ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG

DƯƠNG THẾ ĐẠT - 22520209
ĐOÀN MẠNH ĐỨC- 22520264
NGUYỄN CAO CƯỜNG-22520174

BÁO CÁO ĐỒ ÁN MÔN HỌC
ĐỀ TÀI
MÃ HÓA, KIỂM SOÁT TRUY CẬP VÀ TRUY VẤN TRÊN CLOUD DATABASE TRONG GIAO DỊCH BẤT ĐỘNG SẢN

ENCRYPTION, ACCESS CONTROL AND QUERY ON CLOUD DATABASE IN REAL ESTATE TRANSACTIONS

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG

DƯƠNG THẾ ĐẠT - 22520209
ĐOÀN MẠNH ĐỨC- 22520264
NGUYỄN CAO CƯỜNG-22520174

ĐỒ ÁN MÔN HỌC
ĐỀ TÀI
MÃ HÓA, KIỂM SOÁT TRUY CẬP VÀ TRUY VẤN TRÊN CLOUD DATABASE TRONG GIAO DỊCH BẤT ĐỘNG SẢN

ENCRYPTION, ACCESS CONTROL AND QUERY ON CLOUD DATABASE IN REAL ESTATE TRANSACTIONS

MÔN HỌC: MẬT MÃ HỌC

GIẢNG VIÊN LÝ THUYẾT
TS. NGUYỄN NGỌC TỰ
LỚP: NT219.O22.ANTT

LỜI MỞ ĐẦU

Các công ty bất động sản giữ một vai trò quan trọng trong nền kinh tế của chúng ta. Là một trong những ngành công nghiệp phức tạp, bất động sản không chỉ bao gồm việc mua, bán và quản lý các bất động sản, mà còn phải xử lý các giao dịch mỗi ngày. Với sự phát triển không ngừng của công nghệ, đặc biệt là số hóa các dữ liệu, việc bảo vệ các thông tin, dữ liệu là vô cùng cần thiết.

Truy cập bất hợp pháp vào hệ thống của các công ty bất động sản có thể gây ra những hệ quả khôn lường. Sự tồn tại của các lỗ hổng bảo mật trong hệ thống có thể dẫn đến các rủi ro về việc lộ thông tin cá nhân, gian lận, hoặc thậm chí đe dọa tính bảo mật và toàn vẹn của hệ thống. Chính vì vậy, triển khai các biện pháp bảo mật tiên tiến là cần thiết để đảm bảo thông tin và dữ liệu của hệ thống được bảo vệ một cách an toàn và tin cậy.

Trong đồ án này, chúng em nghiên cứu về *"Mã hóa, kiểm soát truy cập và truy vấn trên cloud database trong giao dịch bất động sản"* sử dụng CP-ABE. CP-ABE là hệ mã hóa cho phép chúng ta mã hóa và giải mã dữ liệu dựa trên tập các thuộc tính của người dùng và chính sách truy cập của dữ liệu. Điều này giúp đảm bảo rằng chỉ những người thuộc chính sách truy cập và đáp ứng đủ các yêu cầu thuộc tính mới có thể truy cập vào dữ liệu.

Qua đồ án này, chúng em hy vọng sẽ cung cấp một phương pháp để giải quyết bài toán bảo vệ thông tin trong hệ thống giao dịch bất động sản và phương pháp có thể được ứng dụng để giải quyết các vấn đề thực tiễn, đảm bảo sự an toàn và sự tin cậy của hệ thống trong thời đại số hóa.

LỜI CẢM ƠN

Để hoàn thành đồ án môn học này, chúng em xin tỏ lòng biết ơn sâu sắc đến Thầy TS. Nguyễn Ngọc Tự đã hướng dẫn, cung cấp kiến thức, kinh nghiệm quý báu trong suốt quá trình thực hiện đồ án. Sự tận tâm và sự đồng hành của thầy đã đóng góp không nhỏ vào việc hoàn thành đồ án môn học của chúng em.

Đồng thời, chúng em cũng muốn gửi lời cảm ơn tới các chuyên gia, các nhóm nghiên cứu và các cá nhân có kiến thức chuyên sâu trong lĩnh vực CP-ABE. Sự chia sẻ kiến thức và kinh nghiệm của quý vị đã giúp chúng em có cái nhìn rõ ràng hơn và đưa ra các giải pháp đáng tin cậy trong việc ….

Lời cảm ơn không thể đủ để bày tỏ sự biết ơn chân thành của nhóm chúng em. Đây là đồ án môn học về lĩnh vực mật mã đầu tay của nhóm nên không thể tránh được có nhiều thiếu sót, rất mong được sự góp ý và chỉ dẫn thêm.

Sau cùng, nhóm chúng em xin gửi đến TS. Nguyễn Ngọc Tự, gia đình, bạn bè thân thiết và mọi người lời kính chúc sức khỏe và lời tri ân chân thành nhất.

TP. Hồ Chí Minh, 2024

Trân trọng./.

Dương Thế Đạt, Đoàn Mạnh Đức, Nguyễn Cao Cường

NHẬN XÉT

(Của giảng viên lý thuyết)

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

MỤC LỤC

CHƯƠNG 1: PHÂN TÍCH HỆ THỐNG 1

1.1. Ngữ cảnh ứng dụng và vấn đề đặt ra: 1

1.1.1 : Ngữ cảnh ứng dụng: 1

1.1.1.1 : Kiến trúc hệ thống bất động sản: 1

1.1.1.2 : Phân quyền truy cập dữ liệu: 1

1.1.2 : Vấn đề đặt ra: 2

1.2. Các bên liên quan 3

1.3. Tài sản cần bảo vệ (Assets): 4

1.4. Các mối đe dọa(Threats): 4

CHƯƠNG 2: THIẾT KẾ GIẢI PHÁP 5

2.1. Các mục tiêu bảo mật (Security Goals): 5

2.2. Giải pháp thực hiện các mục tiêu trên (Solutions): 5

2.2.1. Giải pháp về Authentication & Authorization: 5

2.2.2. Giải pháp về Confidentiality: 6

2.2.2.1 : Mã hóa dữ liệu bằng hệ mã đối xứng(Symmetric cipher): 6

2.2.2.2 : Kiểm soát truy cập dữ liệu bằng CP-ABE encryption 6

2.2.2.3 : Trao đổi khóa bí mật secret key bằng thuật toán Diffie - Hellman 6

CHƯƠNG 3: TRIỂN KHAI GIẢI PHÁP VÀ ĐÁNH GIÁ 9

3.1 Kịch bản triển khai: 9

3.2 Chạy thử nghiệm hệ thống 9

3.3 Đánh giá 18

DANH MỤC HÌNH

Hình 1: Minh họa vấn đề truy cập dữ liệu. 9

Hình 2: Minh họa các bên liên quan của hệ thống. *10*

Hình 3: Mô hình CP-ABE trong giao dịch bất động sản *15*

Hình 4: Mô hình minh họa kịch bản triển khai *17*

CHƯƠNG 1: PHÂN TÍCH HỆ THỐNG

1.1. Ngữ cảnh ứng dụng và vấn đề đặt ra:

1.1.1 : Ngữ cảnh ứng dụng:

1.1.1.1 : Kiến trúc hệ thống của công ty bất động sản:

Trong thực tế, một công ty bất động sản có rất nhiều hệ thống khác nhau với các vai trò khác nhau. Ở ngữ cảnh này, chúng em xem xét ở khía cạnh hệ thống lưu trữ và quản lý tài liệu, nơi lưu trữ dữ liệu liên quan đến các giao dịch, hợp đồng và hồ sơ pháp lý. Hệ thống này có thể truy cập bởi nhiều bên liên quan như dịch vụ hành chính công hay ngân hàng nhằm các mục đích nhất định (đã có sự thỏa thuận bởi các bên).

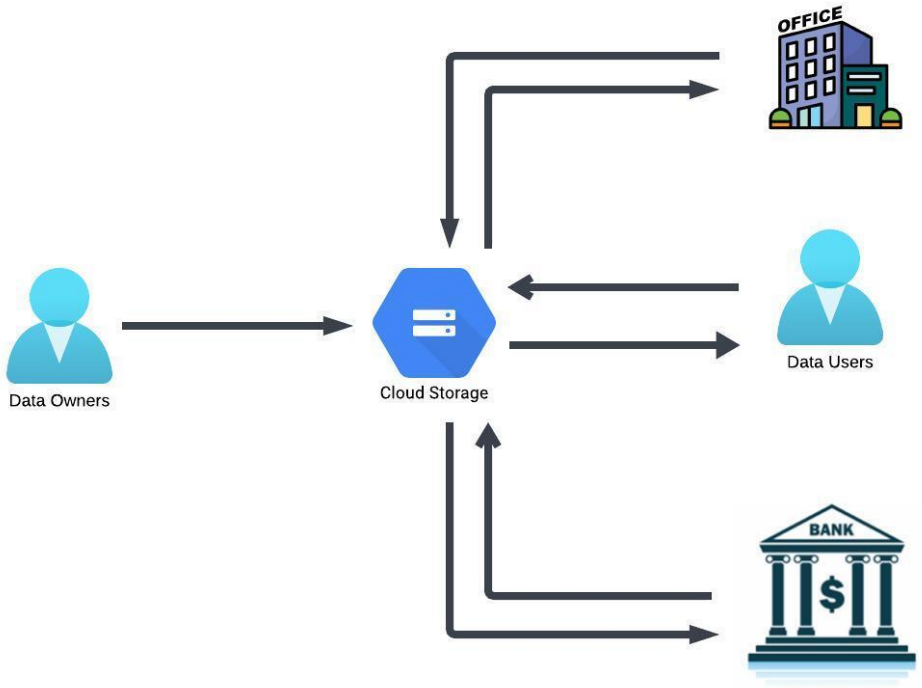
1.1.1.2 : Phân quyền truy cập dữ liệu:

Hệ thống quản lý, lưu trữ dữ liệu không những có thể được truy cập từ bên trong nội bộ công ty mà còn có thể từ bên ngoài bởi các bên có liên quan, với các mục đích khác nhau. Tuy nhiên quyền truy cập và người được truy cập của các bên là khác nhau, và các thao tác trên hệ thống của các bên cũng khác nhau. Ví dụ những người trong công ty có thể upload dữ liệu lên và có thể lấy dữ liệu về, tuy nhiên những người bên ngoài thì không thể upload. Hay những người thuộc về hành chính công chỉ có thể thấy các thông tin liên quan đến hồ sơ pháp lý, ngân hàng chỉ có thể thấy thông tin về giao dịch.

1.1.2 : Vấn đề đặt ra :

Dữ liệu được lưu trữ và chia sẻ công khai ở kho dữ liệu dùng chung (Cloud storage) cho hệ thống quản lý, lưu trữ thông tin bất động sản. Tùy theo nghiệp vụ, thẩm quyền của mỗi cá nhân/bên mà phạm vi truy cập dữ liệu sẽ khác nhau. Vậy làm thế nào để có thể bảo vệ thông tin và kiểm soát quyền truy cập đến các dữ liệu bất động sản trên Cloud Storage?

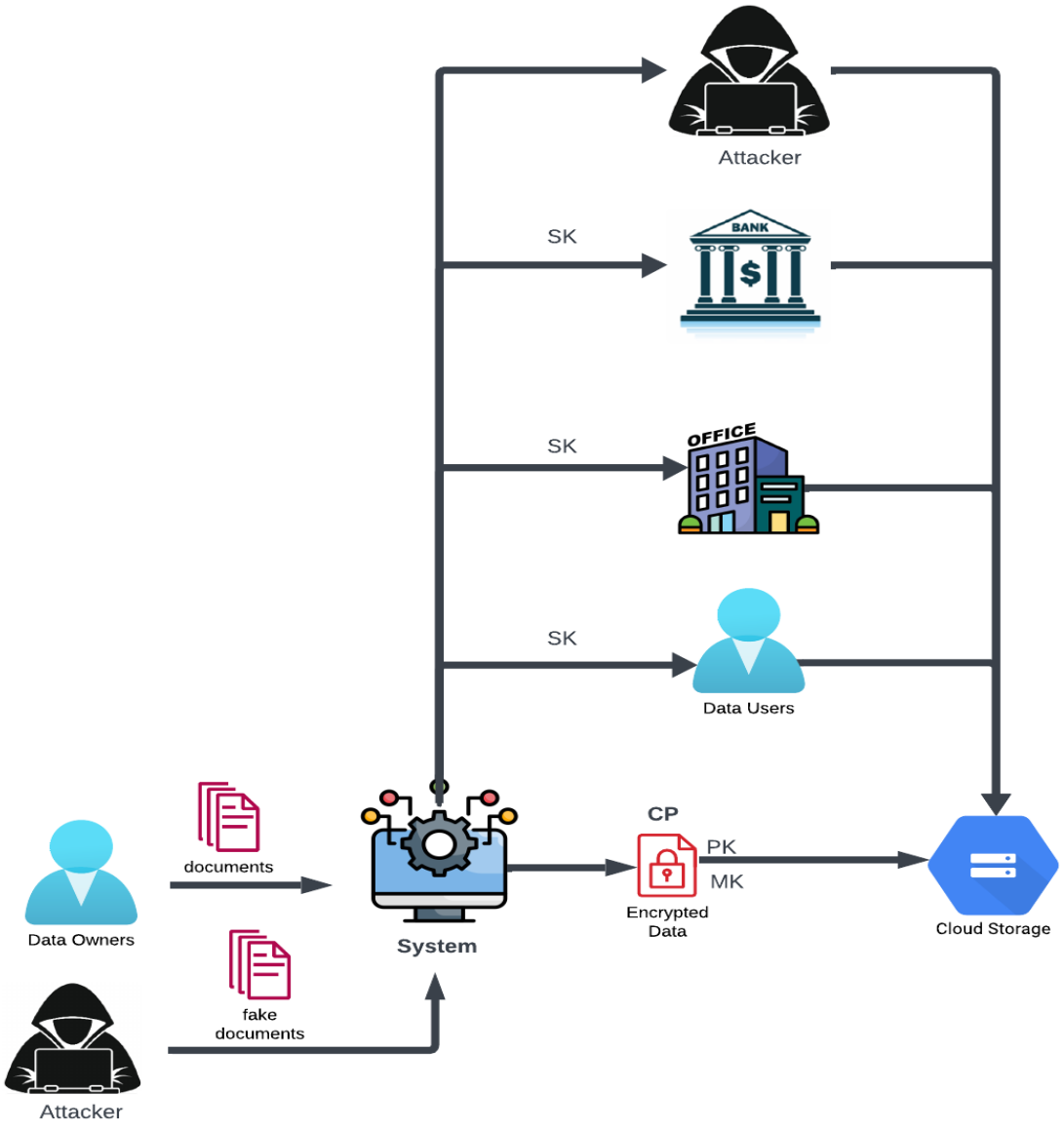
`



Hình 1: Minh họa vấn đề truy cập dữ liệu.

1.2. Các bên liên quan:

Các bên liên quan của hệ thống bất động sản gồm có các nhóm: Data Owners, Data users, Cloud và các mối đe dọa Adversaries.



Hình 2: Minh họa các bên liên quan của hệ thống.

Nhóm Data Owners (chủ sở hữu dữ liệu): là những cá nhân trong hệ thống quản lý của công ty. Họ chịu trách nhiệm quản lý, sở hữu dữ liệu trong hệ thống. Data owner chịu trách nhiệm định nghĩa các chính sách truy cập của những dữ liệu đó.

Nhóm Data Users (nhóm người dùng dữ liệu): là những cá nhân, tổ chức được ủy quyền, cấp phép để truy cập và sử dụng dữ liệu trong hệ thống bất động sản. Data users ở đây bao gồm nhân viên thuộc công ty hay của dịch vụ hành chính công và của ngân hàng.

Trong vài trường hợp data owners cũng là data users, họ tùy vào nghiệp vụ có thể sở hữu dữ liệu và định nghĩa các chính sách truy cập vào tài liệu đó, đồng thời họ cũng là một user, được phép truy cập và sử dụng một số dữ liệu nhất định dựa trên thuộc tính của họ.

Nhóm Adversaries: là các mối đe dọa có ý định tấn công, xâm nhập đến hệ thống bất động sản. Nhóm này có thể là kẻ tấn công bên ngoài (attacker), người bên trong hệ thống bất động sản, đối tác, …

Cloud: đóng vai trò lưu trữ, nơi chia sẻ các dữ liệu của toàn bộ hệ thống bất động sản. Chịu trách nhiệm nhận dữ liệu từ hệ thống để lưu trữ và gửi dữ liệu về khi có yêu cầu. Cloud storage cung cấp khả năng lưu trữ linh hoạt, khả năng mở rộng và khả năng truy cập từ xa cho các bên liên quan khác. Tuy vậy, Cloud storage được đánh giá là một bên liên quan semi-trusted, hệ thống lưu trữ dữ liệu lên Cloud storage nhưng không thể tiết lộ thông tin lên cho bên Cloud storage này biết được vì một số lý do về pháp lý và tính bảo mật thông tin.

1.3. Tài sản cần bảo vệ (Assets):

Hệ thống bất động sản chứa nhiều tài sản nhạy cảm, bao gồm hợp đồng, hồ sơ pháp lý và các giao dịch liên quan đến bất động sản. Các tài sản này cần được bảo vệ để đảm bảo tính bảo mật, ngăn ngừa các rủi ro liên quan đến việc chỉnh sửa, mất mát, và lộ thông tin.

Đồng thời các tài sản này lại được chia sẻ cho bên thứ ba (Cloud storage) để chia sẻ và lưu trữ, nên tài sản càng cần được bảo mật, kiểm soát nghiêm ngặt hơn.

1.4. Các mối đe dọa(Threats):

- **Semi-trusted third party:**

Vì những dữ liệu này được lưu trữ, chia sẻ trên Cloud storage (third party) nên ta cần phải cẩn trọng rằng Cloud cũng là một bên có thể truy cập tới dữ liệu, ngoài những bên liên quan mà ta mong muốn. Đồng thời Cloud cũng có nhiều service khác ngoài service của chúng ta yêu cầu.

- **Fake related parties**:

Adversaries có thể giả mạo các bên liên quan tới hệ thống và truy cập và giải mã dữ liệu.

CHƯƠNG 2: THIẾT KẾ GIẢI PHÁP

2.1. Các mục tiêu bảo mật (Security Goals):

Sau khi tìm hiểu và đưa ra được mục tiêu của các mối đe dọa, nhóm chúng em đề ra 2 mục tiêu bảo mật chính cho hệ thống: Authentication & Authorization và Confidentiality.

- **Authentication & Authorization** (Xác thực và cấp quyền): xác thực người dùng đang muốn truy cập vào là ai, liệu người đó có được phép truy cập vào không. Đồng thời xác định nếu người đó được truy cập vào, họ sẽ được phép làm gì với hệ thống.
- **Confidentiality** (Tính bảo mật): đảm bảo dữ liệu chỉ được chia sẻ cho các bên nghiệp vụ liên quan.

2.2. Giải pháp thực hiện các mục tiêu trên (Solutions):

2.2.1. Giải pháp về Authentication & Authorization:

- Thiết kế module login

Người dùng khi muốn login vào hệ thống bất động sản họ sẽ phải cung cấp thông tin đăng nhập bao gồm username và password. Nếu chưa từng đăng nhập trên hệ thống thì sẽ có mục đăng ký. Hệ thống sẽ kiểm tra thông tin này để xác định người dùng này có quyền truy cập vào hệ thống hay không.

2.2.2. Giải pháp về Confidentiality:

Khi hệ thống vận hành, liệu việc giao tiếp giữa các thực thể trong hệ thống có được an toàn hay không? Và như đã đề cập bên trên, dữ liệu phải được bảo vệ an toàn do chia sẻ lưu trữ ở bên thứ ba (Cloud storage), và tùy vào mỗi đối tượng người dùng, tùy vào vị trí, phòng ban, nghiệp vụ của từng đối tượng mà dữ liệu được phép truy cập để sử dụng là khác nhau. Vậy giải pháp cho các tình huống này là như thế nào?

2.2.2.1 : Mã hóa dữ liệu bằng hệ mã đối xứng(Symmetric cipher):

Dữ liệu cần được mã hóa là rất lớn, nên cần lựa chọn một hệ mã phù hợp. Mã hóa bằng AES (Advanced Encryption Standard) là sự lựa chọn phù hợp để mã hóa dữ liệu, đảm bảo được chi phí tính toán nhẹ, bản mã/bản rõ không chênh lệch kích thước và đạt được các yêu cầu về bảo mật.

Vấn đề kinh điển của hệ mã đối xứng đó là việc phân phối keys (keys

distribution), vậy trong hệ thống này khóa để giải mã (content keys) được phân phối như thế nào và làm sao để chỉ những người có chức năng, nghiệp vụ tương ứng với dữ liệu mới có thể giải mã và sử dụng ?

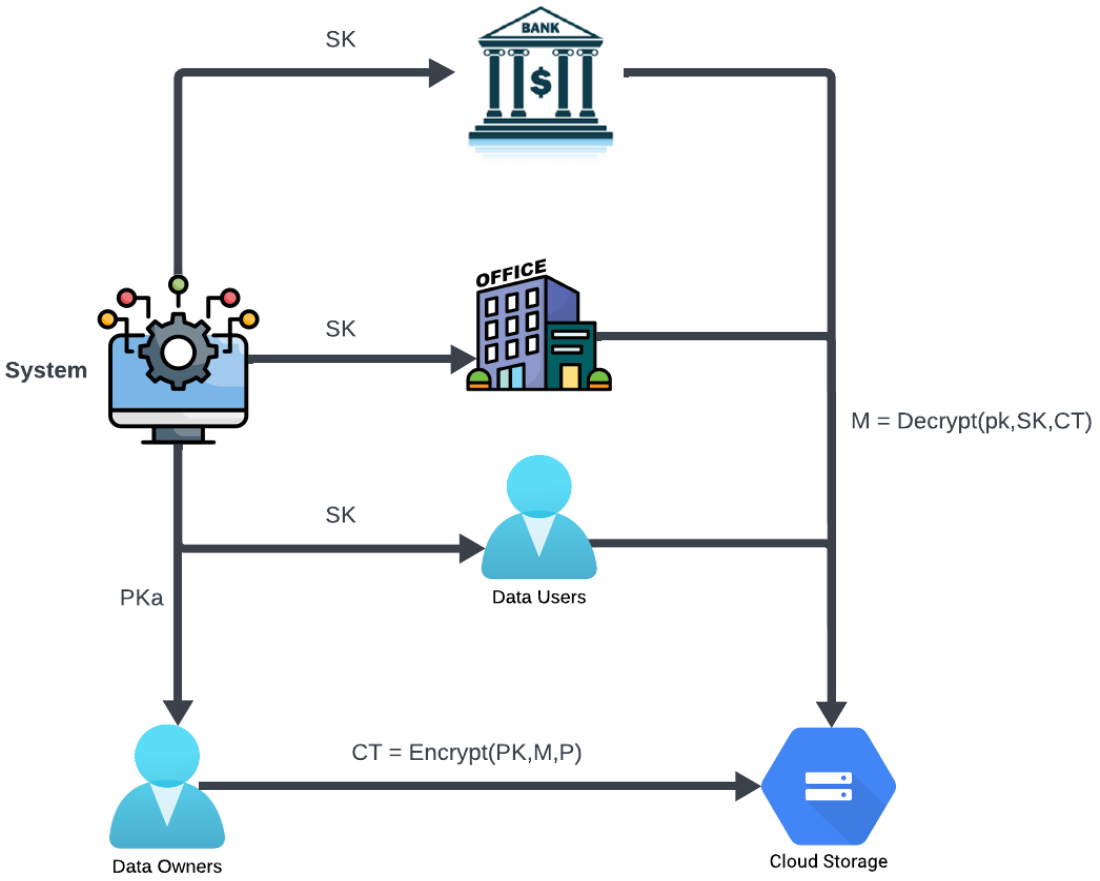
2.2.2.2 : Kiểm soát truy cập dữ liệu bằng CP-ABE:

Để giải quyết vấn đề đặt ra ở mục 2.2.2.1, chúng em chọn áp dụng CP-ABE encryption để mã hóa content keys (keys AES), đảm bảo việc giải mã keys chỉ có thể thực hiện bởi cá nhân/nhóm mang một số thuộc tính (attributes) nhất định, ngăn ngừa rò rỉ thông tin của bất động sản.

Tuy nhiên, vẫn còn trường hợp attacker đánh cắp secret key để giải mã dữ liệu đã mã hóa được tạo từ hệ thống gửi cho người dùng, hệ thống này làm sao giải quyết vấn đề trên ?

2.2.2.3 : Trao đổi khóa bí mật secret key bằng thuật toán Diffie - Hellman:

Nhằm giải quyết vấn đề ở mục 2.2.2.2, chúng em dùng thuật toán Diffie - Hellman để tạo key cho mỗi phiên trao đổi và dùng AES-GCM-256 để mã hóa secret key, cặp key dùng để mã hóa và giải mã secret key sẽ được trao đổi giữa hệ thống và người dùng khi người dùng tạo tài khoản trên hệ thống.



Hình 3: Mô hình CP-ABE trong hệ thống bất động sản.

Chọn scheme bsw07. Dưới đây là phần phân tích scheme:

Phase 1: System Initialization (Thiết lập hệ thống)

Hệ thống tạo ra Global Public Key (PK) và Master Key (MK), trong đó PK được dùng để mã hóa, MK được dùng để giải mã và cả hai được dùng tạo Secret Key (SK) giải mã dữ liệu.

Phase 2: Secret Key Generation

Thuật toán lấy đầu vào là Global Public Key (PK), Master Key (MK) và tập thuộc tính (S) của người dùng. Chính sách truy cập (A) được viết dưới dạng biểu thức Boolean. Sau đó, hệ thống sẽ trả về Secret Key (SK) được tạo từ tập thuộc tính người dùng, khóa này được dùng để giải mã bản mã.

Phase 3: Data Encryption by Owners. (Data owners mã hóa dữ liệu)

Thuật toán mã hóa lấy đầu vào là Global Public Key (PK), dữ liệu (M), chính sách truy cập (A) và cặp khóa trao đổi (P1 và P2). Thuật đầu ra sẽ là bản mã (CT) sao cho chỉ có một tập người dùng có tập thuộc tính (S) thỏa mãn chính sách truy cập mới có thể giải mã. Chúng em nhận thấy policy trong schema gốc còn ở dạng văn bản thô nên có rủi ro khi người dùng lấy về, do đó policy sẽ được mã hóa bằng AES-GCM-256 với cặp khóa P1 và P2.

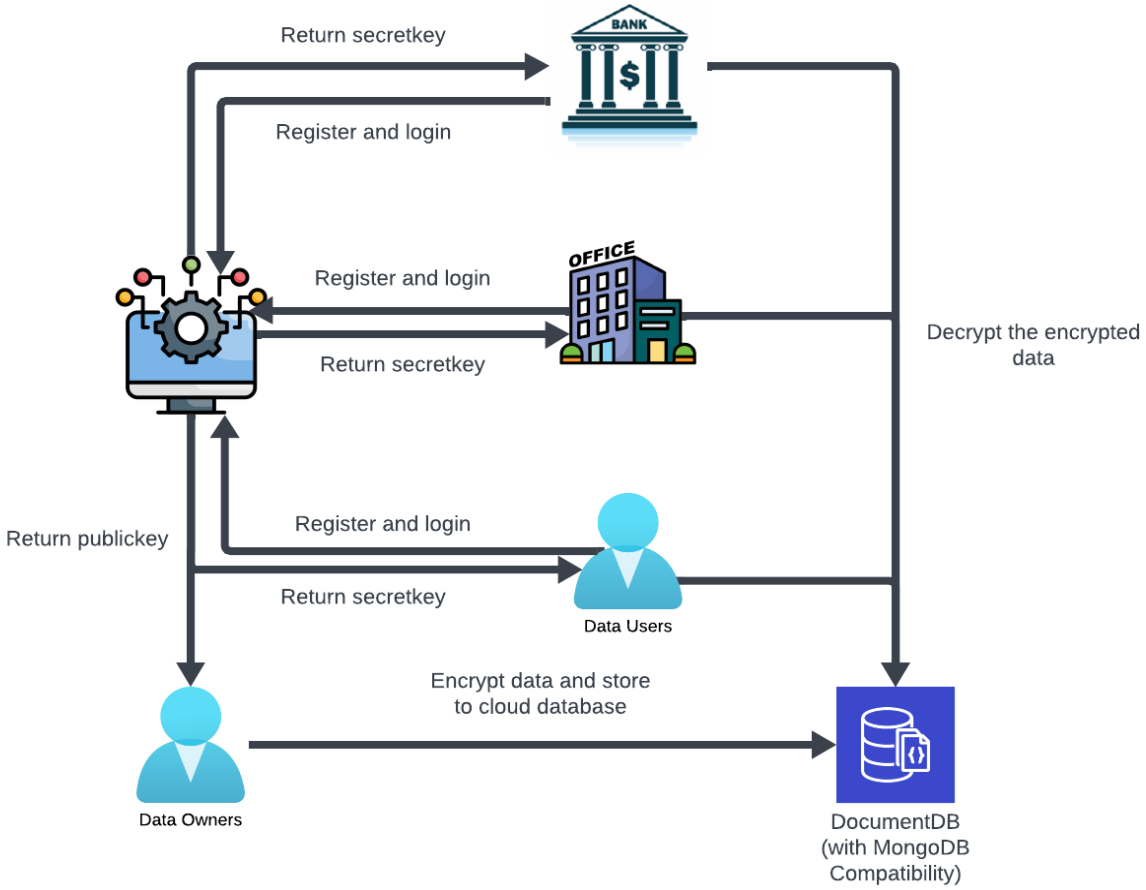
Phase 4: Data Decryption by Users. (Data users giải mã dữ liệu)

Thuật toán giải mã lấy đầu vào là Master Key (MK), bản mã (CT), khóa bí mật (SK) và cặp khóa trao đổi (P1 và P2). Chúng em sẽ dùng cặp khóa trao đổi này để giải mã lại policy sau đó thực hiện xét khóa bí mật có phù hợp với chính sách truy cập hay không. Nếu tập thuộc tính phù hợp với chính sách truy cập A, khóa bí mật sẽ giải mã bản rõ là M.

CHƯƠNG 3: TRIỂN KHAI GIẢI PHÁP VÀ ĐÁNH GIÁ

3.1 Kịch bản triển khai:

Chạy thực nghiệm hệ thống đã xây dựng bằng terminal. Kiểm chứng tính đúng đắn, hiệu quả của mô hình CP-ABE trong việc mã hóa, kiểm soát truy cập và truy vấn dữ liệu trên cloud database.



Hình 4: Mô hình minh họa kịch bản triển khai.

Setup:

- Môi trường thực nghiệm: Linux
- Hệ thống gồm các nodes: 1 Bank server node (data owner), 2 User nodes (với 2 kịch bản khác nhau để minh họa phạm vi truy cập dữ liệu).

Khái quát nhiệm vụ của từng node

3.2 Thử nghiệm hệ thống

```python
class CPabe_BSW07(ABEnc):
    def __init__(self, groupObj):
        ABEnc.__init__(self)
        global util, group
        util = SecretUtil(groupObj, verbose=False)
        group = groupObj

    @Output(pk_t, mk_t)
    def setup(self):
        g, gp = group.random(G1), group.random(G2)
        alpha, beta = group.random(ZR), group.random(ZR)
        # initialize pre-processing for generators
        g.initPP(); gp.initPP()

        h = g ** beta; f = g ** ~beta
        e_gg_alpha = pair(g, gp ** alpha)

        pk = { 'g':g, 'g2':gp, 'h':h, 'f':f, 'e_gg_alpha':e_gg_alpha }
        mk = {'beta':beta, 'g2_alpha':gp ** alpha }
        return (pk, mk)
```

Generate master key and public key

```python
    @Input(pk_t, mk_t, [str])
    @Output(sk_t)
    def keygen(self, pk, mk, S):
        r = group.random()
        g_r = (pk['g2'] ** r)
        D = (mk['g2_alpha'] * g_r) ** (1 / mk['beta'])
        D_j, D_j_pr = {}, {}
        for j in S:
            r_j = group.random()
            D_j[j] = g_r * (group.hash(j, G2) ** r_j)
            D_j_pr[j] = pk['g'] ** r_j
        return { 'D':D, 'Dj':D_j, 'Djp':D_j_pr, 'S':S }
```

Generate private key for data users

```
@Input(pk_t, bytes, str, prikey, pubkey)
@Output(dict)
def encrypt(self, pk, M, policy_str, private, public):
    M_GT = group.random(GT)
    # Symmetric encryption of the message
    sym_key = hashPair(M_GT)
    sym_crypto = SymmetricCryptoAbstraction(sym_key)
    ct_sym = sym_crypto.encrypt(M)

    server_pri = ECC.construct(
        curve=private['curve'],
        point_x=IntegerGMP(int(private['point_x'])),
        point_y=IntegerGMP(int(private['point_y'])),
        d=IntegerGMP(int(private['d']))
    )

    server_pub = ECC.construct(
        curve=public['curve'],
        point_x=IntegerGMP(int(public['point_x'])),
        point_y=IntegerGMP(int(public['point_y']))
    )

    nonce = os.urandom(64)
    encoded_policy = policy_str.encode('utf-8')
    encrypted_policy = policy_encrypt(encoded_policy, nonce, server_pri, server_pub)
    # ABE encryption of the symmetric key
    policy = util.createPolicy(policy_str)
    s = group.random(ZR)
    shares = util.calculateSharesDict(s, policy)

    C = pk['h'] ** s
    C_y, C_y_pr = {}, {}
    for i in shares.keys():
        j = util.strip_index(i)
        C_y[i] = pk['g'] ** shares[i]
        C_y_pr[i] = group.hash(j, G2) ** shares[i]

    ct_abe = { 'C_tilde':(pk['e_gg_alpha'] ** s) * M_GT,
               'C':C, 'Cy':C_y, 'Cyp':C_y_pr, 'policy':encrypted_policy }

    return {'ct_sym': ct_sym, 'ct_abe': ct_abe, 'ct_nonce': nonce}
```

Encrypt data with policy and store to cloud database

```
@Input(sk_t, dict, prikey, pubkey)
@Output(bytes)
def decrypt(self, sk, ct, private, public):
    ct_sym = ct['ct_sym']
    ct_abe = ct['ct_abe']
    nonce = ct['ct_nonce']

    server_pri = ECC.construct(
        curve=private['curve'],
        point_x=IntegerGMP(int(private['point_x'])),
        point_y=IntegerGMP(int(private['point_y'])),
        d=IntegerGMP(int(private['d']))
    )
    server_pub = ECC.construct(
        curve=public['curve'],
        point_x=IntegerGMP(int(public['point_x'])),
        point_y=IntegerGMP(int(public['point_y']))
    )

    decrypted_policy = policy_decrypt(ct_abe['policy'], nonce, server_pri, server_pub).decode('utf-8')
    # ABE decryption to get the symmetric key
    policy = util.createPolicy(decrypted_policy)
    pruned_list = util.prune(policy, sk['S'])
    if pruned_list == False:
        return b'False'
    z = util.getCoefficients(policy)
    A = 1
    for i in pruned_list:
        j = i.getAttributeAndIndex(); k = i.getAttribute()
        A *= ( pair(ct_abe['Cy'][j], sk['Dj'][k]) / pair(sk['Djp'][k], ct_abe['Cyp'][j]) ) ** z[j]

    M_GT = ct_abe['C_tilde'] / (pair(ct_abe['C'], sk['D']) / A)

    sym_key = hashPair(M_GT)
    # Symmetric decryption of the message
    sym_crypto = SymmetricCryptoAbstraction(sym_key)
    M = sym_crypto.decrypt(ct_sym)
    return M
```

Decrypt the cipher

```
# This KDF has been agreed in advance
def kdf(x):
        return SHAKE256.new(x).read(32)

def sessionKeyGen(U_static, V_static):
    session_key = key_agreement(static_priv=U_static,
                                static_pub=V_static,
                                kdf=kdf)
    return session_key

def preprocess_data(group, key):
    new_key = {}
    for field in key:
        if type(key[field]) == list:
            new_key[field] = pickle.dumps(key[field])
        elif type(key[field]) == dict:
            temp = {}
            for N_field in key[field]:
                temp[N_field] = group.serialize(key[field][N_field])
            new_key[field] = temp
        else:
            new_key[field] = group.serialize(key[field])
    return new_key

def recover_data(group, key):
    new_key = {}
    pattern = re.compile(rb'^[0-9]:')
    for field in key:
        if type(key[field]) == dict:
            temp = {}
            for N_field in key[field]:
                temp[N_field] = group.deserialize(key[field][N_field])
            new_key[field] = temp
        elif bool(pattern.match(key[field])) == True:
            new_key[field] = group.deserialize(key[field])
        else:
            new_key[field] = pickle.loads((key[field]))

    return new_key
```

Generate key for private key agreement

```python
def key_agreement_encrypt(group, secret, nonce, server_prikey, client_pubkey):
    session_key = sessionKeyGen(server_prikey, client_pubkey)
    # Encrypt secret with session key (using AES-GCM)
    aesgcm = AESGCM(session_key)
    secret = preprocess_data(group, secret)
    encrypted_secret = {}
    for field in secret:
        if type(secret[field]) == dict:
            temp = {}
            for N_field in secret[field]:
                temp[N_field] = aesgcm.encrypt(nonce, secret[field][N_field], None)
            encrypted_secret[field] = temp
        else:
            encrypted_secret[field] = aesgcm.encrypt(nonce, secret[field], None)

    return encrypted_secret

def key_agreement_decrypt(group, encrypted_secret, nonce, client_prikey, server_pubkey):
    session_key = sessionKeyGen(client_prikey, server_pubkey)
    # Decrypt encrypted secret with session key (using AES-GCM)
    aesgcm = AESGCM(session_key)
    decrypted_secret = {}
    for field in encrypted_secret:
        if type(encrypted_secret[field]) == dict:
            temp = {}
            for N_field in encrypted_secret[field]:
                temp[N_field] = aesgcm.decrypt(nonce, encrypted_secret[field][N_field], None)
            decrypted_secret[field] = temp
        else:
            decrypted_secret[field] = aesgcm.decrypt(nonce, encrypted_secret[field], None)

    decrypted_secret = recover_data(group, decrypted_secret)
    return decrypted_secret
```

Encrypt and decrypt key

```
nguyencuong@nguyencuong-Latitude-E7470:~/Workspace/CryptoCharm/python$ /bin/python3 /home/nguyencuong/Workspace
/CryptoCharm/python/CP-ABE/program.py
Connect MongoDB Successfully!
1. Login

2. Register

Your choice: 1
Enter your phone: 0707636110
Enter your password: nguyencuong
Logged in successfully
1. Push data

2. Get data

Your choice: 1

Access policy: Banker and 0707636123

Path to your file: CP-ABE/files/test.csv

Upload successfully!
```

Register and login for user

```
nguyencuong@nguyencuong-Latitude-E7470:~/Workspace/CryptoCharm/python$ /bin/python3 /home/nguyencuong/Workspace
/CryptoCharm/python/CP-ABE/program.py
Connect MongoDB Successfully!
1. Login

2. Register

Your choice: 1
Enter your phone: 0707636123
Enter your password: minh
Logged in successfully
2. Get data

Your choice: 2
What actions are allowed using your private key?
1. Allow
2. Cancel
1

ID:  6672d277a443cfb771dd6d66
Filename: Lab6LTHT.png
Created At:  2024-06-19 19:43:35.972000

ID:  66730c67abd015243702c8b7
Filename: test.csv
Created At:  2024-06-19 23:50:47.280000

Document Id: 66730c67abd015243702c8b7
```

```
Enter your phone: 0707636123
Enter your password: minh
Logged in successfully
2. Get data

Your choice: 2
What actions are allowed using your private key?
1. Allow
2. Cancel
1

ID:  6672d277a443cfb771dd6d66
Filename: Lab6LTHT.png
Created At:  2024-06-19 19:43:35.972000

ID:  66730c67abd015243702c8b7
Filename: test.csv
Created At:  2024-06-19 23:50:47.280000

Document Id: 66730c67abd015243702c8b7

Encrypted Secret Key:
 {'D': b"!\xff\x9e_\x19\xb8\x00wB\xee\x9f\xc5\x9dZ\x05n{'\x04\xdd\x88\x1f\x02\x9eu\xc7\xdb-\x8b\xad#Zm\xcfn\xd2\xd2\xc4\x92\xdb-*\xa9x\x1c\x8cc\xfcb\xf9\xe2\xc3\xb0\xc1-\xbb\
n17\x97\x9f\xe3\x90\x87\x9b\xa78\x12\xc5\x97|\x9d\xad\x8e\x98\xe2ux\xb4\x12\xaa\\\x80\xe1\xe3@:T\xf1\xe3B\xc2=o\x8c\xad\xab\x15\xb1\x1boab\\\x91\xaa", 'Dj': {'0707636123': b'
!\xff\xb6S\x03\x8f?\x19|\xde\xb4\xe0\x84^5p\x18\x1c\x04\xb9\x9f4\x17\x9aD\x9d\xe5\x1a\x92\x8f \x07}\x8d\r\xd9\xcd\xf4\xdf\xdd\x01H\xbblQ\x8eA\xf5o\xff\xaa\xbd\xce\xaap\xbc:\r
c\xc0\xbf\xef\x93\x90\x80\xac04\xd2\x8a|\xed\x88\xd7\x91\xf1pU\x80\x1e\xd5-\x95\xea\xbcB\tD\xf5\xe3\x8c\x89\xde\x99\xf8\xc7\x01\x99\xbd\xad\\\xa7\r:x\x03', 'NGUYỄN MINH': b'!
\xff\x9eGf\x9c\rwb\xf2\xd5\xed\xbb8\x01|\x02_\r\x8d\x9c&5\xc5\\\xfd\xefz\x91\x99\x18\x1fL\x82<\xcf\xd0\x9e\xb4\xc6Ps\x81aA\x92y\xefU\xdf\xd4\xc1\x88\xf7\x1b\xc0,Gp\xdf\xbc\x8
4\x85\xa4\x85\x8e\x1e7\x80\xcfb\xf5\xa2\xa8\x9c\xd4vC\xb4*\xa7]\xa4\x94\xee\x08]D\xf5\xe3\x93\xfb\x92\xce\x07\x1d\xb7-\xd1d\xd7\x02\xb9h\xa4q', 'BANKER': b'!\xff\xb6a}\xb1/\\
D\xd3\xcf\xd3\xbe$&I5?Y\xb2\x80><\xdeW\xda\x92\x18\xec\x9f,:}\x9f\x1c\xc7\xd8\x9e\xb2\xbf<Z\xb1aU\x9fT\xe7\x1a\xd4\xaf\x9e\xd4\xacp\xe4m>T\xcd\xba\xa4\x81\x82\xa1\xcf\x10S\x8
4\xf1\x07\x82\xb9\x9b\xb7\xd6ks\xac\x11\xaaI\x9f\xcc\xaf`\x00r\xf5\xe3\xcbw\x05\x9d"\x92\xc9\x9d\xb2\xb4V\xc2\x95r\x8f\x89', 'VIETCOMBANK': b"!\xff\xb7q+\x8c\x06l{\x95\xa7\xc2
\xe4Q\x15V})\x1b\xa6\x94:4\xc4_\xc5\xc2\x08\xb7\xac0\x02w\xa1\x1f\xfc\x80\xd6\xb0\x90Pu\xafW\x13\xb7D\x87\x02\xd1\xd5\xa0\x92\xda\x08\xbb\x0e+a\xcc\xe4\x85\xab\xa0\x8c\x81\x
17*\x82\xfa}\x8c\x8f\xd5\x83\xcdoQ\xf7.\xd3C\xa2\xd6\xb5k\x00b\xf5\xe3\xc6\xf0R'A\x19\xced'dQ\xaf\xd1Rj\xb6", 'STAFF': b'!\xff\xaev"\x99\x1dDm\xfc\xcf\xe6\xe4~ A\x18>\'\x8d\xb
c\x028\xe7Q\xcc\xc5)\x84\x84^\x00D\xc14\xe6\xc0\xf2\xd4\xa1#D\xcc%p\xffy\xc3~\xc6\xa8\xc3\xaf\xf4\x15\xe2\x17\x0ew\xc7\x84\xae\xbb\x91\xf9\x93J/\x82\xd3p\xe3\xfc\xb8\x9d\xa4
^\x86\x02\x80*\x85\xd1\x96\x0bDb\xf5\xe3)I\xa7\x04`\x87\xb2\x9c\x81\xc4+\xc7Q9\xba\x8b'}, 'Djp': {'0707636123': b'"\xff\x9e{\x10\x8a(Qh\xde\xb9\x9b\xbeu\x03J<\\,\xbd\x96\x04
&\xe7Q\x99\xc8\x17\x98\xd8CY$\x8d\x0e\xaf\xc0\xd2\xbf\xab\x1ay\x95af\xf7d\xd0`\xfc\xae\xc2\x93\xad/\xa6i2N\xd9\x9f\xe1\x9a\xe1\x8e\x80\x0e)\xc3\xd9f\xee\x87\x87\x8f\xaeos\xfc
?\xd0C\xbd\xe2\xb7Z\x1ab\xf1\xe3SC\xd5{q\x0f\xf2n\x11\x9f\xcf6\xb6B\xdf)', 'NGUYỄN MINH': b'"\xff\xb7V\x1b\xae\x16W;\xcb\xb1\x91\x8f eb \x1d2\xae\xb9\x10\x02\xcaD\xee\xec\x7f
\xe7\xbd\x02-W\x983\xc3\x82\xd0\xd3\xdbMe\xb5S\x0b\xfeE\x8eJ\xfc\xcd\x91\x94\xb3 \xdf>0v\xd7\x91\xa1\xa2\xa1\x8e\xb0;1\xea\xdd\x05\x82\xa7\xaf\x90\xe4`r\xa3\x0f\xabb\xa1\xe2\
xadm\x1dT\xf5\xe3\xa9\xc2Z\xd2\x92Y\xe8L\x08\xefAE\xef\xb5\xb02', 'BANKER': b'"\xff\xb8Y<\x98_o%\xf7\x8e\xcc\x91t\x0b2 \x02Z\xc1\xc25I\xfbm\xd2\xd1,\xba\xad?=\\\xb8\x17\xff\x
c0\xc7\x82\x93\x15d\x8d~h\xb31\xd3K\xf6\xe9\xb8\x9b\xe8t\xf3e\x18a\xe9\x97\xa4\x8b\x98\xbc\xa5H\x0f\xf0\xeaW\xb3\xb8\x84\xc3\xdbLp\xfa}\xaab\x81\xe0\xbc\x0f-r\xf5\xe3\x9c@\x9
2Z\x8d\x0c\xe4\xa4|\xe4\xba\xbc\x89\xe2\xd5\x96', 'VIETCOMBANK': b'"\xff\xb9\x01\x0f\x98\x15F:\xdf\xac\xc2\xe4Rg`\nY]\xbb\x81B#\xf8W\xc6\xd4\x19\xba\x81?\x1az\xce?\xc1\x98\xd
0\xcc\xaa\x1ft\xb2sG\xa2d\xfe\x19\xda\xf2\xc5\xb1\xe9\x00\xe7\x1a0K\xcd\xe0\xee\x9a\xb8\x85\x83\x15?\xf8\xedp\x9d\xa8\xae\x9d\xafD}\x85%\x86w\xdd\xf2\xbdy\x00D\xf5\xe3{\xe5\x
faG7\xafH\x7f\xe0\x03\xe5E_\xb5\xd8\xb0', 'STAFF': b'"\xff\xa5\x7f=\x88,bs\xc2\x96\x82\x81v\x0b3>\\\x18\x8d\x94\x03\x00\x97J\xf0\x9a:\xb0\x81\n\x03=\xbe\x1c\xc1\xe5\xd1\x8f\x
980/\x9d7J\x88/\xfbL\xc5\xf1\xad\xd0\xf6\r\xfb%,Q\x97\x97\x8e\xd2\xb7\xa6\xd5\x1aW\xdf\xef`\xb9\xa0\x8c\x9d\xd1DZ\x88y\xb1\x7f\xaf\xf6\xebm\x1fb\xf1\xe3\x11M\xfa\xa3L\x85\xf5\
xcf\xff+K\x9e\xef\xabU\x0b'}, 'S': b'\x93\xclasI\xcbg6\n\xba\xfe\xf4B?\xdd\x0f}]Z\xdd\xc6DF\x9e\x0c\x857\xc2\xd3\xa4/?V\x16\xfe\x13\xf9\x86\xaa\xa05Um\x98"\x85A\xf8f\xd2\xc8
cs\x93\x17\xc0\x18+C\xe1\x9e\x94\xab\x9c\x80p\xf3c\xe0\xecw\x9c\x8cv\x93\xb9\x05\x02\r\xcbw"\xee\xde\x08\x04>\xbf&\xd5\x1f\xa7'}
```

```
Enter Encrypted Secret Key:
{'D': b"!\xff\x9e_\x19\xb8\x00wB\xee\x9f\xc5\x9dZ\x05n{'\x04\xdd\x88\x1f\x02\x9eu\xc7\xdb-\x8b\xad#Zm\xcfn\xd2\xd2\xc4\x92\xdb-*\xa9x\x1c\x8cc\xfcb\xf9\xe2\xc3\xb0\xc1-\xbb\n
17\x97\x9f\xe3\x90\x87\x9b\xa78\x12\xc5\x97|\x9d\xad\x8e\x98\xe2ux\xb4\x12\xaa\\\x80\xe1\xe3@:T\xf1\xe3B\xc2=o\x8c\xad\xab\x15\xb1\x1boab\\\x91\xaa", 'Dj': {'0707636123': b'!
\xff\xb6S\x03\x8f?\x19|\xde\xb4\xe0\x84^5p\x18\x1c\x04\xb9\x9f4\x17\x9aD\x9d\xe5\x1a\x92\x8f \x07}\x8d\r\xd9\xcd\xf4\xdf\xdd\x01H\xbblQ\x8eA\xf5o\xff\xaa\xbd\xce\xaap\xbc:\rc
\xc0\xbf\xef\x93\x90\x80\xac04\xd2\x8a|\xed\x88\xd7\x91\xf1pU\x80\x1e\xd5-\x95\xea\xbcB\tD\xf5\xe3\x8c\x89\xde\x99\xf8\xc7\x01\x99\xbd\xad\\\xa7\r:x\x03', 'NGUYỄN MINH': b'!\
xff\x9eGf\x9c\rwb\xf2\xd5\xed\xbb8\x01|\x02_\r\x8d\x9c&5\xc5\\\xfd\xefz\x91\x99\x18\x1fL\x82<\xcf\xd0\x9e\xb4\xc6Ps\x81aA\x92y\xefU\xdf\xd4\xc1\x88\xf7\x1b\xc0,Gp\xdf\xbc\x84
\x85\xa4\x85\x8e\x1e7\x80\xcfb\xf5\xa2\xa8\x9c\xd4vC\xb4*\xa7]\xa4\x94\xee\x08]D\xf5\xe3\x93\xfb\x92\xce\x07\x1d\xb7-\xd1d\xd7\x02\xb9h\xa4q', 'BANKER': b'!\xff\xb6a}\xb1/\\D
\xd3\xcf\xd3\xbe$&I5?Y\xb2\x80><\xdeW\xda\x92\x18\xec\x9f,:}\x9f\x1c\xc7\xd8\x9e\xb2\xbf<Z\xb1aU\x9fT\xe7\x1a\xd4\xaf\x9e\xd4\xacp\xe4m>T\xcd\xba\xa4\x81\x82\xa1\xcf\x10S\x84
\xf1\x07\x82\xb9\x9b\xb7\xd6ks\xac\x11\xaaI\x9f\xcc\xaf`\x00r\xf5\xe3\xcbw\x05\x9d"\x92\xc9\x9d\xb2\xb4V\xc2\x95r\x8f\x89', 'VIETCOMBANK': b"!\xff\xb7q+\x8c\x06l{\x95\xa7\xc2
\xe4Q\x15V})\x1b\xa6\x94:4\xc4_\xc5\xc2\x08\xb7\xac0\x02w\xa1\x1f\xfc\x80\xd6\xb0\x90Pu\xafW\x13\xb7D\x87\x02\xd1\xd5\xa0\x92\xda\x08\xbb\x0e+a\xcc\xe4\x85\xab\xa0\x8c\x81\x1
7*\x82\xfa}\x8c\x8f\xd5\x83\xcdoQ\xf7.\xd3C\xa2\xd6\xb5k\x00b\xf5\xe3\xc6\xf0R'A\x19\xced'dQ\xaf\xd1Rj\xb6", 'STAFF': b'!\xff\xaev"\x99\x1dDm\xfc\xcf\xe6\xe4~ A\x18>\'\x8d\xb
c\x028\xe7Q\xcc\xc5)\x84\x84^\x00D\xc14\xe6\xc0\xf2\xd4\xa1#D\xcc%p\xffy\xc3~\xc6\xa8\xc3\xaf\xf4\x15\xe2\x17\x0ew\xc7\x84\xae\xbb\x91\xf9\x93J/\x82\xd3p\xe3\xfc\xb8\x9d\xa4
^\x86\x02\x80*\x85\xd1\x96\x0bDb\xf5\xe3)I\xa7\x04`\x87\xb2\x9c\x81\xc4+\xc7Q9\xba\x8b'}, 'Djp': {'0707636123': b'"\xff\x9e{\x10\x8a(Qh\xde\xb9\x9b\xbeu\x03J<\\,\xbd\x96\x04&
\xe7Q\x99\xc8\x17\x98\xd8CY$\x8d\x0e\xaf\xc0\xd2\xbf\xab\x1ay\x95af\xf7d\xd0`\xfc\xae\xc2\x93\xad/\xa6i2N\xd9\x9f\xe1\x9a\xe1\x8e\x80\x0e)\xc3\xd9f\xee\x87\x87\x8f\xaeos\xfc?
\xd0C\xbd\xe2\xb7Z\x1ab\xf1\xe3SC\xd5{q\x0f\xf2n\x11\x9f\xcf6\xb6B\xdf)', 'NGUYỄN MINH': b'"\xff\xb7V\x1b\xae\x16W;\xcb\xb1\x91\x8f eb \x1d2\xae\xb9\x10\x02\xcaD\xee\xec\x7f\
xe7\xbd\x02-W\x983\xc3\x82\xd0\xd3\xdbMe\xb5S\x0b\xfeE\x8eJ\xfc\xcd\x91\x94\xb3 \xdf>0v\xd7\x91\xa1\xa2\xa1\x8e\xb0;1\xea\xdd\x05\x82\xa7\xaf\x90\xe4`r\xa3\x0f\xabb\xa1\xe2\x
adm\x1dT\xf5\xe3\xa9\xc2Z\xd2\x92Y\xe8L\x08\xefAE\xef\xb5\xb02', 'BANKER': b'"\xff\xb8Y<\x98_o%\xf7\x8e\xcc\x91t\x0b2 \x02Z\xc1\xc25I\xfbm\xd2\xd1,\xba\xad?=\\\xb8\x17\xff\xc
0\xc7\x82\x93\x15d\x8d~h\xb31\xd3K\xf6\xe9\xb8\x9b\xe8t\xf3e\x18a\xe9\x97\xa4\x8b\x98\xbc\xa5H\x0f\xf0\xeaW\xb3\xb8\x84\xc3\xdbLp\xfa}\xaab\x81\xe0\xbc\x0f-r\xf5\xe3\x9c@\x92
Z\x8d\x0c\xe4\xa4|\xe4\xba\xbc\x89\xe2\xd5\x96', 'VIETCOMBANK': b'"\xff\xb9\x01\x0f\x98\x15F:\xdf\xac\xc2\xe4Rg `nY]\xbb\x81B#\xf8W\xc6\xd4\x19\xba\x81?\x1az\xce?\xc1\x98\xd0
\xcc\xaa\x1ft\xb2sG\xa2d\xfe\x19\xda\xf2\xc5\xb1\xe9\x00\xe7\x1a0K\xcd\xe0\xee\x9a\xb8\x85\x83\x15?\xf8\xedp\x9d\xa8\xae\x9d\xafD}\x85%\x86w\xdd\xf2\xbdy\x00D\xf5\xe3{\xe5\xf
aG7\xafH\x7f\xe0\x03\xe5E_\xb5\xd8\xb0', 'STAFF': b'"\xff\xa5\x7f=\x88,bs\xc2\x96\x82\x81v\x0b3>\\\x18\x8d\x94\x03\x00\x97J\xf0\x9a:\xb0\x81\n\x03=\xbe\x1c\xc1\xe5\xd1\x8f\x9
80/\x9d7J\x88/\xfbL\xc5\xf1\xad\xd0\xf6\r\xfb%,Q\x97\x97\x8e\xd2\xb7\xa6\xd5\x1aW\xdf\xef`\xb9\xa0\x8c\x9d\xd1DZ\x88y\xb1\x7f\xaf\xf6\xebm\x1fb\xf1\xe3\x11M\xfa\xa3L\x85\xf5\
xcf\xff+K\x9e\xef\xabU\x0b'}, 'S': b'\x93\xclasI\xcbg6\n\xba\xfe\xf4B?\xdd\x0f}]Z\xdd\xc6DF\x9e\x0c\x857\xc2\xd3\xa4/?V\x16\xfe\x13\xf9\x86\xaa\xa05Um\x98"\x85A\xf8f\xd2\xc8c
s\x93\x17\xc0\x18+C\xe1\x9e\x94\xab\x9c\x80p\xf3c\xe0\xecw\x9c\x8cv\x93\xb9\x05\x02\r\xcbw"\xee\xde\x08\x04>\xbf&\xd5\x1f\xa7'}

Decrypt Secret Key


You can access. Please check your folder
```

Decrypt the data

3.3 Đánh giá

- Mô hình chúng em đã thực nghiệm được việc mã hóa dữ liệu đầu vào, lưu trữ và quản lý được trên cloud database. Khi có user truy vấn đến, mô hình sẽ kiểm tra xem thuộc tính của user có thể giải mã dữ liệu hay không
- Tuy nhiên, mô hình chưa có xác thực từ phía server, nên chưa đảm bảo được tính xác thực rằng user có đang truy cập đến đúng server không.

TÀI LIỆU THAM KHẢO

- **Maryam Almarwani, Boris Konev & Alexei Lisitsa. Data Querying with Ciphertext Policy Attribute Based -**
- **Encryption(2022)https://doi.org/10.48550/arXiv.2209.15103**
- **Junzuo Lai, Robert H. Deng & Jingjiu Li. Expressive CP-ABE with partially hidden access structures**
- **https://dx.doi.org/10.1145/2414456.241446**