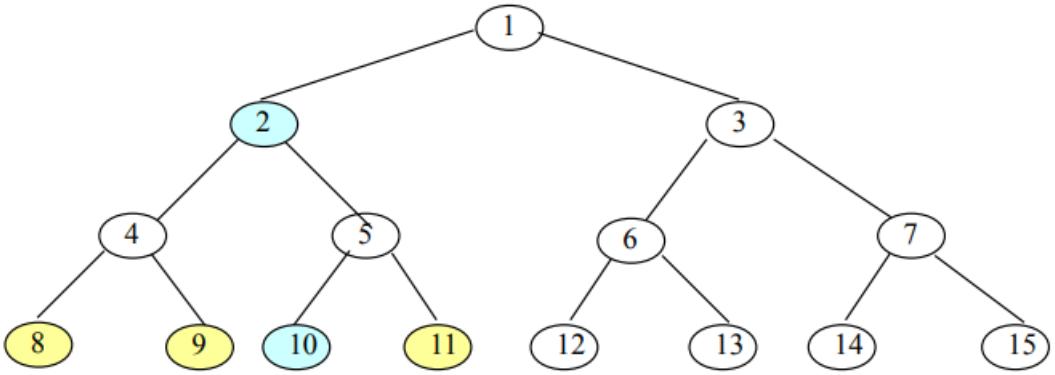
Xét cây con $S_{i,j}$, nhãn của các nodes trong cây con này được sinh ra từ nhãn L_i của node gốc i và các hàm G_L, G_R. Khóa của nhóm $S_{i,j}$ sẽ được tính bởi $G_M(L_{i,j})$ trong đó $L_{i,j}$ là nhãn của node j sinh ra từ L_i và các hàm G_L, G_R. Ví dụ nếu node j là 10 và node i là 2 trên hình vẽ thì $L_{2,10} = G_L(G_R(L_2))$.

Với cách gán như vậy mỗi node là lá và là con cháu của cả i và j (ví dụ node 10) sẽ nhận nhãn của tất cả các node là anh chị em của ông cha của nó trên đường từ node lá đó đến node gốc. Ví dụ ở trên thì node 10 sẽ nhận nhãn của các nodes 4 và 11. Lưu ý rằng, nhãn của các nodes 4 và 11 được sinh ra từ L_2 và các hàm G_L, G_R. Ngoài ra do node 10 không có nhãn của $L_{2,10}$ nên không thể có khóa bí mật của nhóm $S_{2,10}$, trong khi nó có khóa bí mật của các nhóm khác có gốc tại node 2 như $S_{2,4}$, $S_{2,11}$.

Như vậy, khi mã hóa ta chỉ việc phân hoạch người dùng thành các nhóm $S_{i,,j}$ phù hợp, trong đó người dùng không có quyền giải mã sẽ là con cháu của node *j*, sau đó mã hóa khóa phiên *K* bằng khóa bí mật của các nhóm $S_{i,j}$ này.

Về độ hiệu quả, do số lượng các nhãn mà mỗi node lá thuộc cây con $S_{i,,j}$ phải lưu chỉ là độ sâu của cây $S_{i,,j}$ và số lượng node *i* là *logN* nên số nhãn tối đa mà mỗi node lá phải lưu chỉ là $O(log^2N)$, hay độ dài khóa bí mật chỉ là $O(log^2N)$. Người ta cũng đã chứng minh được rằng, số lượng tối đa các nhóm $S_{i,j}$ sau khi phân hoạch chỉ là $2r - 1$ với *r* là số người dùng không có quyền giải mã.

Về tốc độ tính toán thì hệ NNL-2 kém hơn so với hệ NNL-1 vì ở hệ NNL-2 người dùng khi giải mã cần bước tính, tính ra khóa bí mật từ nhãn. Tóm lại, với hệ NNL-2 ta giảm được độ dài bản mã nhưng lại tăng thêm độ dài khóa bí mật và tốc độ giải mã.



Hình 1.2 Hệ NNL-2

Chuyển đổi NNL-1 sang hệ mã hóa khóa công khai [26]:

Hệ mã hóa quảng bá NNL-1 ở trên có nhược điểm chính đó là để mã hóa người lập mã cần phải biết khóa bí mật của tất cả các người dùng. Nhược điểm này không phù hợp với nhiều loại ứng dụng như: Chia sẻ files, mạng xã hội, lưu trữ dữ liệu an toàn,.... Để khắc phục nhược điểm này, các tác giả [26] đã dựa vào kỹ thuật mã hóa dựa trên định danh để chuyển đổi hệ NNL-1 thành hệ mã hóa khóa công khai.

Nhược điểm của các hệ mã hóa khóa công khai như RSA hay Elgamal là khóa công khai chỉ là một con số ngẫu nhiên, không có thông tin gì đặc biệt để đại diện cho chủ thể sở hữu nó. Do vậy, các hệ này cần phải có một trung tâm chứng thực số viết tắt là CA (Center Authority) hay rộng hơn là cơ sở hạ tầng khóa công khai viết tắt là PKI (Public Key Infrastructure) để chứng thực khóa công khai thông qua chứng thực số, điều này dẫn đến tốn kém cho hệ thống. Vì vậy, ý tưởng để khắc phục điểm yếu này được giới thiệu lần đầu bởi Shamir [54], nếu ta dùng khóa công khai bao gồm các thông tin đặc biệt gắn liền với chủ thể sở hữu nó thì ta không cần phải chứng thực khóa công khai.

Chẳng hạn, có thể dùng số chứng minh thư, số hộ chiếu hay địa chỉ email của Bob để làm khóa công khai. Với khóa công khai như vậy, chúng ta không còn cần một CA để đảm bảo rằng khóa công khai này thuộc về Bob. Do đó, vấn đề ở đây không còn cần phải có chứng thực hay không còn cần phải xây dựng PKI nữa, Alice có thể dễ dàng nhận ra rằng đây chính là khóa công khai của Bob. Và để từ một địa chỉ email hay số chứng minh thư bất kỳ dùng làm khóa công khai có thể tạo ra được một khóa bí mật tương ứng. Trong mô hình kỹ thuật mã hóa dựa trên định danh, cần phải có một thực thể làm công việc đó gọi là viết tắt là PKG (Private Key Generator).

Ứng dụng ý tưởng của kỹ thuật mã hóa dựa trên định danh, các tác giả [26] gán cho mỗi node trên cây nhị phân của hệ NNL-1 một bít 0 hoặc 1, do đó mỗi node i sẽ có một định danh ID_i khác nhau, đó là chuỗi bít từ node gốc của cây đến node i. Dùng kỹ thuật mã hóa dựa trên định danh mỗi node i sẽ có một khóa bí mật tương ứng với định danh ID_i đó. Vì vậy, mỗi node bây giờ có hai khóa, khóa công khai chính là định danh ID_i, và khóa bí mật tương ứng với định danh đó. Hệ mã lúc này có thể được mô tả như sau:

Khởi tạo (λ):

Hệ thống chọn một mã hóa dựa trên định danh viết tắt là IBE (Identity-Based Encryption) có khóa bí mật của hệ thống là msk. Giả sử N là tập của tất cả các người dùng trong hệ thống, $|\mathcal{N}| = N$. Tập người dùng này được bố trí trên một cây nhị phân có N lá, mỗi lá là một người dùng.

Tiếp theo gán cho mỗi node trên cây nhị phân một bít 0 hoặc bít 1. Định danh ID_i của node *i* khác nhau sẽ khác nhau, và đó là chuỗi bít từ node gốc của cây đến node *i*.

Tạo khóa (msk, *id*):

Với mỗi node có định danh ID_i, dùng giải thuật tạo khóa của hệ IBE với khóa bí mật hệ thống là msk để tạo ra khóa bí mật SK_{ID_i}. Mỗi người dùng *id* sẽ nhận tất cả khóa của các nodes trên đường từ node lá *id* tới node gốc của cây nhị phân, tương tự như hệ NNL-1.

Mã hóa (*S*, param):

Đầu vào của giải thuật là tập người dùng có khả năng giải mã *S* và khóa công khai của hệ thống. Đặt tập $|\mathcal{R}| = \mathcal{N}/S$, giả sử $|\mathcal{R}| = r$.

Bước đầu tiên, phân hoạch cây nhị phân N lá ở trên ra ω cây nhị phân con riêng biệt không giao nhau $(S_{i_1}, S_{i_2}, \dots, S_{i\omega})$ sao cho $S_{i_1} \cup S_{i_2} \cup, \dots, \cup S_{i_\omega} = S = \mathcal{N}/\mathcal{R}$. Cách phân hoạch minh họa trên Hình 1.1

1. Tìm cây khung nhỏ nhất trong cây nhị phân đầy đủ, có *N* lá ở trên mà chỉ chứa các lá có mặt trong tập $\mathcal{R}$ (cây này gọi là cây Steiner Tree ($\mathcal{R}$)).

2. Tập các cây con $(S_{i_1}, S_{i_2}, \dots, S_{i\omega})$ sẽ là các cây con có gốc là nodes anh em (liền kề) với các nodes trên cây khung nhỏ nhất trên. Ký hiệu các nodes gốc của các cây con này là $(S_{i_1}, S_{i_2}, \dots, S_{i\omega})$.

Mỗi nodes $(S_{i_1}, S_{i_2}, \dots, S_{i\omega})$ có tương ứng một định danh $ID_{i_1}, ID_{i_2}, .., ID_{i_\omega}$ được công bố công khai. Do đó, người lập mã sẽ dùng hệ IBE để mã hóa khóa phiên K dưới các định danh này. Tức là người lập mã sẽ mã hóa khóa phiên ω lần với hệ mã hóa IBE dưới ω định danh $ID_{i_1}, ID_{i_2}, \dots, ID_{i_\omega}$. Cụ thể bản mã sẽ là:

$$\text{Hdr} = \left(\left(IBE\left(ID_{i_1}, K\right), S_{i_1}\right), \left(IBE\left(ID_{i_2}, K\right), S_{i_2}\right), \dots, \left(IBE\left(ID_{i_\omega}, K\right), S_{i_\omega}\right)\right)$$

Lưu ý: Giải thuật mã hóa lúc này không cần biết khóa bí mật của hệ thống **msk**.

Giải mã (Hdr, SK_{id}, param):

Đầu vào của giải thuật là bản mã Hdr của khóa phiên làm việc *K*, khóa bí mật của người dùng SK_{id}, khóa công khai của hệ thống param. Với mỗi người dùng *id*

thuộc tập *S* không thuộc tập $\mathcal{R}$ tức là không thuộc cây ($\mathcal{R}$), thì sẽ thuộc một trong các tập con $(S_{i_1}, S_{i_2}, \dots, S_{i\omega})$, là lá của một trong các cây con có nodes gốc là $(S_{i_1}, S_{i_2}, \dots, S_{i\omega})$. Nghĩa là SK_{id} sẽ chứa một trong các khóa bí mật $\left(SK_{ID_{i_1}}, SK_{ID_{i_2}}, \dots, SK_{ID_{i_\omega}}\right)$. Lúc này, giải thuật trước tiên sẽ tìm node gốc $S_{i_j}, 1 \leq j \leq \omega$ sao cho *id* là lá trên cây con này, sau đó dùng khóa bí mật $SK_{ID_{i_j}}$ để giải bản mã *(IBE* $(SK_{ID_{i_j}}, K)$ để tính khóa phiên *K*.

Lưu ý rằng: Nếu người dùng *id* không thuộc tập *S*, tức là thuộc tập $\mathcal{R}$ hay là thuộc cây ($\mathcal{R}$), thì sẽ không thuộc bất kỳ tập $(S_{i_1}, S_{i_2}, \dots, S_{i\omega})$ nào, tức là không có bất kỳ khóa $(SK_{ID_{i_1}}, SK_{ID_{i_2}}, \dots, SK_{ID_{i_\omega}})$ nào, do đó không thể giải được mã. Với hệ trên ta thấy độ dài khóa bí mật lúc này vẫn là *O(logN* + 1*)*, độ dài bản mã vẫn tuyến tính với $r \log \frac{N}{r}$.

Chuyển đổi NNL-1 sang hệ mã hóa khóa công khai đồng thời rút ngắn độ dài khóa bí mật:

Phan và Trinh [49] mở rộng hơn khi chuyển đổi hệ NNL-1 sang dạng mã hóa quảng bá dựa trên định danh tương tự như [26] nhưng có độ dài khóa bí mật lúc này chỉ là hằng số, không phụ thuộc vào r và N. Kỹ thuật của họ là xây dựng một lược đồ mã hóa mới gọi là GWIBE là mở rộng của hệ IBE có tính chất như sau:

• Nếu mã hóa với định danh là *ID* = (0, 1, ∗, 1) thì bất kỳ khóa bí mật nào tương ứng với định danh *ID* = (0, 1, 0, 1) hoặc *ID* = (0, 1, 1, 1) đều có thể giải mã được.

• Ký tự ∗ có nghĩa là đại diện cho bất kỳ bits nào.

Với sự mở rộng của hệ IBE như vậy, khi thay thế IBE bằng GWIBE trong hệ NNL-1 ở trên Phan và Trinh [49] đã xây dựng được lược đồ mã hóa mới có độ dài khóa bí mật chỉ gồm 3 phần tử.

1.2.2. Hệ mã hóa BGW và các cải tiến

Mã hóa quảng bá BGW [9] và một số cải tiến của hệ [10, 29, 46]. Hệ mã hóa BGW đến nay đã được áp dụng rộng rãi trong thực tế, đặc biệt là trong lĩnh vực truyền

hình trả tiền. Phần đầu tiên, tác giả sẽ trình bày về công cụ ánh xạ song tuyến và giả thuyết bài toán khó viết tắt là (DBDHE) dùng để xây dựng nên hệ mã BGW này.

1.2.2.1. Công cụ ánh xạ song tuyến

Giả sử $\mathbb{G}$, $\widetilde{\mathbb{G}}$ và $\mathbb{G}_T$ là ba nhóm Abelian hữu hạn có bậc là số nguyên tố $p > 2^{\lambda}$ trong đó λ là tham số an toàn và g, g̃ là phần tử sinh của hai nhóm $\mathbb{G}$, $\widetilde{\mathbb{G}}$ một cách tương ứng.

Một hàm $e: \mathbb{G} \times \widetilde{\mathbb{G}} \to \mathbb{G}_T$ gọi là một ánh xạ song tuyến nếu những điều kiện sau là thỏa mãn với mọi $a, b \in \mathbb{Z}_p$:

1. $e(g^a, \tilde{g}^b) = e(g, \tilde{g})^{ab}$
2. $e(g^a, \tilde{g}^b) = 1$ nếu a = 0 hay b = 0
3. $e(g^a, \tilde{g}^b)$ có thể được tính toán một cách hiệu quả.

$(p, \mathbb{G}, \widetilde{\mathbb{G}}, \mathbb{G}_T, e)$ được gọi là một hệ thống ánh xạ song tuyến. Người ta chia hệ thống này ra làm ba loại:

1. Pairings loại 1 nếu $\mathbb{G} = \widetilde{\mathbb{G}}$.

2. Pairings loại 2 nếu $\mathbb{G} \neq \widetilde{\mathbb{G}}$, và có một hàm ánh xạ $\emptyset : \widetilde{\mathbb{G}} \to \mathbb{G}$ được tính toán một cách hiệu quả.

3. Pairings loại 3 nếu $\mathbb{G} \neq \widetilde{\mathbb{G}}$, và không có một hàm ánh xạ $\emptyset : \widetilde{\mathbb{G}} \to \mathbb{G}$ được tính toán một cách hiệu quả.

Lưu ý rằng, khi cài đặt cụ thể ba loại Parings trên bằng đường cong Eliptics thì cài đặt Parings loại 3 hiện nay là hiệu quả nhất.

Giả thuyết DBDHE:

Được phát biểu dưới dạng một bài toán, trong đó kẻ tấn công $\mathcal{A}$ cố phải giải bài toán này trong thời gian *t*, hệ mã hóa BGW [9] được chứng minh là an toàn trong mô hình CPA và dưới giả thuyết DBDHE, trong đó giả thuyết DBDHE được phát biểu như sau:

Định nghĩa 2.1. Giả thuyết (t, n, ε) − DBDHE phát biểu rằng:

Đối với mọi kẻ tấn công $\mathcal{A}$ trong thời gian t cho trước có đầu vào là: $\left(\mathrm{g}, \mathrm{h}, \mathrm{g}^{\alpha^1}, \ldots, \mathrm{g}^{\alpha^n}, \mathrm{g}^{\alpha^{n+2}}, \ldots, \mathrm{g}^{\alpha^{2n}}\right) \in \mathbb{G}^{2n+1}$ và $\mathrm{T} \in \mathbb{G}_{\mathrm{T}}$, $\mathcal{A}$ phải kiểm tra xem T có bằng $\mathrm{e}(\mathrm{g}, \mathrm{h})^{\alpha^{n+1}} \in \mathbb{G}^{\mathrm{T}}$ không.

Lưu ý rằng từ $\left(\mathrm{g}, \mathrm{h}, \mathrm{g}^{\alpha^1}, \ldots, \mathrm{g}^{\alpha^n}, \mathrm{g}^{\alpha^{n+2}}, \ldots, \mathrm{g}^{\alpha^{2n}}\right) \in \mathbb{G}^{2n+1}$, $\mathcal{A}$ không thể trực tiếp tính được giá trị $\mathrm{e}(\mathrm{g}, \mathrm{h})^{\alpha^{n+1}}$.

Như vậy, kẻ tấn công $\mathcal{A}$ với đầu vào cho trước phải đi phân biệt giữa hai phần tử $e(g,h)^{\alpha^{n+1}} \in \mathbb{G}^T$ và T ngẫu nhiên. Hiện nay, vẫn chưa tồn tại thuật toán hiệu quả nào cho $\mathcal{A}$ để giúp $\mathcal{A}$ phân biệt hiệu quả hai phần tử đó. Hay có thể hiểu rằng, giả thuyết DBDHE là một bài toán khó vẫn chưa có lời giải. Khi nói mã hóa BGW an toàn dưới giả thuyết DBDHE tức là hiểu hệ mã hóa BGW chỉ bị phá khi mà có lời giải đúng cho giả thuyết DBDHE ở trên.

Hệ mã hóa BGW [9] được trình bày như sau:

Khởi tạo (λ):

Giả sử $\mathbb{G}$ là một hệ thống ánh xạ song tuyến có bậc là số nguyên tố *p*. Giải thuật chọn phần tử sinh $g \in \mathbb{G}$ và một số mũ ngẫu nhiên $\alpha \in \mathbb{Z}_p$.

Tính $g_i = g^{\alpha^i} \in \mathbb{G}$ với mọi *i* = 1, 2,...*n*, *n* + 2,..., 2*n*. Tiếp theo, giải thuật chọn số mũ ngẫu nhiên $\gamma \in \mathbb{Z}_p$ và đặt $v = g^{\gamma} \in \mathbb{G}$. Khóa công khai của hệ thống là:

$$\text{Param} = (g_1, \ldots, g_n, g_{n+2}, \ldots, g_{2b}, v)$$

Khóa bí mật của hệ thống là msk = α.

Tạo khóa (msk, *i*):

Với đầu vào là khóa bí mật của hệ thống và người dùng thứ *I*, khóa bí mật tương ứng với người dùng thứ *i* sẽ là:

$$d_i = v^{\alpha^i}$$

Mã hóa (*S*, param):

Đầu vào của giải thuật là tập người dùng *S* có khả năng giải mã. Giải thuật chọn số mũ ngẫu nhiên $r \in \mathbb{Z}_p$, đặt khóa phiên $K = e(g_{n+1}, g)^r$ trong đó $e(g_{n+1}, g)$ có thể được tính từ $e(g_n, g_1)$ ở trong khóa công khai của hệ thống.

Tiếp theo, đặt:

$$\text{Hdr} = (g^r, (v \cdot \prod_{j \in S} g_{n+1-j})^r) \quad (1.4)$$

Đầu ra của giải thuật là (Hdr, *K*). Lưu ý rằng, Hdr bao gồm cả tập *S* và như đã trình bày ở mục trước, ta chỉ quan tâm đến việc mã hóa và giải mã khóa phiên *K*.

Giải mã *(Hdr, i, d_i,* param):

Giả sử Hdr = (C_1, C_2), giải thuật với đầu vào là bản mã Hdr và khóa bí mật d_i, khóa công khai param cho đầu ra là khóa phiên *K* như sau:

$$\text{K} = \frac{e(g_i, C_2)}{e\left(d_i \cdot \prod_{j \in S, j \neq i} g_{n+1-j+i}, C_1\right)}$$

Đánh giá độ hiệu quả:

1. Độ dài bản mã chỉ bao gồm 2 phần tử:

$$(g^r \; v \text{à} \left(v \cdot \prod_{j \in S, j \neq i} g_{n+1-j}\right)^r$$

2. Độ dài khóa bí mật dù chỉ bao gồm một phần tử $d_i = v^{\alpha^i}$ nhưng khi giải mã, người giải mã vẫn cần phải biết khóa công khai của hệ thống. Có nghĩa là độ dài của khóa giải mã phải bao gồm cả độ dài của khóa công khai, do độ dài của khóa công khai là tuyến tính với tổng số người dùng trong hệ thống, nên đây vẫn là điểm yếu và hạn chế còn tồn tại của hệ mã này.

3. Quá trình mã hóa và giải mã đều ổn định và hiệu quả.

1.2.2.2. Một số cải tiến của hệ mã BGW [46, 10, 29]

BGW là một trong những hệ mã hóa quảng bá quan trọng và được sử dụng rộng rãi trong thực tế hiện nay. Một số cải tiến của hệ mã này như sau:

• Trong bài báo [46], các tác giả đã vận dụng tính chất của hàm băm để cải thiện độ an toàn của hệ BGW. Cụ thể, các tác giả đã nâng cao được độ an toàn của hệ BGW từ mô hình bảo mật chọn lọc lên mô hình bảo mật thích ứng. Tuy nhiên, lại tồn tại những nhược điểm về độ dài của khóa bí mật lớn, bản mã lớn, tốc độ mã hóa và giải mã kém hiệu quả hơn so với hệ mã gốc BGW.

• Trong bài báo [10], các tác giả đã vận dụng công cụ mới multilinear maps để rút ngắn độ dài khóa giải mã của hệ BGW. Điểm yếu nêu trên của hệ mã BGW được khắc phục và cải tiến trong tài liệu [46]. Độ an toàn của công cụ multilinear maps cho

đến thời điểm hiện nay vẫn đang còn tranh cãi, chưa đưa ra được kết luận cuối cùng. Ngoài ra, hạn chế của cải tiến này là tốc độ mã hóa và giải mã của hệ thống sẽ kém hơn nhiều so với hệ BGW gốc.

• Trong bài báo gần đây [29], các tác giả đã cải tiến hệ BGW ở hai điểm:

Thứ nhất là, đã nâng cao độ an toàn của hệ BGW từ mô hình an toàn yếu lên mô hình an toàn mạnh.

Thứ hai là, an toàn của hệ mã bây giờ không dựa vào giả thuyết DBDHE mà dựa vào giả thuyết tốt hơn là giả thuyết *k-Linear (*được hiểu là bài toán khó k-Linear là *khó* hơn bài toán khó DBDHE*).* Tuy nhiên, hạn chế của hệ mã này vẫn tồn tại, đó là có độ dài khóa công khai dài hơn, tức là độ dài khóa giải mã cũng sẽ dài hơn.

1.2.3. Hệ mã hóa Delerablee và các cải tiến

Một mã hóa quảng bá quan trọng khác là mã hóa Delerablee [25], so với hệ BGW hệ mã hóa Delerablee có những ưu và nhược điểm sau:

Về ưu điểm:

1. Rút ngắn được độ dài khóa công khai của hệ thống ngắn hơn so với BGW. Cụ thể, độ dài của khóa công khai trong hệ Delerablee chỉ tuyến tính với số tối đa người dùng có thể giải mã cho mỗi lần mã hóa, thay vì tuyến tính với tổng số người dùng trong hệ thống như hệ BGW.

2. Là hệ mã hóa quảng bá dựa trên định danh. Có nghĩa là người dùng lúc này có thể dùng bất kỳ thông tin gì gắn với mình để làm khóa công khai. Ví dụ, địa chỉ email, số chứng minh thư, ... do đó, hệ mã không cần đến trung tâm chứng thực khóa công khai trong hệ thống.

Về nhược điểm:

1. Mô hình an toàn của hệ Delerablee kém hơn. Cụ thể, hệ Delerablee cần dùng bộ tiên tri ngẫu nhiên trong chứng minh an toàn. Hệ Delerablee được chứng minh là an toàn dựa vào một giả thuyết bài toán khó yếu hơn so với giả thuyết DBDHE trong hệ BGW, cụ thể hệ Delerablee dựa vào giả thuyết GDDHE.

Giả thuyết GDDHE dùng trong hệ Delerablee được trình bày như sau:

Định nghĩa 2.2. Bài toán khó GDDHE: Giả sử $(p, \mathbb{G}, \widetilde{\mathbb{G}}, \mathbb{G}_T, e)$ là một hệ thống ánh xạ song tuyến, trong đó f và g là hai đa thức nguyên tố cùng nhau bậc t và n. Giả sử $g_0 \in \mathbb{G}$, $h_0 \in \widetilde{\mathbb{G}}$ là hai phần tử sinh. Cho trước:

$$g_0, g_0^{\gamma}, \dots, g_0^{\gamma^{t-1}}, g_0^{\gamma.f(\gamma)}, g_0^{k\gamma.f(\gamma)},$$

$$h_0, h_0^{\gamma}, \dots, h_0^{\gamma^{2n}} h_0^{k.g(\gamma)},$$

và $T \in \mathbb{G}_T$ hãy phân biệt T là $e(g_0, h_0)^{k.f(\gamma)}$ hay là phần tử ngẫu nhiên.

Hệ Delerablee được mô tả như sau:

Khởi tạo (1^{λ}, m): Giả sử rằng λ là tham số an toàn, *m* là số tối đa người dùng có thể giải mã trong một lần mã hóa. $D = (p, \mathbb{G}, \widetilde{\mathbb{G}}, \mathbb{G}_T, e)$ là một hệ thống ánh xạ song tuyến. Giả sử $h \xleftarrow{\$} \widetilde{\mathbb{G}}$, $g \xleftarrow{\$} \mathbb{G}$ là hai phần tử sinh và $\alpha \xleftarrow{\$} \mathbb{Z}_p^*$. Giả sử $\mathcal{H}$ là hàm băm $\mathcal{H}: \{0,1\}^* \to \mathbb{Z}_p^*$. Khóa công khai của hệ thống:

$$\text{param} = \left(D, \left\{h^{\alpha^i}\right\}_{i=0,\dots,m,}\ g^{\alpha}, v = e(g,h), \mathcal{H}\right)$$

Khóa bí mật của hệ thống:

$$\text{msk} = (g, \alpha)$$

Tạo khóa (ID_i, msk, param):

Giả sử rằng $ID_i \in \{0,1\}^*$ là định danh của người dùng. Khóa bí mật tương ứng với định danh này là:

$$sk_{ID_i} = g^{\frac{1}{\alpha + \mathcal{H}(ID_i)}}$$

Mã hóa (param, *S*):

Không mất tính tổng quát, giả sử rằng tập người dùng có khả năng giải mã *S* là bao gồm các chỉ số *i* sao cho người dùng với định danh ID_i có khả năng giải mã được. Giải thuật chọn ngẫu nhiên giá trị $k \in \mathbb{Z}_p^*$ tính khóa phiên *K* :

$$K = e(g,h)^k$$

tiếp theo tính bản mã Hdr = (C_1, C_2) như sau:

$$C_1 = g^{-\alpha.k},\ C_2 = h^{k.\prod_{i \in S}(\alpha + H(ID_i))}$$

Cuối cùng giải thuật cho đầu ra là *K* và Hdr = (C_1, C_2) bao gồm cả tập chỉ số *S*.

Giải mã (sk_{ID_i}, Hdr, param):

Giải thuật đầu tiên kiểm tra rằng $i \in S$ có đúng không. Nếu không báo là không giải mã thành công. Ngược lại, giải thuật tính $K' = h^{\gamma}$ trong đó:

$$\left(\gamma = \frac{1}{\alpha} \prod_{\substack{i' \in S \\ i' \neq i}} (\alpha + \mathcal{H}(ID_{i'})) - \prod_{\substack{i' \in S \\ i' \neq i}} \mathcal{H}(ID_{i'}) \right) \tag{1.5}$$

Chú ý rằng, có thể tính K' từ param. Đặt:

$$B = \prod_{\substack{i' \in S \\ i' \neq i}} H(ID_{i'}) \tag{1.6}$$

Giải thuật cuối cùng cho đầu ra là:

$$K = \left(e(C_1, K') \, . e(sk_{ID_i}, C2)\right)^{\frac{1}{B}} \tag{1.7}$$

Tính đúng đắn:

$$K = \left(e(C_1, K') . e(sk_{ID_i}, C_2)\right)^{\frac{1}{B}} \tag{1.8}$$

$$= \left(e(g^{-\alpha.k}, h^{\gamma}) \, . e\left(g^{\frac{1}{\alpha + \mathcal{H}(ID_i)}}, h^{k.\prod_{i \in S}(\alpha + \mathcal{H}(ID_i))}\right)\right)^{\frac{1}{B}} \tag{1.9}$$

$$= \left(e(g,h)^{k \prod_{\substack{i' \in S \\ i' \neq i}} \mathcal{H}(ID_{i'})} \right)^{\frac{1}{B}} \tag{1.10}$$

$$= e(g,h)^k \tag{1.11}$$

Các cải tiến của hệ Delerablee:

1. Trong bài báo [55], các tác giả đã cải tiến hệ Delerablee bằng cách phân định rõ hai vấn đề, thứ nhất là người mã hóa dữ liệu, thứ hai là người có khả năng tước bỏ quyền giải mã của một số người dùng. Như vậy, người mã hóa cứ việc mã hóa dữ liệu và cung cấp cho người có khả năng tước bỏ quyền giải mã. Người có khả năng tước quyền giải mã, sẽ tùy thuộc vào tình huống cụ thể mà cho phép ai được quyền giải mã và ai không được quyền giải mã. Ngoài ra, người có khả năng tước bỏ quyền giải mã không có khả năng giải mã trừ khi người lập mã cho phép.

2. Trong bài báo [56], các tác giả đã cải tiến hệ Delerablee bằng cách giảm bớt sự phức tạp tính toán cũng như băng thông cho người dùng giải mã. Cụ thể, người dùng giải mã có thể nhờ bên thứ ba là một bên có khả năng tính toán tốt nhưng không được tin tưởng (ví dụ public server) để giải mã một phần bản mã, sau đó gửi kết quả trả về cho người dùng, từ đó người dùng giải tiếp bản mã mà máy chủ công cộng không biết gì về kết quả cuối cùng là bản rõ. Ngoài ra, trong một số ứng dụng cụ thể để giảm băng thông gửi đến người dùng, người lập mã có thể gửi trực tiếp bản mã đến máy chủ, sau đó máy chủ đó sẽ giải mã một phần rồi gửi kết quả cho người dùng. Băng thông để gửi kết quả sau khi giải mã một phần sẽ ít hơn nhiều so với gửi toàn bộ bản mã ban đầu.

1.3. Tình hình nghiên cứu hiện nay của Mã hóa quảng bá

Hệ NNL [44] và các cải tiến [26, 49]: Là mã hóa quảng bá khóa bí mật (tức là chỉ có ai biết khóa bí mật của người dùng mới có thể thực hiện được việc lập mã) dựa trên cấu trúc cây nhị phân, trong đó người dùng là các lá ở trên cây. Hệ NNL dùng hệ mã hóa khóa bí mật (ví dụ AES) để mã hóa và giải mã nên tốc độ mã hóa và giải mã rất nhanh. Độ dài của bản mã và khóa bí mật là $r \cdot logN$ và $logN$ với hệ NNL-1; 2r-1 và log^2N với hệ NNL-2, trong đó N là số tối đa người dùng trong hệ thống và r là số người dùng không có khả năng giải mã đối với bản mã đó.

Một số cải tiến đối với hệ NNL như tác giả Dodis và Fazio viết trong tài liệu [26], sử dụng kỹ thuật mã hóa dựa trên định danh, chuyển cả NNL-1 và NNL-2 sang mã hóa quảng bá khóa công khai (tức là ai cũng có thể thực hiện việc mã hóa, sau này gọi tắt là mã hóa quảng bá), nhưng với chi phí phải trả khi ứng dụng và triển khai trong thực tế, thì hệ trở nên kém hiệu quả hơn do dùng IBE. Thay vì dùng hệ mã hóa khóa đối xứng để mã hóa và giải mã, nhóm tác giả Phan và Trinh [49] mở rộng hơn khi chuyển đổi NNL-1 sang dạng mã hóa quảng bá dựa trên định danh và độ dài khóa bí mật lúc này chỉ là hằng số, không phụ thuộc vào r và N. Với việc dựa trên định danh, lúc này khóa công khai của mỗi người dùng trong hệ thống không còn là một con số ngẫu nhiên nữa, mà nó gắn liền với một định danh cụ thể của người dùng đó,

ví dụ như số chứng minh thư hay địa chỉ email (hệ thống không còn cần dùng cơ sở hạ tầng khóa công khai để cấp chứng thư số cho khóa công khai của người dùng).

Truy vết: Tập người dùng hợp lệ được cấp quyền có thể dùng khóa bí mật của mình để tạo ra các thiết bị giải mã không hợp pháp, sau đó có thể tiết lộ khóa bí mật cho người dùng không được cấp phép. Để giải quyết vấn đề này, cấp thẩm quyền phải có khả năng truy ngược lại được người dùng nào đã làm điều này. Một hệ hỗ trợ khả năng như vậy gọi là hệ hỗ trợ truy tìm dấu vết. Các hệ BE hỗ trợ truy vết hiệu quả hiện nay là [1, 12, 44].

Hệ BGW [9] và các cải tiến [10, 29, 46]: Dựa trên kỹ thuật phép ghép cặp đôi, các tác giả đã đề xuất một hệ mã hóa quảng bá có độ dài bản mã và độ dài khóa bí mật là 2 và 1 phần tử, còn tốc độ giải mã tương đối nhanh. Tuy nhiên, độ dài khóa công khai là lớn, phụ thuộc vào tổng số người dùng trong hệ thống. Các tác giả cũng đề xuất phương pháp cân bằng giữa độ dài của bản mã và khóa công khai, khi cả hai cùng phụ thuộc vào căn bậc hai của tổng số người dùng trong hệ thống. Ngoài ra, an toàn của hệ BGW yếu và dựa trên một giả thuyết mạnh. Các tác giả trong bài báo [29], dựa trên hệ BGW đã đề xuất một cải tiến trong đó họ đề xuất một hệ tương tự BGW nhưng có mức an toàn cao hơn và dựa trên một giả thuyết yếu hơn, điểm yếu của hệ này là có độ dài khóa công khai dài hơn so với hệ BGW. Lưu ý, các hệ kể trên đều không hỗ trợ truy vết.

Hệ Delerablee [25] và các cải tiến [55, 56]: Dựa trên Parings, tác giả đề xuất một hệ mã hóa quảng bá có độ dài bản mã, độ dài khóa bí mật, độ dài khóa công khai và tốc độ giải mã tương tự như hệ BGW. Tuy nhiên, điểm mạnh là hệ Delerablee là hệ mã hóa quảng bá dựa trên định danh. Điểm yếu của hệ này là hệ đạt mức an toàn yếu và ngoài ra hệ còn phải dựa vào giả thuyết là tồn tại hàm băm lý tưởng. Hệ cũng không hỗ trợ truy vết.

Vấn đề phân phối khóa: Trong các hệ mã hóa quảng bá, nhà quản trị hệ thống sẽ chịu trách nhiệm phân phối khóa bí mật cho các người dùng. Điều này dẫn đến hai vấn đề, hoặc là hệ thống có thể bị tấn công dẫn đến hệ thống không hoạt động và các người dùng bị lộ khóa bí mật, hoặc tự bản thân quản trị hệ thống không trung thực.

Để giải quyết vấn đề này, có hai phương pháp đã được các nhà nghiên cứu đề xuất: Một là chia nhỏ các hệ thống thành các hệ thống con [51, 42], người dùng phải nhận được tất cả các khóa bí mật thành phần từ các hệ thống con này để tạo ra khóa bí mật cho riêng mình; hai là, chỉ cần một số lượng nhất định các hệ thống con phối hợp với nhau là có thể cấp khóa bí mật được cho người dùng [48]. Lưu ý rằng, với cách thứ hai ta có thể giải quyết được đồng thời cả hai vấn đề ở trên. Và cho dù một số hệ thống nhỏ bị tấn công hệ thống vẫn có thể hoạt động được, miễn là số lượng các hệ thống không bị tấn công vẫn còn lớn hơn ngưỡng. Nếu số lượng các hệ thống con không trung thực bé hơn ngưỡng, thì những hệ thống con này dù cho phối hợp với nhau vẫn không biết được khóa bí mật của người dùng. Hệ [48] vẫn chưa hiệu quả để có thể áp dụng rộng rãi vào thực tế. Việc xây dựng một hệ mã hóa quảng bá hiệu quả hỗ trợ tính chất phân quyền hiện vẫn còn là vấn đề mở cần được nghiên cứu thêm.

Bảo mật: Với các hệ mã hóa quảng bá có hai mô hình an toàn được xét:

Thứ nhất là mô hình an toàn ở mức yếu, trong đó kẻ tấn công phải xác định trước mục tiêu muốn tấn công là tập người dùng nào, sau đó mới được biết các thông tin khác như khóa công khai. Khóa bí mật của những người dùng không nằm trong tập mục tiêu. Tùy thuộc vào mô hình tấn công là CPA hay CCA mà kẻ tấn công có thể chọn trước một số bản rõ và biết bản mã tương ứng, hay chọn trước một số bản mã và biết bản rõ tương ứng. Cuối cùng, kẻ tấn công phải giải được bản mã được mã hóa cho tập người dùng mục tiêu.

Thứ hai là mức an toàn cao hơn, trong đó cho phép kẻ tấn công sau khi biết hết các thông tin cần thiết mới phải đưa ra tập người dùng muốn tấn công. Tất nhiên, với điều kiện là kẻ tấn công chưa biết khóa bí mật của những người dùng này. Ngoài ra, phải cần quan tâm đến giả thuyết, một hệ mã hóa quảng bá thường được phát biểu là an toàn dưới mô hình nào và dựa trên giả thuyết gì. Các giả thuyết dạng tĩnh như CDH, DDH, BDH,... thường được đánh giá cao hơn các giả thuyết không phải dạng tĩnh. Với giả thuyết dạng tĩnh, đầu vào cho kẻ tấn công chỉ là một hằng số các thông tin, còn giả thuyết không phải dạng tĩnh thì ít nhất là q thông tin. Một số hệ mã hóa quảng bá có thể đạt an toàn trong mô hình an toàn ở mức cao dưới bài toán khó dạng

như tài liệu [30, 2]. Hay gần đây dùng kỹ thuật chứng minh kép để chứng minh như trong bài báo [29]. Thiết kế một hệ BE đạt bảo mật dưới giả thuyết chuẩn dạng tĩnh vẫn còn chưa được giải quyết.

Vấn đề ẩn danh người nhận: Một hướng mở rộng khác của mã hóa quảng bá là vấn đề ẩn danh người nhận gọi là mã hóa quảng bá ẩn danh (anonymous broadcast encryption) [13]. Tức là, kẻ tấn công thu được bản mã nhưng từ bản mã đã lấy được không thể biết ai là người có khả năng giải mã. Điều này có thể đảm bảo an toàn cho người nhận, đặc biệt trong các môi trường mang tính chất nhạy cảm về chính trị, an ninh. Những hệ được đánh giá tốt hiện nay có thể hỗ trợ tính chất này trong tài liệu [13] vẫn còn chưa hiệu quả khi độ dài bản mã còn lớn. Việc đề xuất một hệ mã hóa quảng bá mới hỗ trợ tính chất này vẫn còn là một vấn đề mở để các chuyên gia, các nhà khoa học nghiên cứu.

1.4. Mã hóa quảng bá đa kênh và mã hóa dựa trên thuộc tính

1.4.1. Tổng quan về mã hóa quảng bá đa kênh

Dựa trên BGW, các tác giả ở [47] đã mở rộng mã hóa quảng bá từ việc gửi một thông tin *m* đến một nhóm người dùng S, cho phép cùng lúc gửi nhiều thông tin m_1, m_2,..., m_k đến các tập người dùng khác nhau tương ứng S_1, S_2,. ., S_k. Một hệ mã hóa như vậy được gọi là hệ mã hóa quảng bá đa kênh. Ứng dụng của MCBE trong thực tế ví dụ như truyền hình trả tiền, khi mã mỗi thông tin m_i như là một kênh, và mỗi tập S_i là một nhóm người trả tiền đăng ký xem kênh đó, tức là trung tâm cùng lúc có thể phát sóng rất nhiều kênh. Các tác giả [47] đã đề xuất một hệ mã có độ dài bản mã chỉ 2 phần tử tương tự như hệ BGW (lưu ý rằng, nếu dùng hệ BGW để gửi k thông tin khác nhau đến k tập người dùng khác nhau, độ dài bản mã sẽ là 2k). Tuy nhiên, điểm yếu của hệ mã này là tốc độ giải mã không hiệu quả và là mã hóa quảng bá khóa bí mật. Hệ cũng không hỗ trợ truy vết. Các tác giả trong tài liệu [60] đã cải tiến hệ này bằng cách rút ngắn hơn độ dài của khóa công khai. Các tác giả trong bài báo [15] làm tăng hơn nữa an toàn và hiệu năng của hệ khi đưa ra xây dựng dựa trên Parings loại ba. Gần đây, các tác giả trong bài báo [3] đã giới thiệu hệ mã hóa quảng

bá đa kênh mới có tính chất là mã hóa khóa công khai, tức là người lập mã không cần phải biết bất kỳ tham số bí mật gì khi thực hiện mã hóa.

1.4.2. Tổng quan về Mã hóa dựa trên thuộc tính

Mã hóa dựa trên thuộc tính (ABE) là mã hóa cho phép quá trình mã hóa và giải mã dựa trên thuộc tính. ABE được phân ra làm hai loại: Thứ nhất, là mã hóa dựa trên thuộc tính có chính sách bản mã. Thứ hai, là mã hóa dựa trên thuộc tính có chính sách khóa. Đối với CP-ABE, thông tin *m* sẽ được mã hóa dưới một chính sách nào đó, ví dụ như:

(NV and PKT and e-H) or (NV and PCS and e-H)

Người dùng sở hữu tập thuộc tính nào sẽ nhận khóa bí mật tương ứng với tập thuộc tính đó, miễn là tập thuộc tính thỏa mãn chính sách ở bản mã là người dùng có thể giải mã được. Ngược lại, đối với KP-ABE, thông tin *m* sẽ được mã hóa dưới một tập thuộc tính, ví dụ như:

(NV, PKT, e-H)

Người dùng sẽ sở hữu một chính sách bất kỳ và nhận khóa bí mật tương ứng với chính sách đó, miễn là chính sách đó được thỏa mãn bởi tập thuộc tính ở bản mã là người dùng có thể giải mã được. Trong hai loại, thì CP-ABE có nhiều ứng dụng trong thực tế hơn là KP-ABE. Với các hệ ABE thì chính sách là quan trọng nhất. Cho đến nay, các nhà nghiên cứu đã xem xét một vài loại chính sách khác nhau: Chính sách truy cập, chính sách truy cập đạt ngưỡng, chính sách truy cập linh động, chính sách truy cập tổng quát. Với chính sách truy cập thì trong chính sách chỉ cho phép dùng phép AND, ví dụ:

NV and PKT and e-H

Đối với chính sách truy cập, cứ miễn là người dùng sở hữu được số lượng thuộc tính lớn hơn một ngưỡng nào đó quy định ở bản mã là có thể giải mã được, không quan trọng loại thuộc tính mà người dùng sở hữu. Ví dụ: Ở bước mã hóa nếu dùng chính sách mã hóa thông tin m với ngưỡng là 3 thuộc tính thì bất cứ người dùng nào trong hệ thống sở hữu số thuộc tính ít nhất là 3 là có thể giải mã được. Ta có thể thấy hai loại chính sách này có ứng dụng khá hạn chế. Trong thực tế, ta cần có loại

chính sách mà quy định giải mã có thể biểu diễn được ít nhất là bằng một biểu thức Boolean ví dụ như:

(NV and PKT and e-H) or (NV and PCS and e-H)

Chính sách như vậy gọi là chính sách truy cập linh động, các hệ mã hóa dựa trên thuộc tính hỗ trợ linh động có thể đáp ứng được hầu hết các ứng dụng trong thực tế. Các nhà nghiên cứu còn ý tưởng phát triển việc xây dựng mã hóa dựa trên thuộc tính có thể hỗ trợ chính sách được mô tả bằng một mạch tổng quát, tức là không chỉ là mạch đúng sai như ở trên. Tuy nhiên, những hệ mã như vậy chỉ tồn tại ở dạng lý thuyết vì tính hiệu quả của nó còn xa so với thực tế. Ta có thể liệt kê một số hệ tiêu biểu hỗ trợ chính sách truy cập và ngưỡng truy cập như [21, 22].

Hệ mã đầu tiên hỗ trợ chính sách truy cập linh động được xây dựng dựa trên cấu trúc cây, được giới thiệu bởi Goyal và các tác giả [31]. Tiếp theo đó, dựa vào kỹ thuật chia sẻ bí mật tuyến tính, các hệ tiếp theo được đề xuất trong các tài liệu [4, 5, 6, 7]. Hệ mã hỗ trợ chính sách truy cập tổng quát được giới thiệu ở tài liệu [32, 33]. Trong đó, với hệ mã thứ nhất [33], các tác giả dựa trên công cụ đa tuyến tính, tuy nhiên, sự tồn tại của đa tuyến tính vẫn chưa thực sự rõ ràng. Hệ thứ hai [32], dựa trên giả thuyết LWE nhưng có độ hiệu quả rất thấp.

Cũng như hệ mã hóa quảng bá, các vấn đề cần được quan tâm cho một hệ mã dựa trên thuộc tính là tính hiệu quả của độ dài bản mã, độ dài khóa, tốc độ mã hóa, tốc độ giải mã, độ an toàn CPA hay CCA, vấn đề phân phối khóa, truy vết. Ngoài ra, mã hóa dựa trên thuộc tính còn cần quan tâm đến vấn đề loại bỏ quyền giải mã của người dùng. Trong thực tế, có những trường hợp cho dù người dùng sở hữu đầy đủ thuộc tính thỏa mãn chính sách để có thể giải mã được, nhưng vì lý do nào đó người lập mã không muốn cho người dùng này có thể giải mã được. Hệ mã hóa thuộc tính có hỗ trợ việc loại bỏ người dùng này mà không ảnh hưởng đến quyền giải mã của những người dùng khác được gọi là một mã hóa dựa trên thuộc tính hỗ trợ loại bỏ người dùng. Các hệ này có thể được xây dựng dựa trên sự kết hợp đặc biệt của mã hóa quảng bá và mã hóa dựa trên thuộc tính. Những hệ kết hợp được như vậy gọi là

hệ mã hóa quảng bá dựa trên thuộc tính được giới thiệu trong các công trình, tài liệu được công bố [15, 16, 17].

Với mã hóa dựa trên thuộc tính, ta có thể liệt kê một số hệ tiêu biểu với các ưu điểm khác nhau như sau: Hệ CP-ABE, hoặc có độ dài bản mã là hằng số [4, 8, 15, 18, 21, 22], hoặc có độ dài khóa bí mật là hằng số [17, 16]. Trong đó, các hệ [21, 22] chỉ hỗ trợ chính sách là And-gates hoặc Threshold, các hệ trong tài liệu [4, 8, 15, 18] có thể hỗ trợ chính sách là chính sách linh động. Hệ ABE có tốc độ giải mã nhanh [6, 4, 5, 42], hệ ABE hỗ trợ truy viết [39], phi tập trung hóa ABE hay nhiều trung tâm cấp khóa ABE [41, 19, 61, 51, 43]. Đối với vấn đề an toàn cho hệ mã, với sự kết hợp của kỹ thuật ghép nối [7] và mã hóa cặp [58], các hệ ABE đạt được bảo mật thích ứng và được xây dựng tại các công trình [4, 5, 6, 7]. Hệ ABE hỗ trợ tổng quát, được xây dựng từ tài liệu tham khảo [33] hay từ giả thuyết LWE [32].

Một hướng nghiên cứu mở rộng của ABE hiện nay đang rất được quan tâm là vấn đề tìm kiếm trên dữ liệu đã được mã hóa [11, 14, 23, 24, 34, 37, 45, 57, 59]. Hiện nay, dữ liệu của các doanh nghiệp thường được mã hóa bằng ABE và lưu trên các đám mây, trong khi hàng ngày doanh nghiệp vẫn cần phải làm việc trên khối dữ liệu đã được mã hóa này. Ví dụ trong bài toán tìm kiếm dữ liệu, phương pháp đơn giản nhất là doanh nghiệp sẽ cung cấp toàn bộ khóa bí mật cho một máy chủ công cộng nào đó có năng lực mạnh để giải mã toàn bộ dữ liệu của mình, sau đó tìm kiếm dữ liệu trên dữ liệu đã được giải mã đó. Tuy nhiên, phương pháp này có nhược điểm là máy chủ đó sẽ biết được toàn bộ nội dung dữ liệu của doanh nghiệp, điều mà không một doanh nghiệp nào mong muốn.

Hướng nghiên cứu hiện nay là doanh nghiệp chỉ cung cấp một phần thông tin của khóa bí mật gọi là cửa sập cho máy chủ công khai, để máy chủ dựa vào đó tìm kiếm dữ liệu cần thiết trên khối dữ liệu đang được mã hóa của doanh nghiệp. Sau khi tìm được các bản mã tương ứng doanh nghiệp muốn tìm sẽ gửi trả về cho doanh nghiệp, doanh nghiệp sẽ dùng khóa bí mật của mình để giải mã các bản mã này. Với phương pháp như vậy, máy chủ chỉ biết được thông tin là cửa sập và các bản mã nhưng lại không thể biết được nội dung thực sự dữ liệu của doanh nghiệp. Trong khi

doanh nghiệp vẫn tận dụng được sức mạnh tính toán của máy chủ. Kỹ thuật này cũng được áp dụng vào rất nhiều ứng dụng khác, chẳng hạn như ứng dụng định hướng chuyển tiếp Email của các Gateway.

1.5. Kết luận chương 1

Chương 1 trình bày tổng quát về ba loại mã tiên tiến hiện nay là mã hóa quảng bá, mã hóa quảng bá đa kênh và mã hóa dựa trên thuộc tính. Các loại mã này hỗ trợ quyền giải mã linh động và đang được sử dụng trong rất nhiều loại ứng dụng hiện nay như: Các ứng dụng truyền hình trả tiền, chia sẻ files, social network (facebooks, twitter,...), lưu trữ an toàn dữ liệu trên đám mây cho mọi loại ứng dụng như e-Health, chính phủ điện tử,...

Tình hình nghiên cứu hiện nay của ba loại mã hóa cũng như các hạn chế, tồn tại chưa được khắc phục và vẫn đang được nghiên cứu, đưa ra một số cách tiếp cận khả thi để giải quyết các vấn đề mở này.

CHƯƠNG 2: MÃ HÓA QUẢNG BÁ ĐA KÊNH

Trong chương này, nghiên cứu sinh trình bày về hệ mã hóa quảng bá đa kênh bao gồm: Định nghĩa chung về mã hóa quảng bá đa kênh, mô hình an toàn của một hệ mã hóa quảng bá đa kênh, một số hệ mã hóa quảng bá đa kênh quan trọng và các hạn chế đối với một số hệ mã hóa quảng bá đa kênh hiện nay. Phần cuối là nội dung cụ thể về lược đồ mã hóa quảng bá đa kênh do NCS đề xuất nhằm khắc phục một số hạn chế của mã hóa này.

2.1. Định nghĩa và mô hình an toàn của hệ mã hóa quảng bá đa kênh

Mã hóa quảng bá đa kênh được giới thiệu bởi Pointcheval và các tác giả [47] với mục tiêu mở rộng khái niệm của mã hóa quảng bá từ việc gửi một thông tin m đến một nhóm người dùng S, đến việc cho phép cùng lúc gửi nhiều thông tin $m_1, m_2, ..., m_k$ đến các tập người dùng khác nhau tương ứng $S_1, S_2,..., S_k$. Trong khi đó, độ dài bản mã, độ dài khóa bí mật và tốc độ giải mã đều đã được cải tiến.

2.1.1. Định nghĩa

Một hệ mã hóa quảng bá đa kênh như vậy được định nghĩa như sau:

Khởi tạo (λ):

Đầu vào của giải thuật khởi tạo là tham số an toàn λ. Trong đó, tham số an toàn λ có nghĩa là, để phá được hệ mã này kẻ tấn công cần thực hiện ít nhất 2^{λ} phép toán. Đầu ra của giải thuật là khóa công khai và khóa bí mật của hệ thống. Khóa công khai bao gồm thêm cả các thông tin như số tối đa các kênh của hệ thống, ký hiệu là m; số tối đa các người dùng có thể đăng ký để xem một kênh, ký hiệu là n.

Tạo khóa (msk, $ID_{i,j}$, param):

Đầu vào của giải thuật là khóa bí mật của hệ thống msk, định danh của người dùng thứ i trong kênh thứ j, ký hiệu là $ID_{i,,j}$. Lưu ý: $1 \leq j \leq m$, và i có thể lớn hơn n nhưng tổng số người dùng trong một kênh phải bé hơn hoặc bằng n. Tham số cuối cùng trong quá trình tạo khóa là khóa công khai của hệ thống. Giải thuật sẽ trả về khóa bí mật $sk_{ID_{ij}}$ cho người dùng thứ i. Ngược lại, giải thuật sẽ trả về $\perp$ (giải thuật ngừng và kết quả đầu ra là null).

Lưu ý rằng, nếu người dùng đăng ký xem bao nhiêu kênh thì sẽ nhận tương ứng bấy nhiêu khóa bí mật. Ví dụ: Người dùng i đăng ký xem kênh j, kênh k, kênh h sẽ nhận tương ứng ba khóa bí mật $sk_{ID_{ij}}$, $sk_{ID_{ik}}, sk_{ID_{ih}}$.

Mã hóa (param, $S_{i_1}, S_{i_2}, \dots, S_{i_t}$):

Đầu vào của giải thuật là khóa công khai của hệ thống và các tập người dùng có khả năng giải mã trong từng kênh i_1, i_2,...,i_t. Đầu ra của giải thuật là t khóa phiên làm việc $K_{i_1}, \dots, K_{i_t}$ và bản mã của chúng chứa cả các thông tin $S_{i_1}, S_{i_2}, \dots, S_{i_t}$, ký hiệu là Hdr. Lưu ý rằng, trong thực tế thì khóa phiên làm việc $K_{i_1}, \dots, K_{i_t}$ sau đó sẽ được dùng như khóa bí mật trong hệ mã hóa khóa bí mật (ví dụ AES) để mã hóa dữ liệu của từng kênh. Như vậy, bản mã đầy đủ trong thực tế gọi sẽ bao gồm Hdr và bản mã dữ liệu thực tế được mã hóa dưới các khóa bí mật $K_{i_1}, \dots, K_{i_t}$.

Giải mã ($sk_{ID_{ij}}$, Hdr, param):

Đầu vào của giải thuật là khóa bí mật của người dùng *i* trong kênh *j*, ký hiệu là $sk_{ID_{i,j}}$, bản mã Hdr của các khóa phiên làm việc $K_{i_1}, \dots, K_{i_t}$, khóa công khai của hệ thống. Đầu ra của giải thuật là khóa phiên làm việc K_j nếu như $(i, j) \in S_j$, ngược lại đầu ra của giải thuật là $\perp$. Lưu ý rằng, trong thực tế thì sau khi giải mã tìm được khóa phiên K_j, người dùng sẽ dùng K_j như khóa bí mật để giải mã bản mã đầy đủ bằng hệ mã hóa khóa bí mật để tìm lại dữ liệu thực tế đã được mã hóa của kênh *j*.

Một hệ mã hóa được mã hóa với cơ chế như trên được gọi là hệ mã hóa lai. Lý do mã hóa lai được dùng trong thực tế là do nó tận dụng được cả hai ưu thế của hệ mã hóa khóa công khai truyền thống và hệ mã hóa khóa bí mật. Cụ thể, hạn chế còn tồn tại của hệ mã hóa khóa công khai là có tốc độ mã hóa chậm, trong khi ưu điểm là không cần thống nhất khóa bí mật chung giữa người gửi và người nhận. Còn nhược điểm của mã hóa khóa bí mật là phải thống nhất trước khóa bí mật chung giữa người gửi và người nhận, trong khi ưu điểm là tốc độ mã hóa nhanh. Hệ mã hóa lai là tận dụng lợi thế của cả hai hệ này, cụ thể khóa phiên làm việc $K_{i_1}, \dots, K_{i_t}$ ngắn sẽ được mã hóa bằng mã hóa khóa công khai có tốc độ chậm, còn dữ liệu dài sẽ được mã hóa bằng mã hóa khóa bí mật có tốc độ nhanh dưới khóa phiên $K_{i_1}, \dots, K_{i_t}$. Như vậy, với

hệ mã hóa lai giữa người gửi và người nhận không cần thống nhất trước khóa bí mật chung, dữ liệu được mã hóa bằng hệ mã hóa khóa bí mật.

Cũng giống như hệ mã hóa quảng bá, để đơn giản ta chỉ xét việc mã hóa và giải mã của khóa phiên làm việc $K_{i_1}, \dots, K_{i_t}$, do việc mã hóa và giải mã dữ liệu thực tế dùng $K_{i_1}, \dots, K_{i_t}$ như là khóa bí mật là giống nhau ở tất cả các hệ mã hóa quảng bá đa kênh.

Xét một ví dụ, giả sử hệ thống có 5 kênh (m = 5): 1 (Thể thao), 2 (Phim truyện), 3 (Thế giới động vật), 4 (Tin tức mới), 5 (Âm nhạc). Số lượng tối đa người dùng có thể xem một kênh là 100 (n = 100). Giả sử rằng: Alice là người dùng 1, Bob là người dùng 2 và David là người dùng 3. Nếu Alice thích Phim truyện và Âm nhạc, cô ấy đăng ký để xem kênh 2 và 5 và nhận khóa bí mật tương ứng ($sk_{ID_{1,2}}$, $sk_{ID_{1,5}}$). Tương tự, nếu Bob đăng ký để xem kênh 1 và 4, anh ấy sẽ nhận khóa bí mật tương ứng ($sk_{ID_{2,1}}$, $sk_{ID_{2,4}}$). Nếu David đăng ký để xem kênh 1, 2 và 4, anh ấy sẽ nhận khóa bí mật tương ứng ($sk_{ID_{3,1}}, sk_{ID_{3,2}}, sk_{ID_{3,4}}$).

Để mã hóa cho cả Alice, Bob và David, trung tâm phát sóng sẽ chọn bốn tập như sau:

- Tập S_1 tương ứng với kênh 1, tập này bao gồm các chỉ số: (2, 1), (3, 1).
- Tập S_2 tương ứng với kênh 2, tập này bao gồm các chỉ số: (1, 2), (3, 2).
- Tập S_4 tương ứng với kênh 4, tập này bao gồm các chỉ số: (2, 4), (3, 4).
- Tập S_5 tương ứng với kênh 5, tập này bao gồm các chỉ số: (1, 5).

Dễ dàng thấy rằng, Alice với khóa bí mật $sk_{ID_{1,2}}$ có thể giải mã để tính ra khóa phiên K_2 và từ đó có thể xem được kênh Phim truyện, và với khóa bí mật $sk_{ID_{1,5}}$ có thể giải mã để tính ra khóa phiên K_5, từ đó có thể xem được kênh Âm nhạc. Hoàn toàn tương tự cho Bob và David.

Trong ví dụ trên t = 4, i_1 = 1, i_2 = 2, i_3 = 4, i_4 = 5. Để cho đơn giản về mặt ký hiệu, từ nay về sau ta giả sử rằng t tập là $S_1, S_2, \dots, S_t$, $t \leq m$. Mặt khác, mặc dù số tối đa người dùng đăng ký xem một kênh là 100, tuy nhiên số tối đa người dùng trong hệ thống (Alice, Bob, David, . . .) có thể lớn hơn nhiều so với 100.

2.1.2. Mô hình an toàn

Mô hình an toàn chuẩn cho một hệ mã hóa quảng bá đa kênh được định nghĩa như sau:

Chúng ta xét một kịch bản giữa kẻ tấn công $\mathcal{A}$ và kẻ thách thức $\mathcal{C}$ (đại diện cho sự an toàn của hệ mã).

Tại giai đoạn bắt đầu, kẻ tấn công $\mathcal{A}$ công bố t tập $S_1^*, \dots, S_t^*$ trong đó, với i = 1, ..., t, $|S_i^*| \leq n$, $t \leq m$ và chỉ số $i^* \leq t$ tương ứng với tập $S_{i^*}^*$ mà $\mathcal{A}$ định tấn công cho $\mathcal{C}$.

Khởi tạo (λ): Đầu tiên, $\mathcal{C}$ chạy giải thuật khởi tạo(λ) để tạo ra tham số công khai, khóa bí mật msk của hệ thống. Tiếp theo, $\mathcal{C}$ công bố param cho $\mathcal{A}$ đồng thời giữ bí mật msk. Ngoài ra, $\mathcal{C}$ cũng khởi tạo danh sách rỗng Λ_C, Λ_C là danh sách các chỉ số đã bị $\mathcal{A}$ biết khóa bí mật.

Giai đoạn truy vấn 1: Kẻ tấn công $\mathcal{A}$ có thể tùy ý yêu cầu để biết thông tin sau: $\mathcal{A}$ yêu cầu được biết khóa bí mật của người dùng tương ứng với chỉ số (i, j): Kẻ thách thức $\mathcal{C}$ chạy giải thuật Tạo khóa ($sk_{ID_{i,j}}$, msk): Để tạo ra khóa bí mật $sk_{ID_{i,j}}$, sau đó công bố $sk_{ID_{i,j}}$ cho $\mathcal{A}$ đồng thời thêm chỉ số (i, j) vào danh sách các chỉ số đã bị $\mathcal{A}$ biết khóa bí mật Λ_C.

Giai đoạn thách thức: $\mathcal{C}$ chạy giải thuật mã hóa (param, $S_1^*, S_2^* \dots, S_t^*$): Để thu về bản mã và bản rõ (khóa phiên) là (Hdr*, $K_1^*, K_2^* \dots, K_t^*$). Tiếp theo, $\mathcal{C}$ chọn ngẫu nhiên một bít $b \xleftarrow{\$} \{0,1\}$. Nếu $b = 1$, $\mathcal{C}$ chọn $K_{i^*}^* \xleftarrow{\$} \mathcal{K}$ ($\mathcal{K}$ là không gian của khóa phiên), còn nếu $b = 0$ thì $\mathcal{C}$ giữ nguyên khóa $K_{i^*}^*$. Như vậy, khóa $K_{i^*}^*$ sẽ là số ngẫu nhiên không liên quan gì đến bản mã Hdr* nếu $b = 1$. Cuối cùng, $\mathcal{C}$ công bố (Hdr*, $K_1^*, K_2^* \dots, K_t^*$) cho $\mathcal{A}$.

Giai đoạn truy vấn 2: Kẻ tấn công $\mathcal{A}$ tiếp tục có quyền yêu cầu biết các thông tin như trong giai đoạn truy vấn 1.

Giai đoạn dự đoán kết quả: Kẻ tấn công $\mathcal{A}$ đưa ra dự đoán bít $b' \in \{0,1\}$ cho bít b.

Ta nói rằng, kẻ tấn công $\mathcal{A}$ thắng trong kịch bản trên nếu như $b' = b$ và không tồn tại bất kỳ chỉ số (i, i^*) sao cho $(i, i^*) \in S_{i^*}^*$ và $(i, i^*) \in \Lambda_C$. Ta ký hiệu $\text{Succ}^{IND}(\mathcal{A}) = Pr[b' = b]$ là xác suất mà $\mathcal{A}$ thắng trong kịch bản trên, và lợi thế của nó là:

$$\textbf{Adv}^{IND}(\mathcal{A}) = 2 \times \textbf{Succ}^{IND}(\mathcal{A}) - 1$$
$$= \Pr[1 \leftarrow \mathcal{A}|b = 1] - \Pr[1 \leftarrow |b = 0]$$

Định nghĩa 3.1: Một hệ mã hóa quảng bá đa kênh được gọi là đạt an toàn CPA nếu tất cả các kẻ tấn công chạy trong thời gian đa thức có lợi thế trong kịch bản tấn công ở trên là nhỏ không đáng kể.

Cũng như mã hóa quảng bá, khái niệm an toàn ở trên gọi là an toàn không phân biệt được, tức là kẻ tấn công không có khả năng phân biệt giữa một khóa phiên K đúng và một giá trị ngẫu nhiên. Một khái niệm an toàn yếu hơn gọi là an toàn không tính toán được, tức là kẻ tấn công chỉ không có khả năng tính ra được khóa phiên K, tuy nhiên nó có thể có khả năng phân biệt giữa một khóa phiên K đúng và một giá trị ngẫu nhiên.

2.2. Một số hệ mã hóa quảng bá đa kênh quan trọng

Trong mục này, nghiên cứu sinh giới thiệu một số hệ mã hóa quảng bá đa kênh quan trọng hiện nay.

2.2.1. Hệ mã hóa quảng bá đa kênh - MCBE_1

Như đã trình bày ở chương 1, dựa trên BGW [9], các tác giả trong bài báo [47] đã mở rộng khái niệm của mã hóa quảng bá từ việc gửi một thông tin m đến một nhóm người dùng S, đến việc cho phép cùng lúc gửi nhiều thông tin $m_1, m_2,..., m_k$ đến các tập người dùng khác nhau tương ứng $S_1, S_2,..., S_k$. Ứng dụng của MCBE trong thực tế, ví dụ như truyền hình trả tiền khi mà mỗi thông tin m_i như là một kênh và mỗi tập S_i là một nhóm người trả tiền đăng ký xem kênh đó, tức là trung tâm cùng lúc có thể phát sóng rất nhiều kênh. Các tác giả [47] đã đề xuất một hệ mã có độ dài bản mã chỉ 2 phần tử tương tự như hệ BGW (lưu ý rằng nếu dùng hệ BGW để gửi k thông tin khác nhau đến k tập người dùng khác nhau, độ dài bản mã sẽ là $2k$), tuy nhiên điểm yếu của hệ mã này là tốc độ giải mã chậm và là hệ mã hóa quảng bá khóa bí mật.

Hệ mã hóa BGW: Trước tiên, chúng ta đã biết hệ mã hóa quảng bá BGW [9] bao gồm các giải thuật sau:

Khởi tạo (λ):

Giả sử $\mathbb{G}$ là một hệ thống ánh xạ song tuyến có bậc là số nguyên tố p. Giải thuật chọn phần tử sinh $g \in \mathbb{G}$ và một số mũ ngẫu nhiên $\alpha \in \mathbb{Z}_p$. Tính $g_i = g^{\alpha^i} \in \mathbb{G}$ với mọi i = 1, 2,... , n, n + 2,..., 2n. Tiếp theo, giải thuật chọn số mũ ngẫu nhiên $\gamma \in \mathbb{Z}_p$ *và đặt* $v = g^{\gamma} \in \mathbb{G}$.

Khóa công khai của hệ thống là:

$$\text{param} = (g_1, \dots, g_n, g_{n+2}, \dots, g_{2n}, v) \tag{2.1}$$

Khóa bí mật của hệ thống là msk = α.

Tạo khóa (msk, i):

Với đầu vào là khóa bí mật của hệ thống và chỉ số người dùng thứ i. Khóa bí mật tương ứng với người dùng thứ i sẽ là:

$$d_i = v^{\alpha^i}$$

Mã hóa (S, param):

Đầu vào của giải thuật là tập người dùng S có khả năng giải mã. Giải thuật chọn số mũ ngẫu nhiên $r \in \mathbb{Z}_p$, đặt khóa phiên $K = e(g_{n+1}, g)^r$, trong đó $e(g_{n+1}, g)$ có thể được tính từ $e(g_n, g)$ ở trong khóa công khai của hệ thống.

Tiếp theo, đặt:

$$\text{Hdr} = \left(g^r , (v . \prod_{j \in S} g_{n+1-j})^r \right) \tag{2.2}$$

Đầu ra của giải thuật là (Hdr, K). Lưu ý rằng Hdr bao gồm cả tập S, và như đã trình bày ở mục trước, ta chỉ quan tâm đến việc mã hóa và giải mã khóa phiên K.

Giải mã (Hdr, i, d_i, param):

Giả sử Hdr = (C_1, C_2), giải thuật với đầu vào là bản mã Hdr và khóa bí mật d_i, khóa công khai param cho đầu ra là khóa phiên K như sau:

$$K = \frac{e(g_i, C_2)}{e\left(d_{i\,.\ \prod_{j\in S, j\neq i} g_{n+1-j+i}}\ ,\ C_1\right)} \tag{2.3}$$

Một cách tự nhiên, khi ta muốn dùng hệ mã hóa BGW để mã hóa *m* thông tin khác nhau đến m tập khác nhau $S_1, S_2,..., S_m$, chúng ta có thể kết hợp *m* hệ BGW như sau:

Khởi tạo (λ): Giống như trong hệ BGW.

Mã hóa ($S_1, S_2,..., S_m$, param):

Chọn ngẫu nhiên các giá trị $r_1, \dots, r_m \in \mathbb{Z}_p$, tính:

$$K_1 = e(g_{n+1}, g)^{r_1}, \dots, K_m = e(g_{n+1}, g)^{r_m} \tag{2.4}$$

$$\text{Hdr} = \left(\left(g^{r_1}, (v\,.\prod\nolimits_{j\in S_1} g_{n+1-j})^{r_1}\right), \dots, \left(g^{r_m}, (v\,.\prod\nolimits_{j\in S_m} g_{n+1-j})^{r_m}\right)\right) \tag{2.5}$$

Giải mã ($S_1,..., S_m$, Hdr, *i*, (EK, d_i), *j*):

Phân tích $C_1 = g^{r_j};\ \ C_2 = (v\,.\prod_{j\in S_1} g_{n+1-j})^{r_j}$ từ Hdr sau đó giải mã giống như trong hệ BGW.

Đánh giá độ hiệu quả:

1. Độ dài bản mã Hdr lớn, bao gồm tới *2m* phần tử, do vậy không hiệu quả và không thể dùng được trong thực tế nếu số lượng kênh là lớn.

2. Độ dài khóa bí mật tuy chỉ bao gồm một phần tử $d_i = v^{\alpha^i}$, tuy nhiên khi giải mã thì người giải mã vẫn cần phải biết khóa công khai của hệ thống param. Tức là độ dài của khóa giải mã phải bao gồm cả độ dài của khóa công khai. Bởi vì, độ dài của khóa công khai của hệ thống là tuyến tính với tổng số người dùng trong hệ thống.

3. Cả quá trình mã hóa và giải mã đều hiệu quả.

Ý tưởng xây dựng:

Với cách xây dựng đơn giản ở trên, độ dài của bản mã Hdr là lớn. Các tác giả trong bài báo [47] đầu tiên cố gắng khắc phục điểm yếu này bằng cách dùng lại giá trị ngẫu nhiên, bằng cách này độ dài của bản mã Hdr chỉ còn *m + 1* phần tử:

$$Hdr = \left(g^r, (v \,.\prod_{j \in S_1} g_{n+1-j})^r, \ldots, (v \,.\prod_{j \in S_m} g_{n+1-j})^r \right) \quad (2.6)$$

Tuy nhiên, việc dùng lại giá trị ngẫu nhiên *r* sẽ dẫn đến việc khóa phiên tại tất cả các kênh là như nhau, có nghĩa là chỉ cần người dùng giải mã được một kênh thì sẽ giải mã được tất cả các kênh khác. Hay nói cách khác, chỉ cần người dùng đăng ký một kênh thì sẽ xem được tất cả các kênh còn lại. Để giải quyết vấn đề này các tác giả đã thêm các phần tử $X_i \in \mathbb{G}$ tương ứng với người dùng *i = 1,..., n*, sau đó thay đổi khóa phiên và bản mã Hdr bằng việc dùng x_i, trong đó:

$X_i = g^{x_i}$, *với i* = 1,..., *n*

$$K_1 = e(g_{n+1}, g)^{r+\sum_{j \in S_1} x_j}, \ldots, K_m = e\,(g_{n+1}, g)^{r+\sum_{j \in S_m} x_j}, \quad (2.7)$$

$$Hdr = \left(g^r, (v \,.\prod_{j \in S_1} g_{n+1-j})^{r+\sum_{j \in S_1} x_j}, \ldots, (v \,.\prod_{j \in S_m} g_{n+1-j})^{r+\sum_{j \in S_m} x_j} \right) \quad (2.8)$$

Với cách làm trên, bản mã Hdr đã giảm xuống còn m + 1 phần tử, tuy nhiên như vậy vẫn là dài. Các tác giả đã tiếp tục cải tiến bằng cách nhân tất cả các bản mã thành phần lại với nhau:

$$Hdr = \left(g^r, (v \,.\prod_{j \in S_1} g_{n+1-j})^{r+\sum_{j \in S_1} x_j} \times \ldots \times (v \,.\prod_{j \in S_m} g_{n+1-j})^{r+\sum_{j \in S_m} x_j} \right) \quad (2.9)$$

Như vậy, độ dài bản mã Hdr lúc này chỉ còn là hai phần tử. Tuy nhiên khi thực hiện mã hóa thì người lập mã cần phải biết một số tham số bí mật, do vậy hệ mã không thể là ở dạng mã hóa khóa công khai. Ngoài ra, hệ mã chỉ đạt an toàn ở mức thấp, có thể gọi là mức an toàn cơ bản (kẻ tấn công không thể yêu cầu kẻ thách thức trả lời các câu hỏi về lập mã cũng như giải mã).

Hệ mã thứ nhất, ký hiêu là $MCBE_1$ được mô tả như sau:

Khởi tạo (λ):

Đầu vào của giải thuật là tham số an toàn λ, giải thuật tạo ra khóa công khai **param**, khóa bí mật của hệ thống **msk**, khóa bí mật dùng để mã hóa viết tắt là **EK** và

khóa bí mật của người dùng như sau: Giả sử $\mathbb{G}$ là hệ thống bilinear group bậc $p, g \in \mathbb{G}$ là phần tử sinh.

Chọn ngẫu nhiên $\propto \in \mathbb{Z}_p$. Tính $g_i = g^{\alpha^i} \in \mathbb{G}$với i = 1, 2,..., n, n + 2,..., 2n. Tiếp theo chọn ngẫu nhiên $\gamma \in \mathbb{Z}_p$ và đặt $v = g^{\gamma} \in \mathbb{G}$.

Giải thuật cũng chọn các số ngẫu nhiên $x_1, x_2, \ldots, x_n \in \mathbb{Z}_p$và đặt:

$$X_1 = g^{x_1}, X_2 = g^{x_2}, \ldots, X_n = g^{x_n} \quad (2.10)$$

Khóa bí mật của hệ thống là **msk** = *(g, v, α, γ, x_1, x_2,..., x_n)*, khóa bí mật dùng để mã hóa là **EK** = *(g, v, g_{n+1}, x_1, x_2,..., x_n)*. Khóa công khai của hệ thống là:

$$\text{param} = (g_1, \ldots, g_n, g_{n+2}, \ldots, g_{2n}, X_1, X_2, \ldots, X_n)$$

Khóa bí mật của người dùng thứ i là $d_i = v^{\alpha^i}$, với $i \in \{1, \ldots, n\}$. Lưu ý, khi người dùng đăng ký xem nhiều kênh thì sẽ có nhiều khóa bí mật như trên. Ở đây, để đơn giản về mặt ký hiệu, ta giả sử người dùng chỉ đăng ký xem một kênh duy nhất.

Mã hóa (*S_1,S_2, ... ,S_m*, **EK**):

Chọn ngẫu nhiên $r \xleftarrow{\$} \mathbb{Z}_p$, đặt:

$$K_k = e(g_{n+1}, g)^{r+\sum_{j \in S_k} x_j} \text{ với } k = 1, \ldots, m.$$

Tiếp theo, đặt:

$$Hdr = \left(g^r, \prod_{k=1}^{k=m} (v \cdot \prod_{j \in S_k} g_{n+1-j})^{r+\sum_{j \in S_k} x_j} \right) \quad (2.11)$$

Lưu ý rằng người lập mã biết g_{n+1}, x_1,... , x_n từ **EK**. Cuối cùng giải thuật cho đầu ra là (**Hdr,** *K_1, K_2, ... , K_m)*.

Giải mã (*S_1,... , S_m*, **Hdr**, *i*, *d_i* , *k*):

Phân tích **Hdr** = *(C_1, C_2). Nếu i ∈ S_k* tính:

$$K_k = \frac{e(g_i, C_2)}{e(d_i \cdot \prod_{\substack{j \in S_k \\ j \neq i}} g_{n+1-j+i}, C_1 \cdot \prod_{j \in S_k} X_j) \cdot \prod_{\substack{\ell = 1 \\ \ell \neq k}}^{\ell = m} e(d_i \cdot \prod_{j \in S_\ell} g_{n+1-j+i}, C_1 \cdot \prod_{j \in S_\ell} X_j)} \quad (2.12)$$

$$= \frac{e(g_i, C_2)}{e(d_i \cdot \prod_{\substack{j \in S_k \\ j \neq i}} g_{n+1-j+i}, g^{r+\sum_{j \in S_k} x_j}) \cdot \prod_{\substack{\ell = 1 \\ \ell \neq k}}^{\ell = m} e\left(d_i \cdot \prod_{j \in S_\ell} g_{n+1-j+i}, g^{r+\sum_{j \in S_\ell} x_j}\right)} \quad (2.13)$$

$$= \frac{e\left(g^{\alpha^i}, \prod_{\ell=1}^{\ell=m} \left(v \cdot \prod_{j\in S_\ell} g_{n+1-j}\right)^{r+\sum_{j\in S_\ell} x_j}\right)}{e\left(v^{\alpha^i} \cdot \left(\prod_{\substack{j\in S_k \\ j\neq i}} g_{n+1-j}\right)^{\alpha^i},\ g^{r+\sum_{j\in S_k} x_j}\right) \cdot \prod_{\substack{\ell=m \\ \ell=1 \\ \ell\neq k}} e\left(v^{\alpha^i} \cdot \left(\prod_{j\in S_\ell} g_{n+1-j}\right)^{\alpha^i}, g^{r+\sum_{j\in S_\ell} x_j}\right)} \quad (2.14)$$

$$= \frac{e\left(g^{\alpha^i}, \left(v \cdot \prod_{j\in S_k} g_{n+1-j}\right)^{r+\sum_{j\in S_k} x_j}\right)}{e\left(v^{\alpha^i} \cdot \left(\prod_{\substack{j\in S_k \\ j\neq i}} g_{n+1-j}\right)^{\alpha^i},\ g^{r+\sum_{j\in S_k} x_j}\right)} \cdot \prod_{\substack{\ell=1 \\ \ell\neq k}}^{\ell=m} \frac{e\left(g^{\alpha^i},\ \left(v \cdot \prod_{j\in S} g_{n+1-j}\right)^{r+\sum_{j\in S_\ell} x_j}\right)}{e\left(v^{\alpha^i} \cdot \left(\prod_{j\in S_\ell} g_{n+1-j}\right)^{\alpha^i}, g^{r+\sum_{j\in S_\ell} x_j}\right)} \quad (2.15)$$

$$= \frac{e\left(\left(v \cdot \prod_{j\in S_k} g_{n+1-j}\right)^{\alpha^i},\ g^{r+\sum_{j\in S_k} x_j}\right)}{e\left(\left(v \cdot \prod_{\substack{j\in S_k \\ j\neq i}} g_{n+1-j}\right)^{\alpha^i},\ g^{r+\sum_{j\in S_k} x_j}\right)} \cdot \prod_{\substack{\ell=1 \\ \ell\neq k}}^{\ell=m} \frac{e\left(\left(v \cdot \prod_{j\in S_\ell} g_{n+1-j}\right)^{\alpha^i},\ g^{r+\sum_{j\in S_\ell} x_j}\right)}{e\left(\left(v \cdot \prod_{j\in S_\ell} g_{n+1-j}\right)^{\alpha^i}, g^{r+\sum_{j\in S_\ell} x_j}\right)} \quad (2.16)$$

$$= e\left(g_{n+1-i}^{\alpha^i},\ g^{r+\sum_{j\in S_k} x_j}\right) = e\left(g_{n+1},\ g^{r+\sum_{j\in S_k} x_j}\right) = e(g_{n+1}, g)^{r+\sum_{j\in S_k} x_j}$$

Lưu ý, ta dùng mối liên hệ $d_i = v^{\alpha^i}$, $g_{n+1-j+i} = g_{\ n+1-j}^{\ \alpha^i}$, và $g_{n+1-i}^{\alpha^i} = g_{n+1}$ và khi mã hóa người lập mã cần biết khóa bí mật **EK** nên đây là hệ mã hóa khóa bí mật.

2.2.2. Hệ mã hóa quảng bá đa kênh - MCBE$_2$

Điểm yếu của hệ mã hóa quảng bá đa kênh MCBE$_1$ là mô hình an toàn chỉ đạt ở mức cơ bản khi kẻ tấn công không thể yêu cầu kẻ thách thức trả lời các câu hỏi về mã hóa và giải mã. Để khắc phục nhược điểm này, các tác giả trong bài báo [47] đã cải tiến hệ MCBE$_1$ bằng cách áp dụng tính chất của bộ tiên tri ngẫu nhiên trong xây dựng và chứng minh an toàn của hệ. Hệ MCBE$_2$ được mô tả cụ thể như sau:

Khởi tạo (λ):

Đầu vào của giải thuật là tham số an toàn λ, giải thuật tạo ra khóa công khai param, khóa bí mật của hệ thống **msk**, khóa bí mật dùng để mã hóa **EK** và khóa bí mật của người dùng như sau:

Giả sử $\mathbb{G}$ là hệ thống song tuyến bậc $p, g \in \mathbb{G}$ là phần tử sinh. Chọn ngẫu nhiên $\alpha \in \mathbb{Z}_p$. Tính $g_i = g^{\alpha^i}$ với i = 1, 2, . . . , n, n + 2, . . . ,2n.Tiếp theo, chọn ngẫu nhiên $\gamma \in \mathbb{Z}_p$và đặt $v = g^{\gamma} \in \mathbb{G}$ và $d_n = v^{\alpha^n}$ hệ thống dùng hàm băm $\mathcal{H}$ như một bộ tiên tri ngẫu nhiên. Khóa bí mật của hệ thống là msk = *(g, v, α, γ)*, khóa bí mật dùng để mã hóa là: **EK = msk**, khóa công khai của hệ thống là:

$$\text{param} = (g_1, \ldots, g_n, g_{n+2}, \ldots, g_{2n}, d_n)$$

Khóa bí mật của người dùng thứ i là $d_i = v^{\alpha^i}$, với $i \in \{1, \ldots, n\}$. Giống như hệ mã **MCBE$_1$**, khi người dùng đăng ký xem nhiều kênh thì sẽ có nhiều khóa bí mật

như trên. Ở đây, để đơn giản về mặt ký hiệu, ta giả sử người dùng chỉ đăng ký xem một kênh duy nhất.

Mã hóa *(S_1,..,S_m,* EK):

Chọn ngẫu nhiên $r \in \mathbb{Z}_p$; $S_{m+1} = \{n\}$, với mỗi tập S_i, *với* $i = 1,\ldots, m+1$ tính $Y_i = H(i, g^r)$ ($Y_i = g^{y_i}$, với số mũ chưa biết y_i), và

$$K_i = e(g_{n+1}, Y_i).e(g_{n+1}, g)^r = e(g_{n+1}, g)^{r+y_i}, i = 1, \ldots, m+1 \qquad (2.17)$$

Cuối cùng tính Hdr = (C_1, C_2, C_3) như sau:

$$C_1 = g^r$$

$$C_2 = \prod_{i=1}^{i=m+1} \left(Y_i^{\gamma+\sum_{j\in S_i}\alpha^{n+1-j}} \left(v \cdot \prod_{j\in S_i} g_{n+1-j} \right)^r \right)$$

$$= \prod_{i=1}^{i=m+1} \left(v \cdot \prod_{j\in S_i} g_{n+1-j} \right)^{r+y_i}$$

$$C_3 = \mathcal{H}(C_1 C_2)^r$$

Người lập mã biết α và γ để tính C_2. Giải thuật cho đầu ra

(Hdr, K_1,.., K_{m+1}).

Giải mã *(S_1,.., S_m,Hdr, i, d_i , k*):

Đặt $S_{m+1} = \{n\}$, phân tích:

Hdr = (C_1, C_2, C_3). Nếu $i \in S_k$, kiểm tra $e\big(C_1, \mathcal{H}(C_1, C_2)\big) = e(g, C_3)$,

Nếu đúng tính:

$Y_i = \mathcal{H}(i, g^\gamma)$, *với* $i = 1,\ldots, m+1$.

Và tính:

$$K_k = \frac{e(g_i C_2)}{e\left(d_i . \prod_{\substack{j\in S_k \\ j\neq i}} g_{n+1-j+i}, C_1 . Y_k\right) \prod_{\substack{\ell = 1 \\ \ell \neq k}}^{\ell = m+1} e\left(d_i . \prod_{j\in S_\ell} g_{n+1-j+i}, C_1 . Y_\ell\right)} \qquad (2.18)$$

$$= \frac{e\left(g^{\alpha^i}, \prod_{\ell = 1}^{\ell = m+1} (v . \prod_{j\in S_\ell} g_{n+1-j})^{r+y_\ell}\right)}{e\left(v^{\alpha^i} . (\prod_{\substack{j\in S_k \\ j\neq i}} g_{n+1-j})^{\alpha^i}, g^{r+y_k}\right) . \prod_{\substack{\ell = 1 \\ \ell \neq k}}^{\ell = m+1} e\left(v^{\alpha^i} . \left(\prod_{j\in S_\ell} g_{n+1-j}\right)^{\alpha^i}, g^{r+y_\ell}\right)} \qquad (2.19)$$

$$= \frac{e\left(g^{\alpha^i},\ (v.\prod_{j\in S_k} g_{n+1-j})^{r+y_k}\right)}{e\left(v^{\alpha^i}.\left(\prod_{\substack{j\in S_k\\ j\neq i}} g_{n+1-j}\right)^{\alpha^i},\ g^{r+y_k}\right)} \cdot \prod_{\substack{\ell=1\\ \ell\neq k}}^{\ell=m+1} \frac{e\left(g^{\alpha^i},(v.\prod_{j\in S_\ell} g_{n+1-j})^{g^{r+y_\ell}}\right)}{e\left(v^{\alpha^i}.(\prod_{j\in S_\ell} g_{n+1-j})^{\alpha^i},\ g^{r+y_\ell}\right)} \quad (2.20)$$

$$= \frac{e\left((v.\ \prod_{j\in S_k} g_{n+1-j})^{\alpha^i},\ g^{r+y_k}\right)}{e\left(\left(v.\ \prod_{\substack{j\in S_k\\ j\neq i}} g_{n+1-j}\right)^{\alpha^i},\ g^{r+y_k}\right)} \cdot \prod_{\substack{\ell=1\\ \ell\neq k}}^{\ell=m+1} \frac{e\left((v.\prod_{j\in S_\ell} g_{n+1-j})^{\alpha^i},\ g^{r+y_\ell}\right)}{e\left((v.\prod_{j\in S_\ell} g_{n+1-j})^{\alpha^i},\ g^{r+y_\ell}\right)} \quad (2.21)$$

$$= e\left(g_{n+1-i}^{\alpha^i}, g^{r+y_k}\right) = e(\ g_{n+1}, g^{r+y_k}) = \ e(g_{n+1}, g)^{r+y_k} \quad (2.22)$$

Lưu ý, $d_i = v^{\alpha^i}$, $g_{n+1-j+i} = g_{n+1-j}^{\alpha^i}$, và $g_{n+1-i}^{\alpha^i} = g_{n+1}$. Ngoài ra, khi mã hóa người lập mã cần biết khóa bí mật **EK** nên đây vẫn là hệ mã hóa khóa bí mật.

2.2.3. Một số cải tiến đối với hệ $MCBE_1$ và $MCBE_2$

Có ba cải tiến đối với hệ mã quảng bá đa kênh $MCBE_1$ và $MCBE_2$ như sau:

1. Các tác giả [60] đã cải tiến hệ mã $MCBE_1$ bằng cách rút ngắn hơn độ dài của khóa công khai. Tuy nhiên, độ dài khóa công khai trong hệ [60] vẫn dài, cụ thể là vẫn có độ dài tuyến tính với *n*.

2. Các tác giả [15] làm tăng hơn nữa an toàn và hiệu năng của hệ $MCBE_1$ và $MCBE_2$ khi đưa ra xây dựng hệ $MCBE_1$ và $MCBE_2$ mới dựa trên Parings loại ba.

3. Hai cải tiến ở trên vẫn chưa khắc phục được điểm yếu của hệ $MCBE_1$ và $MCBE_2$ là mã hóa khóa bí mật. Gần đây, các tác giả trong bài báo [3] đã giới thiệu hệ mã hóa quảng bá đa kênh mới có tính chất là mã hóa khóa công khai, tức là người lập mã không cần phải biết bất kỳ tham số bí mật gì khi thực hiện mã hóa.

2.3. Lược đồ mã hóa quảng bá đa kênh đề xuất

Luận án đã đề xuất một lược đồ mã hóa quảng bá đa kênh cải tiến mới có các ưu điểm so với các mã hóa quảng bá đa kênh khác như sau:

Là mã hóa quảng bá đa kênh khóa công khai, tức là người lập mã không cần phải biết bất kỳ tham số bí mật gì khi thực hiện mã hóa.

Có tốc độ giải mã nhanh. Cụ thể, người giải mã chỉ cần tính 2 phép tính parings khi giải mã. Tác giả cài đặt hệ mã và đưa ra các đánh giá cụ thể về thời gian chạy của hệ mã.

Có độ dài bản mã Hdr chỉ gồm hai phần tử, độ dài khóa bí mật của người dùng có số lượng phần tử chính bằng số lượng kênh mà người dùng đăng ký để xem.

Được xây dựng dựa trên hệ mã hóa Delerablee [25] đã trình bày tại mục 1.2.3

2.3.1. Ý tưởng xây dựng

Trong hệ [25], khóa bí mật của người dùng có dạng:

$$g^{\frac{1}{\alpha+ID_u}}$$

Bản mã tương ứng với tập người dùng S có dạng:

$$h^{k.\prod_{i\in S}(\alpha+ID_i)}$$

và miễn là $u \in S$ thì người dùng u có thể tính được khóa phiên $e(g, h)^k$, k là số ngẫu nhiên được chọn mỗi lần lập mã.

Trong tổng thể của hệ mã hóa quảng bá đa kênh, chúng ta có m tập người dùng (tương ứng với m kênh) $S_1,..., S_m$, và mỗi tập có một khóa phiên tương ứng. Ý tưởng là, với mỗi tập người dùng S_i , $i = 1,.., m$, đặt khóa phiên của tập này là $e(g,h)^{k.\beta_i}$, trong đó $\beta_1,..., \beta_m$ là các số ngẫu nhiên được chọn ở thuật toán khởi tạo. Với ý tưởng như vậy, khóa bí mật của người dùng lúc này sẽ có dạng:

$$g^{\frac{\beta_i}{\alpha+ID_{ui}}}$$

Vì mỗi người dùng u có thể đăng ký tối đa m kênh nên mỗi người dùng u sẽ có tối đa m khóa bí mật $g^{\frac{\beta_i}{\alpha+ID_{ui}}}$, $i = 1,..., m$.

2.3.2. Lược đồ mã hóa đề xuất và so sánh

Mã hóa được xây dựng như sau:

Khởi tạo (1^{λ}):

Đầu vào của giải thuật là tham số an toàn λ, đầu ra của giải thuật là khóa bí mật của hệ thống msk, khóa công khai của hệ thống, khóa công khai bao gồm cả m là số tối đa các kênh, n là số tối đa người dùng đăng ký vào một kênh.

Đặt N = $m{\cdot}n$, giải thuật tạo ra hệ thống ánh xạ song tuyến $D = (p, \mathbb{G}, \widetilde{\mathbb{G}}, \mathbb{G}_T, e)$, chọn ngẫu nhiên $h \xleftarrow{\$} \widetilde{\mathbb{G}}$, $g \xleftarrow{\$} \mathbb{G}$ và $\alpha, \beta_1, \dots, \beta_m \xleftarrow{\$} \mathbb{Z}_p^*$.

Giả sử $\mathcal{H}$ là hàm băm sao cho $\mathcal{H}: \{0,1\}^* \to \mathbb{Z}_p^*$. Giải thuật cho đầu ra:

$$\text{param} = \left(D, \left\{h^{\alpha^i}\right\}_{i=0,\dots,N}, \left\{h^{\beta_i \alpha^j}\right\}_{\substack{i=1,\dots,m \\ j=0,\dots,N}}, g^{\alpha}, \left\{e(g,h)^{\beta_i}\right\}_{i=1,\dots,m}, \mathcal{H}\right)$$

và khóa bí mật của hệ thống:

$$\text{msk} = (g, \alpha, \beta_1, \dots, \beta_m)$$

Tạo khóa ($ID_{i,j}$, msk, param) :

Giả sử, định danh của người dùng *i* trong kênh *j* là chuỗi bit bất kỳ $\text{ID}_{i,j} \in \{0, 1\}^*$. Người dùng *i* đăng ký vào kênh *j* sẽ nhận được khóa bí mật tương ứng sau:

$$sk_{ID_{i,j}} = g^{\frac{\beta_j}{\alpha + \mathcal{H}(ID_{i,j})}}$$

Ký hiệu (*i*, *j*) là chỉ số khóa bí mật. Trong hệ thống, mỗi người dùng có thể đăng ký vào nhiều kênh và nhận khóa bí mật tương ứng với từng kênh. Ví dụ người dùng *i* đăng ký vào kênh 1,..., *t*, người dùng *i* sẽ nhận các khóa bí mật $\left\{sk_{ID_{i,j}}\right\}_{j=1,\dots,t}$.

Mã hóa (param, S_1,..,S_t):

Đầu vào của giải thuật là *t* tập chỉ số khóa bí mật của người dùng trong *t* kênh S_1,.., S_t , $t \leq m$. Trong đó S_1,.., S_t là các tập không giao nhau. Ký hiệu $S = \cup_{i=1}^{t} S_i$ là tập đầy đủ tất cả các chỉ số khóa bí mật của người dùng cho một lần mã hóa. Giải thuật chọn ngẫu nhiên $k \in \mathbb{Z}_p^*$, tính khóa phiên bí mật cho *t* kênh như sau:

$$K_i = e(g,h)^{k\beta_i}, i = 1, \dots, t$$

Sau đó, tính bản mã Hdr = (C_1, C_2) như sau:

$$C_1 = g^{-\alpha.k}, C_2 = h^{k.\prod_{(i,j)\in S}\left(\alpha + \mathcal{H}(ID_{i,j})\right)}$$

Cuối cùng, giải thuật cho đầu ra $\text{K} = \{K_i\}_{i=1,\dots,t}$ và $\text{Hdr} = (C_1, C_2)$ bao gồm cả thông tin của tập *S*.

Giải mã ($sk_{ID_{i,j}}$, Hdr, param):

Giải thuật đầu tiên kiểm tra xem chỉ số *(i, j)* ∈ *S* có đúng hay không, nếu sai thì cho đầu ra là ⊥. Nếu đúng là thuộc tập *S*, giải thuật tính giá trị $K' = h^{\gamma}$ trong đó:

$$\gamma = \frac{\beta_j}{\alpha}\left(\prod_{\substack{(i',j')\in S \\ (i',j')\neq(i,j)}} \left(\alpha + \mathcal{H}\left(ID_{i',j'}\right)\right) - \prod_{\substack{(i',j')\in S \\ (i',j')\neq(i,j)}} \mathcal{H}\left(ID_{i',j'}\right)\right) \quad (2.24)$$

Giải thuật có thể tính K' từ các thông tin có trong khóa công khai.

Đặt:

$$B = \prod_{\substack{(i',j')\in S \\ (i',j')\neq(i,j)}} \mathcal{H}\left(ID_{i',j'}\right)$$

Giải thuật cuối cùng cho đầu ra:

$$K_j = \left(e(C_1, K').e\left(sk_{ID_{i,j}}, C_2\right)\right)^{\frac{1}{B}}$$

Tính đúng đắn:

$$K_j = \left(e(C_1, K').e\left(sk_{ID_{i,j}}, C_2\right)\right)^{\frac{1}{B}} \quad (2.25)$$

$$= \left(e(g^{-\alpha.k}, h^{\gamma}).e\left(g^{\frac{\beta_j}{\alpha+\mathcal{H}(ID_{i,j})}}.h^{k.\prod_{(i,j)\in S}\left(\alpha+\mathcal{H}(ID_{i,j})\right)}\right)\right)^{\frac{1}{B}} \quad (2.26)$$

$$= \left(e(g,h)^{k\beta_j \prod_{\substack{(i',j')\in S \\ (i',j')\neq(i,j)}} \mathcal{H}\left(ID_{i',j'}\right)}\right)^{\frac{1}{B}} \quad (2.27)$$

$$= e(g,h)^{k\beta_j} \quad (2.28)$$

So sánh với các hệ mã khác:

Để đánh giá độ hiệu quả của hệ mã hóa đa kênh đề xuất, nghiên cứu sinh lập Bảng 2.1 so sánh với một số hệ mã hóa quảng bá đa kênh khác, trong đó:

Header là độ dài của bản mã.

S-key là độ dài khóa bí mật.

P-key là độ dài khóa công khai.

Dec time là thời gian giải mã.

Security là đánh giá an toàn của hệ mã. Lưu ý rằng, với mô hình bảo mật chọn lọc, kẻ tấn công phải thông báo trước tập người dùng mà kẻ tấn công định tấn công trước khi biết các thông tin khác như: Khóa bí mật của các người dùng khác, chọn bản mã biết bản rõ,… Như vậy có nghĩa là, quyền của kẻ tấn công sẽ yếu hơn so với mô hình thích nghi, khi kẻ tấn công không cần phải thông báo trước tập người dùng mà kẻ tấn công định tấn công.

Ngoài ra, mô hình an toàn dùng bộ tiên tri ngẫu nhiên thì hệ thống cũng sẽ kém an toàn hơn vì dùng ROM có nghĩa là giả thuyết hàm băm là lý tưởng không có bất kỳ khó khăn nào. Tuy nhiên, trong cài đặt thực tế như họ hàm băm SHA thì không có hàm băm nào là lý tưởng.

Trong Bảng 2.1 mô hình có CCA nghĩa là kẻ tấn công có quyền chọn tùy ý bản mã và biết bản rõ tương ứng, còn mô hình không có CCA thì kẻ tấn công không có quyền này.

Tóm lại, mô hình an toàn mạnh nhất sẽ là CCA và yếu nhất sẽ là Selective+ROM.

Setting là kiểu mã hóa bí mật (Secret-key) hay công khai (Public-key).

	Header	S-key	P-key	Dec time	Security	Setting
[15]-1	$2\|\mathbb{G}\|$	$m\|\widetilde{\mathbb{G}}\|$	$3mn\|\mathbb{G}\|+ 2mn\|\widetilde{\mathbb{G}}\|$	$(m + 1)P$	Selective	Bí mật
[15]-2	$3\|\mathbb{G}\|+1\|\mathbb{G}\|$	$m\|\widetilde{\mathbb{G}}\|$	$2mn\|\mathbb{G}\| + 2mn\|\widetilde{\mathbb{G}}\|$	$(m + 2)P$	Selective-CCA+ROM	Bí mật
[60]	$2\|\mathbb{G}\|$	$m\|\widetilde{\mathbb{G}}\|$	$2mn\|\mathbb{G}\|+ 2mn\|\widetilde{\mathbb{G}}\|$	$(m + 1)P$	Selective	Bí mật
[3]	$2\|\mathbb{G}\|$	$m\|\mathbb{G}\|$	$(mn + m) \|\mathbb{G}\|$	$2P$	Selective	Công khai
MCBE đề xuất	$1\|\mathbb{G}\|+1\|\widetilde{\mathbb{G}}\|$	$m\|\mathbb{G}\|$	$m^2n\|\widetilde{\mathbb{G}}\|$	$2P$	Selective +ROM	Công khai

Bảng 2.1. So sánh một số hệ mã hóa đa kênh với MCBE đề xuất

Trong đó:

n là số tối đa người dùng trong một kênh,

m là số tối đa kênh,

P là một phép tính Parings.

$|\mathbb{G}|$ là kích thước của một phần tử trong nhóm $\mathbb{G}$,

$|\widetilde{\mathbb{G}}|$ là kích thước của một phần tử trong nhóm $\widetilde{\mathbb{G}}$.

Hệ của nghiên cứu sinh là hệ mã hóa khóa công khai trong khi 3 hệ [15]-1 và [15]-2 và [60] đều là mã hóa bí mật. Độ dài của bản mã hệ nghiên cứu sinh đề xuất có độ dài ngắn nhất. Thời gian giải mã của hệ nghiên cứu sinh đề xuất chỉ là 2 phép toán Perings (2P) còn các hệ khác lớn hơn hoặc bằng.

2.3.3. Đánh giá an toàn

Trong mục này, tác giả sẽ chứng minh rằng hệ mã đạt an toàn CPA dưới giả thuyết (hay còn được hiểu là bài toán khó) GDDHE tương tự như trong bài báo [25], trong đó giả thuyết GDDHE ở đây được định nghĩa như sau:

Định nghĩa 2.2: Bài toán khó GDDHE: Giả sử $(p, \mathbb{G}, \widetilde{\mathbb{G}}, \mathbb{G}_T, e)$ là hệ thống ánh xạ song tuyến, chọn ngẫu nhiên $t, m, n, k, \beta_1, \dots, \beta_m \overset{\$}{\leftarrow} \mathbb{Z}_p^*$ sao cho $t > N = m \cdot n$

Đặt $f(X) = \prod_{i=1}^{t}(X + x_i)$ và $g(X) = \prod_{i=t+1}^{t+n}(X + x_i)$, $x_i \overset{\$}{\leftarrow} \mathbb{Z}_p^*$, i = 1,.., N, là hai đa thức nguyên tố cùng nhau có bậc tương ứng là t và n. Đặt đa thức:

$q(X) = \prod_{i=n+1}^{n+t}(X + x_i)$ và hai phần tử sinh là $g_0 \in \mathbb{G}$, $h_0 \in \widetilde{\mathbb{G}}$. Cho trước:

$$\vec{Y} = (g_0, \{g_0^{\frac{\beta_m f(\alpha)}{\alpha + x_i}}\}_{i=1,\dots,n}, \{g_0^{\beta_i \cdot \alpha^j}\}_{\substack{i=1,\dots,m-1 \\ j=0,\dots,t-1}}, g_0^{\alpha \cdot f(\alpha)}, g_0^{k \cdot \alpha \cdot f(\alpha)}$$
$$h_0 h_0^{\alpha}, \dots, h_0^{\alpha^{2t+n}}, \left\{h_0^{\beta_i \alpha^j}\right\}_{\substack{i=1,\dots,m \\ j=0,\dots,2t+n}}, h_0^{k \cdot q(\alpha)}, e(g_0 h_0)^{k \cdot \beta_1 \cdot f(\alpha)}, \dots, e(g_0 h_0)^{k \cdot \beta_{m-1} \cdot f(\alpha)}) \quad (2.29)$$

hãy phân biệt giữa $T = e(g_0, h_0)^{k \cdot \beta_m \cdot f(\alpha)} \in \mathbb{G}_T$ và một phần tử ngẫu nhiên $T = R \in \mathbb{G}_T$.

Để phân biệt giá trị *T*, kẻ tấn công $\mathcal{A}$ mà cho đầu ra một bit $b \in \{0, 1\}$ có lợi thế ϵ trong việc giải bài toán GDDHE ở trên nếu:

$$\left|Pr\left[\mathcal{A}\left(\vec{Y}, T = e(g_0 h_0)^{k \cdot \beta_m \cdot f(\alpha)}\right) = 0\right] - Pr\left[\mathcal{A}\left(\vec{Y}, T = R\right) = 0\right]\right| \geq \epsilon$$

Định nghĩa 2.3: Bài toán GDDHE là một bài toán khó nếu không tồn tại kẻ tấn công nào chạy trong thời gian đa thức mà có lợi thế ϵ đáng kể trong việc phân biệt T.

Chứng minh độ khó của bài toán GDDHE ở trên như sau:

Chứng minh: Đầu tiên viết lại bài toán GDDHE ở dạng số mũ:

$$P = \left\{1, \{\beta_m \frac{f(\alpha)}{\alpha + x_i}\}_{i=1,\dots,n}, \{\beta_i \alpha^j\}_{\substack{i=1,\dots,m-1 \\ j=0,\dots,t-1}}, \alpha f(\alpha), k\alpha f(\alpha)\right\}$$

$$Q = \left\{1, \alpha, \dots, \alpha^{2t+n}, \{\beta_i \alpha^j\}_{\substack{i=1,\dots,m \\ j=0,\dots,2t+n}}, kq(\alpha)\right\}$$

$$R = \{k\beta_1 f(\alpha), \dots, k\beta_{m-1} f(\alpha)\}$$

$$f = k\beta_m f(\alpha)$$

Giả sử rằng f không độc lập tuyến tính với *(P, Q, R)*, tức là kẻ tấn công có thể tìm được các giá trị $b_{i,j}$, c_i sao cho đẳng thức sau đúng

$$f = \sum_{\substack{p_i \subset P \\ q_j \subset Q}} b_{i,j}.p_i.q_j + \sum_{r_i \subset R} c_i r_i$$

Dùng cả *k* và β_m để phân tích *f*. Do cả *k* và β_m là các số ngẫu nhiên được chọn trước nên suy ra kẻ tấn công cần tìm a_i, b_j sao cho biểu thức sau đúng:

$$k\beta_m f(\alpha) = k\beta_m(b_0 \alpha f(\alpha)\alpha^0 + \cdots + b_{2t+n}\alpha f(\alpha)\alpha^{2t+n}) + a_1 \frac{f(\alpha)}{\alpha + x_1} q(\alpha) + \cdots + a_n \frac{f(\alpha)}{\alpha + x_n} q(\alpha) \quad (2.30)$$

$$k\beta_m f(\alpha) = k\beta_m f(\alpha)\left(G(\alpha) + a_1 \frac{q(\alpha)}{\alpha + x_1} + \cdots + a_n \frac{q(\alpha)}{\alpha + x_n}\right) \quad (2.31)$$

$$1 = G(\alpha) + a_1 \frac{q(\alpha)}{\alpha + x_1} + \cdots + a_n \frac{q(\alpha)}{\alpha + x_n} \quad (2.32)$$

Trong đó *G* là đa thức có tính chất *G*(0) = 0. Do đó, ta dễ dàng thấy 1 phải được suy ra từ:

$$a_1 \frac{q(\alpha)}{\alpha + x_1} + \cdots + a_n \frac{q(\alpha)}{\alpha + x_n} \quad (2.33)$$

Tuy nhiên, do *q(α)* và mỗi đa thức $\alpha + x_i$ là nguyên tố cùng nhau với mọi *i* = 1, . . . , *n*, nên đẳng thức trên không thể đúng đối với mọi cách chọn a_i, hay nói cách khác f

là độc lập tuyến tính với *(P, Q, R).* Do vậy, từ *(P, Q, R)* không thể tính được *f*, hay kẻ tấn công không thể phân biệt được *T* trong bài toán GDDHE.

Định lý 2.4: Hệ mã quảng bá đa kênh đạt an toàn CPA dưới bài toán GDDHE ở trên.

Chứng minh*:* Giả sử $\mathcal{A}$ là kẻ tấn công hệ mã và *S* là kẻ tấn công bài toán GDDHE ở trên. Chứng minh rằng nếu tồn tại kẻ tấn công $\mathcal{A}$ tấn công thành công hệ mã với lợi thế nhiều, thì cũng sẽ tồn tại $\mathcal{S}$ tấn công bài toán GDDHE ở trên với nhiều lợi thế.

Khởi tạo: Đầu tiên $\mathcal{S}$ nhận đầu vào là đầu vào của bài toán GDDHE ở trên, $\mathcal{S}$ cần phân biệt giá trị *T* là:

$T = e(g_0, h_0)^{k.\beta_m.f(\alpha)} \in \mathbb{G}_T$ và một phần tử ngẫu nhiên $T = R \in \mathbb{G}_T$.

Luận án sẽ chỉ ra rằng $\mathcal{S}$ có thể mô phỏng $\mathcal{A}$ và dùng kết quả của $\mathcal{A}$ để phân biệt *T*. Cụ thể, $\mathcal{S}$ sẽ nhận từ $\mathcal{A}$ m^* *(*$m^* \le m$*)* tập chỉ số $S_1^*, \dots, S_m^*$ và chỉ số i^* để xác định tập $S_{i^*}^*$ mà $\mathcal{A}$ muốn tấn công.

Giả sử ta viết tập $S_{i^*}^*$ dưới dạng tập các định danh:

$S_{i^*}^* = \{ID_{1,i^*}, \dots, ID_{s^*,i^*}\}, s^* \le n$. Không mất tính tổng quát, $\mathcal{S}$ đặt β_m ở trong bài toán là β_{i^*} trong tập $S_{i^*}^*$. Đối với các tập β_i, i = 1,...,m^* khác vẫn giữ nguyên. Trong trường hợp $m^* = m$, $\mathcal{S}$ đặt β_{i^*} ở trong bài toán là β_{m^*} trong tập S_m^*. Các tham số khác như *m*, *n*, *N* được định nghĩa như ở trên.

Ký hiệu $\tilde{s} = \sum_{i=1, i\neq i^*}^{m^*} |S_i^*|$ để tạo ra các tham số cho hệ thống $\mathcal{S}$ ngầm đặt $g = g_0^{f(\alpha)}$ và nhận $g^\alpha = g_0^{\alpha f(\alpha)}$ từ giả thuyết. Với $h_0, h_0^\alpha, \dots, h_0^{\alpha^{2t+n}}$ trong tay từ giả thuyết, $\mathcal{S}$ tính:

$$h = h_0^{\prod_{i=n+\tilde{s}+1}^{t}(\alpha+x_i)\prod_{i=t+s^*+1}^{t+n}(\alpha+x_i)}$$

và sau đó $\{h^{\alpha^i}\}_{i=1,\dots,N}$

Lưu ý, trong quá trình băm, $\mathcal{S}$ sẽ dùng x_i, i = 1,.., n, để trả lời tất cả các yêu cầu truy vấn băm $\mathcal{H}(ID_{j,i^*})$ trong đó $ID_{j,i^*} \notin S_{i^*}^*$ (ID_{j,i^*}trong kênh i^* nhưng không thuộc $S_{i^*}^*$). $\mathcal{S}$ cũng dùng x_i , $i = n + 1,\dots, n + \tilde{s}$, để trả lời tất cả các truy vấn băm $\mathcal{H}(ID_{j,i})$ trong đó $ID_{j,i^*} \in \{S_i^*\}_{\substack{i=1,\dots,m^* \\ i\neq i^*}}$. $\mathcal{S}$ cũng dùng x_i, $i = t + 1,.. , t + s^*$ để trả lời tất cả các truy vấn băm $\mathcal{H}(ID_{j,i^*})$ trong đó $ID_{j,i^*} \in S_{i^*}^*$

Tiếp theo, để tính $e(g,h)^{\beta_i}, i = 1, \dots, m, \mathcal{S}$ tính:

$$e(g,h)^{\beta_i} = e\left(g_0 h_0^{\beta_i \cdot \prod_{i=n+\tilde{s}+1}^{t}(\alpha+x_i) \prod_{i=t+s^*+1}^{t+n}(\alpha+x_i).f(\alpha)}\right) \quad (2.34)$$

Lưu ý rằng, $\mathcal{S}$ biết $\left\{h_0^{\beta_i,\alpha^j}\right\}_{\substack{i=1,\dots,m \\ j=0,\dots,2t+n}}$ từ giả thuyết. Với $\left\{h_0^{\beta_i,\alpha^j}\right\}_{\substack{i=1,\dots,m \\ j=0,\dots,2t+n}}$ đã biết trong tay, $\mathcal{S}$ cũng có thể tính $\left\{h_0^{\beta_i,\alpha^j}\right\}_{\substack{i=1,\dots,m \\ j=0,\dots,N}}$ Cuối cùng, $\mathcal{S}$ cung cấp khóa công khai cho $\mathcal{A}$.

Giai đoạn truy vấn 1: Trong giai đoạn này, $\mathcal{S}$ cần trả lời hai loại truy vấn:

1. Truy vấn băm.

2. Truy vấn để biết khóa bí mật $ID_{j,i}$ trong đó $ID_{j,i} \notin S_{i^*}^*$

Đối với truy vấn băm: $\mathcal{A}$ có thể truy vấn trên bất kỳ định danh nào để biết được giá trị băm trên định danh đó. $\mathcal{S}$ tạo ra một danh sách $\mathcal{L}$ bao gồm: $(ID_{i,j}x_u, sk_{ID_{i,j}}) \in \{0,1\}^*, \mathbb{Z}_p^*, \mathbb{G}$. Ban đầu $\mathcal{L}$ bao gồm bộ ba $(*, *, *)$ hay $(ID_{i,j}, x_u, *)$, giá trị rỗng sẽ được ký hiệu bằng *. Cụ thể, nếu $ID_{j,i} \notin S_{i^*}^*$ ($ID_{j,i*}$ trong kênh i^* nhưng không trong $S_{i^*}^*$), $\mathcal{S}$ dùng x_u, u = 1,.., n. Nếu $ID_{i,j} \in \{S_{i^*}^*\}_{\substack{i=1,\dots,m^* \\ i \neq i^*}}$, $\mathcal{S}$ dùng x_u, $u = n + 1$,.., $n + \tilde{s}$. Đối với $ID_{j,i*} \in S_{i^*}^*$, $\mathcal{S}$ dùng x_u, $u = t + 1, \dots, t + s^*$

Với mỗi truy vấn băm tương ứng với $ID_{j,i}$, S đầu tiên kiểm tra xem $ID_{j,i}$ đã xuất hiện trong danh sách chưa? Nếu không, $\mathcal{S}$ chọn x_u, $u = n + \tilde{s} + 1$,.., t, giá trị chưa bao giờ xuất hiện trong L và bộ ba $(ID_{i,j}, x_u, *)$, trong L và trả về x_u cho $\mathcal{A}$. Ngược lại, $\mathcal{S}$ đơn giản là tìm bộ ba $(ID_{i,j}, x_u, *)$, và trả về x_u cho $\mathcal{A}$.

Đối với truy vấn yêu cầu biết khóa bí mật: $\mathcal{A}$ đầu tiên gửi $ID_{j,i} \notin S_{i^*}^*$ cho $\mathcal{S}$, $\mathcal{S}$ tạo ra khóa bí mật $sk_{ID_{i,j}}$ như sau:

1. Nếu $sk_{ID_{i,j}}$ đã được truy vấn trước đây, $\mathcal{S}$ tìm $sk_{ID_{i,j}}$ từ $\mathcal{L}$ và trả về $sk_{ID_{i,j}}$ cho $\mathcal{A}$.

2. Nếu $j = i^*$, có nghĩa là $ID_{i,j} \notin S_{i^*}^*$ nhưng thuộc về kênh i^*.

$\mathcal{S}$ sẽ dùng $g_0^{\frac{\beta_m f(\alpha)}{\alpha+x_u}} = g^{\frac{\beta_m}{\alpha+x_u}}$, u = 1,..., n, từ giả thuyết như $sk_{ID_{i,j}}$ để trả lời $\mathcal{A}$ (mỗi lần dùng một giá trị khác nhau), và sau đó thêm bộ ba $(ID_{i,j}, x_u, sk_{ID_{i,j}})$ vào trong $\mathcal{L}$.

3. Ngược lại, $\mathcal{S}$ kiểm tra xem $ID_{i,j}$ đã xuất hiện trong danh sách hay chưa? Nếu không $\mathcal{S}$ chọn x_u, $u = n + \tilde{s} + 1,\ldots, t$, giá trị chưa bao giờ xuất hiện trong $\mathcal{L}$, sau đó tính:

$$sk_{ID_{i,j}} = g_0^{\frac{\beta_j \cdot f(\alpha)}{\alpha + x_u}} = g^{\frac{\beta_j}{\alpha + x_u}} \tag{2.35}$$

Lưu ý rằng, $\mathcal{S}$ có thể tính vì $u = n + \tilde{s} + 1,\ldots, t$, do đó đa thức $\frac{f_\alpha}{\alpha + x_u}$ có bậc $t-1$, hơn nữa $\mathcal{S}$ có $\{g_0^{\beta_j, \alpha^i}\}_{i=0,\ldots,t-1}$ trong tay từ giả thuyết. Tiếp theo, $\mathcal{S}$ thêm bộ ba ($ID_{i,j}$, x_u, $sk_{ID_{i,j}}$) vào trong $\mathcal{L}$ và trả về $sk_{ID_{i,j}}$cho $\mathcal{A}$. Trong trường hợp $ID_{i,j}$ đã xuất hiện trong danh sách, $\mathcal{S}$ đơn giản là tìm x_u tương ứng từ danh sách $\mathcal{L}$ sau đó tính $sk_{ID_{i,j}}$ như trên. Cuối cùng $\mathcal{S}$ thêm bộ ba ($ID_{i,j}$, x_u, $sk_{ID_{i,j}}$) vào trong $\mathcal{L}$ và trả về $sk_{ID_{i,j}}$cho $\mathcal{A}$.

Giai đoạn thách thức: $\mathcal{S}$ tính:

$$C_1 = g_0^{-k.\alpha.f(\alpha)} = g^{-\alpha.k}$$

Và

$$\begin{aligned} C_2 &= h_0^{k.q(\alpha)} \\ &= h_0^{k.\prod_{i=n+1}^{n+\tilde{s}}(\alpha+x_i)\prod_{i=n+\tilde{s}+1}^{t}(\alpha+x_i).\prod_{i=t+1}^{t+s^*}(\alpha+x_i)\prod_{i=t+s^*+1}^{t+n}(\alpha+x_i)} \qquad (2.36) \\ &= h^{k.\prod_{i=n+1}^{n+\tilde{s}}(\alpha+x_i)\prod_{i=t+1}^{t+s^*}(\alpha+x_i)} \qquad (2.37) \end{aligned}$$

Để tính các khóa phiên, $\mathcal{S}$ biết $\left\{h_0^{\beta_i,\alpha^j}\right\}_{\substack{i=1,\ldots,m \\ j=0,\ldots,2t+n}}$ nên tính:

$$K_i^* = e(g_0 h_0)^{k\beta_i f(\alpha)\prod_{i=n+\tilde{s}+1}^{t} x_i \prod_{i=t+s^*+1}^{t+n} x_i}.e\left(g_0^{k\alpha f(\alpha)}, h_0^{\beta_i p(\alpha)}\right), \quad i = 1,\ldots,m^*, i \neq i^* \tag{2.38}$$

Trong đó:

$$p(\alpha) = \frac{1}{\alpha}\left(\prod_{i=n+\tilde{s}+1}^{t}(\alpha + x_i)\prod_{i=t+s^*+1}^{t+n}(\alpha + x_i) - \prod_{i=n+\tilde{s}+1}^{t} x_i \prod_{i=t+s^*+1}^{t+n} x_i\right) \tag{2.39}$$

Tiếp theo $\mathcal{S}$ tính:

$$K_i^* = T^{\prod_{i=n+\tilde{s}+1}^{t} x_i \prod_{i=t+s^*+1}^{t+n} x_i}.e(g_0^{k\alpha f(\alpha)}, h_0^{\beta_{i^*} p(\alpha)}) \tag{2.40}$$

Lưu ý: Nếu $T = e(g_0, h_0)^{k\beta_{i*} f(\alpha)}$ thì $K_{i^*}^* = e(g, h)^{k\beta_{i*}}$. Nếu T là ngẫu nhiên, $K_{i^*}^*$ cũng là ngẫu nhiên.

Giai đoạn dự đoán:

$\mathcal{A}$ cho đầu ra dự đoán b' cho b. Nếu $b' = b$, $\mathcal{S}$ cho đầu ra là bit 0 (tức là nếu $T = e(g_0, h_0)^{k\beta_{i*}f(\alpha)}$). Ngược lại, $\mathcal{S}$ cho đầu ra là bit 1 (tức là, T là phần tử ngẫu nhiên trong $\mathbb{G}_T$). Vì quá trình $\mathcal{S}$ mô phỏng $\mathcal{A}$ là hợp lệ nên ta suy ra rằng, lợi thế của $\mathcal{S}$ để phá bài toán GDDHE là $\text{Adv}^{\text{IND}}(\mathcal{A})/2$. Điều này dẫn đến một khả năng là nếu như tồn tại $\mathcal{A}$ thì cũng sẽ tồn tại $\mathcal{S}$, tức là nếu như bài toán GDDHE mà khó, có nghĩa là không tồn tại $\mathcal{S}$ thì cũng sẽ không tồn tại $\mathcal{A}$.

2.3.4. Cài đặt và đánh giá hiệu quả

Để cài đặt hệ mã, trong phần mã hóa chúng ta thấy rằng không dễ dàng để tính

$$C_2 = h^{k.\prod_{(i,j)\in S}\left(\alpha+\mathcal{H}(ID_{ij})\right)}$$

từ khóa công khai. Thay vì đó ta áp dụng công thức sau:

$$\prod_{i=1}^{m}(X + a_i) = \sum_{j=0}^{m}\left(\sum_{1\le i_1\le i_2<\cdots<i_j\le m} a_{i_1}, a_{i_2} \dots a_{i_j}\right)X^{m-j} \tag{2.41}$$

và các tham số của đa thức:

$$s_j = \sum_{1\le i_1\le i_2<\cdots<i_j\le m} a_{i_1}, a_{i_2} \dots a_{i_j} \tag{2.42}$$

tất cả là đối xứng a_1,.., a_m và được gọi là những đa thức đối xứng của giá trị a_i. Tham số s_j cho phép viết lại:

$$C_2 = h^{k\sum_{j=0}^{m} s_j.\alpha^{m-j}} = \left(\prod_{j=0}^{m}\left(h^{\alpha^{m-j}}\right)\right)^k \tag{2.43}$$

Ta có thể tính C_2 do tập $\left\{h^{\alpha^i} | i = 0, \dots, n\right\}$ có trong khóa công khai của hệ thống. Tương tự như vậy, đối với giải thuật giải mã, cũng dùng cách như trên để tính K' do $h^{\beta_i\alpha^j}$ có trong khóa công khai.

Tham số s_j ở trên có thể được tính nhanh bằng cách dùng giải thuật quy hoạch động như sau: Đặt $s_{k,j}$ là tổng của j tổ hợp của a_1, a_2, . . . , a_k. Tức là:

$$s_{k,j} = \sum_{1 \le i_1 \le i_2 < \cdots < i_j \le k} a_{i_1}, a_{i_2} \ldots a_{i_j} \tag{2.44}$$

Lưu ý: Tham số $s_j = s_{m,j}$. Tổng $s_{k,j}$ có thể được chia làm hai phần, phần thứ nhất chứa đựng a_k và phần thứ hai không chứa a_k. Chúng ta có:

$$s_{k,j} = \left(\sum_{1 \le i_1 \le i_2 < \cdots < i_j \le k-1} a_{i_1}, a_{i_2} \ldots a_{i_j} \right) + \left(\sum_{1 \le i_1 \le i_2 < \cdots < i_{j-1} \le k-1} a_{i_1}, a_{i_2} \ldots a_{i_{j-1}} \right) . a_k \tag{2.45}$$

Kéo theo mối liên hệ:

$$s_{k,j} = s_{k-1,j} + s_{k-1,j-1} . a_k \tag{3.46}$$

Trong đó $s_{k,0} = 1$ đối với $k \geq 0$ và $s_{0,j} = 0$ đối với $j > 1$. Từ mối liên hệ này, bằng việc xây dựng bảng từ $s_{0,0}$ cho tới $s_{m,m}$, có thể tính các tham số $s_j = s_{m,j}$ với độ phức tạp là $O(m^2)$.

Cài đặt MCBE đề xuất ở trên bằng ngôn ngữ C và dùng thư viện PBC [40]. Mã nguồn của chương trình cài đặt có ở địa chỉ *https://github.com/tranvinhduc/MCBE* . Cài đặt trên máy tính xách tay với bộ vi xử lý Intel Core i7-4600U @ 2.1 GHz. Đo kết quả trung bình 1000 lần. Trên máy tính này thư viện PBC tính một Parings khoảng 0.9ms, một phép mũ trên đường cong elliptic của nhóm $\mathbb{G}$ khoảng xấp xỉ 1.3ms.

Với hệ MCBE, thời gian thực hiện mã hóa chủ yếu là đi tính khóa phiên *Ki* và thành phần C_2, tương ứng với việc cần tính *m* và *N* phép mũ trong nhóm $\mathbb{G}$. Thời gian giải mã cần tính *N* phép mũ trong $\mathbb{G}$ và hai phép Parings.

Kết quả thực nghiệm được trình bày trên Bảng 2.2. Như nhận định, cả hai giải thuật mã hóa và giải mã chạy khá nhanh, và tốc độ tăng tuyến tính với *N*.

m	N	Encrypt	Decrypt
10	20	29ms	25ms
10	40	55ms	50ms
10	80	106ms	102ms
20	40	56ms	50ms
20	80	108ms	102ms
20	160	211ms	207ms
25	50	70ms	64ms
25	100	136ms	129ms
25	200	266ms	260ms

Bảng 2.2: Thực nghiệm cài đặt hệ MCBE đề xuất

Trong đó m là số kênh, mỗi kênh có n người dùng, và $N = n \times m$ là tổng số của tất cả các đăng ký trong tất cả các kênh.

2.4. Kết luận chương 2

Các nghiên cứu của chương 2 được công bố trong công trình số 4, tại các bài báo tạp chí công bố trong luận án.

Trong chương này tác giả đã trình bày về mã hóa quảng bá đa kênh, bao gồm định nghĩa tổng quát cho mã hóa quảng bá đa kênh, định nghĩa mô hình an toàn chuẩn cho một hệ mã hóa quảng bá đa kênh. Nội dung chương cũng trình bày thêm về một số hệ mã hóa quảng bá đa kênh quan trọng hiện nay bao gồm các hệ $MCBE_1$, $MCBE_2$ cùng các cải tiến gần đây của các hệ này. Điểm yếu chung của một số hệ mã hóa quảng bá đa kênh là vấn đề người lập mã cần biết các tham số bí mật. Gần đây, các tác giả trong bài báo tham khảo số [3] đã giới thiệu một mã hóa quảng bá đa kênh mà người lập mã không cần biết các tham số bí mật. Chương này, Nghiên cứu sinh trình bày hệ mã hóa quảng bá đa kênh đề xuất có cùng tính chất và độ hiệu quả tương đương như hệ [3]. Tuy nhiên, dùng phương pháp hoàn toàn khác là dựa trên kỹ thuật của hệ Delerablee. Nghiên cứu sinh cũng trình bày chứng minh chi tiết rằng đề xuất mới là đạt an toàn .

Với một số hệ mã hóa quảng bá đa kênh hiện nay, điểm yếu còn lại là vấn đề tập trung hóa, tức là chỉ có một trung tâm cung cấp khóa bí mật cho toàn bộ người dùng trong hệ thống, điều đó dẫn đến mất an toàn nếu như trung tâm này bị tấn công hay thậm chí gian dối. Nghiên cứu, thiết kế các hệ phi tập trung hóa và vẫn giữ được sự hiệu quả cần thiết vẫn là vấn đề mở hiện nay đối với các hệ mã hóa quảng bá đa kênh. Ngoài ra, còn một vấn đề mở cần giải quyết là các hệ mã hóa quảng bá đa kênh hiện nay có độ dài khóa bí mật cũng như khóa công khai dài. Với những ứng dụng như IoT, khi năng lực người dùng yếu thì đây là vấn đề thực sự cần phải giải quyết.

CHƯƠNG 3: HỆ MÃ HÓA DỰA TRÊN THUỘC TÍNH

Chương 3 trình bày giới thiệu chung về hệ mã hóa dựa trên thuộc tính bao gồm: Định nghĩa chung về mã hóa dựa trên thuộc tính, mô hình an toàn của mã hóa dựa trên thuộc tính, một số hệ mã hóa dựa trên thuộc tính quan trọng hiện nay. Nội dung chương, tác giả sẽ trình bày 02 hệ mã hóa dựa trên thuộc tính mới được đề xuất, đó cũng chính là đóng góp mới trong luận án.

3.1. Định nghĩa và mô hình an toàn của hệ mã hóa dựa trên thuộc tính

Mã hóa dựa trên thuộc tính được giới thiệu bởi Sahai và Waters [53], là mở rộng của mã hóa quảng bá, trong đó cho phép điều kiện giải mã linh động hơn so với mã hóa quảng bá. Vấn đề khó khăn với mã hóa quảng bá là người lập mã phải biết cụ thể tập người dùng có thể giải mã được tại thời điểm lập mã, tuy nhiên trong thực tế người lập mã không phải lúc nào cũng biết được điều này. Ví dụ, công ty FPT lưu trữ dữ liệu của họ trên đám mây, họ muốn lưu trữ một văn bản cho phép các nhân viên của phòng kỹ thuật và phòng hỗ trợ khách hàng, đồng thời tham gia trong dự án e-Health có thể giải mã được.

Với kỹ thuật mã hóa quảng bá, công ty FPT phải biết ngay tại thời điểm mã hóa văn bản là những nhân viên cụ thể nào của hai phòng trên tham gia vào dự án e-Health. Tuy nhiên, trong thực tế, do tính chất công việc dự án e-Health có thể thêm nhân viên từ các phòng trên để kịp thời giải quyết công việc. Điều đó dẫn đến tồn tại hạn chế đó là, công ty FPT phải thực hiện lại quá trình mã hóa văn bản và đẩy lên trên Cloud, gây tốn kém, mất thời gian… Hiển nhiên là không hợp lý.

Mã hóa dựa trên thuộc tính được phát triển để giải quyết những vấn đề như vậy, trong một hệ thống mã hóa thuộc tính. Tùy ý, ta có thể định nghĩa một tập các thuộc tính. Ví dụ, trong công ty FPT có dự án e-Health (e-H), phòng kỹ thuật (PKT), phòng chăm sóc khách hàng (PCS), nhân viên (NV), trưởng phòng (TP),... Là các thuộc tính. Nếu người dùng X thuộc phòng kỹ thuật, là nhân viên và tham gia dự án e-Health thì sẽ nhận các thuộc tính là PKT, e-H, NV và nhận khóa bí mật tương ứng với các thuộc tính này. Công ty FPT khi mã hóa văn bản chỉ đơn giản là thực hiện việc mã hóa trong đó quy định rằng những nhân viên của hai phòng này và làm trong

dự án e-Health có thể giải mã được mà không cần biết cụ thể là nhân viên nào. Điều kiện giải mã có thể được mô tả bằng một biểu thức boolean như sau:

(NV and PKT and e-H) or (NV and PCS and e-H) Khi một nhân viên mới thuộc một trong hai phòng này tham gia dự án, người này sẽ nhận thêm thuộc tính là dự án e-Health và nhận khóa bí mật tương ứng, khi đó nhân viên mới này sẽ có khả năng giải mã vì đáp ứng được điều kiện giải mã.

3.1.1. Định nghĩa

Như vậy, với một hệ mã hóa dựa trên thuộc tính có hai khái niệm quan trọng, điều kiện giải mã hay còn gọi là chính sách truy cập, thường có các dạng như sau:

– AND-gates: là biểu thức boolean chỉ bao gồm duy nhất phép AND, ví dụ như: (NV and PKT and e-H), do vậy, muốn giải mã được người dùng phải có đồng thời các thuộc tính NV, PKT và e-H;

– Threshold: Không quan tâm đến thuộc tính là gì, miễn là người dùng sở hữu số thuộc tính lớn hơn một ngưỡng nào đó do người lập mã quy định là có thể giải mã;

– Boolean: Có dạng như một biểu thức Boolean với các phép AND, OR,...

Thuộc tính: Tùy thuộc vào việc cài đặt hệ mã hóa vào ứng dụng cụ thể nào mà ta sẽ định nghĩa các thuộc tính tương ứng. Như ví dụ trên, ta có các thuộc tính là dự án e-Health (e-H), phòng kỹ thuật (PKT), phòng chăm sóc khách hàng (PCS), nhân viên (NV), trưởng phòng (TP),...

Nếu chính sách truy cập nằm ở bản mã, và người dùng dựa vào thuộc tính để nhận khóa bí mật (như ví dụ trên) thì được gọi là hệ mã hóa dựa trên thuộc tính có chính sách ở bản mã. Còn ngược lại, nếu mỗi người dùng trong hệ thống có tương ứng một chính sách truy cập và nhận khóa bí mật tương ứng với chính sách này trong khi việc mã hóa dựa trên các thuộc tính thì hệ mã hóa dựa trên thuộc tính đó được gọi là hệ mã hóa dựa trên thuộc tính có chính sách ở khóa.

Sau đây chúng ta xem xét định nghĩa của một hệ mã hóa dựa trên thuộc tính có chính sách ở bản mã, với hệ KP-ABE là hoàn toàn tương tự. Do hệ CP-ABE có ứng dụng thực tế quan trọng hơn so với hệ KP-ABE, nên từ nay về sau trong luận án chủ yếu trình bày về hệ CP-ABE, việc ánh xạ sang hệ KP-ABE sẽ được bỏ qua. Một

hệ mã hóa dựa trên thuộc tính có chính sách ở bản mã như vậy, được hiểu và trình bày chung như sau:

Khởi tạo $(\lambda, n, \{S_u\}_{u\in\mathcal{U}}$:

Đầu vào của giải thuật khởi tạo là tham số an toàn λ, số tối đa các thuộc tính của hệ thống *n*, danh sách các thuộc tính của từng người dùng *u* trong hệ thống, $\mathcal{U}$ là danh sách người dùng. Trong đó tham số an toàn λ nghĩa là để phá được hệ mã này kẻ tấn công cần thực hiện ít nhất 2^{λ} phép toán. Đầu ra của giải thuật là khóa công khai và khóa bí mật của hệ thống.

Tạo khóa (msk, *u*, S_u, param):

Đầu vào của giải thuật là khóa bí mật của hệ thống, định danh của người dùng thứ *u*, danh sách các thuộc tính S_u của người dùng *u*, và khóa công khai của hệ thống. Giải thuật sẽ trả về khóa bí mật sk_u của người dùng *u*. Giải thuật này cũng có thể được tích hợp luôn vào giải thuật khởi tạo ở trên.

Mã hóa ($\mathbb{A}$, param):

Đầu vào của giải thuật là chính sách mã hóa $\mathbb{A}$ và khóa công khai của hệ thống. Đầu ra của giải thuật là khóa phiên *K* và bản mã Hdr bao gồm cả mô tả của $\mathbb{A}$.

Giải mã (sk_u, Hdr, param):

Đầu vào của giải thuật là khóa bí mật của người dùng *u*, bản mã Hdr và khóa công khai của hệ thống. Đầu ra của giải thuật là khóa phiên làm việc *K* nếu như S_u thỏa mãn $\mathbb{A}$. Ngược lại nếu S_u không thỏa mãn $\mathbb{A}$ thì đầu ra là $\perp$.

Một hệ mã hóa với cơ chế như trên được gọi là hệ mã hóa lai. Lý do, hệ mã hóa lai được dùng trong thực tế, là do nó tận dụng được cả hai ưu thế của mã hóa khóa công khai truyền thống và mã hóa khóa bí mật. Cụ thể, nhược điểm của mã hóa khóa công khai là có tốc độ mã hóa chậm, trong khi ưu điểm là không cần thống nhất khóa bí mật chung giữa người gửi và người nhận. Còn nhược điểm của mã hóa khóa bí mật là phải thống nhất trước khóa bí mật chung giữa người gửi và người nhận, trong khi ưu điểm là tốc độ mã hóa nhanh. Hệ mã hóa lai là tận dụng lợi thế của cả hai hệ mã hóa này. Cụ thể, khóa phiên làm việc *K* ngắn sẽ được mã hóa bằng hệ mã hóa khóa công khai có tốc độ chậm, còn dữ liệu dài sẽ được mã hóa bằng hệ mã hóa

khóa bí mật có tốc độ nhanh dưới khóa phiên *K*. Như vậy, với hệ mã hóa lai giữa người gửi và người nhận không cần thống nhất trước khóa bí mật chung, dữ liệu được mã hóa bằng hệ mã hóa khóa bí mật.

Và cũng giống như hệ mã hóa quảng bá, để đơn giản ta chỉ xét việc mã hóa và giải mã của khóa phiên làm việc *K*, do việc mã hóa và giải mã dữ liệu thực tế dùng *K* như là khóa bí mật là giống nhau ở tất cả các hệ mã hóa dựa trên thuộc tính.

3.1.2. Mô hình an toàn

Mô hình an toàn chung cho một hệ mã hóa dựa trên thuộc tính được định nghĩa như sau:

Xét một kịch bản giữa kẻ tấn công $\mathcal{A}$ và kẻ thách thức $\mathcal{C}$ (đại diện cho sự an toàn của hệ mã), chúng ta thấy:

Giai đoạn đầu của kịch bản, kẻ thách thức $\mathcal{C}$ và kẻ tấn công $\mathcal{A}$ được cho trước danh sách *n* thuộc tính của hệ thống. Tiếp theo, $\mathcal{A}$ cho đầu ra là chính sách bản mã $\mathbb{A}^*$ và danh sách thuộc tính $\{S_{u^*}\}_{u^* \in \mathcal{U}}$ của người dùng u^* mà $\mathcal{A}$ sẽ tấn công.

Khởi tạo: $\mathcal{C}$ chạy giải thuật Khởi tạo $(\curlywedge, n, \{S_u\}_{u \in \mathcal{U}})$ và gửi cho $\mathcal{A}$ khóa công khai của hệ thống.

Giai đoạn truy vấn 1: Kẻ tấn công $\mathcal{A}$ có thể tùy ý để yêu cầu biết khóa bí mật sk_u tương ứng với những người dùng u sao cho S_u không thỏa mãn chính sách giải mã $\mathbb{A}^*$ (lưu ý rằng nếu như S_u thỏa mãn $\mathbb{A}^*$thì $\mathcal{A}$ có thể dễ dàng dùng luôn khóa bí mật này để giải bản mã thách thức, như vậy mô hình an toàn không có nghĩa). $\mathcal{C}$ gửi sk_u cho $\mathcal{A}$.

Giai đoạn thách thức: Kẻ thách thức $\mathcal{C}$ chạy giải thuật **Mã hóa** ($\mathbb{A}^*$, param): Để thu về Hdr và khóa phiên $K \in \mathcal{K}$. Tiếp theo, kẻ thách thức chọn ngẫu nhiên một bít b, đặt $K' = K$ nếu $b = 0$, $K' \xleftarrow{\$} \mathcal{K}$; Nếu $b = 1$ và cuối cùng gửi $(K'$, Hdr$)$ cho $\mathcal{A}$.

Giai đoạn truy vấn 2: Tương tự như giai đoạn truy vấn 1.

Giai đoạn dự đoán: Kẻ tấn công $\mathcal{A}$ cho đầu ra là bít dự đoán b'.

Thông thường, $\mathcal{A}$ chiến thắng nếu $b = b'$, và lợi thế của nó là:

$$Adv^{ind}(\curlywedge, n, \{S_u\}_{u \in U}, \mathcal{A}) = |2Pr[b = b'] - 1|$$

Hệ mã hóa được định nghĩa an toàn như sau:

Định nghĩa 3.1: Một hệ mã hóa dựa trên thuộc tính, được gọi là đạt an toàn CPA, nếu tất cả các kẻ tấn công chạy trong thời gian đa thức có lợi thế trong kịch bản tấn công ở trên là nhỏ không đáng kể.

Cũng như mã hóa quảng bá, khái niệm an toàn ở trên gọi là an toàn không phân biệt được khóa, tức là kẻ tấn công không có khả năng phân biệt giữa một khóa phiên *K* đúng và một giá trị ngẫu nhiên. Một khái niệm an toàn yếu hơn gọi là an toàn không tính toán được khóa, tức là kẻ tấn công chỉ không có khả năng tính ra được khóa phiên *K*. Tuy nhiên, nó có thể có khả năng phân biệt giữa một khóa phiên *K* đúng và một giá trị ngẫu nhiên. Ngoài ra, nếu như kẻ tấn công $\mathcal{A}$ không phải công bố người dùng mà nó muốn tấn công ở giai đoạn khi bắt đầu kịch bản tấn công thì mô hình an toàn được gọi là an toàn mạnh.

3.2. Một số hệ mã hóa dựa trên thuộc tính nền tảng quan trọng hiện nay

3.2.1. Hệ mã hóa dựa trên thuộc tính của Rouselakis-Waters năm 2013

Rouselakis và Waters viết trong tài liệu tham khảo [52] vào năm 2013, tại một trong những hội nghị quan trọng nhất trong ngành an toàn bảo mật thông tin ACM CCS đã trình bày hệ mã dựa trên định danh mới có các tính chất quan trọng sau:

Số lượng tối đa các thuộc tính có trong hệ thống là bằng đúng số nguyên tố *p*, như vậy với số nguyên tố *p* khoảng 160 bít, thì số lượng tối đa các thuộc tính có thể là 2^{160}. Hệ có số lượng các thuộc tính lớn như vậy gọi là hệ có tính chất không giới hạn số thuộc tính. Hệ mã đạt được một số tính chất như:

- Độ dài của khóa công khai chỉ là hằng số.
- Các giải thuật mã hóa và giải mã hiệu quả.

Tuy nhiên, điểm yếu của hệ mã này là:

- Độ dài bản mã còn dài.
- Đạt an toàn chưa cao, cụ thể chỉ là CPA.
- Độ dài khóa bí mật còn dài.

Về mặt kỹ thuật, hệ Rouselakis-Waters13 dựa trên ma trận chia sẻ bí mật tuyến tính được mô tả như sau:

Giả sử p là số nguyên tố và $\mathcal{U}$ là tập các thuộc tính. Nếu $\mathbb{A}$ là một chính sách mã hóa dựa trên $\mathcal{U}$, thì ta có thể tìm một ma trận LSS, $M \in \mathbb{Z}_p^{\ell \times n}$ và một hàm ρ mà ánh xạ các dòng trong ma trận M với các thuộc tính trong $\mathcal{U}$ mà các thuộc tính này xuất hiện trong $\mathbb{A}$, tức là hàm ρ có dạng $\rho \in \mathcal{F}([\ell] \to \mathcal{B})$.

Cặp *(M, ρ)* được gọi là một chính sách mã hóa LSS. Và khi vector $\vec{y} = (s, y_2, \dots, y_n)^{\perp} \overset{\$}{\leftarrow} \mathbb{Z}_p^n$ với số bí mật s cần chia sẻ, thì vector chia sẻ bí mật sẽ là $\overrightarrow{\lambda} = M.\vec{y}$

Đặt S là một tập các thuộc tính mà thỏa mãn chính sách mã hóa $\mathbb{A}$ (hay chính là *(M, ρ)), I* là tập các dòng của ma trận M mà ánh xạ qua hàm ρ xuất hiện trong S, tức là $I = \{i | i \in [\ell] \wedge \rho(i) \in S\}$.

Như vậy, người ta đã chứng minh được rằng: Tồn tại các hằng số $\{\omega_i\}_{i \in I}$ trong $\mathbb{Z}_p$ sao cho với mọi giá trị chia sẻ hợp lệ $\left\{\lambda_i = \left(M.\vec{y}\right)_i\right\}_{i \in I}$ của thành phần bí mật s thì $\sum_{i \in I} \omega_i \lambda_i = s$ và các hằng số $\{\omega_i\}_{i \in I}$ có thể được tính toán được.

Hệ Rouselakis-Waters 13 trong tài liệu tham khảo [52] được mô tả như sau:

Khởi tạo (1^{λ}):

Đầu vào của giải thuật là tham số an toàn λ, giải thuật tạo ra khóa công khai và khóa bí mật của hệ thống như sau:

Đầu tiên biểu diễn tập các thuộc tính là $\mathcal{U} = \mathbb{Z}_p$, với p là số nguyên tố, tạo ra hệ thống ánh xạ song tuyến $D = (p, \mathbb{G}, \mathbb{G}_T, e)$.

Chọn ngẫu nhiên $g, u, h, \omega, v \overset{\$}{\leftarrow} \mathbb{G}$ và $\alpha \overset{\$}{\leftarrow} \mathbb{Z}_p$. Cuối cùng cho đầu ra:

$$\text{param} = (D, g, u, h, \omega, v, e(g,g)^{\alpha}) \tag{3.1}$$

và

$$\text{msk} = (\alpha)$$

Tạo khóa (msk, $S = (A_1, A_2, \dots, A_k) \subseteq \mathbb{Z}_p$):

Đầu vào là tập các thuộc tính S và khóa bí mật của hệ thống. Giải thuật đầu tiên chọn k+1 phần tử ngẫu nhiên $r, r_1, \dots, r_k \overset{\$}{\leftarrow} \mathbb{Z}_p$

Khóa bí mật $sk = (K_0, K_{1,} K_{j,2}, K_{j,3}), _{j \in [k]}$ sau đó được tính như sau:

$$K_0 = g^{\alpha}\omega^{r}, K_1 = g^{r}, \left(K_{j,2} = g^{r_j}, K_{j,3} = (u^{A_j}h)^{r_j}v^{-r}\right)_{j\in[k]} \quad (3.2)$$

Mã hóa$((M,\rho) \in \mathbb{Z}_p^{\ell\times n}, \mathcal{F}(\lceil\ell\rceil \to \lceil\mathbb{Z}_p\rceil)$, param):

Đầu vào của giải thuật là chính sách mã hóa (M,ρ) và khóa công khai của hệ thống. Giải thuật đầu tiên, chọn một vector ngẫu nhiên $\vec{\mathcal{Y}} = (s, \mathcal{Y}_2, \dots, \mathcal{Y}_n)^{\perp} \overset{\$}{\leftarrow} \mathbb{Z}_p^{n\times 1}$. Lưu ý: *s* là thành phần bí mật của ma trận LSS. Do vậy, Vector chia sẻ thành phần bí mật *s* là:

$$\vec{\lambda} = (\lambda_1, \dots, \lambda_\ell)^{\perp} = M\vec{\mathcal{Y}}$$

Giải thuật tiếp theo, chọn ℓ số ngẫu nhiên $t_1, \dots, t_\ell \overset{\$}{\leftarrow} \mathbb{Z}_p$, Hdr được tính như sau:

$$K = e(g,g)^{\alpha s}, C_0 = g^{s}, \left(C_{i,1} = \omega^{\lambda_i}v^{t_i}, C_{i,2} = \left(u^{\rho(i)}h\right)^{-t_i}, C_{i,3} = g^{t_i}\right)_{i\in[\ell]} \quad (3.3)$$

Cuối cùng, giải thuật cho đầu ra khóa phiên bí mật *K* và bản mã:

Hdr = $\left(C_0, \left(C_{i,1}, C_{i,2}, C_{i,3}\right)_{i\in[\ell]}\right)$ bao gồm cả mô tả của *(M, ρ).*

Giải mã (*sk*, Hdr, param):

Đầu vào của giải thuật là khóa bí mật của người dùng *sk*, Hdr và khóa công khai của hệ thống. Giải thuật đầu tiên, tìm tập các dòng *I* trong ma trận *M*.

I = {i: ρ(i) ∈ S}. Tiếp theo, giải thuật tính các hằng số $\{\omega_i\}_{i\in I}$

Sao cho $\sum_{i\in I}\omega_i M_i = (1,0,\dots,0)$. Trong đó, M_i là dòng thứ *i* của ma trận *M*. Chú ý rằng, nếu *S* là tập các thuộc tính mà thỏa mãn chính sách mã hóa *(M, ρ)* thì các hằng số ω_i như trên sẽ tồn tại.

Cuối cùng, giải thuật tính:

$$K = \frac{e(C_0K_0)}{\prod_{i\in I}\left(e(C_{i,1},K_1)e(C_{i,2},K_{j,2})e(C_{i,3},K_{j,3})\right)^{\omega_i}} \quad (3.4)$$

trong đó *j* là chỉ số của thuộc tính *ρ(i)* trong tập S.

3.2.2. Hệ mã hóa dựa trên thuộc tính của Agrawal-Chase17

Cũng tại hội nghị ACM CCS vào năm 2017, Agrawal và Chase trong tài liệu tham khảo số [5], đã đề xuất một hệ mã hóa dựa trên thuộc tính có các tính chất quan trọng sau:

• Đạt an toàn ở mức cao, cụ thể là CPA

• Không giới hạn kích thước của chính sách mã hóa. Lưu ý rằng, chính sách mã hóa thường là một biểu thức Boolean. Trong các hệ khác thì biểu thức boolean này có kích thước giới hạn được quy định ở giai đoạn khởi tạo. Các tham số khác như: Khóa công khai của hệ thống sẽ phụ thuộc vào kích thước của biểu thức boolean. Nếu để quá lớn thì hệ thống sẽ không hiệu quả, nếu để bé thì có thể không đáp ứng được yêu cầu về mã hóa. Hệ chỉ giải quyết được vấn đề này khi không quy định kích thước tối đa của biểu thức boolean.

• Độ dài của khóa công khai là hằng số.

• Số lượng tối đa các thuộc tính có trong hệ thống là không giới hạn, cụ thể là bằng đúng số nguyên tố p. Vậy, với số nguyên tố p khoảng 160 bít, thì số lượng tối đa các thuộc tính là 2^{160}.

• Các giải thuật mã hóa và giải mã hiệu quả

Tuy nhiên, điểm yếu của hệ mã này là:

• Độ dài bản mã còn dài

• Độ dài khóa bí mật còn dài

Cụ thể hệ Agrawal-Chase17 được mô tả như sau:

Khởi tạo (1^{λ}):

Đầu vào của giải thuật là tham số an toàn λ, giải thuật tạo ra khóa công khai và khóa bí mật của hệ thống như sau:

Đầu tiên, biểu diễn tập các thuộc tính là $\mathcal{U} = \mathbb{Z}_p$, với p là số nguyên tố, tạo ra hệ thống ánh xạ song tuyến $D = (p, \mathbb{G}, \mathbb{G}_T, e)$.

Chọn ngẫu nhiên $g, h \xleftarrow{\$} \mathbb{G}, a_1, a_2, d_1, d_2, d_3, b_1, b_2 \xleftarrow{\$} \mathbb{Z}_p^*$ và $\mathcal{H}$ là một hàm băm. Cuối cùng cho đầu ra:

$$\text{param} = (D, h, H_1 = h^{a_1}, H_2 = h^{a_2}, T_1 = e(g,h)^{d_1 a_1 + d_3},\ T_2 = e(g,h)^{d_2 a_2 + d_3}, \mathcal{H}) \tag{3.5}$$

và

$$\text{msk} = (g, h, a_1, a_2, b_1, b_2, g^{d_1}, g^{d_2}, g^{d_3}) \tag{3.6}$$

Tạo khóa (msk, *S*):

Đầu vào là tập các thuộc tính *S* và khóa bí mật của hệ thống. Giải thuật đầu tiên chọn hai phần tử ngẫu nhiên $r, r_1 \overset{\$}{\leftarrow} \mathbb{Z}_p$ và tính:

$$sk_0 = (h^{b_1 r_1}, h^{b_2 r_2}, h^{r_1+r_2}) \tag{3.7}$$

Tiếp theo, dùng *h*, b_1, b_2 từ msk, với mọi $\mathcal{Y} \in S$ và *t* = 1, 2 tính:

$$sk_{\mathcal{Y},t} = \mathcal{H}(\mathcal{Y}1t)^{\frac{b_1 r_1}{a_t}} \cdot \mathcal{H}(\mathcal{Y}2t)^{\frac{b_2 r_2}{a_t}} \cdot \mathcal{H}(\mathcal{Y}3t)^{\frac{r_1+r_2}{a_t}} \cdot g^{\frac{\sigma_{\mathcal{Y}}}{a_t}} \tag{3.8}$$

trong đó $\sigma_{\mathcal{Y}} \overset{\$}{\leftarrow} \mathbb{Z}_p$. Đặt $sk_{\mathcal{Y}} = \left(sk_{\mathcal{Y},1}, sk_{\mathcal{Y},2}, g^{-\sigma_{\mathcal{Y}}}\right)$ tính:

$$sk'_t = g^{d_t} \cdot \mathcal{H}(011t)^{\frac{b_1 r_1}{a_t}} \cdot \mathcal{H}(012t)^{\frac{b_2 r_2}{a_t}} \cdot \mathcal{H}(013t)^{\frac{r_1+r_2}{a_t}} \cdot g^{\frac{\sigma'}{a_t}} \tag{3.9}$$

với *t* = 1, 2 và $\sigma' \overset{\$}{\leftarrow} \mathbb{Z}_p$. Đặt $sk' = \left(sk_1, sk_2, g^{d_3}, g^{-\sigma'}\right)$

Cuối cùng cho đầu ra khóa bí mật của người dùng là $sk = \left(sk_0, \{sk_{\mathcal{Y}}\}_{\mathcal{Y} \in S'}, sk'\right)$.

Mã hóa$((M, \rho) \in \mathbb{Z}_p^{\ell \times n}, \mathcal{F}(\lceil \ell \rceil \to \lceil \mathbb{Z}_p \rceil)$**,** param):

Đầu vào của giải thuật là chính sách mã hóa (M, ρ) và khóa công khai của hệ thống. Giải thuật đầu tiên chọn ngẫu nhiên $s_1, s_2 \in \mathbb{Z}_p$, tính:

$$ct_0 = \left(H_1^{s_1}, H_2^{s_2}, h^{s_1+s_2}\right) \tag{3.10}$$

Sau đó, với i = 1,.., ℓ *và* k = 1, 2, 3 tính:

$$ct_{i,k} = \mathcal{H}(\rho(i)k1)^{s_1} \cdot \mathcal{H}(\rho(i)k2)^{s_2} \cdot \prod_{j=1}^{n} [\mathcal{H}(0jk1)^{s_1} \cdot \mathcal{H}(0jk2)^{s_2}]^{(M)_{i,j}} \tag{3.11}$$

Trong đó $(M)_{i,j}$ là phần tử thứ *(i, j)* của ma trận *M*. Đặt $ct_i = \left(ct_{i_1}, ct_{i_2}, ct_{i_3}\right)$ tính khóa phiên:

$$K = T_1^{s_1} \cdot T_2^{s_2}$$

Giải thuật cho đầu ra là bản mã Hdr = *(*ct_0*,* ct_1*,...,*ct_l*)* bao gồm cả mô tả của (M, ρ), và khóa phiên là *K*.

Giải mã (*sk*, Hdr, param):

Đầu vào của giải thuật là khóa bí mật của người dùng s*k*, bản mã Hdr và khóa công khai của hệ thống.

Giải thuật đầu tiên tìm tập các dòng *I* trong ma trận *M* . Sao cho *I* = {$i : \rho(i) \in S$}. Tiếp theo, giải thuật tính các hằng số $\{\omega_i\}_{i\in I}$ sao cho $\sum_{i\in I} \omega_i M_i = (1,0,\dots,0)$. Trong đó, M_i là dòng thứ *i* của ma trận *M*. Chú ý rằng, nếu *S* là tập các thuộc tính mà thỏa mãn chính sách mã hóa (M, ρ), thì các hằng số ω_i như trên sẽ tồn tại.

Cuối cùng, giải thuật tính:

$$X_1 = e\left(sk'_1 . \prod_{i\in I} sk_{\rho(i),1}^{\omega_i}, ct_{0,1}\right) . e\left(sk'_2 . \prod_{i\in I} sk_{\rho(i),2}^{\omega_i}, ct_{0,2}\right) . e\left(sk'_3 . \prod_{i\in I} sk_{\rho(i),3}^{\omega_i}, ct_{0,3}\right) \quad (3.12)$$

Và

$$X_2 = e\left(\prod_{i\in I} ct_{i,1}^{\omega_i}, sk_{0,1}\right) . e\left(\prod_{i\in I} ct_{i,2}^{\omega_i}, sk_{0,2}\right) . e\left(\prod_{i\in I} ct_{i,3}^{\omega_i}, sk_{0,3}\right) \quad (3.13)$$

Và cuối cùng tính:

$$K = \frac{X_1}{X_2}$$

Lưu ý rằng, $sk_{0,1}, sk_{0,2}, sk_{0,3}$ là phần tử thứ nhất, thứ hai, thứ ba của sk_0 tương tự đối với ct_0.

3.3. Mã hóa dựa trên thuộc tính (CP-ABE-01) đề xuất

Mục này tác giả đề xuất một lược đồ mã hóa dựa trên thuộc tính có các ưu điểm sau:

• Có độ dài bản mã Hdr là 3 phần tử.

• Hỗ trợ chính sách giải mã là một biểu thức boolean, cụ thể là có dạng CNF.

• Tốc độ mã hóa và giải mã là hiệu quả. Để giải mã, người dùng cần tính *2m* pairings. Chi tiết hơn, NCS cài đặt và trình bày so sánh đề xuất trong luận án với các hệ mã hóa khác trong Bảng 2.2.

Điểm yếu của đề xuất là:

• Độ dài của khóa bí mật vẫn dài, cụ thể là độ dài của khóa bí mật là tuyến tính với *N*. Trong đó, *N* là tích của số các thuộc tính trong hệ thống *n* và số các mệnh đề *m* trong biểu thức CNF.

• Độ dài của khóa công khai vẫn dài, và số lượng tối đa thuộc tính có trong hệ thống là giới hạn.

• Hệ chỉ đạt an toàn CPA.

3.3.1. Ý tưởng xây dựng

Ý tưởng chính, là biến đổi một hệ mã hóa quảng bá đa kênh thành một hệ mã hóa dựa trên thuộc tính. Với mục đích đó tác giả xem mỗi tập S_i trong hệ MCBE đã đề xuất trình bày ở chương 3, như một mệnh đề $\tilde{\beta}_i$ trong biểu thức boolean CNF (là chính sách giải mã). Khóa phiên *K* trong hệ CP-ABE lúc này chính là tích của tất cả các khóa phiên con trong hệ MCBE. Cụ thể khóa phiên $K = \prod_{i=1}^{t} e(g,h)^{k.\beta_i}$.

Tiếp theo, tác giả xem mỗi chỉ số $i \in \{1,..., n\}$ trong hệ MCBE như là một thuộc tính trong hệ CP-ABE. Ngoài ra, để cho mỗi thuộc tính có thể được dùng lại nhiều lần trong chính sách bản mã, mỗi thuộc tính có *m* bản sao, tức là nếu một thuộc tính khi được dùng lại thì sẽ dùng một bản sao khác. Như vậy, mỗi thuộc tính có thể dùng lại tối đa *m* lần, *m* là số tối đa các mệnh đề trong biểu thức boolean CNF. Nếu một người dùng trong hệ CP-ABE sở hữu một thuộc tính $i \in \{1,..., n\}$, người dùng đó sẽ nhận khóa bí mật tương ứng với chỉ số $\{i, j\}_{j=1,...,m}$ trong hệ MCBE.

Để có khả năng giải mã (tức là tính khóa phiên *K*), người dùng phải tính được tất cả các khóa phiên con $e(g,h)^{k.\beta_i}, i = 1, \dots, t$. Trường hợp đó có thể dẫn đến một khả năng, người dùng liên kết lại với nhau có thể giải mã. Bởi vì khi họ liên kết với nhau thì có thể sẽ tính được tập thuộc tính thỏa mãn chính sách bản mã. Để giải quyết vấn đề hợp tác giải mã của người dùng, tức là mỗi người dùng kết hợp lại với nhau cũng chưa đưa ra được tập thuộc tính để thỏa mãn chính sách bản mã. Để hạn chế người dùng để họ không tính được chính sách bản mã (không có quyền giải mã), ta dùng một thành phần ngẫu nhiên khác nhau cho mỗi lần tạo khóa bí mật cho người dùng *u*.

3.3.2. Mã hóa đề xuất và so sánh

Mã hóa đề xuất được xây dựng như sau:

Khởi tạo$(1^{\lambda}, n, \{S_u\}_{u \in \mathcal{U}})$: Giả sử rằng λ là tham số an toàn, *n* là số tối đa các thuộc tính trong hệ thống. Ký hiệu *m* là số tối đa các mệnh để trong biểu thức boolean, CNF, đặt $N = m \cdot n$. Mỗi người dùng u sở hữu một tập các thuộc tính $S_u \subset \{1,..., n\}$. Mỗi thuộc tính A_i, $i = 1,.., n$, có *m* bản sao của chính nó, đặt $\mathcal{B} = \{A_{1,1}, \dots, A_{n,m}\}$ là

tập tất cả các thuộc tính, ký hiệu $\mathcal{B}_u = (A_{i,j} \in \mathbb{Z}_p^*)_{\substack{i\in S_u \\ j=1,\dots,m}}$. Giải thuật tạo ra tham số công khai cho hệ thống và khóa bí mật cho người dùng $u \in \mathcal{U}$ như sau: (lưu ý, ở đây gộp luôn giải thuật khởi tạo và tạo khóa):

Giải thuật đầu tiên tạo ra hệ thống ánh xạ song tuyến $D = (p, \mathbb{G}, \widetilde{\mathbb{G}}, \mathbb{G}_T, e)$, chọn ngẫu nhiên $h \xleftarrow{\$} \widetilde{\mathbb{G}}, g \xleftarrow{\$} \mathbb{G}$ và $\alpha, \gamma, \beta_1, \dots, \beta_m \xleftarrow{\$} \mathbb{Z}_p^*$. Tiếp theo, giải thuật cho đầu ra tham số công khai của hệ thống:

$$\text{param} = \left(D, \mathcal{B}, \{h^{\alpha^j}\}_{j=0,\dots,N}, \{h^{\beta_j}\}_{j=1,\dots,m}, g^{\alpha}, \{e, (g,h)^{\gamma\beta_i}\}_{i=1,\dots,m}\right) \quad (3.14)$$

Để tạo ra khóa bí mật sk_u, giải thuật đầu tiên chọn ngẫu nhiên $s_u \xleftarrow{\$} \mathbb{Z}_p^*$, sau đó tính:

$$sk_u = \left(\{g^{\frac{\beta_j s_u}{\alpha + A_{i,j}}}\}_{\substack{i\in S_u \\ j=1,\dots,m}}, \{h^{\alpha^i \beta_j s_u}\}_{\substack{i=0,\dots,N \\ j=1,\dots,m}}, g^{s_u+\gamma}\right) \quad (3.15)$$

Lưu ý rằng sk_u cũng bao gồm S_u.

Mã hóa (param, $\tilde{\beta} = \tilde{\beta}_1 \wedge \cdots \wedge \tilde{\beta}_t$):

Đầu vào của giải thuật là khóa công khai param và chính sách mã hóa $\tilde{\beta}$.

Giải thuật đầu tiên chọn ngẫu nhiên $k \in \mathbb{Z}_p^*$ sau đó tính khóa phiên:

$$K = e(g,h)^{k\gamma \sum_{i=1}^{t} \beta_i}$$

Lưu ý: $t \leq m$ và giải thuật có $\{e(g,h)^{\gamma\beta_i}\}_{i=1,\dots,m}$ từ param.

Để tính Hdr, giải thuật tính Hdr = (C_1, C_2, C_3), trong đó:

$$C_1 = g^{-\alpha.k} \; ; C_2 = h^{k \sum_{i=1}^{t} \beta_i}$$

Và

$$C_3 = h^{k.\prod_{i\in\tilde{\beta}_1}(\alpha + A_{i,1}) \dots \prod_{i\in\tilde{\beta}_t}(\alpha + A_{i,t})}$$

Cuối cùng giải thuật cho đầu ra K và Hdr = (C_1, C_2, C_3) bao gồm cả $\tilde{\beta}$.

Giải mã (sk_u, Hdr, param):

Giải thuật đầu tiên kiểm tra xem S_u có thỏa mãn $\tilde{\beta}$ không?

Trường hợp 1: S_u không thỏa mãn $\tilde{\beta}$, giải thuật trả về $\perp$.

Trường hợp 2: S_u có thỏa mãn $\tilde{\beta}$, giải thuật tính khóa phiên thành phần K_1 như sau:

Giải thuật đầu tiên chọn $i' \in (\tilde{\beta}_1 \cap S_u)$ sau đó tính $K'_1 = h^{\emptyset}$ trong đó:

$$\emptyset = \frac{\beta_1 s_u}{\alpha}\left(\prod_{\substack{i\in\widetilde{\beta}_1\\ i\neq i'}}(\alpha + A_{i,1}) \dots \prod_{i\in\widetilde{\beta}_t}(\alpha + A_{i,t}) - \prod_{\substack{i\in\widetilde{\beta}_1\\ i\neq i'}} A_{i,1} \dots \prod_{i\in\widetilde{\beta}_t} A_{i,t}\right) \quad (3.17)$$

Giải thuật cũng có thể tính K_1' từ $\left\{h^{\alpha^i \beta_j s_u}\right\}_{\substack{i=0,\dots,N\\ j=1,\dots,m}}$.

Đặt:

$$B_1 = \prod_{\substack{i\in\widetilde{\beta}_1\\ i\neq i'}} A_{i,1} \dots \prod_{i\in\widetilde{\beta}_t} A_{i,t}$$

Tính:

$$K_1 = \left(e(C_1 K_1') \,.\, e(g^{\frac{\beta_1 s_u}{\alpha + A_{i',1}}}, C_3)\right)^{\frac{1}{B_1}} \quad (3.18)$$

$$= \left(e\left(g^{-\alpha.k} h^{\theta}\right).e\left(g^{\frac{\beta_1 s_u}{\alpha + A_{i',1}}}, h^{k.\prod_{i\in\widetilde{\beta}_1}(\alpha + A_{i,1}) \dots \prod_{i\in\widetilde{\beta}_t}(\alpha + A_{i,t})}\right)\right)^{\frac{1}{B}} \quad (3.19)$$

$$= \left(e(g,h)^{k\beta_1 s_u \prod_{\substack{i\in\widetilde{\beta}_1\\ i\neq i'}} A_{i,1} \dots \prod_{i\in\widetilde{\beta}_t}(\alpha + A_{i,t})}\right)^{\frac{1}{B_1}} \quad (3.20)$$

$$= e(g,h)^{k\beta_1 s_u} \quad (3.21)$$

Tương tự, tính K_2,.., K_t, sau đó tính:

$$K' = \prod_{i=1}^{t} K_i = e(g,h)^{k s_u \sum_{i=1}^{t} \beta_i} \quad (3.22)$$

Cuối cùng tính:

$$K = \frac{e(g^{s_u + \gamma}, C_2)}{K'} = \frac{e\left(g^{s_u + \gamma}, h^{k\sum_{i=1}^{t}\beta_i}\right)}{e(g,h)^{k s_u \sum_{i=1}^{t}\beta_i}} = e(g,h)^{k\gamma \sum_{i=1}^{t}\beta_i} \quad (3.23)$$

So sánh với các hệ mã khác:

Để đánh giá độ hiệu quả của lược đồ mã hóa dựa trên thuộc tính đề xuất, NCS lập Bảng 3.1 so sánh với các hệ mã hóa dựa trên thuộc tính hiện có mà có cùng tính chất là có độ dài bản mã là hằng số, trong đó:

• **Header** là độ dài của Hdr.

• **S-key** là độ dài khóa bí mật.

• **P-key** là độ dài khóa công khai.

• **Acce Policy** là chính sách giải mã.

• **Setting** là kiểu mã hóa bí mật (**Secret-key**) hay công khai (**Public-key**).

	Acce Policy	CNF	S-key	P-key	Setting
[27]	AND-gates	$O(1)$	$O(1)$	$O(n^2)$	Public-key
[35]	Threshold	$O(1)$	$O(n)$	$O(n)$	Public-key
[8]	LSS	$O(1)$	$O(k^4.\ell^4)$	$O(k^2.\ell^2)$	Public-key
[4]	LSS	$O(1)$	$O(n.\ell^2)$	$O(n.\ell)$	Public-key
[6]	LSS	$O(1)$	$O(n.\ell^2)$	$O(n.\ell)$	Public-key
[15]-3	CNF	$O(1)$	$O(m.n)$	$O(m.n)$	Secret-key
[15]-4	CNF	$O(1)$	$O(m.n)$	$O(1)$	Secret-key
[52]	LSS	$O(\ell)$	$O(k)$	$O(1)$	Public-key
[5]	LSS	$O(\ell)$	$O(k)$	$O(1)$	Public-key
CP-ABE đề xuất	CNF	O(1)	$O(m^2.n)$	$O(m.n)$	Public-key

Bảng 3.1. So sánh một số hệ mã hóa dựa trên thuộc tính đã có với mã hóa đề xuất.

Trong đó, n là số tối đa các thuộc tính trong hệ thống, m là số tối đa các mệnh đề trong CNF, k là số tối đa các thuộc tính trong một khóa bí mật (số tối đa các thuộc tính mà một người dùng có thể sở hữu), ℓ là số dòng trong ma trận LSS, tương đương với n. LSS là ma trận tuyến tính chia sẻ khóa bí mật tuyến tính, có dạng như một biểu thức Boolean.

Hiện nay trên thực tế chưa có công trình nào công bố về mã hóa dựa trên thuộc tính mà khắc phục được hết những hạn chế của hệ mã hóa dựa trên thuộc tính. Ý

tưởng đề xuất trong luận án có lợi thế chính, đó là mã hóa công khai, độ dài bản mã ngắn, hỗ trợ chính sách giải mã linh động. Tốc độ mã hóa và giải mã nhanh hơn một số hệ khác đã công bố.

Tuy nhiên điểm yếu của đề xuất: Độ dài khóa bí mật vẫn dài hơn các hệ cùng bảng so sánh, độ dài khóa công khai dài và số lượng tối đa các thuộc tính có trong hệ thống là giới hạn.

3.3.3. Đánh giá an toàn

Dựa trên mô hình an toàn đã được định nghĩa ở trên, để chứng minh an toàn chúng ta phải chứng minh rằng: Cho trước kẻ tấn công tất cả các thông tin như bản mã thách thức, khóa công khai của hệ thống và khóa bí mật của tất cả những người dùng không có khả năng giải mã. Kẻ tấn công phải không có khả năng phân biệt được giữa một phần tử ngẫu nhiên và khóa phiên K (tức là kẻ tấn công không biết bất cứ thông tin gì về khóa phiên K).

Ký hiệu P, Q, R là danh sách các đa thức bao gồm: Tất cả các phần tử nằm trong bản mã thách thức, khóa công khai của hệ thống và khóa bí mật của tất cả những người dùng không có khả năng giải mã. Ký hiệu f là đa thức chứa các phần tử của khóa phiên làm việc (thực tế là đa thức có duy nhất một phần tử), chúng ta phải chứng minh rằng f là độc lập với (P, Q, R), tức là kẻ tấn công có (P, Q, R) nhưng không thể suy ra được f. Trong Hình 3.1 mô phỏng các đa thức (P, Q, R) và f dưới dạng một bài toán GDDHE, tức là cho trước (P, Q, R) và tính f.

$$P = \left\{ \alpha, k\alpha, \left\{ s_u + \gamma, \left\{ \frac{\beta_j s_u}{\alpha + A_{i,j}} \right\}_{\substack{i \in S_u \\ j=1,\dots,m}} \right\}_{u \in \tilde{\mathcal{U}}} \right\}$$

$$Q = \left\{ k \sum_{i=1}^{t} \beta_i , k \prod_{i \in \tilde{\beta}_1} (\alpha + A_{i,1}) \dots \prod_{i \in \tilde{\beta}_t} (\alpha + A_{i,t}) , \{\alpha^j, \beta_i\}_{\substack{j=0,\dots,N \\ i=1,\dots,m}}, \{\alpha^i, \beta_j s_u\}_{\substack{j=0,\dots,N \\ i=1,\dots,m;\ u \in \tilde{\mathcal{U}}}} \right\}$$

$$R = \left\{ \{\gamma \beta_j\}_{j=1,\dots,m} \right\}, \qquad f = k\gamma \sum_{i=1}^{t} \beta_i$$

$\tilde{\mathcal{U}}$ là tập của các người dùng không có khả năng giải mã, S_u không thỏa mãn $\tilde{\beta} = \tilde{\beta}_1 \wedge \dots \wedge \tilde{\beta}_t$ với tất cả $u \in \tilde{\mathcal{U}}$.

Hình 3.1: Bài toán khó (P, Q, R, f) – GDDHE

An toàn của hệ mã được phát biểu thông qua định lý sau:

Định lý 3.2. Nếu tồn tại một kẻ tấn công chạy trong thời gian đa thức $\mathcal{A}$ có lợi thế $Adv^{ind}(.)$ để phá mã CP-ABE ở trên, thì cũng tồn tại một kẻ tấn công có cùng lợi thế Adv^{ind} giải quyết được bài toán GDDHE định nghĩa ở trên.

Như vậy, định lý ở trên chỉ ra rằng nếu bài toán GDDHE là khó, thì ý tưởng đề xuất của luận án là an toàn.

***Chứng minh*:** Giả sử $\mathcal{B}$ là kẻ tấn công bài toán *(P, Q, R, f)* − GDDHE, và $\mathcal{A}$ là kẻ tấn công hệ mã. Đầu tiên, $\mathcal{B}$ sẽ nhận được đầu vào của bài toán GDDHE như mô tả trong Hình 3.1 và một phần tử *K*. Nếu bít $b = 0$ thì $K = e(g, h)^f$. Nếu bít $b = 1$ thì *K* là một phần tử ngẫu nhiên trong $\mathbb{G}_T$.

Để cho đơn giản về mặt ký hiệu, ta ký hiệu đầu vào của bài toán GDDHE bằng $g^{P(\dots)}, h^{Q(\dots)}, g_T^{R(\dots)}$. Mục tiêu của B là đi dự đoán bít b, với mục tiêu đó, B sẽ dùng $g^{P(\dots)}, h^{Q(\dots)}, g_T^{R(\dots)}$có trong tay để mô phỏng $\mathcal{A}$, sau đó dùng đầu ra của $\mathcal{A}$ để đi dự đoán bít *b*.

Cụ thể, với $g^{P(\dots)}, h^{Q(\dots)}, g_T^{R(\dots)}$ trong tay, đầu tiên ở giai đoạn khởi tạo, $\mathcal{B}$ cung cấp cho $\mathcal{A}$ khóa công khai và khóa bí mật của tất cả những người dùng không thể giải mã được (lưu ý rằng, tất cả các thông tin này đều có sẵn trong $g^{P(\dots)}, h^{Q(\dots)}, g_T^{R(\dots)}$.

Tiếp theo, trong giai đoạn thách thức, $\mathcal{B}$ cung cấp cho $\mathcal{A}$ bản mã thách thức Hdr cùng với khóa *K* ở trên. Ở giai đoạn dự đoán, $\mathcal{A}$ sẽ cho đầu ra là bít dự đoán cho *b*, $\mathcal{B}$ lúc này dùng luôn dự đoán của $\mathcal{A}$ để đưa ra dự đoán bit *b* cho mình. Do sự mô phỏng của $\mathcal{B}$ cho $\mathcal{A}$ là đúng đắn, nên dễ thấy rằng nếu $\mathcal{A}$ dự đoán đúng thì $\mathcal{B}$ cũng dự đoán đúng, và nếu $\mathcal{B}$ sai thì $\mathcal{A}$ cũng sai, hay nói theo cách khác, lợi thế của cả $\mathcal{B}$ và $\mathcal{A}$ là như nhau.

Chứng minh rằng: Bài toán GDDHE ở trên là khó, có nghĩa phải chứng minh f là độc lập tuyến tính với *(P, Q, R).*

Bổ đề 3.3. Bài toán GDDHE được định nghĩa tại Hình 3.1 là khó, có nghĩa f là độc lập tuyến tính với *(P, Q, R).*

***Chứng minh**:* Giả sử *f* không độc lập tuyến tính với *(P, Q, R),* tức là ta có thể tìm được các hằng số $b_{i,,j}$, c_i sao cho biểu thức sau đây là thỏa mãn:

$$f = \sum_{\substack{p_i \subset P \\ q_{j \subset Q}}} b_{i,j} \,.\, p_i \,.\, q_j + \sum_{r_i \subset R} c_i . r_i \tag{3.27}$$

Chúng ta sẽ dùng *k* để phân tích *f.* Vì *k* là được chọn một cách ngẫu nhiên nên chúng ta sẽ bỏ qua tích $b_{i,,j} \cdot p_i \cdot q_j$ có chứa k^2 hoặc không chứa *k* và cũng bỏ qua trường hợp mà q_j không chia hết cho $\alpha + A_{i,j}$. Tóm lại, cần tìm các hằng số sao cho biểu thức sau là thỏa mãn:

$$\begin{aligned} k\gamma \sum_{i=1}^{t} \beta_i = {} & \sum_{i=1}^{N+1} a_i k\alpha^i + \sum_{u \in \tilde{\mathcal{U}}} \sum_{j=1}^{m} \sum_{i=1}^{N+1} b_{u,j,i}\, k\alpha^i \beta_j s_u + k\alpha \sum_{i=1}^{m} c_{1,i}\beta_i \\ & + \sum_{u \in \tilde{\mathcal{U}}} d_u . k(\gamma + s_u) \sum_{i=1}^{t} \beta_i + c_2 . k\alpha \prod_{i \in \tilde{\beta}_1} (\alpha + A_{i,1}) \dots \prod_{i \in \tilde{\beta}_t} (\alpha + A_{i,t}) \\ & + \sum_{u \in \tilde{\mathcal{U}}} e_u . k(\gamma + s_u) \prod_{i \in \tilde{\beta}_1} (\alpha + A_{i,1}) \dots \prod_{i \in \tilde{\beta}_t} (\alpha + A_{i,t}) \\ & + \sum_{u \in \tilde{\mathcal{U}}} \sum_{i \in S_u} \sum_{j=1}^{m} \frac{f_{u,i,j} k\beta_j s_u}{\alpha + A_{i,j}} \prod_{i' \in \tilde{\beta}_1} (\alpha + A_{i',1}) \dots \prod_{i' \in \tilde{\beta}_t} (\alpha + A_{i',t}) \end{aligned} \tag{3.28}$$

Mỗi γ, β_i, s_u, $i = 1,\ldots, m$, $u \in \tilde{\mathcal{U}}$, là được chọn một cách ngẫu nhiên và s_u không xuất hiện ở vế trái của biểu thức, điều đó dẫn đến $k\gamma \sum_{i=1}^{t} \beta_i$ phải đến từ $\sum_{u\in\tilde{\mathcal{U}}} d_u . k(\gamma + s_u) \sum_{i=1}^{t} \beta_i$, và tất cả các phần tử liên quan đến s_u phải bị triệt tiêu. Do vậy, các biểu thức sau phải đồng thời được thỏa mãn với tất cả $u \in \tilde{\mathcal{U}}$:

$$1 = \sum_{u\in\tilde{\mathcal{U}}} d_u \tag{3.29}$$

$$\begin{aligned} 0 = &\sum_{j=1}^{m}\sum_{i=1}^{N+1} b_{u,j,i} k\alpha^i \beta_j s_u + d_u k s_u \sum_{i=1}^{t} \beta_i \\ &+ e_u k s_u \prod_{i\in\tilde{\beta}_1}(\alpha + A_{i,1}) \ldots \prod_{i\in\tilde{\beta}_t}(\alpha + A_{i,t}) \\ &+ \sum_{i\in S_u}\sum_{j=1}^{m} \frac{f_{u,j,i} k \beta_j s_u}{\alpha + A_{i,j}} \prod_{i'\in\tilde{\beta}_1}(\alpha + A_{i',1}) \ldots \prod_{i'\in\tilde{\beta}_t}(\alpha + A_{i',t}) \end{aligned} \tag{3.30}$$

Hoặc

$$1 = \sum_{u\in\tilde{\mathcal{U}}} d_u \tag{3.31}$$

$$\begin{aligned} 0 = &\sum_{j=1}^{m}\sum_{i=1}^{N+1} b_{u,j,i} \alpha^i \beta_j + d_u \sum_{i=1}^{t} \beta_i + e_u \prod_{i\in\tilde{\beta}_1}(\alpha + A_{i,1}) \ldots \prod_{i\in\tilde{\beta}_t}(\alpha + A_{i,t}) \\ &+ \sum_{i\in S_u}\sum_{j=1}^{m} \frac{f_{u,j,i} \beta_j}{\alpha + A_{i,j}} \prod_{i'\in\tilde{\beta}_1}(\alpha + A_{i',1}) \ldots \prod_{i'\in\tilde{\beta}_t}(\alpha + A_{i',t}) \end{aligned} \tag{3.32}$$

Vì *mỗi* β_j , $j = 1,\ldots, m$, cũng được chọn ngẫu nhiên nên biểu thức sau phải thỏa mãn:

$$\begin{aligned} 0 = &\sum_{j=1}^{m}\sum_{i=1}^{N+1} b_{u,j,i} \alpha^i \beta_j + d_u \sum_{i=1}^{t} \beta_i \\ &+ \sum_{i\in S_u}\sum_{j=1}^{m} \frac{f_{u,j,i} \beta_j}{\alpha + A_{i,j}} \prod_{i'\in\tilde{\beta}_1}(\alpha + A_{i',1}) \ldots \prod_{i'\in\tilde{\beta}_t}(\alpha + A_{i',t}) \end{aligned} \tag{3.33}$$

Thấy rằng $d_u \sum_{i=1}^{t} \beta_i$ phải đến từ:

$$\sum_{i\in S_u} \sum_{j=1}^{m} \frac{f_{u,j,i}\beta_j}{\alpha + A_{i,j}} \prod_{i'\in\tilde{\beta}_1}(\alpha + A_{i',1}) \ \ldots \ \prod_{i'\in\tilde{\beta}_t}(\alpha + A_{i',t}). \tag{3.34}$$

Tuy nhiên, vì S_u không thỏa mãn $\tilde{\beta}$, điều đó có nghĩa là tồn tại ít nhất một $\alpha + A_{i,j}$, j = 1,.., t mà không được chia hết bởi $\prod_{i' \in \tilde{\beta}_1}(\alpha + A_{i',1}) \dots \prod_{i' \in \tilde{\beta}_t}(\alpha + A_{i',t})$, hay tồn tại ít nhất một β_j, j = 1,...,t, không đến từ:

$$\sum_{i \in S_u} \sum_{j=1}^{m} \frac{f_{u,j,i}\beta_j}{\alpha + A_{i,j}} \prod_{i' \in \tilde{\beta}_1}(\alpha + A_{i',1}) \dots \prod_{i' \in \tilde{\beta}_t}(\alpha + A_{i',t})$$

Hay biểu thức ở trên không thể tồn tại. Do vậy, kết luận f là độc lập tuyến tính với *(P, Q, R).*

3.3.4. Cài đặt và đánh giá hiệu quả

Để cài đặt hệ mã, trong phần mã hóa chúng ta thấy rằng không dễ dàng để tính

$$C_2 = h^{k.\prod_{(i,j) \in S}\left(\alpha + \mathcal{H}(ID_{i,j})\right)} \tag{3.35}$$

từ khóa công khai. Thay vào đó, áp dụng công thức sau:

$$\prod_{i=1}^{m}(X + a_i) = \sum_{j=0}^{m}\left(\sum_{1 \le i_1 < i_2 < \dots < i_j \le m} a_{i_1} a_{i_2} \dots a_{i_j}\right) X^{m-j} \tag{3.36}$$

và các tham số của đa thức:

$$s_j = \sum_{1 \le i_1 < i_2 < \dots < i_j \le m} a_{i_1} a_{i_2} \dots a_{i_j} \tag{3.37}$$

tất cả là đối xứng a_1,..., a_m và được gọi là những đa thức đối xứng của giá trị a_i. Tham số s_j cho phép viết lại:

$$C_2 = h^{k.\sum_{j=0}^{m} s_j.\alpha^{m-j}} = \left(\prod_{j=0}^{m}\left(h^{\alpha^{m-j}}\right)^{s_j}\right)^k \tag{3.38}$$

Có thể tính C_2 do tập $\{h^{\alpha^i} \mid i = 0,.., N\}$ có trong param. Đối với giải thuật giải mã, ta cũng dùng cách trên để tính K' do $h^{\beta_i \alpha^j}$ cũng có trong param.

Tham số s_j ở trên có thể được tính nhanh bằng cách dùng giải thuật quy hoạch động như sau:

Đặt $s_{k,j}$ là tổng của j tổ hợp của a_1, a_2,..., a_k. Tức là,

$$s_{k,j} = \sum_{1 \le i_1 < i_2 < \dots < i_j \le k} a_{i_1} a_{i_2} \dots a_{i_j} \tag{3.39}$$

Lưu ý rằng, tham số $s_j = s_{m,j}$. Tổng $s_{k,j}$ có thể được chia làm hai phần, phần thứ nhất chứa đựng a_k và phần thứ hai không chứa a_k. Chúng ta có:

$$s_{k,j} = \left(\sum_{1 \le i_1 < i_2 < \cdots < i_j \le k-1} a_{i_1} a_{i_2} \ldots a_{i_j} \right) + \left(\sum_{1 \le i_1 < i_2 < \cdots < i_{j-1} \le k-1} a_{i_1} a_{i_2} \ldots a_{i_{j-1}} \right) . a_k \quad (3.40)$$

Và:

$s_{k,j} = s_{k-1,j} + s_{k-1,j-1} . a_k$ trong đó $s_{k,0} = 1$ đối với $k \geq 0$ và $s_{0,j} = 0$ đối với $j > 1$.

Từ mối liên hệ này, bằng việc xây dựng bảng từ $s_{0,0}$ cho tới $s_{m,m}$, có thể tính các tham số $s_j = s_{m,j}$ với độ phức tạp là $O(m^2)$.

Cài đặt hệ CP-ABE đề xuất ở trên bằng ngôn ngữ C và dùng thư viện PBC [40]. Mã nguồn của chương trình cài đặt có ở địa chỉ:

https://github.com/tranvinhduc/MCBE

Tác giả cài đặt trên máy tính xách tay với bộ vi xử lý Intel Core i7-4600U @ 2.1 GHz. Đo kết quả trung bình 1000 lần. Trên máy tính này, thư viện PBC tính một Parings khoảng 0.9ms, một phép mũ trên đường cong elliptic của nhóm $\mathbb{G}$ khoảng xấp xỉ 1.3ms.

Với lược đồ mã hóa của CP-ABE, thời gian chủ yếu khi mã hóa là để tính C_2 và C_3, với tương ứng cần m và N phép mũ trong nhóm $\mathbb{G}$. Hầu hết thời gian giải mã là để tính K_1, K_2, ..., K_m trong đó m là số các mệnh đề trong một chính sách giải mã. Với mỗi K_i cần N phép mũ trong $\mathbb{G}$.

Cài đặt dùng chính sách giải mã $\tilde{\beta}$ có dạng biểu thức CNF sau:

$$\tilde{\beta} = \tilde{\beta}_1 \wedge \ldots \wedge \tilde{\beta}_m, \ \left|\tilde{\beta}\right| = N, \ \left|\tilde{\beta}_i\right| = N/m$$

Bảng 3.2 mô tả kết quả cài đặt CP-ABE của luận án. Kết quả cài đặt thực nghiệm đúng với kết quả phân tích ở trên.

Tóm lại, thực nghiệm đã chỉ ra rằng đề xuất CP-ABE của luận án, đáp ứng được yêu cầu về sự hiệu quả khi triển khai trong thực tế.

m	N	Mã hóa	Giải mã
10	20	27ms	237ms
10	40	54ms	510ms
10	80	107ms	1.03s
20	40	55ms	1.04s
20	80	103ms	2.01s
20	160	209ms	4.1s
25	50	68ms	1.6s
25	100	127ms	3.1s
25	200	259ms	6.4s

Bảng 3.2: Kết quả thực nghiệm cài đặt hệ CP-ABE đề xuất.

Trong đó m là số mệnh đề trong biểu thức boolean CNF), N là số các thuộc tính trong hệ thống.

3.4. Đề xuất thứ hai (CP-ABE-02) về mã hóa dựa trên thuộc tính

3.4.1. Ý tưởng xây dựng và so sánh

Hiện nay dữ liệu của các công ty/doanh nghiệp thường được mã hóa và lưu trên các đám mây. Để đảm bảo tính linh động thì trong các hệ mã hóa hiện có, mã hóa dựa trên thuộc tính thường được lựa chọn. Hàng ngày, các công ty/doanh nghiệp vẫn cần phải làm việc trên khối dữ liệu đã được mã hóa này, chẳng hạn như việc tìm kiếm dữ liệu. Có hai cách để các công ty/doanh nghiệp có thể làm, đó là:

Phương pháp thứ nhất: Công ty/doanh nghiệp sẽ cung cấp toàn bộ khóa bí mật cho một máy chủ nào đó có năng lực mạnh để giải mã toàn bộ dữ liệu của mình, sau đó tìm kiếm dữ liệu trên khối dữ liệu đã được giải mã đó rồi trả về kết quả cho công ty/doanh nghiệp. Tuy nhiên, phương pháp này có nhược điểm là máy chủ đó sẽ biết được toàn bộ nội dung dữ liệu của công ty/doanh nghiệp, điều mà không một công ty/doanh nghiệp nào mong muốn;

Phương pháp thứ hai: Công ty/doanh nghiệp tự lấy về hoàn toàn dữ liệu và giải mã, sau đó tìm kiếm trên dữ liệu đã được giải mã. Phương pháp này hiển nhiên là không hợp lý vì năng lực máy tính của công ty/doanh nghiệp là không đáp ứng được. Để giải quyết vấn đề, một hướng nghiên cứu mở rộng của ABE hiện đang rất được quan tâm là tìm kiếm trên dữ liệu đã được mã hóa được viết trong các tài liệu [11, 14, 23, 24, 34, 37, 45, 57, 59]. Với hướng nghiên cứu này, công ty/doanh nghiệp chỉ cung cấp một phần thông tin của khóa bí mật gọi là cửa sập cho máy chủ, để máy chủ dựa vào đó tìm kiếm dữ liệu cần thiết trên khối dữ liệu đang được mã hóa của doanh nghiệp. Sau khi tìm được các bản mã tương ứng doanh nghiệp muốn tìm sẽ gửi trả về cho doanh nghiệp, doanh nghiệp sẽ dùng khóa bí mật của mình để giải mã các bản mã này. Với phương pháp như vậy, máy chủ chỉ biết được thông tin là cửa sập và các bản mã, chứ không thể biết được nội dung thực sự dữ liệu của doanh nghiệp. Trong khi đó, doanh nghiệp vẫn tận dụng được sức mạnh tính toán của máy chủ. Kỹ thuật này cũng được áp dụng vào rất nhiều ứng dụng khác, ví dụ như ứng dụng định hướng chuyển tiếp Email của các Gateway.

Trong luận án này đề xuất một lược đồ mã hóa dựa trên thuộc tính đồng thời có tính chất tìm kiếm trên khối dữ liệu đã được mã hóa. Đề xuất của nghiên cứu sinh có ưu và nhược điểm sau:

Về ưu điểm:

- Tích hợp một mã hóa dựa trên thuộc tính và một hệ cho phép tìm kiếm trên dữ liệu đã được mã hóa. Tức là trong đề xuất, người dùng chỉ cần sở hữu một khóa bí mật duy nhất cho cả hai hệ, cũng như hệ thống chỉ cần một khóa công khai duy nhất cho cả hai hệ thống. Lưu ý rằng, trong một số hệ khác hai hệ này là tách biệt do đó sẽ kém hiệu quả hơn.

- Trong đề xuất này của luận án, hệ thống người dùng có thể tự mình tạo ra cửa sập. So với các hệ khác để tạo ra cửa sập người dùng thường phải nhờ trợ giúp từ một bên thứ 3 dẫn đến phức tạp và kém an toàn hơn.

- Về độ hiệu quả, đề xuất được xây dựng dựa trên công trình nghiên cứu CP-ABE [42], nên cũng kế thừa cơ bản các tính chất của hệ này như độ dài khóa bí mật

ngắn, độ dài bản mã ngắn, tốc độ giải mã nhanh và hỗ trợ hệ thống phân phối khóa an toàn.

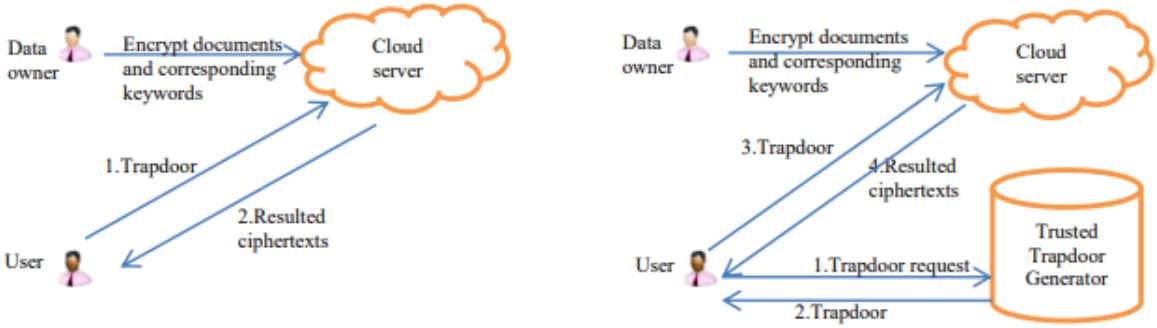
Tuy nhiên đề xuất của NCS cũng có những nhược điểm sau:

• Không đạt được an toàn cho cửa sập (cho đến nay chỉ có hệ [24] đạt được một phần tính chất này).

• Độ dài khóa công khai cũng như cửa sập còn dài.

• Tốc độ tìm kiếm dữ liệu vẫn chưa thực sự hiệu quả.

So sánh mô hình hoạt động đề xuất của nghiên cứu sinh với một số hệ hiện có trong Hình 3.



Hình 3.2 So sánh mô hình hoạt động giữa hệ tìm kiếm đề xuất và các hệ khác

3.4.2. Lược đồ mã hóa đề xuất thứ 2 dựa trên thuộc tính

Lược đồ mã hóa đề xuất thứ 2 dựa trên thuộc tính được mô tả chi tiết như sau:

Khởi tạo $(v, \mathcal{B})$:

Giả sử $N = |\mathcal{B}|$ là số tối đa các thuộc tính có trong hệ thống, $(p, \mathbb{G}, \mathbb{G}_T, e(\cdot,\cdot))$ là một hệ thống Billinear Map. Thuật toán chọn ngẫu nhiên một phần tử sinh $g \in \mathbb{G}$, và các giá trị $a, \alpha, \lambda \in \mathbb{Z}_p$, tính g^a, g^α, g^λ. Thuật toán tiếp tục tạo ra $2N$ phần tử trong nhóm $\mathbb{G}$ tương ứng với N thuộc tính trong hệ thống $h_1, \ldots, h_N, \tilde{h}_1, \ldots, \tilde{h}_N$. Giả sử $\mathcal{H}, \widetilde{\mathcal{H}}$ là các hàm băm sao cho $\mathcal{H}: \{0,1\}^* \to \mathbb{G}$ và $\widetilde{\mathcal{H}}: \mathbb{G}_T \times \{0,1\}^* \to \mathbb{Z}_p$. Giả sử tập hợp các từ khóa có trong hệ thống là $W = (\omega_1, \omega_2, \omega_3, \ldots)$, trong đó mỗi $\omega_i \in$

$\{0,1\}^*$. Lưu ý rằng tập W là không giới hạn, chúng ta có thể thêm mới từ khóa bất kỳ lúc nào nếu muốn. Để cho đơn giản về mặt ký hiệu, thuật toán bỏ qua W trong danh sách tham số hệ thống. Cuối cùng, khóa bí mật của hệ thống là $\text{MSK} = (g^{\alpha}, \curlywedge)$ và khóa công khai của hệ thống là:

$$\text{param} = (g, g^{a}, g^{\curlywedge}, e(g,g)^{\alpha}, h_1, \ldots, h_N, \tilde{h}_1, \ldots, \tilde{h}_N, \mathcal{H}, \widetilde{\mathcal{H}})$$

Tạo khóa$(u, \mathcal{B}(u), \text{MSK}, \text{param})$:

Giả sử $\mathcal{B}(u)$ là tập thuộc tính của người dùng u, giải thuật tạo khóa chọn $s_u \overset{\$}{\leftarrow} \mathbb{Z}_p$, tính khóa bí mật cho người dùng u là $d_{u0} = (d_{u_0}, d'_{u_0}, \{d_{u_i}\}_{i \in B(u)}, \curlywedge)$, trong đó:

$$d_{u_0} = g^{\alpha} \cdot g^{a \cdot s_u}, d'_{u_0} = g^{s_u}, \{d_{u_i} = h_i^{s_u}\}_{i \in \mathcal{B}(u)}$$

Người dùng u chỉ cần giữ bí mật d_{u_0}và $\curlywedge$, phần còn lại của khóa có thể lưu giữ ở bất kỳ đâu không cần giữ bí mật, điều đó có nghĩa là khóa bí mật mà người dùng cần lưu giữ có chỉ là hai phần tử do đó có độ dài là ngắn.

Mã hóa$(\mathcal{M}, \mathbb{A}, param)$:

Đầu vào của giải thuật là dữ liệu $\mathcal{M}$, chính sách mã hóa $\mathbb{A}$, và khóa công khai của hệ thống. Giải sử $\mathbb{A}$ là biểu thức boolean β và kích thước của β là $|\beta|$. Đầu tiên, giải thuật mã hóa biểu diễn β dưới dạng biểu thức dạng DNF $\beta = (\beta_1 \vee \cdots \vee \beta_m)$, trong đó mỗi β_i là một tập các thuộc tính, $i = 1, \ldots, m$.

Thuật toán chọn ngẫu nhiên giá trị $s \overset{\$}{\leftarrow} \mathbb{Z}_p$, tính C, C_0 như sau:

$$C = \mathcal{M} \cdot e(g,g)^{\alpha \cdot s}, C_0 = g^s$$

Tiếp theo, giải thuật so sánh giữa giá trị m và$|\beta|$, nếu $m \leq |\beta|$ giải thuật tính:

$$C_1 = (g^{a}) \prod_{i \in \beta_1} h_i)^s, \ldots, C_m = (g^{a} \prod_{i \in \beta_m} h_i)^s$$

Ngược lại, giải thuật xây dựng ma trận M biểu diễn biểu thức β, và một ánh xạ ρ sao cho $(M, \rho) \in (\mathbb{Z}_p^{\ell \times n}, \mathcal{F}([\ell] \to [N]))$. Giải thuật sau đó chọn một vector ngẫu nhiên $\vec{v} = (s, \mathcal{Y}_2, .., \mathcal{Y}_n) \in \mathbb{Z}_p^n$. Cho $i = 1, \ldots, \ell$, tính $\curlywedge_i = \vec{v} \cdot M_i$, trong đó: M_i là vector tương ứng với dòng thứ i của ma trận M. Giải thuật tiếp tục tính:

$$C_i = g^{a.\curlywedge_i} h_{\rho(i)}^{-s}, i = 1, \ldots, \ell$$

Cuối cùng, giải thuật cho đầu ra hoặc là $ct = (C, C_0, \ldots, C_m)$ cùng với mô tả của β trong trường hợp $m \leq |\beta|$, hoặc là $ct = (C, C_0, \ldots, C_\ell)$ cùng với mô tả của (M, ρ) trong trường hợp ngược lại.

Giải mã(ct, d_u, param):

Thuật toán với đầu vào là khóa bí mật d_u, bản mã ct và khóa công khai của hệ thống, trước tiên sẽ phân tích bản mã ct, kiểm tra xem số phần tử có trong bản mã ct. Nếu số phần tử chính xác là $m + 1$ phần tử, thuật toán sẽ biểu diễn ct dưới dạng $(C_0, C_1, \ldots, C_m)$. Sau đó, tìm chỉ số j sao cho $\beta_j \subset \mathcal{B}(u)$ và tính:

$$\frac{e(C_0, d_{u_0} \prod_{i \in \beta_j} d_{u_i})}{e(d'_{u_0}, C_j)} = \frac{e(g^s, g^\alpha (g^a \prod_{i \in \beta_j} h_i)^{s_u})}{e(g^{s_u}, (g^a \prod_{i \in \beta_j} h_i)^s)} = e(g, g)^{\alpha \cdot s} = K$$

Cuối cùng tính: $\mathcal{M} = C \cdot K^{-1}$

Ngược lại, thuật toán tạo ra tập $I \subset \{1,2,..,\ell\}$ sao cho $I = \{i: \rho(i) \in B(u)\}$. Đặt $\{\omega_i \in \mathbb{Z}_p\}_{i \in I}$ là tập các hằng số, sao cho nếu $\{\lambda_i\}$ là các giá trị chia sẻ đúng của bất kỳ thành phần bí mật s tương ứng với ma trận M thì $\sum_{i \in I} \omega_i \lambda_i = s$. Lưu ý rằng, từ mối liên hệ $\sum_{i \in I} \omega_i M_i = (1,0,..,0)$ trong đó M_i là dòng thứ i của ma trận M, thuật toán có thể tính được các hằng số này. Thuật toán tiếp tục biểu diễn ct dưới dạng $(C, C_0, \ldots, C_\ell)$ và tính:

$$e(\prod_{i \in I} C_i^{-\omega_i}, d'_{u_0}) \cdot e(C_0, d_{u_0} \prod_{i \in I} d_{u_{\rho(i)}}^{-\omega_i}) = K.$$

sau đó tính: $\mathcal{M} = C \cdot K^{-1}$

Tính Cửa sập $(d_u, W_i = (\widetilde{\omega}_{i_1}, \ldots, \widetilde{\omega}_{i_k}), \text{param})$:

Giả sử mỗi $\widetilde{\omega}_{i_j} \in \{0,1\}^*, j \in [k]$, là một sự kết hợp của tập các từ khóa.

Người dùng ngẫu nhiên chọn các giá trị $r_1, .., r_k \in \mathbb{Z}_p$, tính cửa sập $\text{tds} = (\{\text{tds}_{0,j}, \text{tds}_{1,j}, \{\text{tds}_{2,j,\ell}\}_{\ell \in \mathcal{B}_u}\}_{j \in [k]}, \text{tds}_0, \{\text{tds}_i\}_{i \in \mathcal{B}(u)}, \widetilde{W}_i)$

$$= (\{g^\alpha g^{as_u} g^{ar_j} (\{g^\alpha \mathcal{H}(\widetilde{\omega}_{i_j}))^\lambda, g^{r_j}, \{\tilde{h}_\ell^{r_j}\}_{\ell \in \mathcal{B}_u}\}_{j \in [k]}, g^{s_u}, \{h_i^{s_u}\}_{i \in \mathcal{B}_u}, \widetilde{W}_i)$$

trong đó $\widetilde{W}_i$ là giá trị mô tả của W_i. Người dùng sau đó gửi $(\{\text{tds}_{0,j}\}_{j \in [k]}, \widetilde{W}_i)$ cho máy chủ và lưu giữ công khai phần còn lại của tds. Như vậy, độ dài của cửa sập sẽ tuyến tính với số lượng sự kết hợp của các từ khóa mà người dùng muốn tìm kiếm.

Mã hóa từ khóa $(\mathcal{KF}, \mathbb{A}', \text{param})$:

Giả sử chính sách mã hóa là:

$\mathbb{A}' = \beta = (\beta_1 \vee \cdots \vee \beta_m)$ và $\mathcal{KF} = (kf_1 \vee \cdots \vee kf_{m'})$, trong đó mỗi β_i là một tập các thuộc tính và kf_i là một sự kết hợp của các từ khóa.

Lưu ý rằng: $\beta_i \neq \beta_j, kf_{i'} \neq kf_{j'}, \forall i, j \in [m], i', j' \in [m']$

Thuật toán chọn ngẫu nhiên $s \xleftarrow{\$} \mathbb{Z}_p$, sau đó tính:

$$C_0 = g^s, C_1 = (g^a \prod_{i \in \beta_1} h_i)^s, \ldots, C_m = (g^a \prod_{i \in \beta_m} h_i)^s,$$

$$\tilde{C}_1 = (g^a \prod_{i \in \beta_1} \tilde{h}_i)^s, \ldots, \tilde{C}_m = (g^a \prod_{i \in \beta_m} \tilde{h}_i)^s$$

Tiếp theo, tính:

$$X_i \; e(g,g)^{\alpha \cdot s} \cdot e(g, g^a \, \mathcal{H}(kf_i))^{\cdot s}, i = 1, \ldots, m'$$

Sau đó tính:

$$K_1 = \widetilde{\mathcal{H}}(X_1, kf_1), \ldots, K_{m'} = \widetilde{\mathcal{H}}(X_{m'}, kf_{m'}).$$

Cuối cùng, thuật toán cho đầu ra:

$$ct' = (C_0, \ldots, C_, \tilde{C}_1, \ldots, \tilde{C}_m, K_1, \ldots, K_{m'})$$

cùng với bản mô tả của β

Tìm kiếm $(\text{tds}, ct', \text{param})$:

Máy chủ tìm $\ell \in [m]$, sao cho $\beta_\ell \subset \mathcal{B}(u)$, sau đó tính $(X_j, Y_j), j = 1, \ldots, k$

$$X_j = \frac{e(C_0, \text{tds}_{0,j} \prod_{i \in \beta_\ell} \text{tds}_i \cdot \text{tds}_{2,j,i})}{e(\text{tds}_0, C_\ell) \cdot e(\text{tds}_{1,j}, \tilde{C}_\ell)}$$

$$= \frac{e(g^s, g^\alpha g^{as_u} g^{ar_j} g^{a\cdot} \mathcal{H}(\widetilde{w}_{ij})^{\cdot} \prod_{i \in \beta_\ell} h_i^{s_u} \tilde{h}_i^{r_j}}{e(g^{s_u}, (g^a \prod_{i \in \beta_\ell} h_i)^s) \cdot e(g^{r_j}, (g^a \prod_{i \in \beta_\ell} \tilde{h}_i)^s)}$$

$$= e(g,g)^{\alpha \cdot s} \cdot e\left(g, g^a \mathcal{H}\left(\widetilde{w}_{i_j}\right)\right)^{\cdot s}$$

$$Y_j = \mathcal{H}(X_j, \widetilde{w}_{i_j})$$

Nếu tồn tại một cặp $(i, j), i \in [m'], j \in [k]$ sao cho: $K_i = Y_j$ thì máy chủ cho đầu ra là "yes". Ngược lại, máy chủ cho đầu ra là "no". Lưu ý, máy chủ không cần

thiết phải tính rất cả các cặp $(X_j, Y_j), j = 1, \ldots, k$, miễn là máy chủ tìm được một cặp $(i, j), i \in [m'], j \in [k]$ sao cho $K_i = Y_j$, máy chủ cho đầu ra là "yes" và dừng lại.

Tính đúng đắn: Chúng ta thấy rằng, nếu tồn tại một cặp $\widetilde{w}_{i_j} \in W_i$ và $kf_t \in \mathcal{KF}$ sao cho $\widetilde{w}_{i_j} = kf_t$, thì:

$$X_t = e(g,g)^{\alpha \cdot s} \cdot e(g, g^a \mathcal{H}(kf_t))^{\lambda \cdot s} = e(g,g)^{\alpha \cdot s} \cdot e(g, g^a \mathcal{H}(\widetilde{w}_{i_j}))^{\lambda \cdot s} = X_j,$$

có nghĩa là:

$$K_t = \widetilde{\mathcal{H}}(X_t, kf_t) = \widetilde{\mathcal{H}}(X_j, \widetilde{w}_{i_j}) = Y_j$$

3.4.3. Đánh giá an toàn dữ liệu

Đề xuất được chứng minh an toàn dưới bài toán khó BDHE, bài toán khó BDHE được trình bày như sau:

Định nghĩa 3.4. Bài toán BDHE: Giả sử $(p, \mathbb{G}, \mathbb{G}_T, e)$ là một hệ thống ánh xạ song tuyến chọn $a, t, s, q, \theta, r_1, \ldots, r_0 \xleftarrow{\$} \mathbb{Z}_p$, và một phần tử sinh $g \in \mathbb{G}$. Cho trước:

$$\vec{Y} = g, g^s, g^a, \ldots, g^{a^q}, g^{a^{q+2}}, \ldots, g^{a^{2q}}, g^{s(at+a)}, g^{at}, g^{a^2 t}, \ldots, g^{a^q t}, g^{a^{q+2} t}, \ldots, g^{a^{2q} t},$$
$$g^{a^{q+1}}\ g^{a.r_1}, \ldots, g^{a^{q+1}}, g^{a.r_\theta}, g^{r_1}, \ldots g^{r_\theta})$$

và một giá trị $T \in \mathbb{G}_T$, xác định $T = e(g,g)^{a^{q+1} s} \in \mathbb{G}_T$ hay T chỉ là một giá trị ngẫu nhiên trong tập $\mathbb{G}_T$.

Mục này, luận án chứng minh lược đồ mã hóa được đề xuất ở trên đạt an toàn dữ liệu, tức là kẻ tấn công từ bản mã sẽ không biết bất kỳ thông tin gì về bản rõ.

Định lý 3.5. Giả sử rằng β^* là chính sách mã hóa mà kẻ tấn công muốn tấn công, và từ β^* xây dựng ma trận tương ứng L' kích thước $\ell' \times n'$ và hàm ánh xạ ρ'. Biểu diễn $\beta^* = \beta_1^* \vee \cdots \vee \beta_m^*$. Trong đó $\beta_i^*, i = 1, \ldots, m$ là các tập không giao nhau và sau đó xây dựng ma trận tương ứng L^* kích thước $\ell^* \times n^*$ và hàm ánh xạ ρ^*. Nếu những ma trận này thỏa mãn $\ell', n', \ell^*, n^* \leq q$, và nếu $\theta \geq k^* . q^*$ trong đó k^* và q^* là số tối đa các phép kết hợp từ khóa trong một cửa sập và số tối đa các truy vấn của sập mà kẻ tấn công có thể truy vấn, mã hóa trên đạt an toàn dữ liệu dưới giả thuyết bài toán BDHE là khó.

***Chứng minh*:** Phương pháp chứng minh là phương pháp phản chứng, tức là giả sử tồn tại một kẻ tấn công $\mathcal{A}$ phá vỡ an toàn của hệ mã trên, thì cũng sẽ tồn tại một kẻ tấn công $\mathcal{S}$ giải được bài toán khó BDHE. Với phương pháp như vậy, ban đầu $\mathcal{S}$ được cho trước các giá trị như trong định nghĩa bài toán khó BDHE, sau đó $\mathcal{S}$ cần phân biệt $T = e(g,g)^{a^{q+1}s} \in \mathbb{G}_T$ hay T chỉ là một giá trị ngẫu nhiên trong tập $\mathbb{G}_T$.

Với mục đích như vậy, đầu tiên $\mathcal{S}$ sẽ mô phỏng $\mathcal{A}$ sau đó dùng kết quả của $\mathcal{A}$ để dự đoán T. Cụ thể, đầu tiên $\mathcal{S}$ sẽ nhận chính sách bản mã mà $\mathcal{A}$ muốn tấn công β^*. Giả sử kích thước của β^* là $|\beta^*|$. $\mathcal{S}$ biểu diễn β^* dưới dạng $\beta^* = \beta_1^* \vee \cdots \vee \beta_m^*$ trong đó $\beta_i^*, i = 1, \ldots, m$ là các tập không giao nhau. Tiếp theo, $\mathcal{S}$ so sánh giữa m và $|\beta^*|$, có hai trường hợp xảy ra. Trường hợp đầu tiên là $m \leq |\beta^*|$.

Từ chính sách bản mã $\beta^*\beta_1^* \vee \cdots \vee \beta_m^*, \mathcal{S}$ xây dựng ma trận $(M_{\ell^* \times n^*}^*, \rho^*)$ sao cho $\ell^*, n^* \leq q$ để tạo ra các tham số cho hệ thống. $\mathcal{S}$ chọn ngẫu nhiên $\alpha' \overset{\$}{\leftarrow} \mathbb{Z}_p$ và đặt $\alpha = \alpha' + a^{q+1}$

Sau đó tính $e(g,g)^{\alpha} = e(g^{a}, g^{a^q}) e(g,g)^{\alpha'}$. $\mathcal{S}$ tiếp tục tìm tập các dòng của ma trận M^*: $I_1, .., I_m$ trong đó $\{\rho(i), i \in I_j\} = \beta_j^*$ (chú ý rằng $I_j, j = 1, .., m$ là các tập không giao nhau vì β_j^* là các tập không giao nhau). Lúc này, β^* có thể được biểu diễn lại như: $(\wedge \rho(i))_{i \in I_1} \vee (\wedge \rho(i))_{i \in I_2} \vee \cdots \vee (\wedge \rho(i))_{i \in I_m}$. Để tạo ra các phần tử $h_1, \ldots, h_N, \mathcal{S}$ định nghĩa vector:

$$\vec{y} = (t, ta, ta^2, \ldots, ta^{n^*-1})^{\perp} \mathbb{Z}_p^{n^*}$$

Đặt $\vec{\Lambda} = (\lambda_1, \ldots, \lambda_{\ell^*}) = M^* \cdot \vec{y}$ là véc tơ chia sẻ thành phần bí mật, do đó với $j = 1, \ldots, \ell^*$

$$\lambda_j = \sum_{i \in [n^*]} M_{j,i}^* t a^{i-1}$$

$\mathcal{S}$ có thể tìm các tập $\{\omega_i\}_{1 \leq i \leq \ell^*}$ sao cho với mọi $j = 1, \ldots, m$

$$\sum_{i \in I_j} \omega_i \cdot \lambda_i = t$$

Lưu ý, S có thể tìm các tập $\{\omega_i\}_{1 \leq i \leq \ell^*}$ vì từ tính chất của ma trận chia sẻ tuyến tính tồn tại $\{\omega_i\}_{1 \leq i \leq \ell^*}$ sao cho với mọi $j = 1, \ldots, m$

$$\sum_{i \in I_j} \omega_i \cdot M_i^* = (1,0,\ldots,0)$$

Với mỗi $h_j, 1 \leq j \leq N$, trong đó tồn tại một chỉ số $i \in [\ell^*]$ sao cho $j = \rho(i)$ (hàm ρ là ánh xạ 1-1), $\mathcal{S}$ chọn $z_j \xleftarrow{\$} \mathbb{Z}_p$ và tính: Chú ý rằng $\mathcal{S}$ biết ma trận M^* và g^{ta^k} trong đó $k \in [n^*]$ từ giả thuyết đầu vào của bài toán khó BDHE .

$$h_j = g^{z_j} \cdot g^{\omega_i \sum_{k \in [n^*]} M_{i,k}^* \, ta^k} = g^{z_j} \cdot g^{a\omega_i \gtrdot_i}$$

Ngược lại, $\mathcal{S}$ chọn $z_j \xleftarrow{\$} \mathbb{Z}_p$ và tính $h_j = g^{z_j}$. Chú ý rằng $\{h_j\}_{j=1,\ldots,N}$ là phân bố ngẫu nhiên do việc chọn ngẫu nhiên z_j

Để tạo ra các tham số $g^{\gtrdot}$ và $\tilde{h}_1, \ldots, h_N, \mathcal{S}$ đơn giản chọn ngẫu nhiên $\gtrdot, \tilde{z}_1, \ldots, \tilde{z}_N$ từ $\mathbb{Z}_p$ để tính $g^{\gtrdot}$ và $\tilde{h}_1 = g^{\tilde{z}_1}, \ldots, \tilde{h}_N = g^{\tilde{z}_N}$. $\mathcal{S}$ cũng chọn các hàm băm $\mathcal{H}, \widetilde{\mathcal{H}}$. Cuối cùng, $\mathcal{S}$ gửi khóa công khai:

$$\text{param} = (g, g^a, g^{\gtrdot}, e(g,g)^{\alpha}, h_1, \ldots, h_N, \tilde{h}_1, \ldots, \tilde{h}_N, \mathcal{H}, \widetilde{\mathcal{H}})$$

Giai đoạn truy vấn 1: Trong giai đoạn này, $\mathcal{S}$ cần trả lời ba dạng truy vấn sau:

Thứ nhất, là truy vấn về khóa bí mật, với yêu cầu khóa bí mật này không thể có khả năng giải mã được bản mã mà được tạo ra dưới chính sách β^*.

Thứ hai, là truy vấn để biết cửa sập tương ứng với tất cả các khóa bí mật (thậm chí với cả những khóa bí mật mà có khả năng giải mã được bản mã mà được tạo ra dưới chính sách β^*).

Thứ ba, là truy vấn để biết một phần khóa bí mật mà có khả năng giải mã được bản mã, được tạo ra dưới chính sách β^*.

Đối với truy vấn 1, $\mathcal{A}$ đầu tiên gửi tập các chỉ số của thuộc tính $S \subset [N]$ tới $\mathcal{S}$ với điều kiện rằng tập các thuộc tính kết hợp với S không thỏa mãn ma trận M^* (có nghĩa khóa bí mật tương ứng với tập thuộc tính này, không có khả năng giải mã bản mã thách thức). $\mathcal{S}$ tìm véc tơ $\vec{x} = (x_1, \ldots, x_{n^*}) \in \mathbb{Z}_p^{n^*}$ sao cho $x_1 = -1$ và với mọi i trong đó $\rho^*(i) \in S$ tích $\langle \vec{x} \cdot M_i^* \rangle = 0$. Dựa trên tính chất của ma trận chia sẻ, véc tơ $\vec{x}$ sẽ tồn tại. $\mathcal{S}$ tiếp tục chọn $\zeta \xleftarrow{\$} \mathbb{Z}_p$ và giá trị s_u được tính như sau:

$$s_u = \zeta + x_1 a^q + x_2 a^{q-1} + \cdots + x_{n^*} a^{q-n^*} + 1$$

$\mathcal{S}$ tính:

$$d_{u_0} = g^{\alpha'} g^{a\zeta} \prod_{i=2,\dots,n^*} (g^{a^{q+1-i}})^{x_i} = g^{\alpha} \cdot g^{a \cdot s_u}$$

Vì $x_1 = -1$ nên $g^{a \cdot s_u}$ chứa thành phần $g^{-a^{q+1}}$ đây là giá trị có thể triệt tiêu thành phần $g^{\ a^{q+1}}$trong g^{α}.

Với vector $\vec{x}$ đã biết, $\mathcal{S}$ tính:

$$d'_{u_0} = g^{s_u} = g^{\zeta} \prod_{i=1,\dots,n^*} (g^{a^{q+1-i}})^{x_i}$$

Với mỗi $j \in S$ phải không có $i \in [\ell^*]$ thỏa mãn $\rho^*(i) = j$. $\mathcal{S}$ biết giá trị z_j và tính $h_j^{s_u} = (g^{s_u})^{z_j}$

Với mỗi $j \in S$ sao cho có $i \in [\ell^*]$ thỏa mãn $\rho^*(i) = j$. $\mathcal{S}$ tính:

$$h_j^{s_u} = (g^{s_u})^{z_j} \cdot g^{(\zeta + x_1 a^q + x_2 a^{q-1} + \cdots + x_{n^*} a^{q-n^*+1}) \omega_i \sum_{k \in [n^*]} M^*_{i,k} t a^k}$$

Vì giá trị tích $\langle \vec{x} \cdot M_i^* \rangle = 0$, do đó $\mathcal{S}$ không cần thiết phải biệt giá trị thành phần $g^{a^{q+1}t}$ để tính $h_j^{s_u}$, tất cả các giá trị khác $\mathcal{S}$ đã biết từ đầu vào của bài toán khó. Nếu j không thuộc tập S và tồn tại $i \in [\ell^*]$ sao cho $\rho^*(i) = j, \mathcal{S}$ không thể tính $h_j^{s_u}$ vì $\langle \vec{x} \cdot M_i^* \rangle \neq 0$. Cuối cùng, $\mathcal{S}$ cũng biết $\succ$.

Đối với truy vấn thứ hai thì có hai trường hợp sau:

$\mathcal{A}$ yêu cầu cửa sập tương ứng với khóa bí mật không có khả năng giải mã bản mã thách thức. Trong trường hợp này, $\mathcal{S}$ chỉ đơn giản chạy giải thuật cửa sập để tạo ra cửa sập vì $\mathcal{S}$ đã biết khóa bí mật.

$\mathcal{A}$ yêu cầu cửa sập tương ứng với khóa bí mật có khả năng giải mã bản mã thách thức. Giả sử rằng $\mathcal{A}$ cung cấp một tập các phép kết hợp từ khóa:

$$W_i = (\widetilde{\omega}_{i_1}, \dots, \widetilde{\omega}_{i_k})$$

Để tính: $\{\mathrm{tds}_{0,j}, \mathrm{tds}_{1,j}\}_{j \in [k]}, \mathcal{S}$ chọn $s_u \xleftarrow{\$} \mathbb{Z}_p$ sau đó tính:

$$\mathrm{tds}_{0,j} = g^{\alpha'} g^{a^{q+1}} g^{ar_t} g^{as_u} (g^a \mathcal{H}(\widetilde{w}_{i_j}))^{\succ} = g^{\alpha} g^{ar_t} g^{as_u} (g^a \mathcal{H}(\widetilde{w}_{i_j}))^{\succ}$$

và đặt $\mathrm{tds}_{1,j} = g^{r_t}$. Lưu ý, đối với mỗi $\{\mathrm{tds}_{0,j}, \mathrm{tds}_{1,j}\}; \mathcal{S}$ dùng các cặp giá trị khác nhau $\{g^{a^{q+1}} g^{a.r_t}, g^{r_t}\}, t \in [\theta]$ từ giả thuyết của bài toán khó.

Để tính $\{\{\mathrm{tds}_{2,j,\ell}\}_{\ell\in B(u)}\}_{j\in[k]}, \mathcal{S}$ tính: $\tilde{h}_\ell^{r_j} = (g^{r_j})^{\tilde{z}_\ell}$, vì $\mathcal{S}$ biết g^{r_j} từ giả thuyết bài toán khó và $\tilde{z}_\ell$ khi tính $\tilde{h}_\ell$.

Đối với $\mathrm{tds}_0, (\mathrm{tds}_i)_{i\in\mathcal{B}(u)}$ thì $\mathcal{S}$ dễ dàng tính vì $\mathcal{S}$ biết s_u. Lưu ý, vì $\mathcal{S}$ biết s_u, $\mathcal{S}$ có thể tính $h_i^{s_u}$ với bất kỳ tập $\mathcal{B}(u)$ nào.

Đối với loại truy vấn thứ ba, $\mathcal{S}$ chọn $s_u \xleftarrow{\$} \mathbb{Z}_p$ và tính tất cả các thành phần của d_u ngoại trừ d_{u_0}, vì giá trị $g^{a^{q+1}}$ chỉ xuất hiện trong d_{u_0}. Cuối cùng, $\mathcal{S}$ trả giá trị $d'_{u_0}, \{d_i\}_{i\in\mathcal{B}_u}$ về.

Giai đoạn thách thức: Kẻ tấn công $\mathcal{A}$ gửi hai bản rõ $\mathcal{M}_0^*, \mathcal{M}_1^*$ cho $\mathcal{S}$. $\mathcal{S}$ chọn ngẫu nhiên một bit *b*, tính:

$$C^* = \mathcal{M}_b^* \cdot T \,.\, e(g^s, g^{\alpha'}), C_0^* = g^s$$

và các phần tử khác:

$$(C_1^*, \ldots, C_m^*) = \left(g^{s(a+at)} g^{\sum_{i\in I_1} sz_{\rho^*(i)}}, \ldots, g^{s(a+at)} g^{\sum_{i\in I_m} sz_{\rho^*(i)}}\right)$$

$$= \left((g^a \cdot \prod_{i\in I_1} g^{z_{\rho^*(i)}} \cdot g^{a\omega_i \gtrdot i})^s, \ldots, (g^a \cdot \prod_{i\in I_1} g^{z_{\rho^*(i)}} \cdot g^{a\omega_i \gtrdot i})^s \right)$$

$$= \left((g^a \prod_{i\in I_1} h_{p^*(i)})^s, \ldots, g^a \prod_{i\in I_m} h_{p^*(i)})^s \right)$$

$$= \left((g^a \prod_{i\in \beta_1^*} h_i)^s, \ldots, (g^a \prod_{i\in \beta_m^*} h_i)^s \right)$$

trong đó tập $\{\rho^*(i), i \in I_j\}$ *là* $\beta_j^*, j \in [m]$. Chú ý, nếu $T = e(g,g)^{a^{q+1}s}$ thì ct^* là bản mã hợp lệ.

Giai đoạn truy vấn thứ hai: Tương tự như giai đoạn truy vấn thứ nhất.

Giai đoạn dự đoán kết quả: $\mathcal{A}$ đưa kết quả dự đoán b' của nó cho $\mathcal{S}$, $\mathcal{S}$ cho đầu ra dự đoán của nó là 0. Tương ứng với dự đoán rằng $T = e(g,g)^{a^{q+1}s}$ nếu $b' = b$. Ngược lại, $\mathcal{S}$ cho đầu ra là 1 tương ứng với dự đoán rằng T là một phần tử ngẫu nhiên trong tập $\mathbb{G}_T$.

Khi $T = e(g,g)^{a^{q+1}s}, \mathcal{S}$ mô phỏng chính xác kẻ tấn công $\mathcal{A}$, do đó:

$$\Pr[\mathcal{S}(\vec{Y}, T = e(g,g)^{a^{q+1}s}) = 0]$$
$$= \frac{1}{2} + \mathrm{Adv}_{\mathcal{A}}^{Se}$$

Khi T là một phần tử ngẫu nhiên $\mathcal{M}_b^*$ hoàn toàn ngẫu nhiên dưới tầm nhìn của $\mathcal{A}$, do đó:

$$Pr[\mathcal{S}(\vec{Y}, T = R) = 0] = \frac{1}{2}$$

Vì vậy, nếu $\mathcal{A}$ có khả năng phá hệ mã thì $\mathcal{S}$ cũng có khả năng giải bài toán khó BDHE với cùng xác suất.

Trường hợp thứ hai: $m > |\beta_*|$. Từ chính sách mã hóa β_*, $\mathcal{S}$ xây dựng ma trận $(M^*_{\ell^* \times n^*}, \rho^*)$ sao cho cả hai $\ell^*, n^* \leq q$. Để tính các tham số cho hệ thống $\mathcal{S}$ chọn $\alpha' \overset{\$}{\leftarrow} \mathbb{Z}_p$ và đặt $\alpha = \alpha' + a^{q+1}$, sau đó tính $e(g,g)^{\alpha} = e(g^{a}, g^{a^q}) e(g,g)^{\alpha'}$; Để tính các giá trị $h_1,...,h_N$, đối với mỗi h_j, $1 \leq j \leq N$, trong đó tồn tại chỉ số $i \in [\ell^*]$ sao cho $j = \rho^*(i)$, $\mathcal{S}$ chọn $z_j \overset{\$}{\leftarrow} \mathbb{Z}_p$ và tính $h_j = g^{z_j} \cdot g^{\sum_{k \in [n^*]} M^*_{i,k} a^k}$ ($\mathcal{S}$ biết ma trận $\mathcal{M}_*$ và g^{a^k} trong đó $k \in [n^*]$ từ giả thuyết của bài toán khó BDHE)

Ngược lại, $\mathcal{S}$ chọn $z_j \overset{\$}{\leftarrow} \mathbb{Z}_p$ và tính $h_j = g^{z_j}$.

Để tính các giá trị tham số khác như g^{γ} và $\tilde{h}_1, \ldots, \tilde{h}_N$, $\mathcal{S}$ làm tương tự như trường hợp thứ nhất. $\mathcal{S}$ cũng chọn hàm băm $\mathcal{H}, \widetilde{\mathcal{H}}$. Cuối cùng, $\mathcal{S}$ gửi param cho $\mathcal{A}$

$$\text{param} = (g, g^a, g^{\gamma}, e(g,g)^{\alpha}, h_1, \ldots, h_N, \tilde{h}_1, \ldots, \tilde{h}_N, \mathcal{H}, \widetilde{\mathcal{H}})$$

Giai đoạn truy vấn thứ nhất: Trong giai đoạn này tương tự như trường hợp thứ nhất, $\mathcal{S}$ vẫn cần trả lời ba loại truy vấn như trên.

Đối với loại truy vấn thứ nhất, $\mathcal{A}$ đầu tiên gửi tập các chỉ số của tập thuộc tính $\mathcal{S} \subset [N]$ cho $\mathcal{S}$ với yêu cầu tập các thuộc tính mà tương ứng với $\mathcal{S}$ là không thỏa mãn ma trận $\mathcal{M}^*$; $\mathcal{S}$ tìm véc tơ $\vec{x} = (x_1, .., x_{n^*}) \in \mathbb{Z}_p^{n^*}$ sao cho: $x_1 = -1$ và với mọi i trong đó $\rho^*(i) \in \mathcal{S}$ giá trị tích $\langle \vec{x} \cdot M_i^* \rangle = 0$. $\mathcal{S}$ tiếp tục chọn $\zeta \overset{\$}{\leftarrow} \mathbb{Z}_p$ và định nghĩa giá trị s_u như sau:

$$s_u = \zeta + x_1 a^q + x_2 a^{q-1} + \cdots + x_{n^*} a^{q-n^*+1}$$

$\mathcal{S}$ tính:

$$d_{u_0} = g^{\alpha'} g^{a\zeta} \prod_{i=2,\ldots,n^*} (g^{a^{q+1-i}})^{x_i} = g^{\alpha} \cdot g^{a \cdot s_u}$$

Do $x_1 = -1$ nên $g^{a \cdot s_u}$ chứa thành phần $g^{-a^{q+1}}$. Đây là giá trị sẽ triệt tiêu giá trị $g^{a^{q+1}}$ trong g^{a}. Với véc tơ $\vec{x}$ đã biết, $\mathcal{S}$ tiếp tục tính:

$$d'_{u_0} = g^{s_u} = g^{\zeta} \prod_{i=2,\ldots,n^*} (g^{a^{q+1-i}})^{x_i}$$

Với mỗi $j \in \mathcal{S}$ sao cho không có $i \in [\ell^*]$ thỏa mãn $\rho^*(i) = j$. $\mathcal{S}$ biết giá trị z_j và tính:

$$h_j^{s_u} = (g^{s_u})^{z_j}$$

Với mỗi $j \in \mathcal{S}$ sao cho có một $i \in [\ell^*]$ thỏa mãn $\rho^*(i) = j$. $\mathcal{S}$ tính:

$$h_j^{s_u} = (g^{s_u})^{z_j} \cdot g^{(\zeta + x_1 a^q + x_2 a^{q-1} + \cdots + x_{n^*} a^{q-n^*+1}) \sum_{k \in [n^*]} M^*_{i,k} a^k}$$

Lưu ý, vì tích $\langle \vec{x} \cdot M_i^* \rangle = 0$ do đó $\mathcal{S}$ không cần biết thành phần $g^{a^{q+1}}$ để tính $h_j^{s_u}$, tất cả các thành phần khác $\mathcal{S}$ đã biết từ giả thuyết bài toán khó, và $\mathcal{S}$ cũng biết $\succ$.

Đối với loại truy vấn thứ hai; $\mathcal{S}$ trả lời tương tự như ở trường hợp thứ nhất.

Giai đoạn thách thức: Kẻ tấn công $\mathcal{A}$ gửi hai bản rõ $\mathcal{M}_0^*, \mathcal{M}_1^*$ cho $\mathcal{S}$; $\mathcal{S}$ chọn ngẫu nhiên một bit *b*, tính:

$$C^* = M_b^* \cdot T \cdot e(g^s, g^{\alpha'}), C_0^* = g^s$$

Để tính các giá trị khác, $\mathcal{S}$ đầu tiên chọn $y'_2, \ldots, y'_{n^*} \xleftarrow{\$} \mathbb{Z}_p$, đặt véc tơ

$$\vec{v} = (s, sa + y'_2, sa^2 + y'_3, \ldots, sa^{n^*-1} + y'_{n^*}) \in \mathbb{Z}_p^{n^*}$$

sau đó tính:

$$C_i^* = \left(\prod_{j=2}^{n^*} (g^a)^{M^*_{i,j} y'_j} \right) (g^s)^{-z_{\rho^*(i)}}, i = 1, \ldots, \ell$$

Vì $\succ_i = \vec{v} . M_i^*$, do đó:

$$g^{a \succ_i} = g^{a(s.M^*_{i,1} + (sa + y'_2).M^*_{i,2} + \cdots + (sa^{n^*-1} + y'_{n^*}) M^*_{i,n^*})}$$

và

$$(h_{\rho^*(i)})^{-s} = \left(g^{z_{\rho^*(i)}} \cdot g^{\sum_{j \in [n^*]} M^*_{i,j} a^j} \right)^{-s}$$

Suy ra:

$$C_i^* = \left(\prod_{j=2}^{n^*} (g^a)^{M_{i,j}^* y_j'}\right)(g^s)^{-z_{\rho^*(i)}} = g^{a\gamma_i} \cdot (h_{\rho^*(i)})^{-s}$$

Giai đoạn dự đoán kết quả: Tương tự như ở trường hợp 1.

3.4.4. Đánh giá an toàn từ khóa

Trong phần này, sẽ chứng minh rằng ý tưởng đề xuất đạt an toàn về từ khóa (kẻ tấn công không biết bất kỳ thông tin gì về từ khóa được dùng để tìm kiếm thông tin) đối với cả hai kẻ tấn công, một là ở bên trong hệ thống (máy chủ), hai là kẻ tấn công thông thường ở bên ngoài hệ thống.

An toàn trước kẻ tấn công từ bên trong hệ thống:

Định lý 3.6: Giả sử rằng $\beta^* = \beta_1^* \vee ... \vee \beta_m^*$ là chính sách bản mã thsách thức, và từ β^* xây dựng được ma trận L^* có kích thước $\ell^* \times n^*$ và ánh xạ ρ^*. Nếu ma trận trên thỏa mãn $\ell^* \leq q, n^* \leq q$, hệ mã sẽ đạt được an toàn từ khóa trước kẻ tấn công bên trong hệ thống dưới giả thuyết rằng bài toán BDHE là khó.

***Chứng minh*:** Phương pháp chứng minh là phương pháp phản chứng. Tức là, giả sử tồn tại một kẻ tấn công $\mathcal{A}$ phá vỡ an toàn từ khóa của hệ mã trên, thì cũng sẽ tồn tại một kẻ tấn công $\mathcal{S}$ giải được bài toán khó BDHE . Với phương pháp như vậy, ban đầu $\mathcal{S}$ được cho trước các giá trị như trong định nghĩa bài toán khó BDHE, sau đó $\mathcal{S}$ cần phân biệt $T = e(g,g)^{a^{q+1}s} \in \mathbb{G}_T$ hay T chỉ là một giá trị ngẫu nhiên trong tập $\mathbb{G}_T$.

Tương tự, như chứng minh ở phần trên, ban đầu $\mathcal{S}$ biết được giả thuyết đầu vào của bài toán khó và chính sách $\beta^* = \beta_1^* \vee ... \vee \beta_m^*$ và

$\mathcal{KF}_0^* = kf_{0,1}^*, ..., kf_{0,m'}^*$ và $\mathcal{KF}_1^* = (kf_{1,1}^*, ..., kf_{1,m'}^*)$ từ $\mathcal{A}$.

Chú ý rằng $\beta_i^*, i = 1, ..., m$ là các tập không giao nhau. Từ $\beta^* = \beta_1^* \vee ... \vee \beta_m^*$; $\mathcal{S}$ xây dựng ma trận $(\mathcal{M}_{\ell^* \times n^*}^*, \rho^*)$ sao cho cả hai $\ell^*, n^* \leq q$.

Để tính các tham số khác cho hệ thống, $\mathcal{S}$ chọn $\alpha' \xleftarrow{\$}, \mathbb{Z}_p$ và đặt $\alpha = \alpha' + \alpha^{q+1}$, sau đó tính $e(g,g)^\alpha = e(g^a, g^{a^q}) e(g,g)^{\alpha'}$.

$\mathcal{S}$ tiếp theo tìm các tập các dòng của ma trận M^*: $I_1, \dots, I_m$ trong đó $\rho(i), i \in I_j = \beta_j^*$ (chú ý rằng $I_j j = 1, \dots, m$ là các tập không giao nhau vì β_j^* là các tập không giao nhau). Lúc này, β^* có thể được viết lại là: $(\wedge \rho(i))_{i \in I_1} \vee (\wedge \rho(i))_{i \in I_2} \vee \cdots \vee (\wedge \rho(i))_{i \in I_m}$.

Để tính các giá trị $h_1, \dots, h_N, \tilde{h}_1, \dots, \tilde{h}_N, \mathcal{S}$ đặt vector:

$$\vec{y} = (t, ta, ta^2, \dots, ta^{n^*-1})^{\perp} \in \mathbb{Z}_p^{n^*}$$

Đặt $\vec{\Lambda} = (\lambda_1, \dots, \lambda_{\ell^*}) = M^* \cdot \vec{y}$ là véc tơ chia sẻ, do đó với $j = 1, \dots, \ell^*$

$$\lambda_j = \sum_{i \in [n^*]} M_{j,i}^* ta^{i-1}$$

$\mathcal{S}$ tìm tập $\{\omega_i\}_{1 \le i \le \ell^*}$ sao cho với mọi $j = 1, \dots, m$

$$\sum_{i \in I_j} \omega_i \cdot \lambda_i = t$$

Cần lưu ý, $\mathcal{S}$ có thể tìm tập $\{\omega_i\}_{1 \le i \le \ell^*}$ vì từ tính chất của ma trận chia sẻ tuyến tính, ta có thể tìm $\{\omega_i\}_{1 \le i \le \ell^*}$ sao cho với mọi $j = 1, \dots, m$

$$\sum_{i \in I_j} \omega_i \cdot M_i^* = (1, 0, \dots, 0)$$

Với mỗi $h_j, \tilde{h}_j, 1 \le j \le N$, trong đó tồn tại một chỉ số $i \in [\ell^*]$ sao cho $j = \rho^*(i)$, $\mathcal{S}$ chọn $z_j, \tilde{z}_j \xleftarrow{\$} \mathbb{Z}_p$ và tính:

$$h_j = g^{z_j} \cdot g^{\omega_i \sum_{k \in [n^*]} M_{i,k}^* ta^k} = g^{z_j} \cdot g^{a \omega_i \lambda_i}$$

$$\tilde{h}_j = g^{\tilde{z}_j} \cdot g^{\omega_i \sum_{k \in [n^*]} M_{i,k}^* ta^k} = g^{\tilde{z}_j} \cdot g^{a \omega_i \lambda_i}$$

Lưu ý, $\mathcal{S}$ biết ma trận M^* và g^{ta^k} trong đó $k \in [n^*]$ từ giả thuyết bài toán khó BDHE .

Ngược lại, $\mathcal{S}$ chọn $z_j, \tilde{z}_j \xleftarrow{\$} \mathbb{Z}_p$ và tính $h_j = g^{z_j}, \tilde{h}_j = g^{\tilde{z}_j}$. Chú ý rằng $\{h_j, \tilde{h}_j\}_{j=1,\dots N}$ là phân bố ngẫu nhiên do việc chọn ngẫu nhiên $z_j, \tilde{z}_j$

Để tính g^{λ}, $\mathcal{S}$ đặt $\lambda = -a^q$ và tính $g^{\lambda} = \left(g^{a^q}\right)^{-1}$. $\mathcal{S}$ cũng chọn các hàm băm $\mathcal{H}, \widetilde{\mathcal{H}}$; Cuối cùng $\mathcal{S}$ gửi param cho $\mathcal{A}$.

$$\text{param} = (g, g^a, g^{\gamma}, e(g,g)^{\alpha}, h_1, \ldots, h_N, \tilde{h}_1, \ldots, \tilde{h}_N, \mathcal{H}, \widetilde{\mathcal{H}})$$

Giai đoạn truy vấn thứ nhất: Trong giai đoạn này $\mathcal{S}$ cần trả lời năm loại truy vấn:

1. Truy vấn từ hàm băm;

2. Truy vấn cửa sập $\left(\mathcal{B}_u, W_i = \left(\tilde{\omega}_{i_1}, \ldots, \tilde{\omega}_{i_k}\right)\right)$ trong đó W_i không thỏa mãn $\mathcal{KF}_0^*$ hay $\mathcal{KF}_1^*$, có nghĩa rằng không tồn tại bất kỳ bộ ba $\left(i_j, b, b'\right)$ sao cho $\tilde{\omega}_{i_j} = kf_{b,b'}^*$

3. Truy vấn cửa sập $(\mathcal{B}_u, W_i)$ trong đó W_i thỏa mãn $\mathcal{KF}_0^*$ hay $\mathcal{KF}_1^*$, nhưng $\mathcal{B}_u$ không thỏa mãn β^*.

4. Truy vấn cửa sập một phần $(\mathcal{B}_u, W_i)$

Trong đó W_i thỏa mãn $\mathcal{KF}_0^*$ hay $\mathcal{KF}_1^*$ và $\mathcal{B}_u$ thỏa mãn β^*. Chú ý, người dùng chỉ giữ $\left(\left\{\text{tds}_{0,j}\right\}_{j\in[k]}, \widetilde{\text{W}}_\text{i}\right)$ bí mật và công bố phần còn lại của tds.

Có nghĩa rằng $\mathcal{A}$ có thể biết $\left(\{\text{tds}_{1,j}, \{\text{tds}_{2,j,\ell}\}_{\ell\in\mathcal{B}_u}\}_{j\in[k]}, \text{tds}_0, \{\text{tds}_i\}_{i\in\mathcal{B}_u}\right)$ cho bất kỳ $\left(\mathcal{B}_u, W_i = \left(\tilde{\omega}_{i_1}, \ldots, \tilde{\omega}_{i_k}\right)\right)$

5. Truy vấn một phần khóa bí mật $\mathcal{B}_u$ với bất kỳ tập $\mathcal{B}_u$. Lý do là $\mathcal{S}$ chỉ giữ mình d_{u_0} bí mật.

• Đối với truy vấn từ hàm băm: $\mathcal{S}$ tạo ra hai danh sách $\mathcal{L}, \tilde{\mathcal{L}}$, ban đầu $\mathcal{L}, \tilde{\mathcal{L}}$ là rỗng. Với mỗi truy vấn tương ứng với $\tilde{\omega}_i$ mà không thỏa mãn $\mathcal{KF}_0^*$ hay $\mathcal{KF}_1^*$, $\mathcal{S}$ đầu tiên kiểm tra xem $\tilde{\omega}_i$ đã được truy vấn trước đây chưa. Nếu chưa, $\mathcal{S}$ chọn $\mathcal{Y}_i \overset{\$}{\leftarrow}, \mathbb{Z}_p$ và thêm bộ ba $(\tilde{\omega}_i, g^{\mathcal{Y}_i}, \mathcal{Y}_i) \in (\{0,1\}^*, \mathbb{G}, \mathbb{Z}_p)$ vào $\mathcal{L}$ và trả về cho $g^{\mathcal{Y}_i}$. Ngược lại, $\mathcal{S}$ đơn giản là tìm bộ ba $(\tilde{\omega}_i, g^{\mathcal{Y}_i}, \mathcal{Y}_i)$ và trả về $g^{\mathcal{Y}_i}$ cho $\mathcal{A}$. Trong trường hợp $\tilde{\omega}_i$ thỏa mãn $\mathcal{KF}_0^*$ hay $\mathcal{KF}_1^*$; $\mathcal{S}$ đầu tiên kiểm tra xem $\tilde{\omega}_i$ đã được truy vấn trước đây chưa. Nếu chưa, $\mathcal{S}$ chọn $\mathcal{Y}_i \overset{\$}{\leftarrow} \mathbb{Z}_p$ và thêm bộ ba $(\tilde{\omega}_i, g^{-a} \cdot g^{\mathcal{Y}_i}, \mathcal{Y}_i)$ vào trong $\mathcal{L}$ và trả về $g^{-a} \cdot g^{\mathcal{Y}_i}$ cho $\mathcal{A}$. Ngược lại, $\mathcal{S}$ đơn giản là tìm bộ ba $(\tilde{\omega}_i, g^{-a} \cdot g^{\mathcal{Y}_i}, \mathcal{Y}_i)$ và trả về $g^{-a} \cdot g^{\mathcal{Y}_i}$ cho.

Với mỗi truy vấn hàm băm tương ứng với $\left(K_i, kf_j\right)$ trong đó $K_i \in \mathbb{G}_T, kf_j \in \{0,1\}^*$, $\mathcal{S}$ đầu tiên kiểm tra xem $\left(K_i, kf_j\right)$ đã được truy vấn trước đây hay chưa. Nếu

chưa, $\mathcal{S}$ chọn $\mathcal{Y}_{i_j} \overset{\$}{\leftarrow} \mathbb{Z}_p$và thêm bộ ba ($K_i$, kf_j, $\mathcal{Y}_{i_j}$) vào trong $\tilde{\mathcal{L}}$. Ngược lại, $\mathcal{S}$ tìm bộ ba (K_i , kf_j, $\mathcal{Y}_{i_j}$) và trả về $\mathcal{Y}_{i_j}$cho $\mathcal{A}$.

• Đối với loại truy vấn thứ hai: $\mathcal{A}$ đầu tiên gửi $W_i = (\tilde{\omega}_{i_1}, \ldots, \tilde{\omega}_{i_k})$ và $\mathcal{B}(u)$ cho $\mathcal{S}$ với ràng buộc rằng W_i không thỏa mãn $\mathcal{KF}_0^*$ hay $\mathcal{F}_1^*$.

Để tính mỗi $\text{tds}_{0,j}, j \in [k]$, $\mathcal{S}$ đầu tiên kiểm tra xem $\tilde{\omega}_{i_j}, j \in [k]$ đã được truy vấn trước đây hay chưa. Nếu chưa, $\mathcal{S}$ chọn $\mathcal{Y}_{i_j} \overset{\$}{\leftarrow} \mathbb{Z}_p$ và thêm bộ ba $\left(\tilde{\omega}_i, g^{\mathcal{Y}_{i_j}}, \mathcal{Y}_{i_j}\right)$ vào trong $\mathcal{L}$.

Trong cả hai trường hợp, $\mathcal{S}$ đều biết $\mathcal{Y}_{i_j}$ từ $\mathcal{L}$, và $\mathcal{H}(\tilde{\omega}_{i_j}) = g^{\mathcal{Y}_{i_j}}$ vì W_i không thỏa mãn $\mathcal{KF}_0^*$ hay $\mathcal{KF}_1^*$. Tiếp theo, $\mathcal{S}$ chọn $s_u, r_j \overset{\$}{\leftarrow} \mathbb{Z}_p$ và tính:

$$\text{tds}_{0,j} = g^{\alpha'} g^{as_u} g^{ar_j} (g^{\lambda})^{\mathcal{Y}_{i_j}} = g^{\alpha} g^{as_u} g^{ar_j} (g^{a}\mathcal{H}(\tilde{\omega}_{i_j}))^{\lambda}$$

Chú ý rằng: $g^{\alpha'} = g^{\alpha'} \cdot g^{a^{q+1}} \cdot g^{-a^{q+1}} = g^{\alpha} \cdot g^{a.\lambda}$, vì $g^{\lambda} = g^{-a^q}$

Vì $\mathcal{S}$ biết $s_u, r_j, j \in [k]$, $\mathcal{S}$ có thể tính phần còn lại của cửa sập với bất kỳ tập $\mathcal{B}(u)$ nào. Cuối cùng, $\mathcal{S}$ trả về tds cho.

Đối với loại truy vấn thứ ba: $\mathcal{A}$ đầu tiên gửi $W_i = (\tilde{\omega}_{i_1}, \ldots, \tilde{\omega}_{i_k})$ và $\mathcal{B}(u)$ cho $\mathcal{S}$ với ràng buộc rằng $B(u)$ không thỏa mãn β^* và W_i thỏa mãn $\mathcal{KF}_0^*$ hay $\mathcal{KF}_1^*$. $\mathcal{S}$ đầu tiên tìm véc tơ $\vec{x} = (x_1, \ldots, x_{n^*}) \in \mathbb{Z}_p^{n^*}$ sao cho $x_1 = -1$ và với mọi *i* trong đó $\rho^*(i) \in \mathcal{B}(u)$ giá trị tích $\langle \vec{x} \cdot M_i^* \rangle = 0$; $\mathcal{S}$ tiếp tục chọn $\zeta \overset{\$}{\leftarrow} \mathbb{Z}_p$ và đặt giá trị s_u như sau:

$$s_u = \zeta + x_1 a^q + x_2\, a^{q-1} + \cdots + x_{n^*}\, a^{q-n^*+1}$$

$\mathcal{S}$ tiếp theo tính:

$$d_{u_0} = g^{\alpha'} g^{a\zeta} \prod_{i=2,\ldots,n^*} (g^{a^{q+1-i}})^{x_i} = g^{\alpha} \cdot g^{a \cdot s_u}$$

$$d'_{u_0} = g^{\zeta} \prod_{i=2,\ldots,n^*} (g^{a^{q+1-i}})^{x_i} = g^{s_u}$$

Với mỗi $j \in \mathcal{B}(u)$ sao cho không có $i \in [\ell^*]$ thỏa mãn $\rho^*(i) = j$. $\mathcal{S}$ biết giá trị z_j và tính:

$$h_j^{s_u} = (g^{s_u})^{z_j}$$

Với mỗi $j \in \mathcal{B}(u)$ sao cho có $i \in [\ell^*]$ thỏa mãn $\rho^*(i) = j$. $\mathcal{S}$ tính:

$$h_j^{s_u} = (g^{s_u})^{z_j} \cdot g^{(\zeta + x_1 a^q + x_2 a^{q-1} + \cdots + x_{n^*} a^{q-n^*+1}) \omega_i \sum_{k \in [n^*]} M_{i,k}^* t a^k}$$

Chú ý, giá trị tích $\langle \vec{x} \cdot M_i^* \rangle = 0$, do đó $\mathcal{S}$ không cần thiết phải biết giá trị $g^{a^{q+1}t}$ để tính $h_j^{s_u}$, tất cả các giá trị khác $\mathcal{S}$ đã biết từ giả thuyết bài toán khó.

$\mathcal{S}$ tiếp theo đặt g^{s_u} và $\{h_j^{s_u}\}_{j \in \mathcal{B}_u}$ như tds_0 và $\{\text{tds}_i\}_{j \in \mathcal{B}_u}$, một cách tương ứng, để tính $\{\text{tds}_{0,j}, \text{tds}_{1,j}, \{\text{tds}_{2,j,\ell}\}_{\ell \in \mathcal{B}_u}\}_{j \in [k]}$;

$\mathcal{S}$ xem xét hai trường hợp:

1. Nếu $\tilde{\omega}_{i_j}$ thỏa mãn $\mathcal{KF}_0^*$ hay $\mathcal{KF}_1^*$, có nghĩa là tồn tại ít nhất một bộ ba (i_j, b, b') sao cho $\tilde{\omega}_{i_j} = kf_{b,b'}^*$; $\mathcal{S}$ đầu tiên kiểm tra xem $\tilde{\omega}_{i_j}$ đã được truy vấn trước đây hay chưa. Nếu chưa, $\mathcal{S}$ chọn $\mathcal{Y}_{i_j} \xleftarrow{\$} \mathbb{Z}_p$ và thêm bộ ba $(\tilde{\omega}_{i_j}, g^{-a} g^{\mathcal{Y}_{i_j}}, \mathcal{Y}_{i_j})$ vào trong $\mathcal{L}$. Trong cả hai trường hợp $\mathcal{S}$ đều biết $\mathcal{Y}_{i_j}$từ $\mathcal{L}$, và $\mathcal{H}(\tilde{\omega}_{i_j}) = g^{-a} g^{\mathcal{Y}_{i_j}}$. Tiếp theo, $\mathcal{S}$ chọn $r_j \xleftarrow{\$} \mathbb{Z}_p$và tính:

$$\text{tds}_{0,j} = g^{\alpha} g^{a s_u} g^{a r_j} (g^{-a^q})^{\mathcal{Y}_{i_j}} = g^{\alpha} g^{a s_u} g^{a r_j} (g^{\alpha} \mathcal{H}(\tilde{\omega}_{i_j}))^{\gamma}$$

Chú ý rằng: $\gamma = -a^q$. Với giá trị r_j đã biết, $\mathcal{S}$ dễ dàng tính $\text{tds}_{1,j}, \{\text{tds}_{2,j,\ell}\}_{\ell \in \mathcal{B}_u}$

2. Nếu $\tilde{\omega}_{i_j}$ không thỏa mãn $\mathcal{KF}_0^*$ hay $\mathcal{KF}_1^*$

$\mathcal{S}$ kiểm tra xem $\tilde{\omega}_{i_j}$ đã được truy vấn trước đây hay chưa. Nếu chưa, $\mathcal{S}$ chọn $\mathcal{Y}_{i_j} \xleftarrow{\$} \mathbb{Z}_p$và thêm bộ ba $(\tilde{\omega}_{i_j}, g^{\mathcal{Y}_{i_j}}, \mathcal{Y}_{i_j})$ vào trong $\mathcal{L}$. Trong cả hai trường hợp, $\mathcal{S}$ đều biết $\mathcal{Y}_{i_j}$ từ $\mathcal{L}$, và $\mathcal{H}(\tilde{\omega}_{i_j}) = g^{\mathcal{Y}_{i_j}}$. Tiếp theo, $\mathcal{S}$ chọn $\zeta_j \xleftarrow{\$} \mathbb{Z}_p$ và đặt giá trị r_j như sau:

$$r_j = \zeta_j + x_1 a^q + x_2 a^{q-1} + \cdots + x_{n^*} a^{q-n^*+1}$$

sau đó, tương tự tính $g^{\alpha}, g^{a r_j}, g^{r_j}, \{\tilde{h}_{\ell}^{r_j}\}_{\ell \in \mathcal{B}_u}$ như ở trên (chú ý rằng r_j đóng vai trò như s_u). $\mathcal{S}$sau đó đặt $g^{-r_j}, \{\tilde{h}_{\ell}^{-r_j}\}_{\ell \in \mathcal{B}_u}$ như $\text{tds}_{1,j}, \{\text{tds}_{2,j,\ell}\}_{\ell \in \mathcal{B}_u}$ một cách tương ứng, và tính:

$$\text{tds}_{0,j} = g^{\alpha} g^{as_u}\, g^{-\alpha}\, g^{-ar_j} g^{\alpha\prime}\, (g^{-a^q})^{y_{i_j}} = g^{\alpha} g^{as_u}\, g^{-ar_j}\, (g^{a}\ \mathcal{H}(\widetilde{\omega}_{i_j}))^{\lambda}$$

Chú ý rằng: $g^{\alpha} = g^{\alpha'}\ g^{a^{q+1}}$ và $(g^{a}\ \mathcal{H}(\widetilde{\omega}_{i_j}))^{\lambda} = g^{-a^{q+1}}\ (g^{-a^q})^{y_{i_j}}$

Cuối cùng, trả về cho tds.

• **Đối với truy vấn thứ tư:** Thấy rằng, thành phần g^{α} chỉ xuất hiện trong $\text{tds}_{0,j}$, do đó $\mathcal{S}$ chỉ đơn giản là chọn $s_u, \{r_j\}_{j\in[k]} \overset{\$}{\leftarrow} \mathbb{Z}_p$ và tính tds ngoại trừ $\text{tds}_{0,j}$

• **Đối với truy vấn thứ năm:** Thấy rằng, thành phần g^{α} chỉ xuất hiện trong d_{u_0}, do đó $\mathcal{S}$ chỉ đơn giản là chọn $s_u, \overset{\$}{\leftarrow} \mathbb{Z}_p$ và tính d_u ngoại trừ d_{u_0}

Giai đoạn thách thức: $\mathcal{S}$ chọn một bít ngẫu nhiên b, tính:

$$(C_0^*\ , C_1^*, \ldots, C_m^*) = \left(g^s, g^{s(a+at)}\ g^{\sum_{i\in I_1} sz_{\rho^*(i)}}, \ldots, g^{s(a+at)}\ g^{\sum_{i\in I_m} sz_{\rho^*(i)}}\right)$$

$$= \left(g^s\ , (g^a \prod_{i\in\beta_1^*} h_i)^{\ s}, \ldots, (g^a \prod_{i\in\beta_m^*} h_i)^{\ s}\right)$$

$$\left(\tilde{C}_1^*, \ldots, \tilde{C}_m^*\right) = \left(g^{s(a+at)}\ g^{\sum_{i\in I_1} s\tilde{z}_{\rho^*(i)}}, \ldots, g^{s(a+at)}\ g^{\sum_{i\in I_m} s\tilde{z}_{\rho^*(i)}}\right)$$

$$= \left((g^a \prod_{i\in\beta_1^*} \tilde{h}_i)^{\ s}, \ldots, (g^a \prod_{i\in\beta_m^*} \tilde{h}_i)^{\ s}\right)$$

Để tính $\{K_i^*\}_{i\in[m\prime]}$, $\mathcal{S}$ kiểm tra xem $kf_{b,i}^*$ đã được truy vấn trước đây hay chưa. Nếu chưa, $\mathcal{S}$ chọn $\mathcal{Y}_i \overset{\$}{\leftarrow} \mathbb{Z}_p$ và thêm bộ ba $(kf_{b,i}^*\ , g^{-a}\ g^{y_i}, \mathcal{Y}_i\ )$vào trong $\mathcal{L}$. Ngược lại, $\mathcal{S}$ tìm $\left(kf_{b,i}^*, g^{-a}\ g^{y_i}, \mathcal{Y}_i\right)$ từ $\mathcal{L}$.

Trong cả hai cách $\mathcal{S}$ đều biết $\mathcal{Y}_i$ từ $\mathcal{L}$, và $\mathcal{H}\left(kf_{b,i}^*\ \right) = g^{-a}\ g^{y_i}$; S tính:

$$X_i^* = T.e(g^s, g^{\alpha'}) \cdot e(g^s, (g^{\lambda})^{\ y_i})$$

$$= T \cdot e\left(g^s\ , g^{\alpha'}\right).e(g^s, (g^a\ g^{-a} g^{y_i})^{\lambda})$$

$$= T \cdot\ e(g^s\ , g^{\alpha'}) \cdot\ e(g^{\lambda}, g^a \mathcal{H}(kf_{b,i}^*))^s$$

sau đó tính: $K_i^*\ = \widetilde{\mathcal{H}}(X_i^*\ , kf_{b,i}^*)$.

Cuối cùng, $\mathcal{S}$ cho đầu ra $ct'^* = \left(C_0^*\ , \{C_i^*\ \}_{i\in[m]}\}, \{\tilde{C}_i^*\}_{i\in[m]}\}\ , \{K_i^*\}_{\ i\in[m\prime]}\right)$.

Chú ý rằng, nếu $T = e(g,g)^{a^{q+1}s}$ thì ct'^* là bản mã hợp lệ.thì ct'^* là bản mã hợp lệ.

Giai đoạn truy vấn thứ hai: Tương tự như giai đoạn truy vấn thứ nhất.

Giai đoạn dự đoán: $\mathcal{A}$ gửi kết quả dự đoán là bít b' cho $\mathcal{S}, \mathcal{S}$ cho đầu ra là bít 0 tương ứng với việc dự đoán rằng $T = e(g,g)^{a^{q+1}s}$ nếu $' = b$; ngược lại, $\mathcal{S}$ cho đầu ra là bít 0 tương ứng với việc dự đoán rằng T là phần tử ngẫu nhiên trong $\mathbb{G}_T$. Khi $T = e(g,g)^{a^{q+1}s}$ thì $\mathcal{S}$ tạo ra sự mô phỏng hoàn hảo cho $\mathcal{A}$ do đó:

$$\Pr\left[\mathcal{S}(\vec{Y}, T = e(g,g)^{a^{q+1}s}) = 0\right] = \frac{1}{2} + \text{Adv}_{\mathcal{A}}^{IS}$$

Khi T là phần tử ngẫu nhiên thì giá trị $\{K_i^*\}_{i\in[m']}$ hoàn toàn ngẫu nhiên dưới tầm nhìn của $\mathcal{A}$, do đó:

$$Pr[S(\vec{Y}, T = R) = 0] = \frac{1}{2}$$

Vì thế, nếu $\mathcal{A}$ có thể phá được hệ mã thì $\mathcal{S}$ có thể giải được bài toán khó BDHE với cùng xác xuất thành công. An toàn trước kẻ tấn công từ bên ngoài hệ thống.

Định lý 3.7: Giả sử rằng $\beta^* = \beta_1^* \vee \cdots \vee \beta_{\mathrm{m}}^*$ là chính sách bản mã thách thức, và từ β^* xây dựng ma trận L^* có kích thước $\ell^* \times \mathrm{n}^*$ và ánh xạ ρ^*. Nếu ma trận này thỏa mãn $\ell^* \leq \mathrm{q}, \mathrm{n}^* \leq \mathrm{q}$, hệ trên đạt an toàn từ khóa trước kẻ tấn công bên ngoài dưới giả thuyết bài toán BDHE là bài toán khó.

Chứng minh: Phương pháp chứng minh là phương pháp phản chứng, tức là giả sử tồn tại một kẻ tấn công $\mathcal{A}$ phá vỡ an toàn từ khóa của hệ mã trên, thì cũng sẽ tồn tại một kẻ tấn công $\mathcal{S}$ giải được bài toán khó BDHE. Với phương pháp như vậy, ban đầu $\mathcal{S}$ được cho trước các giá trị như trong định nghĩa bài toán khó BDHE, sau đó $\mathcal{S}$ cần phân biệt $T = e(g,g)^{a^{q+1}s} \in \mathbb{G}_T$ hay T chỉ là một giá trị ngẫu nhiên trong tập $\mathbb{G}_T$.

Bước tính các tham số cho hệ thống hoàn toàn tương tự như ở phần chứng minh trên, ngoại trừ việc $\mathcal{S}$ chọn $\gtrless \xleftarrow{\$} \mathbb{Z}_p$ và tính $g^{\gtrless}$.

Giai đoạn truy vấn 1: Trong giai đoạn này $\mathcal{S}$ cần trả lời bốn loại truy vấn sau:

1. Truy vấn từ hàm băm.

2. Truy vấn khóa bí mật tương ứng với tập thuộc tính $\mathcal{B}_u$, trong đó tập thuộc tính $\mathcal{B}_u$ không thỏa mãn chính sách bản mã β^*

3. Truy vấn cửa sập một phần $(\mathcal{B}_u, W_i)$ với mọi tập $(\mathcal{B}_u, W_i)$

4. Truy vấn một phần khóa bí mật $\mathcal{B}_u$ với bất kỳ tập $\mathcal{B}_u$. Lý do là $\mathcal{S}$ chỉ giữ mình d_{u_0} bí mật.

• Đối với truy vấn từ hàm băm: Hoàn toàn tương tự như chứng minh trên.

• Đối với loại truy vấn thứ hai: $\mathcal{S}$ trước tiên tìm véc tơ $\overrightarrow{x} = (x_1, \ldots, x_{n^*}) \in \mathbb{Z}_p^{n^*}$ sao cho $x_1 = -1$ và với mọi *i* trong đó $\rho^*(i) \in \mathcal{B}(u)$, giá trị tích $\langle \overrightarrow{x} \cdot M_i^* \rangle = 0$. $\mathcal{S}$ tiếp tục chọn $\zeta \overset{\$}{\leftarrow} \mathbb{Z}_p$ và đặt s_u như sau:

$$s_u = \zeta + x_1 a^{\,q} + x_2\, a^{q-1} + \cdots + x_{n^*}\, a^{q-n^*+1}$$

$\mathcal{S}$ tính:

$$d_{u_0} = g^{\alpha\prime}\ g^{a\zeta} \prod_{i=2,\ldots,n^*} \left(g^{a^{q+1-i}}\right)^{x_i} = g^{\alpha} \cdot g^{\,a \cdot s_u}$$

$$d'_{u_0} = g^{\zeta} \prod_{i=1,\ldots,n^*} \left(g^{a^{q+1-i}}\right)^{x_i} = g^{s_u}$$

Với mỗi $j \in \mathcal{B}(u)$ sao cho không có $i \in [\ell^*]$ thỏa mãn $\rho^*(i) = j$. $\mathcal{S}$ biết giá trị z_j nên tính:

$$h_j^{s_u} = (g^{s_u})^{z_j}$$

Với mỗi $j \in \mathcal{B}(u)$ sao cho có một $i \in [\ell^*]$ thỏa mãn $\rho^*(i) = j$. $\mathcal{S}$ tính:

$$h_j^{s_u} = (g^{s_u})^{z_j} . g^{\left(\zeta + x_1 a^{\,q} + x_2\, a^{q-1} + \cdots + x_{n^*}\, a^{q-n^*+1}\right)\omega_i \sum_{k \in [n^*]} M^*_{i,k} t a^k}$$

Vì $\langle \vec{x} . M_i^* \rangle = 0$ nên $\mathcal{S}$ không cần thiết phải biết giá trị $g^{a^{q+1}t}$ để tính $h_j^{s_u}$, tất cả các giá trị khác $\mathcal{S}$ đã biết từ giả thuyết bài toán khó. Cuối cùng, do $\mathcal{S}$ biết $\asymp$ nên $\mathcal{S}$ trả về du cho $\mathcal{A}$.

• Đối với loại truy vấn thứ ba và thứ tư, hoàn toàn tương tự như chứng minh ở trên. Giai đoạn thách thức: $\mathcal{S}$ chọn một bít ngẫu nhiên b, tính:

$$(C_0^*, C_1^* \ldots C_m^*) = \left(g^s g^{s(a+at)} g^{\sum_{i \epsilon I_1} s z_{p^*(i)}}, \ldots, g^{s(a+at)} g^{\sum_{i \epsilon I_m} s z_{p^*(i)}}\right)$$

$$= \left(g^s, (g^a \prod_{i \in \beta_1^*} h_i)^s, \ldots, (g^a \prod_{i \in \beta_m^*} h_i)^s\right)$$

$$\left(\tilde{C}_1^* \ldots \tilde{C}_m^*\right) = g^{s(a+at)} g^{\sum_{i \epsilon I_1} s \tilde{z}_{p^*(i)}}, \ldots, g^{s(a+at)} g^{\sum_{i \epsilon I_m} s \tilde{z}_{p^*(i)}}$$

$$= \left((g^a \prod_{i \in \beta_1^*} \tilde{h}_i)^s, \dots, (g^a \prod_{i \in \beta_m^*} \tilde{h}_i)^s \right)$$

Để tính $\{K_i^*\}_{i \in [m]'}$ $\mathcal{S}$ tính:

$$X_i^* = T.e(g^s g^{\alpha'}).e\left(g^s g^a \mathcal{H}(kf_{b,i}^*)\right)^{\lambda} = T.e(g^s g^{\alpha'}).e\left(g^{\lambda} g^a \mathcal{H}(kf_{b,i}^*)\right)^s$$

Sau đó tính $\{K_i^*\} = \widetilde{\mathcal{H}}(X_i^* kf_{b,i}^*)$

Cuối cùng, $\mathcal{S}$ cho đầu ra $ct'^* = (C_0^*, \{C_i^*\}_{i \in [m]}, \{\tilde{C}_i^*\}_{i \in [m]}, \{K_i^*\}_{i \in [m']})$

Lưu ý, nếu T $= e(g,g)^{a^{q+1}s}$ thì ct'^* là bản mã hợp lệ

Giai đoạn truy vấn thứ 2: Hoàn toàn như giai đoạn truy vấn thứ nhất

Giai đoạn dự đoán: $\mathcal{A}$ gửi kết quả dự đoán là bít b' cho $\mathcal{S}$, $\mathcal{S}$ cho đầu ra là bít 0 tương ứng với việc dự đoán rằng $T = e(g,g)^{a^{q+1}s}$ nếu $b' = b$; ngược lại, $\mathcal{S}$ cho đầu ra là bít 0 tương ứng với việc dự đoán rằng T là phần tử ngẫu nhiên trong $\mathbb{G}_T$

Khi $T = e(g,g)^{a^{q+1}s}$ thì $\mathcal{S}$ tạo ra sự mô phỏng hoàn hảo cho $\mathcal{A}$, do đó:

$$\Pr\left[S\left((\overrightarrow{Y}, T = e(g,g)^{a^{q+1}s}\right) = 0\right] = \frac{1}{2} + \text{Adv}_{\mathcal{A}}^{os}$$

Khi T là phần tử ngẫu nhiên, thì giá trị $\{K_i^*\}_{i \in [m']}$ hoàn toàn ngẫu nhiên dưới tầm nhìn của $\mathcal{A}$, do đó:

$$\Pr\left[S\left((\overrightarrow{Y}, T = R\right) = 0\right] = \frac{1}{2}$$

Vậy nếu $\mathcal{A}$ có thể phá được hệ mã thì $\mathcal{S}$ có thể giải được bài toán khó BDHE với cùng xác suất thành công.

3.5. Kết luận chương 3

Toàn bộ nội dung chương 3, đã trình bày về hệ mã hóa dựa trên thuộc tính bao gồm: Định nghĩa tổng quát, định nghĩa mô hình an toàn chuẩn, trình bày một số hệ mã hóa dựa trên thuộc tính hiện nay bao gồm các hệ RouselakisWaters13 và Agrawal-Chase17. Đối với mã hóa dựa trên thuộc tính, tham số quan trọng nhất đó là độ dài của bản mã.

Tại chương 3, luận án đã trình bày hai đóng góp trong hệ mã hóa dựa trên thuộc tính đề xuất, đó là:

Thứ nhất, độ dài bản mã là hằng số, cụ thể chỉ là hai phần tử (tối ưu trong các hệ mã đang có hiện nay) và cũng trình bày chứng minh ý tưởng đề xuất đạt an toàn.

Thứ hai, trình bày về lược đồ mã hóa dựa trên thuộc tính đề xuất, có hỗ trợ tính chất tìm kiếm trên dữ liệu đã được mã hóa, đồng thời giống như hệ thứ nhất, NCS cũng trình bày chứng minh chi tiết rằng hệ đề xuất thứ hai đạt an toàn.

KẾT LUẬN VÀ KIẾN NGHỊ

Trong luận án Nghiên cứu sinh trình bày ba đóng góp chính, trong đó hai đóng góp đầu nằm trong công trình số 4, đóng góp thứ ba nằm trong công trình số 2. Các công trình số 3 và số 1 là nghiên cứu kèm theo trong quá trình làm luận án.

Các kết quả đạt được và đóng góp của luận án:

1. Đóng góp thứ nhất: Đề xuất một lược đồ mã hóa quảng bá đa kênh dựa trên hệ Delerablee [25] có độ hiệu quả và an toàn tương tự như hệ [47, 15] nhưng ở dạng mã hóa công khai, không còn ở dạng bí mật. Được công bố tại công trình số 4.

2. Đóng góp thứ hai: Đề xuất lược đồ CP-ABE mới, có khóa bí mật ngắn hơn các hệ CP-ABE khác. Các hệ khác có cùng tính chất, độ dài bản mã là hằng số. Điểm yếu của đề xuất so với các hệ mã này là, có mức độ an toàn yếu hơn các hệ khác cso cùng tính chất. Nội dung đề xuất được công bố tại công trình số 4.

3. Đóng góp thứ 3: Đề xuất mới, dựa trên hệ ABE hiện có [42], xây dựng một lược đồ ABE mới hỗ trợ tìm kiếm trên dữ liệu đã được mã hóa; được công bố tại công trình số 2.

Các hướng nghiên cứu tiếp theo dự kiến như sau:

1. Xây dựng MCBE có độ dài khóa bí mật ngắn hơn các hệ hiện có mà vẫn giữ được độ dài bản mã là hằng số.

2. Xây dựng phi tập trung hóa MCBE, hiện nay vẫn chưa tồn tại hệ phi tập trung hóa MCBE nào.

3. Xây dựng CP-ABE có tính chất là cả đội dài bản mã và độ dài khóa bí mật đều là hằng số. Lưu ý rằng, các hệ mã hóa quảng bá đã có tính chất này nên tồn tại hệ CP-ABE như vậy là khả thi.

CÁC CÔNG TRÌNH CÔNG BỐ TRONG LUẬN ÁN

1.Trinh Viet Cuong, Trinh Van Anh, Do Thi Thu Hien, Do Thi Thanh Hien, Tran Cam Van, Tran Vinh Duc. Anonymous Key Leakage Attack on Attribute-based Encryption. *Kỷ yếu hội thảo quốc gia @ năm 2018.*

2. Van Anh Trinh, Viet Cuong Trinh. A Ciphertext-policy Attribute-based Searchable Encryption Scheme in Non-interactive Model. *Journal of Computer Science and Cybernetics, Volume 35, Pages 233-249, 2019,*

3. Van Anh Trinh, Viet Cuong Trinh. One-Verifier Signature Scheme and Its Applications. *In Proceeding of The 10th International Symposium on Information and Communication Technology - SoICT 2019, December 4 – 6, 2019, Ha Noi - Ha Long Bay, Viet Nam.*

4. Minh Ha Le, Vinh Duc Tran, Van Anh Trinh, Viet Cuong Trinh. Compacting Ciphertext in Multi-Channel Broadcast Encryption and Attribute-Based Encryption. *Theoretical Computer Science, Volume 804, 12 January 2020, Pages 219-235. (ISI)*

TÀI LIỆU THAM KHẢO

1. S. Agrawal, S. Bhattacherjee, D. H. Phan, D. Stehle, and S. Yamada. Efficient public trace and revoke from standard assumptions *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, CCS 2017, Dallas, TX, USA, October 30 - November 03,2017. ACM 2017, ISBN 978-1-4503-4946-8.

2. K. Acharya and R. Dutta. Adaptively secure broadcast encryption with dealership. In ICISC 16: *19th International Conference on Information Security and Cryptology, Lecture Notes in Computer Science*, pages 161—177 Springer, Heidelberg, Germany, 2017.

3. K. Acharya and R. Dutta. Constructions of Secure Multi-Channel Broadcast Encryption Schemes in Public Key Framework. *CANS 2018: International Conference on Cryptology and Network Security*, Lecture Notes in Computer Science 11124, Springer 2018, ISBN 978-3-030-00434-7, Naples, Italy.

4. Shashank Agrawal and Melissa Chase. A study of pair encodings: Predicate encryption in prime order groups. In Eyal Kushilevitz and Tal Malkin, editors, *TCC 2016-A: 13th Theory of Cryptography Conference, Part II,* volume 9563 of *Lecture Notes in Computer Science*, pages 259–288, Tel Aviv, Israel, January 10–13, 2016. Springer, Heidelberg, Germany.

5. S. Agrawal and M. Chase. FAME: Fast Attribute-based Message Encryption. In Bhavani M. Thuraisingham, David Evans, Tal Malkin, Dongyan Xu, editors, *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017. ACM 2017, ISBN 978-1-4503-4946-8.*

6. Shashank Agrawal and Melissa Chase. Simplifying design and analysis of complex predicate encryption schemes. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *Advances in Cryptology* – EUROCRYPT 2017, Part I, volume 10210 of *Lecture Notes in Computer Science*, pages 627–656, Paris, France, May 8–12, 2017. Springer, Heidelberg, Germany.

7. Nuttapong Attrapadung. Dual system encryption via doubly selective security: Framework, fully secure functional encryption for regular languages, and more. In Phong Q. Nguyen and Elisabeth Oswald, editors, *Advances in Cryptology – EUROCRYPT 2014*, volume 8441 of Lecture *Notes in Computer Science*, pages 557–577, Copenhagen, Denmark, May 11–15, 2014. Springer, Heidelberg, Germany.

8. Nuttapong Attrapadung, Goichiro Hanaoka, and Shota Yamada. Conversions among several classes of predicate encryption and applications to ABE with various compactness tradeoffs. In Tetsu Iwata and Jung Hee Cheon, editors, *Advances in Cryptology – ASIACRYPT 2015, Part I*, volume 9452 of *Lecture Notes in Computer Science*, pages 575–601, Auckland, New Zealand, November 30 – December 3, 2015. Springer, Heidelberg, Germany.

9. D. Boneh, C. Gentry, and B. Waters. Collusion resistant broadcast encryption with short ciphertexts and private keys. In V. Shoup, editor, *Advances in Cryptology – CRYPTO 2005*, volume 3621 of *Lecture Notes in Computer Science*, pages 258–275, Santa Barbara, CA, USA, Aug. 14–18, 2005. Springer, Heidelberg, Germany.

10. D. Boneh, B. Waters, and M. Zhandry. Low overhead broadcast encryption from multilinear maps. In J. A. Garay and R. Gennaro, editors, *Advances in Cryptology CRYPTO 2014, Part I*, volume 8616 of *Lecture Notes in Computer Science*, pages 206–223, Santa Barbara, CA, USA, Aug. 17–21, 2014. Springer, Heidelberg, Germany

11. D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano. Public key encryption with keyword search. In C. Cachin and J. Camenisch, editors, *Advances in Cryptology – EUROCRYPT 2004*

12. D. Boneh, A. Sahai, and B. Waters. Fully collusion resistant traitor tracing with short ciphertexts and private keys. In S. Vaudenay, editor, *Advances in Cryptology – EUROCRYPT 2006*, volume 4004 of *Lecture Notes in Computer Science*, pages 573–592, St. Petersburg, Russia, May 28 – June 1, 2006. Springer, Heidelberg, Germany.

13. Libert B., Paterson K.G., Quaglia E.A. Anonymous Broadcast Encryption: Adaptive Security and Efficient Constructions in the Standard Model. In: Fischlin

M., Buchmann J., Manulis M. (eds) *Public Key Cryptography – PKC 2012. PKC 2012. Lecture Notes in Computer Science*, vol 7293. Springer, Berlin, Heidelberg

14. D. Cash, S. Jarecki, C. S. Jutla, H. Krawczyk, M.-C. Rosu, and M. Steiner. Highlyscalable searchable symmetric encryption with support for Boolean queries. In R. Canetti and J. A. Garay, editors, *Advances in Cryptology – CRYPTO 2013, Part I*

15. S. Canard, D. H. P. D. Pointcheval, and V. C. Trinh. A new technique for compacting ciphertext in multi-channel broadcast encryption and attribute-based encryption. *Theoretical Computer Science,* Volume 723, 2018, Pages 51-72, ISSN 0304-3975, 2018.

16. Sébastien Canard, Duong Hieu Phan, and Viet Cuong Trinh. An Attribute-based Broadcast Encryption Scheme For Lightweight Devices. IET Information Security: Volume 12, Issue 1, January 2018, p. 52-59 DOI: 10.1049/iet-ifs.2017.0157, Print ISSN 1751-8709, Online ISSN 1751-8717.

17. S. Canard, D. H. Phan, and V. C. Trinh. A new technique for compacting secret key in attribute-based broadcast encryption. In CANS 16: *15th International Conference on Cryptology and Network Security, Lecture Notes in Computer Science*, pages 594–603. Springer, Heidelberg, Germany, 2016.

18. S. Canard and V. C. Trinh. Constant-size ciphertext attribute-based encryption from multi-channel broadcast encryption. In: Ray I., Gaur M., Conti M., Sanghi D., Ka makoti V. (eds) Information Systems Security. ICISS 2016. Lecture Notes in Computer Science, vol 10063. Springer. https://doi.org/10.1007/978-3-319-49806-5-10

19. Chase, M., Chow, S.S. Improving privacy and security in multi-authority attributebased encryption. *Proceedings of the 16th ACM Conference on Computer and Communications Security* - CCS ’09. pp. 121-130. ACM, New York, NY, USA (2009)

20. J. Chen, R. Gay, and H. Wee. Improved dual system abe in prime-order groups via predicate encodings. In E. Oswald and M. Fischlin, editors, *Proceedings of EUROCRYPT*, LNCS 9057, pages 595–624. Springer, 2015.

21. Cheng Chen, Jie Chen, Hoon Wei Lim, Zhenfeng Zhang, Dengguo Feng, San Ling, and Huaxiong Wang. Fully secure attribute-based systems with short ciphertexts/signatures and threshold access structures. In Ed Dawson, editor, *Topics in Cryptology – CT-RSA 2013*, volume 7779 of *Lecture Notes in Computer Science*, pages 50–67, San Francisco, CA, USA, February 25 – March 1, 2013. Springer, Heidelberg, Germany.

22. Cheng Chen, Zhenfeng Zhang, and Dengguo Feng. Efficient ciphertext policy attributebased encryption with constant-size ciphertext and constant computation-cost. In Xavier Boyen and Xiaofeng Chen, editors, ProvSec 2011: *5th International Conference on Provable Security*, volume 6980 of *Lecture Notes in Computer Science*, pages 84– 101, Xi'an, China, October 16–18, 2011. Springer, Heidelberg, Germany.

23. H. Cui, R. Deng, J. Liu, and Y. Li. Attribute-based encryption with expressive and authorized keyword search. ACISP, May 2017, LNCS 10342, DOI: 10.1007/978-3-319- 60055-0-6, 2017.

24. H. Cui, Z. Wan, R. Deng, G. Wang, and Y. Li. Efficient and expressive keyword search over encrypted data in the cloud. IEEE Trans. Dependable Secure Comput, Issue: 99, 2016.

25. C. Delerablée. Identity-based broadcast encryption with constant size ciphertexts and private keys. In K. Kurosawa, editor, *Advances in Cryptology – ASIACRYPT 2007*, volume 4833 of *Lecture Notes in Computer Science*, pages 200–215, Kuching, Malaysia, Dec. 2–6, 2007. Springer, Heidelberg, Germany.

26. Y. Dodis and N. Fazio. Public key trace and revoke scheme secure against adaptive chosen ciphertext attack. In Y. Desmedt, editor, *PKC 2003: 6th International Workshop on Theory and Practice in Public Key Cryptography, volume*

2567 of Lecture Notes in Computer Science, pages 100–115, Miami, USA, Jan. 6–8, 2003. Springer, Heidelberg, Germany.

27. K. Emura, A. Miyaji, A. Nomura, K. Omote, and M. Soshi. A ciphertext-policy attribute-based encryption scheme with constant ciphertext length schemes. In Feng Bao, Hui Li, and Guilin Wang, editors, *ISPEC 2009: 5th International Conference on Information Security Practice and Experience, volume 5451 of Lecture Notes in Computer Science*, pages 13–23, Xi'an, China, April 13–15 2009. Springer, Heidelberg, Germany.

28. A. Fiat and M. Naor. Broadcast encryption. In D. R. Stinson, editor, *Advances in Cryptology – CRYPTO'93*, volume 773 of *Lecture Notes in Computer Science*, pages 480–491, Santa Barbara, CA, USA, Aug. 22–26, 1994. Springer, Heidelberg, Germany.

29. R. Gay, L. Kowalczyk, and H. Wee. Tight adaptively secure broadcast encryption with short ciphertexts and keys In *Proceeding of International Conference on Security and Cryptography for Networks*, SCN 2018.

30. C. Gentry and B. Waters. Adaptive security in broadcast encryption systems (with short ciphertexts). In A. Joux, editor, *Advances in Cryptology – EUROCRYPT 2009*, volume 5479 of *Lecture Notes in Computer Science*, pages 171–188, Cologne, Germany, Apr. 26–30, 2009. Springer, Heidelberg, Germany

31. Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for fine-grained access control of encrypted data. In Ari Juels, Rebecca N. Wright, and Sabrina De Capitani di Vimercati, editors, *ACM CCS 06: 13th Conference on Computer and Communications Security*, pages 89–98, Alexandria, Virginia, USA, October 30 – November 3, 2006. ACM Press. Available as Cryptology ePrint Archive Report 2006/309.

32. Sergey Gorbunov, Vinod Vaikuntanathan, and Hoeteck Wee. Attribute-Based Encryption for Circuits. *J. ACM 62, 6, Article 45 (December 2015), 33 pages. DOI: https://doi.org/10.1145/2824233*

33. Sanjam Garg, Craig Gentry, Shai Halevi, Amit Sahai, and Brent Waters. Attributebased encryption for circuits from multilinear maps. In Ran Canetti and Juan A. Garay, editors, *Advances in Cryptology – CRYPTO 2013, Part II*, volume 8043 of *Lecture Notes in Computer Science*, pages 479–499, Santa Barbara, CA, USA, August 18–22, 2013. Springer, Heidelberg, Germany

34. Jinguang Han, Ye Yang, Joseph K. Liu, Jiguo Li,Kaitai Liang,Jian Shen. Expressive attribute-based keyword search with constant-size ciphertext. *In Soft Computing journal, August 2018*, Volume 22, Issue 15, pp 5163–5177.

35. J. Herranz, F. Laguillaumie, and C. Ràfols. Constant size ciphertexts in threshold attribute-based encryption. In P. Q. Nguyen and D. Pointcheval, editors, *PKC 2010: 13th International Conference on Theory and Practice of Public Key Cryptography, volume 6056 of Lecture Notes in Computer Science*, pages 19–34, Paris, France, May 26– 28, 2010. Springer, Heidelberg, Germany.

36. Xiaoming Hu, Wenan Tan, Huajie Xu, Jian Wang, and Chuang Ma. Strong Designated Verifier Signature Schemes with Undeniable Property and Their Applications. *Security and Communication Networks Volume 2017*, Article ID 7921782, 9 pages, https://doi.org/10.1155/2017/7921782. 2017

37. A. Kiayias, O. Oksuz, A. Russell, Q. Tang, and B. Wang. Efficient encrypted keyword search for multi-user data sharing. Lecture Notes in Computer Science, ESORICS 2016. Springer, 2016

38. J. Lai, X. Zhou, R. H. Deng, Y. Li, and K. Chen. Expressive search on encrypted data. In K. Chen, Q. Xie, W. Qiu, N. Li, and W.-G. Tzeng, editors, ASIACCS 13: 8th *ACMSymposium on Information, Computer and Communications Security*, pages 243–252, Hangzhou, China, May 8–10, 2013. ACM Press.

39. Z. Liu and D. S. Wong. Practical Attribute-Based Encryption: Traitor Tracing, Revocation and Large Universe *The Computer Journal, vol. 59, no. 7, pp. 983-1004* July 2016. doi: 10.1093/comjnl/bxv101

40. B. Lynn. The Stanford Pairing Based Crypto Library. Available from http://crypto. stanford.edu/pbc

41. Chuangui Ma, Aijun Ge, and Jie Zhang. Fully Secure Decentralized Ciphertext-Policy Attribute-Based Encryption in Standard Model. *Proceedings of Information Security and Cryptology: Inscrypt January* 2019. DOI: 10.1007/978-3-030-14234-6-23, Springer Berlin Heidelberg, Berlin, Heidelberg (2019)

42. Q. M. Malluhi, A. Shikfa, and V. C. Trinh. A ciphertext-policy attribute-based encryption scheme with optimized ciphertext size and fast decryption. In *ASIACCS 17: 12th ACM Symposium on Information, Computer and Communications Security*, pages 230–240. ACM Press, 2017.

43. Q. M. Malluhi, A. Shikfa, V. D. Tran, and V. C. Trinh. Decentralized ciphertext-policy attribute-based encryption schemes for lightweight devices. *In Computer Communications Volume 145, September 2019*, Pages 113-125

44. D. Naor, M. Naor, and J. Lotspiech. Revocation and tracing schemes for stateless receivers. In J. Kilian, editor, *Advances in Cryptology – CRYPTO 2001*, volume 2139 of *Lecture Notes in Computer Science*, pages 41–62, Santa Barbara, CA, USA, Aug. 19– 23, 2001. Springer, Heidelberg, Germany

45. Jianting Ning, Zhenfu Cao, Xiaolei Dong, Kaitai Liang, Hui Ma, Lifei Wei. Auditable σTime Outsourced Attribute-Based Encryption for Access Control in Cloud Computing. *In IEEE Transactions on Information Forensics and Security*, Volume: 13 , Issue: 1 , Jan. 2018.

46. D. H. Phan, D. Pointcheval, S. F. Shahandashti, and M. Strefler. Adaptive CCA broadcast encryption with constant-size secret keys and ciphertexts. In W. Susilo, Y. Mu, and J. Seberry, editors, ACISP 12: *17th Australasian Conference on Information Security and Privacy, volume 7372 of Lecture Notes in Computer Science*, pages 308–321, Wollongong, NSW, Australia, July 9–11, 2012. Springer, Heidelberg, Germany

47. D. H. Phan, D. Pointcheval, and V. C. Trinh. Multi-channel broadcast encryption. In K. Chen, Q. Xie, W. Qiu, N. Li, and W.-G. Tzeng, editors, ASIACCS 13: 8th ACM *Symposium on Information, Computer and Communications Security*, pages 277–286, Hangzhou, China, May 8–10, 2013. ACM Press.

48. D. H. Phan, D. Pointcheval, and M. Strefler. Decentralized Dynamic Broadcast Encryption. In J. Lopez and G. Tsudik, editors, *SCN 2012: International Conference on Security and Cryptography for Networks*, volume 7485 of *Lecture Notes in Computer Science*, pages 166-183. Springer, Heidelberg, Germany.

49. D. H. Phan and V. C. Trinh. Identity-based trace and revoke schemes. In X. Boyen and X. Chen, editors, *ProvSec 2011 5th International Conference on Provable Security*, volume 6980 of *LNCS Lecture Notes in Computer Science*, pages 204–221. Springer, Oct. 2011.

50. T. V. X. Phuong, G. Yang, W. Susilo, and X. Chen. Attribute based broadcast encryption with short ciphertext and decryption key. *In Proceedings of ESORICS, LNCS 9327*, pages 252–269. Springer, 2015.

51. Y. Rouselakis and B. Waters. Efficient statically-secure large-universe multi-authority attribute-based encryption. In *FC 2015: 19th International Conference on Financial Cryptography and Data Security, Lecture Notes in Computer Science*, pages 315–332. Springer, Berlin, Germany, 2015.

52. Y. Rouselakis and B. Waters. Practical constructions and new proof methods for large universe attribute-based encryption. In A.-R. Sadeghi, V. D. Gligor, and M. Yung, editors, *ACM CCS 13: 20th Conference on Computer and Communications Security*, pages 463–474, Berlin, Germany, Nov. 4–8, 2013. ACM Press.

53. A. Sahai and B. R. Waters. Fuzzy identity-based encryption. In R. Cramer, editor, *Advances in Cryptology – EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 457–473, Aarhus, Denmark, May 22–26, 2005. Springer, Heidelberg, Germany.

54. A. Shamir. Identity-based cryptosystems and signature schemes. In G. R. Blakley and D. Chaum, editors, *Advances in Cryptology* - CRYPTO’84

55. W. Susilo, R. Chen, F. Guo, G. Yang, Y. Mu, and Y.-W. Chow. Recipient revocable identity-based broadcast encryption: How to revoke some recipients in IBBE without knowledge of the plaintext. *In ASIACCS 16: 11th ACM Symposium on*

Information, Computer and Communications Security, pages 201–210. ACM Press, 2016

56. Jongkil Kim, Seyit Camtepe, W. Susilo, Surya Nepal, Joonsang Baek. Identity-Based Broadcast Encryption with Outsourced Partial Decryption for Hybrid Security Models in Edge Computing. *In ASIACCS 2019: 14th ACM Symposium on Information, Computer and Communications Security*, pages 55–66. ACM Press, 2019.

57. D. X. Song, D. Wagner, and A. Perrig. Practical techniques for searches on encrypted data. IEEE SP Berkeley, California, USA, May 14-17, pages 44–55, 2000.

58. Brent Waters (2009). Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In Shai Halevi, editor, *Advances in Cryptology – CRYPTO 2009*, volume 5677 of *Lecture Notes in Computer Science*, pages 619–636, Santa Barbara, CA, USA, August 16–20, . Springer, Heidelberg, Germany.

59. Y. Wang, J. Wang, S. Sun, J. Liu, W. Susilo, and X. Chen (2017). Towards multi-user searchable encryption supporting boolean query and fast decryption. ProvSec, LNCS 10592,

60. X. W. Zhao and H. Li (2013). Improvement on a multi-channel broadcast encryption scheme. Applied Mechanics and Materials, Vols. 427-429, pp. 2163-2169,

61. Liu, Z., Cao, Z., Huang, Q., Wong, D.S., Yuen, T.H (2011). Fully secure multi-authority ciphertext-policy attribute-based encryption without random oracles.. *Computer Security ESORICS 2011: 16th European Symposium on Research in Computer Security, Leuven, Belgium, September 12-14,2011*, pages 278-297. Springer Berlin Heidelberg, Berlin, Heidelberg.