ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

BÁO CÁO ĐỒ ÁN:

BẢO MẬT CƠ SỞ DỮ LIỆU

Mã Hóa và Quản Lý Điểm Sinh Viên

ĐỒ ÁN CUỐI KỲ

MÔN MÃ HÓA MẬT MÃ

21127577 – Trịnh Hoàng An

I. Giới Thiệu

Tiêu đề: Mã Hóa và Quản Lý Điểm Sinh Viên Dựa trên Cloud Firestore và Mã Hóa RSA

Mục đích đồ án:

Đồ án phát triển một hệ thống quản lý điểm sinh viên, tập trung vào việc bảo mật thông tin điểm số. Hệ thống cho phép giáo viên nhập điểm sinh viên và mã hóa chúng trước khi lưu trữ trên cơ sở dữ liệu Cloud Firestore, trong khi sinh viên có thể truy cập và xem điểm số của mình sau khi được giải mã.

Phương pháp phát triển:

- Sử dụng Cloud Firestore của Firebase làm cơ sở dữ liệu trực tuyến để lưu trữ thông tin điểm số.
- Áp dụng thuật toán mã hóa RSA bằng phương pháp số dư Trung Hoa để mã hóa và giải mã điểm số, đảm bảo rằng chỉ những người có quyền truy cập mới có thể hiểu và xử lý thông tin.
- Tạo giao diện người dùng cho phép nhập liệu điểm số và xem điểm đã giải mã một cách an toàn và tiện lợi.

Đồ án này nhằm mục đích tăng cường bảo mật thông tin trong quản lý điểm số sinh viên, đồng thời cung cấp một phương thức truy cập dữ liệu thuận tiện và an toàn cho cả giáo viên và sinh viên.

II. Cách thức hoạt động của chương trình

1. Các thư viện cần cài đặt và môi trường làm việc

Đồ án được thực hiện trên môi trường Python , được chạy trên google colab.

Các thư viện cần cài đặt để chạy được thuật toán là :

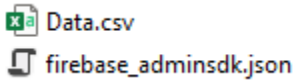
+ firebase-admin : Dùng để gửi và lấy dữ liệu từ fire base

+ cryptography : Dùng để viết các hàm tính toán chuyển đổi dữ liệu

+ Pandas : Dùng để đọc file dữ liệu để gửi lên firebase

2. Cách thức hoạt động và chạy của thuật toán.

- Muốn chạy được thuật toán phải up 2 file quan trọng lên google colab :



+ File đầu tiên là file Data.csv, file này chứa bảng điểm ban đầu của sinh viên

+ File tiếp theo là file firebase_adminsdk.json, dung để yêu cầu quyền truy cập admin của cloud, hỗ trợ lưu và lấy dữ liệu lên cloud(firebase)

```
# Tạo khóa công khai và riêng tư
private_key = rsa.generate_private_key(
    public_exponent=65537,
    key_size=2048,
    backend=default_backend()
)
public_key = private_key.public_key()
```

- Đây là hàm tạo khóa công khai và khóa riêng tư. Dùng để sinh các mã khóa để chạy các hàm sau

```
from firebase_admin import credentials, firestore, initialize_app

# Cấu hình Firebase
cred = credentials.Certificate('firebase_adminsdk.json')
db = firestore.client()
for index, row in df.iterrows():
    mssv_str = str(row['MSSV'])
    doc_ref = db.collection('sinh_vien').document(mssv_str)
    doc_ref.set({
        'ten': row['Họ Và tên'],
        'diem': row['Điểm']
    })
```

- Đây là hàm cấu hình nên bảng dữ liệu trong firebase, dung để gửi thông tin ban đầu lên firebase.

```python
m_values = [7793, 7817, 7919]

def mod_inverse(a, m):
    """Tính nghịch đảo modulo."""
    m0, x0, x1 = m, 0, 1
    while a > 1:
        q = a // m
        m, a = a % m, m
        x0, x1 = x1 - q * x0, x0
    return x1 + m0 if x1 < 0 else x1

def encrypt(data, m_values):
    """Mã hóa dữ liệu sử dụng Phương pháp số dư Trung Hoa."""
    M = 1
    for m in m_values:
        M *= m

    result = 0
    for i in range(len(data)):
        Mi = M // m_values[i]
        ei = mod_inverse(Mi, m_values[i])
        result += ei * data[i] * Mi
    return result % M

def decrypt(c, m_values, index):
    """Giải mã một phần tử cụ thể của dữ liệu."""
    return c % m_values[index]

def encrypt_and_convert_to_bytes(data, m_values):
    """Mã hóa dữ liệu và chuyển đổi sang bytes."""
    encrypted_data = encrypt(data, m_values)
    return encrypted_data.to_bytes((encrypted_data.bit_length() + 7) // 8, byteorder='big')


def bytes_to_int(bytes_data):
    """Chuyển đổi bytes thành int."""
    return int.from_bytes(bytes_data, byteorder='big')
```

- Đây là các hàm chính dùng để mã hóa và giải mã thông tin, vì điểm của sinh viên là dạng float (dạng số thập phân có 1 chữ số sau dấu “,”), nên phải nhân cho 10 và chuyển đổi về int để thực hiện phương pháp số dư Trung Hoa. và hàm **bytes_to_int** dùng để chuyển đổi từ bytes thành int khi học sinh muốn xem điểm thì lấy điểm từ firebase rồi chuyển đổi

```python
for index, row in df.iterrows():
    mssv_str = str(row['MSSV'])

    # Chuyển điểm số thành int sau khi nhân với 10
    diem_int = int(row['Điểm'] * 10)

    # Mã hóa điểm số và chuyển đổi sang bytes
    encrypted_score_bytes = encrypt_and_convert_to_bytes([diem_int], m_values)

    # Lưu thông tin lên Firebase
    doc_ref = db.collection('sinh_vien').document(mssv_str)
    doc_ref.set({
        'ten': row['Họ Và tên'],
        'diem_ma_hoa': encrypted_score_bytes
    })
```

- Hàm này dùng để gửi điểm số lên firebase sau khi đã mã hóa điểm số của học sinh trong file Data.csv.

```python
def input_teacher_data():
    """Nhập thông tin từ giáo viên và gửi lên Firebase."""
    mssv = input("Nhập MSSV: ")
    ten = input("Nhập tên sinh viên: ")
    diem = float(input("Nhập điểm: "))
    diem = round(diem,1)
    diem_int = int(round(diem * 10))  # Chuyển điểm số sang int sau khi nhân với 10

    # Mã hóa điểm số và chuyển đổi sang bytes
    encrypted_score_bytes = encrypt_and_convert_to_bytes([diem_int], m_values)

    # Lưu vào Firebase
    mssv_str = str(mssv)
    doc_ref = db.collection('sinh_vien').document(mssv_str)
    doc_ref.set({
        'ten': ten,
        'diem_ma_hoa': encrypted_score_bytes
    })
    print(f"Đã mã hóa và lưu điểm cho sinh viên {ten}.")


def view_student_score():
    mssv = input("Nhập MSSV của bạn: ")
    mssv_str = str(mssv)
    doc_ref = db.collection('sinh_vien').document(mssv_str)
    doc = doc_ref.get()
    if doc.exists:
        encrypted_score = doc.to_dict()['diem_ma_hoa']
        decrypted_score = decrypt(bytes_to_int(encrypted_score), m_values, 0)

        # Chuyển đổi lại thành điểm số float và chia cho 10
        final_score = round(decrypted_score / 10, 1)
        print(f"Điểm của bạn là: {final_score}")
    else:
        print("Không tìm thấy dữ liệu cho MSSV này.")


user_role = input("Bạn là giáo viên hay sinh viên? (teacher/student): ").lower()

if user_role == 'teacher':
    input_teacher_data()
elif user_role == 'student':
    view_student_score()
else:
    print("Vai trò không xác định!")
```

- Đây là hai hàm dung để xác định được xem bạn là giáo viên hay sinh viên :

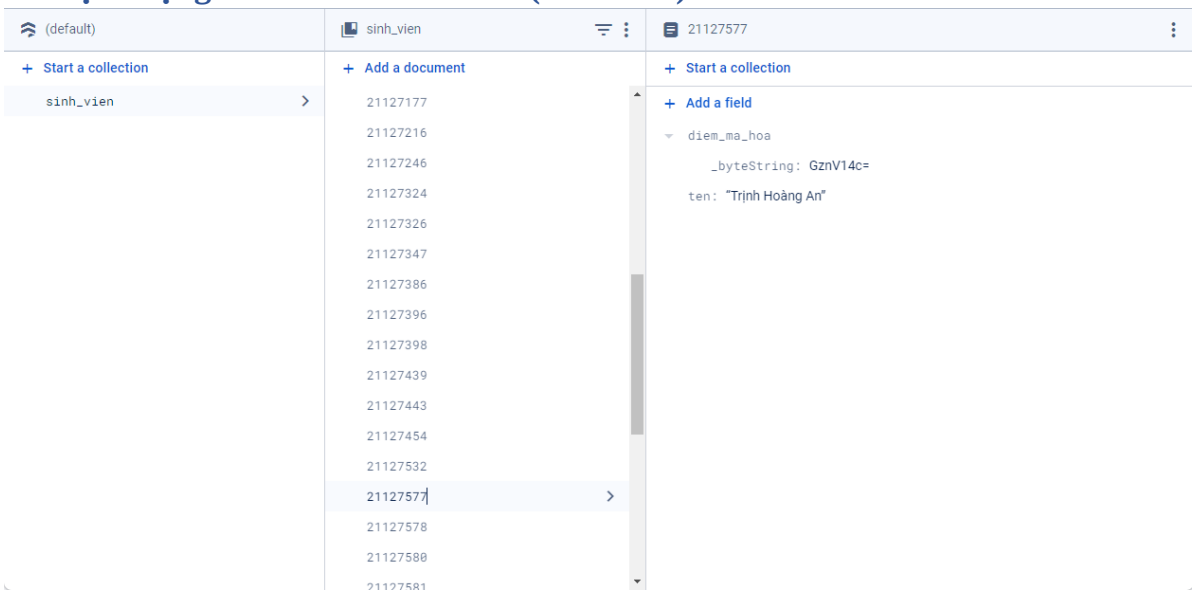
+ Nếu là giáo viên thì sẽ nhập các phần sau để đưa điểm số : MSSV, Họ và tên sinh viên, điểm số của sinh viên . Khi nhập điểm thì điểm sẽ tự làm tròn về dạng 1 chữ số thập phân sau đó thì sẽ bắt đầu mã hóa rồi gửi lên firebase

```
Bạn là giáo viên hay sinh viên? (teacher/student): student
Nhập MSSV của bạn: 22147039
Điểm của bạn là: 8.7
```

```
Bạn là giáo viên hay sinh viên? (teacher/student): teacher
Nhập MSSV: 22147039
Nhập tên sinh viên: Nguyễn Thị An Hòa
Nhập điểm: 8.65892
Đã mã hóa và lưu điểm cho sinh viên Nguyễn Thị An Hòa.
```

+ Nêu là học sinh thì nhập MSSV để xem điểm số của học sinh đó. Điểm sẽ được lấy từ firebase, rồi giải mã sau đó sẽ cho học sinh xem được điểm của mình .

3. Định dạng data base trên cloud (Firebase)



Data base sẽ được lưu dưới dạng :

Sinh_viên :

+ MSSV :

- diem_ma hoa :

  _bytestring : điểm đã được mã hóa

- ten : Tên sinh viên

III. Hướng phát triển.

Đồ án mới được thực hiện thô theo hướng xử lí mã hóa dữ liệu. Sẽ có nhiều thứ cần phát triển trong đồ án này :

+ Giáo viên và học sinh sẽ được cung cấp 1 tài khoản khác nhau dùng để đăng nhập, Giáo viên sẽ xem được toàn bộ điểm của học sinh và sửa điểm cho học sinh, Học sinh sẽ chỉ xem được điểm của chính mình.

+ Bảng điểm sẽ được cập nhật rộng hơn sẽ không chỉ có 1 cột điểm tổng mà sẽ có các điểm thành phần giúp giáo viên và học sinh dễ quan sát hơn

+ Dữ liệu mã hóa còn khá đơn giản, cải tiến phương pháp để dữ liệu được bảo mật tốt hơn.

+ Thiết kế frontend giúp giáo viên và sinh viên dễ dàng tiếp cận với chương trình hơn.

IV. Reference (Tham khảo)

https://docs.google.com/document/d/14gbGl4KjlSCWVkz7B36UDTeqn-mdCBh9/edit

https://firebase.google.com/docs/firestore?hl=fr

V. Mục Lục