Đại học Quốc gia thành phố Hồ Chí Minh
Trường Đại học Công nghệ Thông tin

Lab 1

PHỤC VỤ MỤC ĐÍCH GIÁO DỤC
FOR EDUCATIONAL PURPOSE ONLY

Thuật toán mã hoá DES

Thực hành môn Mật mã học

Tháng 3/2023

Lưu hành nội bộ

<Nghiêm cấm đăng tải trên internet dưới mọi hình thức>

Mọi góp ý về tài liệu, vui lòng gửi về email inseclab@uit.edu.vn

BEE SECURE
INSECLAB

A. TỔNG QUAN

1. Mục tiêu

- Hiểu được thuật toán DES và mode CBC
- Lập trình sử dụng được thư viện cryptopp trên đa nền tảng (window và linux)
- Tìm hiểu được một vài cuộc tấn công trên các thuật toán này

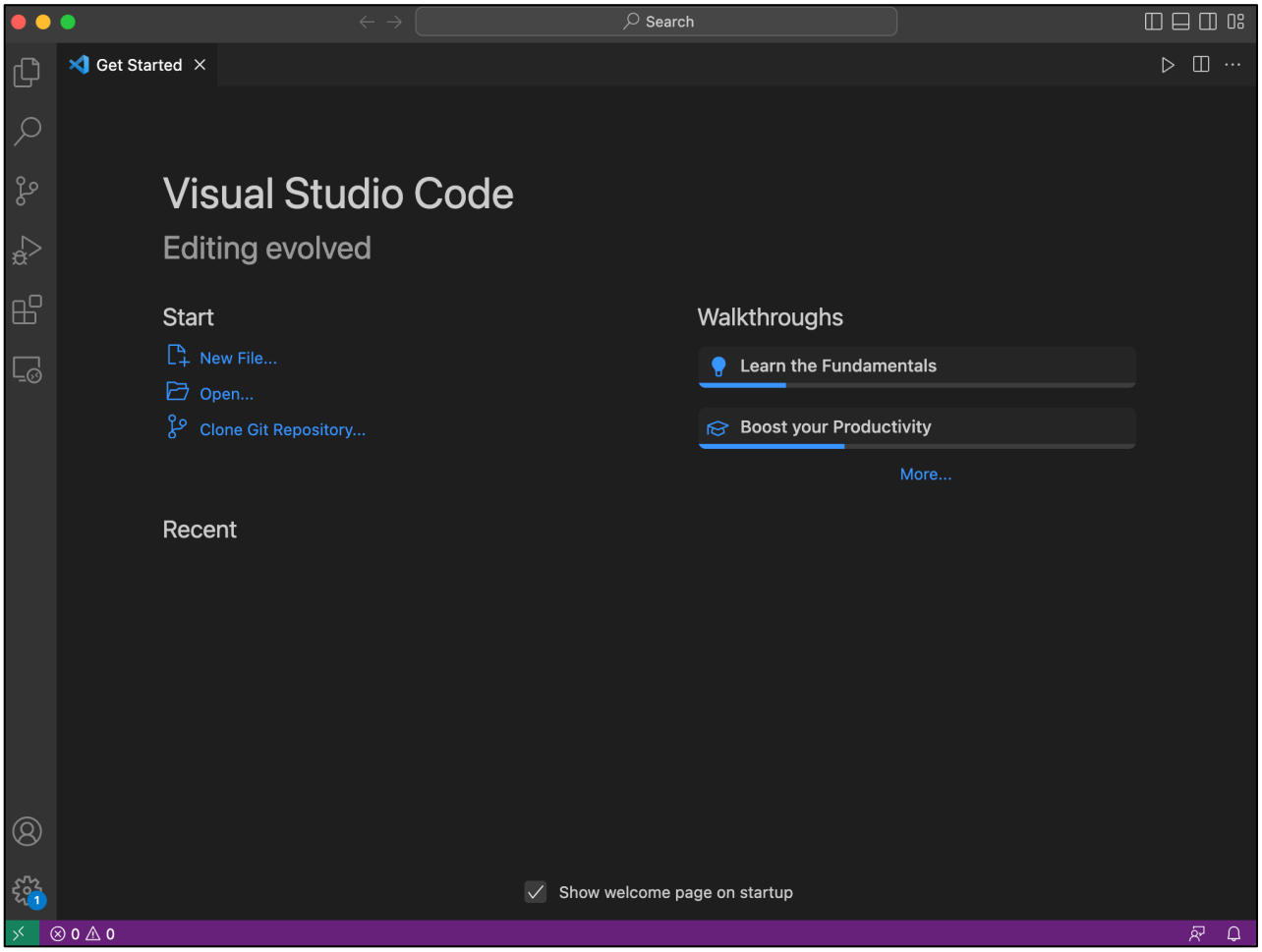
2. Thời gian thực hành

- Thực hành tại lớp: 5 tiết tại phòng thực hành.
- Hoàn thành báo cáo kết quả thực hành: tối đa 13 ngày.

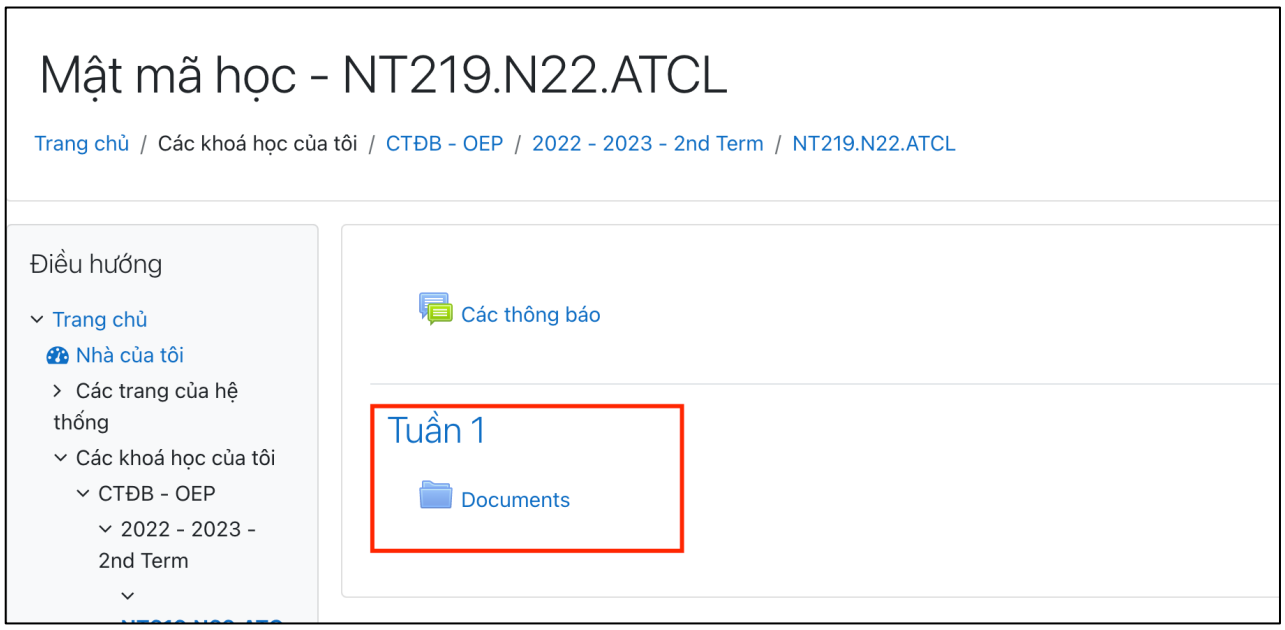
B. CHUẨN BỊ MÔI TRƯỜNG

1. Phần mềm visual studio code

- Tải phần mềm tại: https://code.visualstudio.com



- Thư viện cryptopp được cung cấp tại website môn học



- Tiến hành build thư viện để sử dụng. Truy cập vào thư mục chứa thư viện cryptopp, sau đó gọi lệnh **make**.

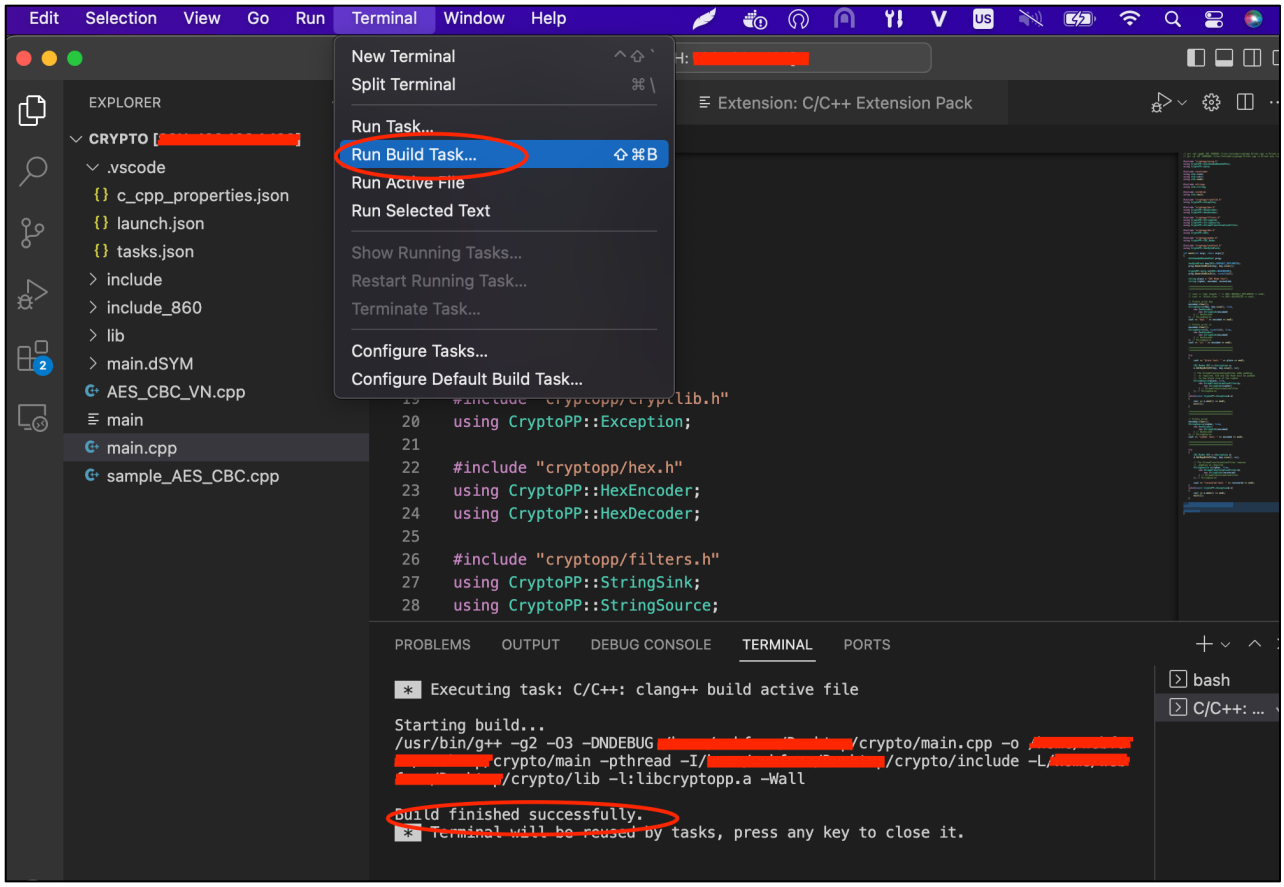
```
ar: creating libcryptopp.a
ranlib libcryptopp.a
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c adhoc.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c test.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c bench1.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c bench2.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c bench3.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c datatest.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c dlltest.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c fipsalgt.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c validat0.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c validat1.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c validat2.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c validat3.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c validat4.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c validat5.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c validat6.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c validat7.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c validat8.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c validat9.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c validat10.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c regtest1.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c regtest2.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c regtest3.cpp
g++ -DNDEBUG -g2 -O3 -fPIC -pthread -pipe -c regtest4.cpp
g++ -o cryptest.exe -g2 -O3 -fPIC -pthread -pipe adhoc.o test.o bench1.o bench2.o bench3.o datatest.o dlltest.o fipsalgt.o validat0.o validat1.o validat2.o v
alidat3.o validat4.o validat5.o validat6.o validat7.o validat8.o validat9.o validat10.o regtest1.o regtest2.o regtest3.o regtest4.o ./libcryptopp.a
```

- Sau khi build xong sẽ xuất hiện 1 tập tin libcryptopp.a. Tập tin này được sử dụng như là thư viện để lập trình các thuật toán mã hoá tiếp theo.
- Tạo file tasks.json trong thư mục .vscode để hướng dẫn vscode build các đoạn code tiếp theo.

.vscode > {} tasks.json > [ ] tasks > {} 0 > [ ] args

```
7       "type": "cppbuild",
8       "label": "C/C++: clang++ build active file",
9       "command": "/usr/bin/g++",
10      "args": [
11          "-g2",
12          "-O3",
13          "-DNDEBUG",
14          "${file}",
15          "-o",
16          "${fileDirname}/${fileBasenameNoExtension}",
17          "-pthread",
18          "-I${workspaceFolder}/include",
19          "-L${workspaceFolder}/lib",
20          "-l:libcryptopp.a",
21          "-Wall"
22      ],
```

- Tiến hành build thử tập tin **sample_DES_CBC.cpp.** Kết quả thông báo **build finished successfully.** là quá trình thêm thư viện thành công.



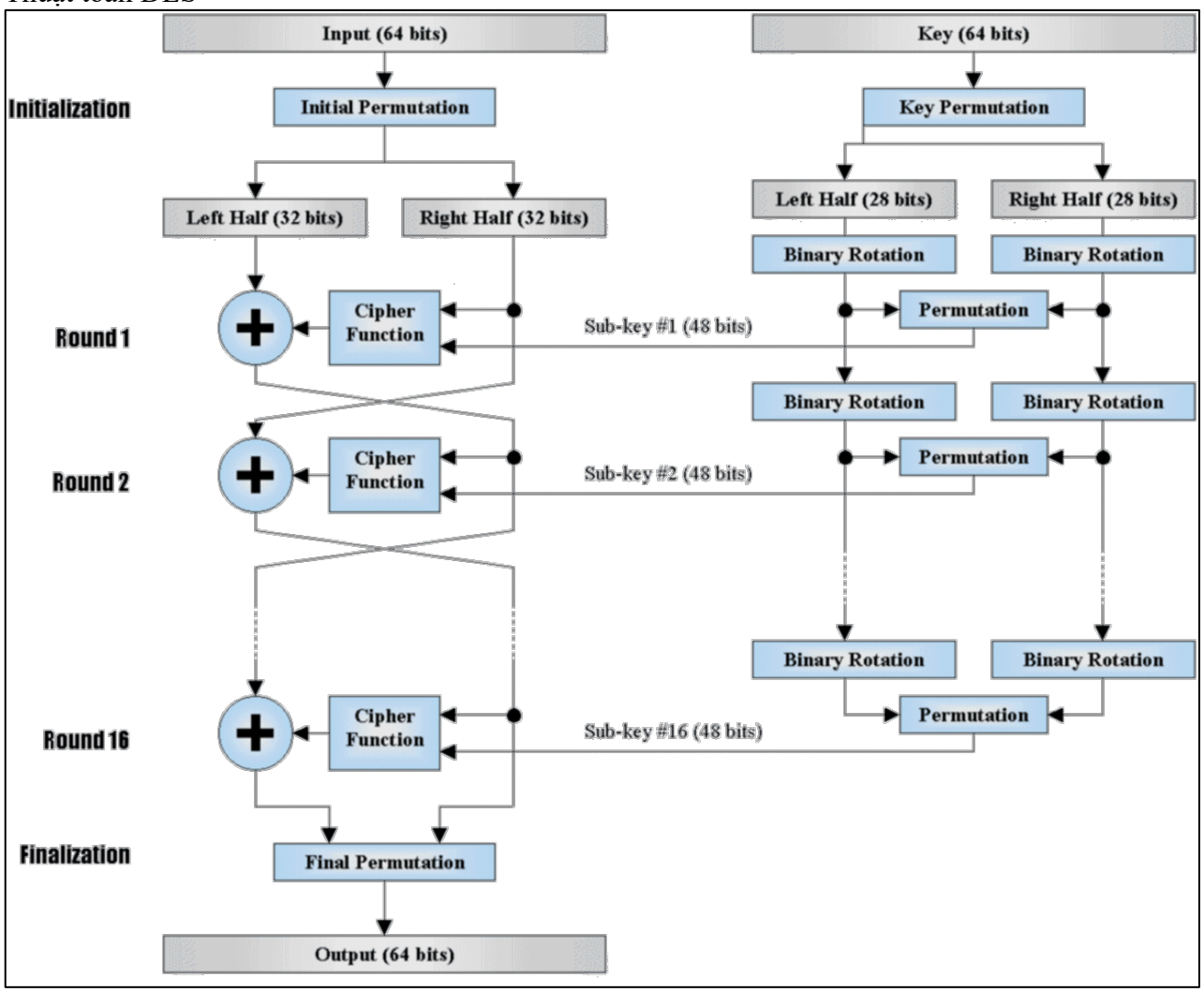
2. Hệ điều hành

- Sử dụng cả hệ điều hành linux và window để kiểm tra thuật toán.

C. THỰC HÀNH

1. Tìm hiểu mã hoá DES, AES sử dụng thư viện cryptopp

Thuật toán DES



a) Mã hoá:

- Như mỗi lược đồ mã hoá khác, thuật toán DES trên cũng có 2 phần đầu vào cho hàm mã hoá: mã rõ để cho việc mã hoá và khoá.
- Trong trường hợp này khoá có chiều dài 56-bit và mã rõ có chiều dài 64-bit

Chậm lại và suy nghĩ 1: Permuted choice 1 đã làm gì để từ 64-bit key thành 56-bit key?

- Nhìn vào bên trái lược đồ, ta có thể thấy tiến trình xử lý mã rõ sẽ bao gồm 3 giai đoạn. Đầu tiên, 64-bit sẽ được truyền vào IP để sắp xếp lại các bit từ đó sản xuất ra đầu vào hoán vị.
- Các giai đoạn sẽ được lặp lại 16 lần với cùng 1 hàm.

- Đầu ra của lần lặp lại cuối cùng là nghịch đảo của hoán vị ban đầu trước khi được đưa ra như đầu ra mã hoá của lược đồ.
- Phần bên phải của lược đồ trình bày cách mà 56-bit của khoá được sử dụng. Đầu tiên khoá được đưa vào hàm hoán vị. Sau đó mỗi vòng, một khoá con sẽ được tạo ra bởi lựa chọn hoán vị và dịch chuyển vòng trái. Hàm hoán vị sẽ là giống nhau ở mỗi vòng, nhưng với khoá con khác nhau được tạo ra bởi dịch chuyển lặp lại của các bit khoá.

b) Giải mã

- Giải mã sẽ sử dụng cùng thuật toán với mã hoá, ngoại trừ quá trình tạo khoá con được làm ngược lại. Thêm vào đó quá trình khởi tạo và kết thúc hoán vị cũng được làm ngược lại

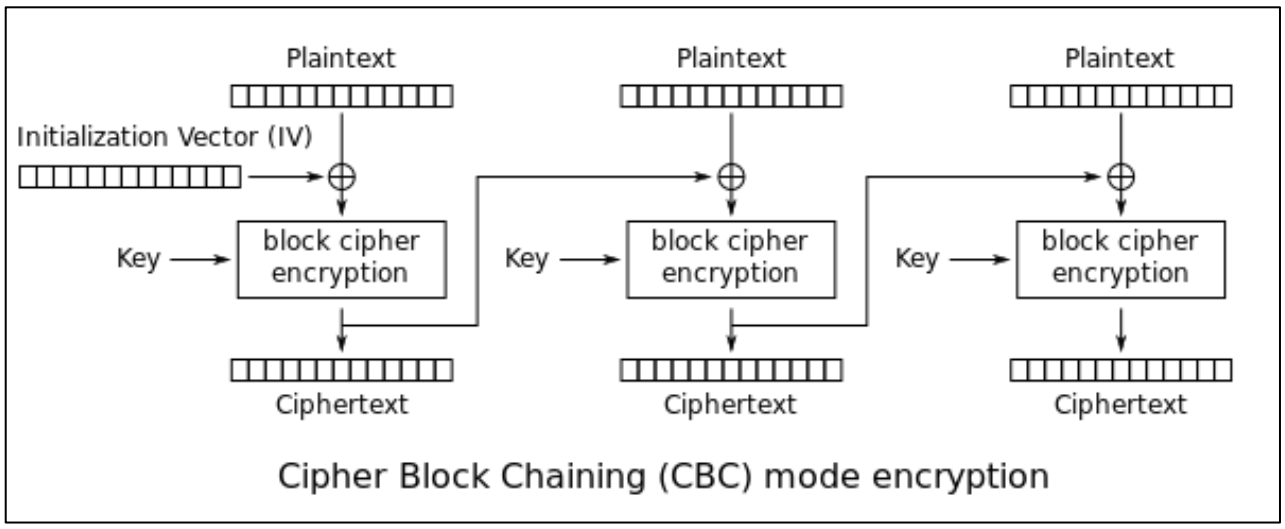
c) Thực hành viết chương trình mã hoá sử dụng thuật toán DES bằng thư viện cryptopp sử dụng mode CBC

- **Bước 1**: Khởi tạo key, iv, plaintext:

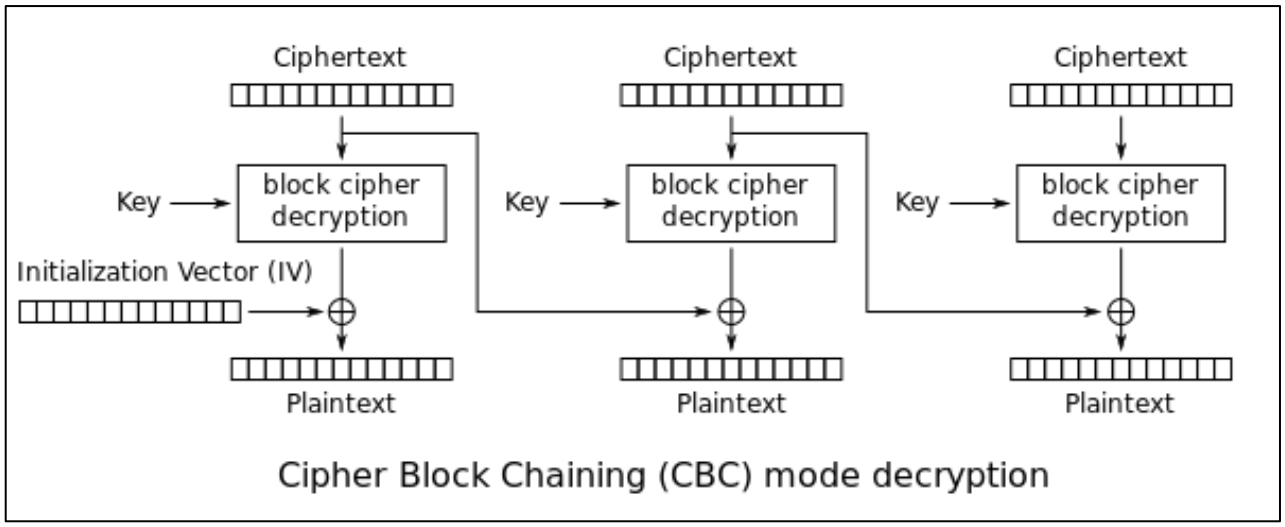
```
AutoSeededRandomPool prng;

SecByteBlock key(DES::DEFAULT_KEYLENGTH);
prng.GenerateBlock(key, key.size());

CryptoPP::byte iv[DES::BLOCKSIZE];
prng.GenerateBlock(iv, sizeof(iv));

string plain = "CBC Mode Test";
string cipher, encoded, recovered;
```

Chậm lại và suy nghĩ 2: DES::DEFAULT_KEYLENGTH và DES::BLOCKSIZE bằng bao nhiêu?

- **Bước 2**: Mã hoá:



- **Bài tập 1:** Debug chương trình và ghi nhận lại hoạt động chính trong quá trình mã hoá của DES với mode CBC
- **Bước 3**: Giải mã:



- **Bài tập 2:** Debug chương trình và ghi nhận lại hoạt động chính trong quá trình giải mã của DES với mode CBC
- Các phần bài tập về lập trình thuật toán DES mode CBC
- **Bài tập 3:** plaintext hỗ trợ đầu vào bao gồm các kí tự thuộc UTF-16
- **Bài tập 4:** Đầu vào plaintext được nhập thủ công vào chương trình.
- **Bài tập 5:** Secret Key và IV nhập vào thủ công từ chương trình

2. Tấn công thuật toán và lược đồ

1. Differential cryptanalysis attack (có thể tìm hiểu thêm và báo cáo)
2. Linear cryptanalysis attack(có thể tìm hiểu thêm và báo cáo)

3. Bài tập luyện tập

- **Bài tập luyện tập 1:** Đánh giá hiệu năng của thuật toán DES với mode CBC
  1. Trường hợp 1: Dữ liệu nhỏ hơn 64-bit
  2. Trường hợp 2: Dữ liệu dạng utf-16
  3. Trường hợp 3: Dữ liệu lớn hơn 1MB
  4. Báo cáo với 2 thông số Cycles Per Byte và MiB/Second. Có thể tham khảo công cụ đánh giá tại https://www.cryptopp.com/wiki/Benchmarks
- **Bài tập luyện tập 2:** Đưa ra điểm yếu của thuật toán DES và viết chương trình tấn công tìm ra được plaintext của thuật toán với các điều kiện sau:
  1. IV, key cố định (giả định là chỉ biết được IV, key là bí mật)
  2. Plaintext có độ dài đủ lớn và cố định.

3. Chỉ cần trình bày logic của chương trình, không cần thực hiện thành công quá trình tấn công.

- **Bài tập luyện tập 3:** Đưa ra điểm yếu của mode CBC và viết chương trình tấn công. (Chỉ cần trình bày logic của chương trình, không cần thực hiện thành công quá trình tấn công)

D. YÊU CẦU & ĐÁNH GIÁ

- Sinh viên tìm hiểu và thực hành theo hướng dẫn, thực hiện theo nhóm đã đăng ký.
- Nộp báo cáo kết quả gồm Code, CSDL được export và chi tiết những việc (Report) mà nhóm đã thực hiện, quan sát thấy và kèm ảnh chụp màn hình kết quả (nếu có); giải thích cho quan sát (nếu có).
- Báo cáo:
  - File .PDF. Tập trung vào nội dung, không mô tả lý thuyết.
  - Đặt tên theo định dạng: [Mã lớp]-LabX_MSSV1.
  - Ví dụ: [NT219.K11.ANTN.1]-Lab1_1852xxxx-.
  - Nếu báo cáo có nhiều file, nén tất cả file vào file .ZIP với cùng tên file báo cáo.
  - Nộp file báo cáo trên theo thời gian đã thống nhất tại courses.uit.edu.vn.

Bài sao chép, trễ, … sẽ được xử lý tùy mức độ vi phạm.

HẾT

Chúc các bạn hoàn thành tốt!