

- cd vào thư mục Downloads, sau đó:

*git clone [https://github.com/NgocTaif/destego\\_dct\\_code](https://github.com/NgocTaif/destego_dct_code)*

- cd vào thư mục /home/student/labtainer/labtainer-student gõ lệnh:

*imodule file:///home/student/Downloads/destego\_dct\_code/destego\_dct\_code.tar*

- Chạy lab: labtainer -r destego\_dct\_code
- Bài labs gồm hai terminal alice và bob.
- Đầu tiên, bật ssh trên cả hai terminal alice và bob: *sudo systemctl start ssh* (nếu có lỗi thì *sudo systemctl stop xinetd* rồi sau đó start lại ssh).

- Trên terminal alice, gửi hai file sang terminal bob bằng lệnh (mật khẩu: ubuntu):

*scp encrypted\_file.bin recovered\_img.png ubuntu@174.20.0.10:/home/ubuntu*

- Trên terminal bob, thực hiện *ls* kiểm tra đã có hai file trên chưa.

- **Task 1:**

- Trên terminal bob, gõ lệnh:  
*python3 decrypt\_message.py encrypted\_file.bin private\_key.pem*
- Nếu hiển thị: “*Giai ma thanh cong: decrypted\_file.txt*” thì thành công, sau đó thử đọc thử file *decrypted\_file.txt* để biết vị trí các ma trận điểm ảnh được giấu bit, và số lượng bit được giấu trong mỗi ma trận đó: *nano decrypted\_file.txt*

- **Task 2:**

- Thực hiện chuyển ảnh (*recovered\_img.png*) về một ma trận điểm ảnh:  
*python3 preprocessor/create\_img\_matrix.py recovered\_img.png*
- Nếu hiển thị: “Chuyen thanh cong ma tran diem anh: 'image\_matrix.txt'.” thì thành công, hãy thử đọc file *image\_matrix.txt*: *nano image\_matrix.txt*

- **Task 3:**

- Thực hiện trích xuất các ma trận điểm ảnh có kích thước 8x8 từ ma trận điểm ảnh tổng trong *image\_matrix.txt*
- Vị trí các ma trận 8x8 được ghi trong *decrypted\_file.txt*
- Trong bài này, ta sẽ thực hiện trích xuất 6 ma trận 8x8 tại các vị trí: (0,0), (0,1), (0,2), (1,0), (1,1), (1,2).

- Trên terminal bob, gõ các lệnh:

```
python3 extract_matrix.py image_matrix.txt 0 0
```

```
python3 extract_matrix.py image_matrix.txt 0 1
```

```
python3 extract_matrix.py image_matrix.txt 0 2
```

```
python3 extract_matrix.py image_matrix.txt 1 0
```

```
python3 extract_matrix.py image_matrix.txt 1 1
```

```
python3 extract_matrix.py image_matrix.txt 1 2
```

- Nếu hiển thị: “Trích xuất thành công ma trận tại vị trí: ...” thì trích xuất thành công.

#### - Task 4:

- Thực hiện biến đổi DCT và lượng tử hoá lại các ma trận đã trích xuất trước đó.

- Trên terminal bob, gõ các lệnh sau để biến đổi DCT:

```
python3 dct_transform/dct_discrete_cosine.py extracted_matrix/matrix_8x8_vitri_0_0.txt
```

```
python3 dct_transform/dct_discrete_cosine.py extracted_matrix/matrix_8x8_vitri_0_1.txt
```

```
python3 dct_transform/dct_discrete_cosine.py extracted_matrix/matrix_8x8_vitri_0_2.txt
```

```
python3 dct_transform/dct_discrete_cosine.py extracted_matrix/matrix_8x8_vitri_1_0.txt
```

```
python3 dct_transform/dct_discrete_cosine.py extracted_matrix/matrix_8x8_vitri_1_1.txt
```

```
python3 dct_transform/dct_discrete_cosine.py extracted_matrix/matrix_8x8_vitri_1_2.txt
```

- Nếu hiển thị: “Đã biến đổi DCT: 'extracted\_matrix/matrix\_8x8\_vitri\_...txt'.” thì thành công.

- Tiếp tục, gõ các lệnh sau để lượng tử hoá các ma trận trên:

```
python3 dct_transform/dct_quantized.py extracted_matrix/matrix_8x8_vitri_0_0.txt
```

```
python3 dct_transform/dct_quantized.py extracted_matrix/matrix_8x8_vitri_0_1.txt
```

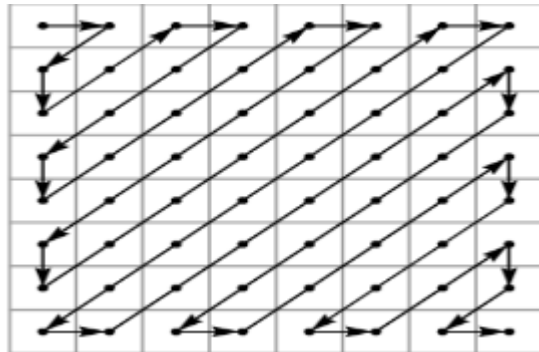
```
python3 dct_transform/dct_quantized.py extracted_matrix/matrix_8x8_vitri_0_2.txt
```

```
python3 dct_transform/dct_quantized.py extracted_matrix/matrix_8x8_vitri_1_0.txt
```

```
python3 dct_transform/dct_quantized.py extracted_matrix/matrix_8x8_vitri_1_1.txt
```

```
python3 dct_transform/dct_quantized.py extracted_matrix/matrix_8x8_vitri_1_2.txt
```

- Nếu hiển thị: “Da luong tu hoa: 'extracted\_matrix/matrix\_8x8\_vitri\_....txt” thì thành công.
- Sau khi đã biến đổi DCT và lượng tử hoá thành công 6 ma trận trên, hãy thực hiện đọc các ma trận này và xem các bit đã được giấu trong đó.
- Trong bài này, mỗi ma trận trong 6 ma trận trên sẽ được thay thế 4 bit cuối theo ma trận zigzag bằng 4 bit nhị phân của thông điệp giấu.
- Ví dụ như theo chiều ma trận zigzag thì ta sẽ lấy bốn bit cuối của ma trận trích xuất (đã biến đổi DCT, lượng tử hoá) như sau: 0100.



```

ubuntu@bob: ~
File Edit View Search Terminal Help
GNU nano 4.8 extracted_matrix/matrix_8x8_vitri_0_0.txt
14 7 -4 0 -1 0 0 0
7 2 -1 0 0 0 0 0
-2 -1 0 0 0 0 0 0
-1 0 0 0 0 0 0 0
-1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 0
[ Read 8 lines ]
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Paste Text ^T To Spell ^_ Go To Line

```

- Kết quả ghép 4 bit thu được từ 6 ma trận ta sẽ được dải bit 24 bit như sau:

010000010101010001010100

#### - Task 5:

- Giải mã dải 24 bit thu được trước đó sang ASCII để thu được thông điệp:  
*python3 take\_data.py <24 bits thu được>*
  - Nếu hiển thị: “The decoded message is: ATT” thì thành công.
- Kiểm tra checkwork.