

# BÁO CÁO THỰC HÀNH

Môn học: Pháp chứng kỹ thuật số

Lab 3: Steganography & Steganaly

GVHD: Đoàn Minh Trung

## 1. THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT334.021.ATTN

STT	Họ và tên	MSSV	Email
1	Phạm Ngọc Thơ	21522641	21522641@gm.uit.edu.vn
2	Hà Thị Thu Hiền	21522056	21522056@gm.uit.edu.vn

## 2. NỘI DUNG THỰC HIỆN:<sup>1</sup>

STT	Công việc	Kết quả tự đánh giá
1	Kịch bản 1-4 (đã báo cáo ở lớp)	100%
2	Kịch bản 5	100%
3	Kịch bản 6	100%
4	Kịch bản 7	100%
5	Kịch bản 8	100%
6	Kịch bản 9	100%
7	Kịch bản 10	100%

Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.

<sup>1</sup> Ghi nội dung công việc, các kịch bản trong bài Thực hành

# BÁO CÁO CHI TIẾT

## 1. Kịch bản 5:

- Tài nguyên thực hiện: qn001.jpg


- Yêu cầu – Gợi ý: Tìm thông điệp (flag) được ẩn giấu. Thông tin flag liên quan đến Đội tuyển bóng đá nam Việt Nam.

Đáp án:

- Để tìm flag giấu trong file, có thể thực hiện bằng chức năng Seek của Jphswin, tuy nhiên cần password. Do đó em sẽ dùng tool stekbread để scan password trước:

```
C:\Users\NgThow\Downloads\Lab3_Forensic\stegdetect04_session03>stegbreak -r rule
s.ini -f meddict.dic qn001.jpg
Corrupt JPEG data: bad Huffman code
Loaded 1 files...
qn001.jpg : negative
Processed 1 files, found 0 embeddings.
Time: 222 seconds: Cracks: 2423746, 10917.8 c/s
```

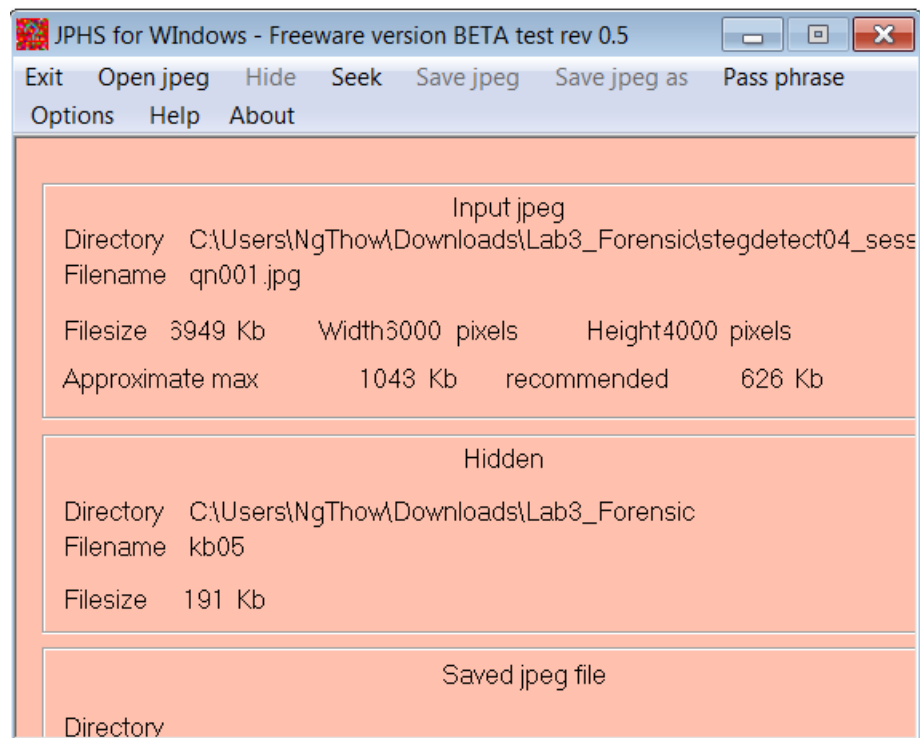
Wordlist meddict.dic không in ra được thông tin gì cả. Em tra ChatGPT để tìm wordlist khác:

 **ChatGPT**

Dưới đây là một số wordlist phổ biến được sử dụng trong các tấn công từ điển và kiểm tra mật khẩu:

1. RockYou: Một trong những danh sách từ điển phổ biến nhất, được tạo ra từ dữ liệu bị rò rỉ từ dịch vụ trò chơi trực tuyến RockYou.
2. SecLists: Một bộ sưu tập lớn các danh sách từ điển và các tệp từ điển khác, bao gồm các từ điển chuyên biệt cho các mục đích khác nhau như mật khẩu, tên người dùng, và các từ điển dành cho tấn công steganography.
3. CrackStation: Một dự án mã nguồn mở cung cấp một tập hợp lớn các từ điển cho việc tấn công từ điển.
4. Hashcat: Hashcat là một công cụ mạnh mẽ cho tấn công từ điển và tấn công brute-force. Nó đi kèm với một số wordlist được cung cấp sẵn, bao gồm cả rockyou và một số từ điển khác.
5. Probable-Wordlists: Bộ sưu tập từ điển phổ biến được sử dụng cho các cuộc tấn công từ điển, bao gồm các từ điển dựa trên ký tự thông thường, tên riêng, và các từ dựa trên ngôn ngữ cụ thể.
6. CIRT Word Lists: Một bộ sưu tập các danh sách từ điển được tạo ra bởi CIRT (Computer Incident Response Team) của đại học Carnegie Mellon, bao gồm các từ điển chuyên biệt cho các mục đích tấn công khác nhau.

- Như đề xuất, em sẽ dùng rockyou, kết quả scan chỗ mục password bị trống, tức là Seek không cần pass, chuyển sang Jphswin:



- Em seek và lưu kết quả vào file Kb05, mở trên word được đoạn văn bản phía dưới, tuy nhiên đọc thì flag ở đâu omg@@:

### Tuyển Việt Nam luôn biết làm chủ thế trận

Trang bóng đá *Okezone Bola* miêu tả chi tiết về trận đấu diễn ra trên sân Hàng Đẫy: "Đội tuyển Việt Nam nhập cuộc khá tốt với chủ đích tấn công trực diện. Kết quả là đội chủ nhà hoàn toàn nắm giữ thế trận trong khi Campuchia phụ thuộc nhiều vào những pha phản công".

"Mặc dù làm chủ trận đấu, trên thực tế, đoàn quân của HLV Park Hang-seo lại gặp khó khăn khi không thể khoan thủng hàng phòng ngự đối phương ở đầu hiệp 1. Phải đến phút thứ 39, nỗ lực của Việt Nam mới được đền đáp khi Tiến Linh tận dụng lợi thế từ đường chuyển của Trọng Hoàng để xâm nhập vào vòng cấm và đánh đầu ghi bàn", tờ báo của Indonesia bình luận.

Những bàn thắng tiếp theo đến như một lẽ tất nhiên khi đội chủ nhà cội bô được áp lực ghi bàn và thoải mái dẫn dắt trận đấu. "Các học trò của ông Park chỉ mất thêm 3 phút để Quang Hải ghi tên mình lên bảng tỷ số sau khi tận dụng pha kiến thiết của Nguyễn Phong Hồng Duy", tờ *Bola Sport* viết.



- Thử kiểm tra xem có file ẩn trong file doc này không bằng **binwalk**:

```
ngthow@master-node:~$ binwalk kb05.docx
```

DECIMAL	HEXADECIMAL	DESCRIPTION
0	0x0	Zip archive data, at least v2.0 to extract, compressed size: 359, uncompressed size: 1363, name: [Content_Types].xml
928	0x3A0	Zip archive data, at least v2.0 to extract, compressed size: 239, uncompressed size: 590, name: _rels/.rels
1728	0x6C0	Zip archive data, at least v2.0 to extract, compressed size: 3915, uncompressed size: 19670, name: word/document.xml
5690	0x163A	Zip archive data, at least v2.0 to extract, compressed size: 264, uncompressed size: 949, name: word/_rels/document.xml.rels
6276	0x1884	Zip archive data, at least v1.0 to extract, compressed size: 178935, uncompressed size: 178935, name: word/media/image1.jpg
185262	0x2D3AE	Zip archive data, at least v2.0 to extract, compressed size: 1538, uncompressed size: 7076, name: word/theme/theme1.xml
186851	0x2D9E3	Zip archive data, at least v2.0 to extract, compressed size: 1118, uncompressed size: 3160, name: word/settings.xml
188016	0x2DE70	Zip archive data, at least v2.0 to extract, compressed size: 3267, uncompressed size: 31584, name: word/styles.xml
191328	0x2EB60	Zip archive data, at least v2.0 to extract, compressed size: 471, uncompressed size: 2670, name: word/webSettings.xml
191849	0x2ED69	Zip archive data, at least v2.0 to extract, compressed size: 576, uncompressed size: 1968, name: word/fontTable.xml
192473	0x2EFD9	Zip archive data, at least v2.0 to extract, compressed size: 386, uncompressed size: 747, name: docProps/core.xml
193170	0x2F292	Zip archive data, at least v2.0 to extract, compressed size: 479, uncompressed size: 992, name: docProps/app.xml
194731	0x2F8AB	End of Zip archive, footer length: 22

- File doc thực tế được nén theo định dạng zip. Do đó để mở được nội dung bên trong, em sẽ đổi đuôi file thành .zip và extract. Thu được:



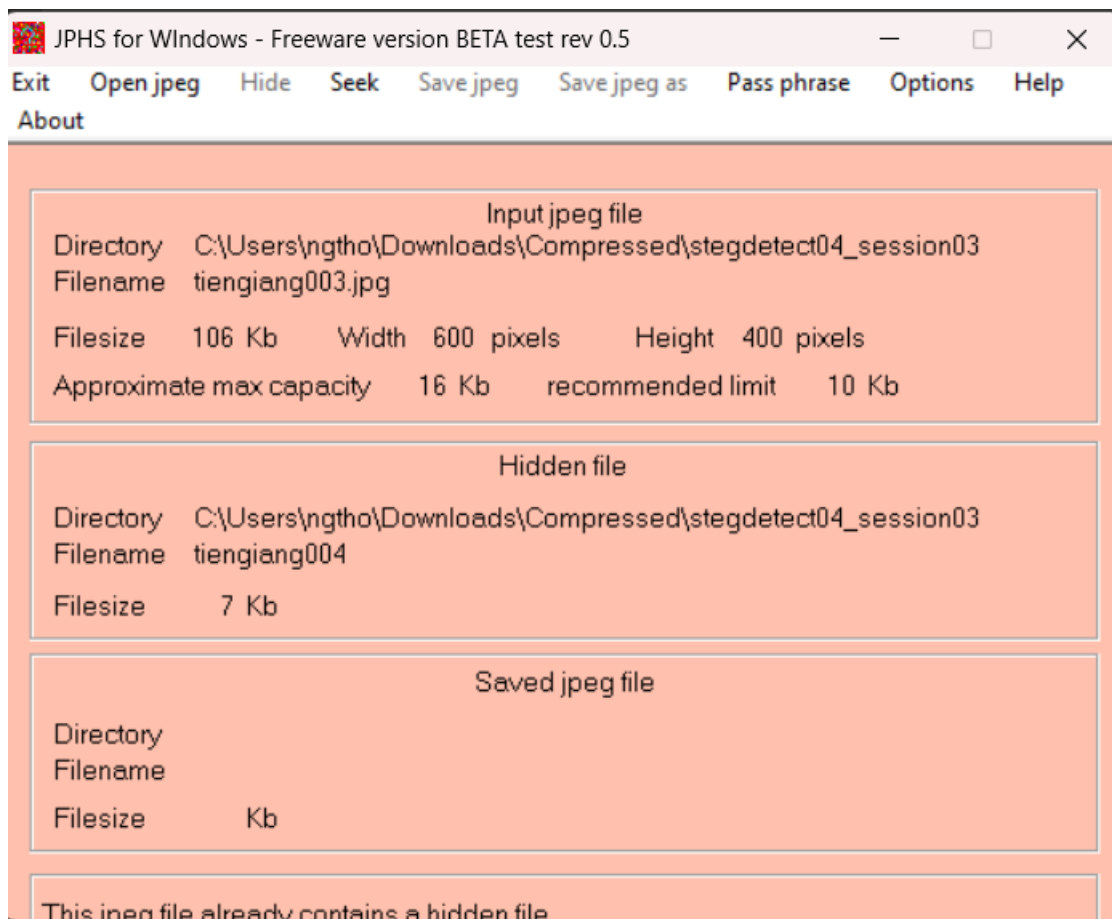


**Kịch bản 06. Thực hiện phân tích:**

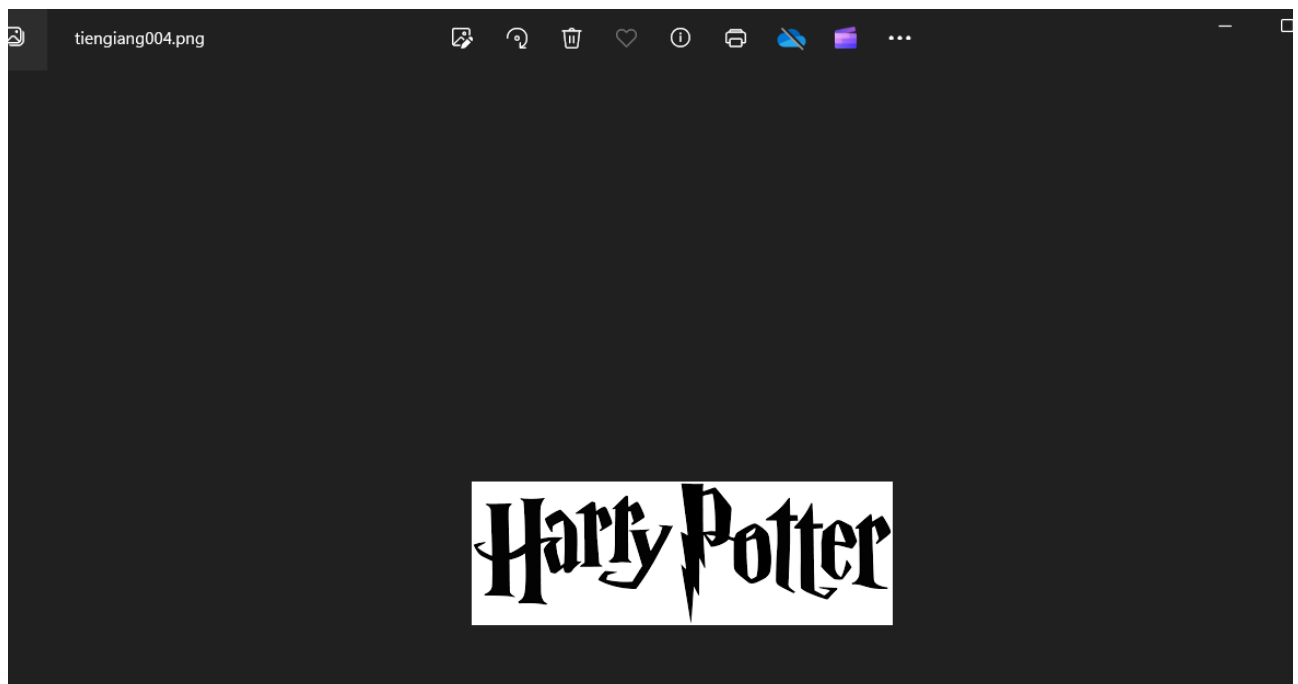
- Đáp án:**

- ```
C:\Users\ngtho\Downloads\Compressed\stegdetect04_session03>.\stegbreak.exe -r rules.ini -f .\rockyou.txt .\tiengiang003.jpg
Loaded 1 files...
.\tiengiang003.jpg : jphide[v5]()
Processed 1 files, found 1 embeddings.
Time: 0 seconds: Cracks: 4751,      Inf c/s
```

- Dùng JPHS để trích xuất file ảnh, do không có password nên Seek sẽ để trống phần password:



- File trích xuất được em lưu là *tiengiang004*, sau khi thử thay đổi file với các đuôi khác nhau, thì đến png sẽ mở được file:



- Dùng *strings* để tìm chuỗi trong file, tìm được chuỗi sau:



```

Z~n"S
.[7d
TN3%d
`RN#
' * '{
j5b
M_<B
;9b0
0j2~
;6h8
%0L.
MS: `
wB?/
0L=C
dE>V
?j)~
~<md
7jqX
#-6x
Zcl,3
AWWE
@kjhj
uyDY$
Gm#Lm
i)7n
;DD
^nG L
P[]*s
2~jv
OCQDZ
- g{
Pllp
u} Ic
jg3U
c)U>
IEND
wherE shOuld onE ReaLly looK for thIS flag
ngthow@master-node:~$

```

- Đề bài có gợi ý “Thuật toán dùng tìm ra flag liên quan đến việc thay thế các kí tự trong chuỗi ban đầu thành chuỗi chỉ gồm 2 kí tự a và b.”, nên đây có thể là kỹ thuật Bacon Cipher chỉ biểu diễn dưới dạng A và B. Nếu “A” là chữ in thường, “B” là chữ in hoa thì được:

AAAAB BABBAA AAB BAABAA ABAB ABA ABAA AAAA

- Dùng tool để giải mã, được flag là: BYDELTA

★ SEARCH A TOOL ON DCode BY KEYWORDS:  
e.g. type 'caesar'

★ BROWSE THE [FULL DCode TOOLS' LIST](#)

results

| ↑↓               | ↑↓      |
|------------------|---------|
| A=A, B=B (abc 1) | BYDELTA |
| A=A, B=B (abc 2) | SA      |
| A=B, B=A (abc 1) | XO?     |
| A=B, B=A (abc 2) | ?J??VN? |

### BACONIAN CIPHER DECODER

★ BACON CIPHERTEXT ?

AAAAB BABBAA AAB BAABAA ABAB ABA ABAA AAAA

▶ DECRYPT BACON



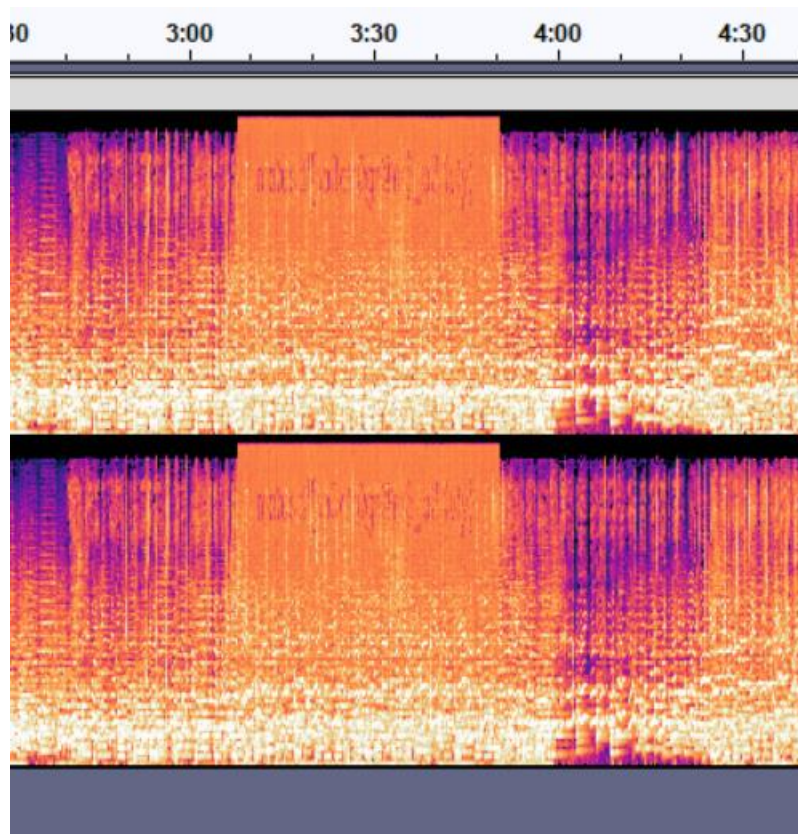
### 3. Kịch bản 7:

#### Kịch bản 07. Thực hiện phân tích, tìm thông tin ẩn giấu:

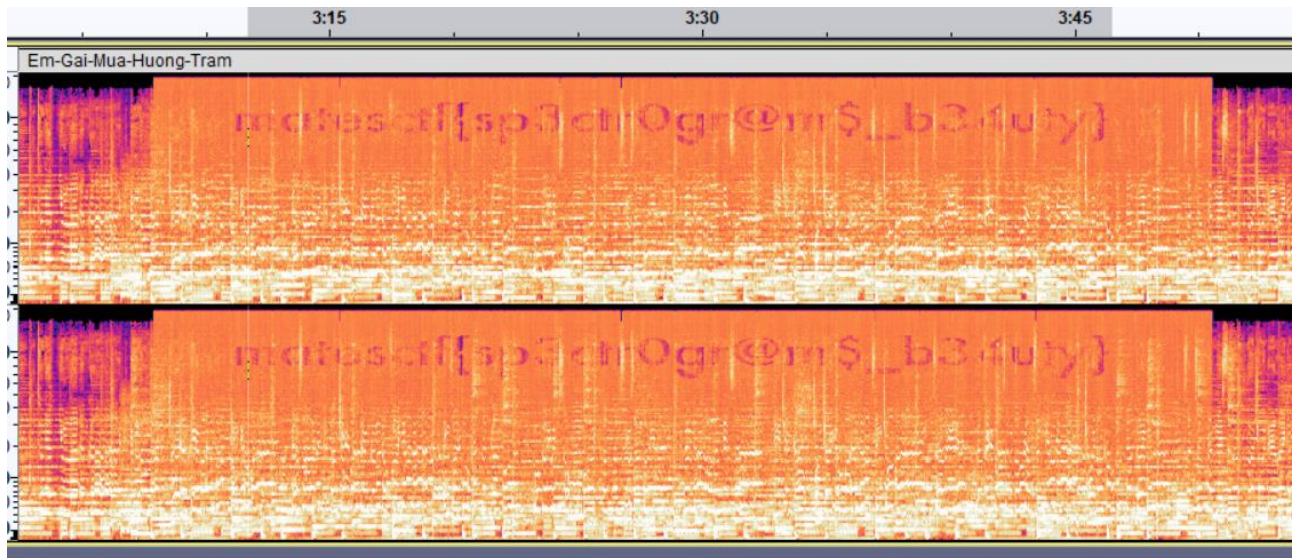
- Tài nguyên: kb07-res (Tìm thông tin ẩn giấu trong Em-Gai-Mua-Huong-Tram.mp3, capture-the-flag.png)

Đáp án:

- Để kiểm tra file âm thanh, em dùng tool Audacity. Mở file sau đó chọn chế độ Spectrogram, ta sẽ thấy beat hiện lên dưới dạng biểu đồ màu sắc. Có một đoạn của biểu đồ dường như là chữ xuất hiện:



- Flag: `matesctf{sp3ctr0gr@m$_b34uty}`



#### 4. Kịch bản 08: Thực hiện phân tích, tìm thông tin ẩn giấu:

##### Kịch bản 08. Thực hiện phân tích, tìm thông tin ẩn giấu:

- Tài nguyên: LoveLetter.txt
- Yêu cầu – Gợi ý: Có gì đó đáng ngờ trong bức thư tình mà bạn đang đọc. Nhân viên điều tra cũng nghĩ rằng bức thư tình này chứa một thông điệp bí mật nào đó. Hãy tìm thông điệp được ẩn giấu (flag). Flag có dạng "FLAG-\*
- Link CTF: <https://ringzer0ctf.com/challenges/215>

- Bây giờ, ta có tài nguyên là LoveLetter.txt thì chúng ta sẽ xem trong đó có gì trước:

```
(hahien@kali) ~/Downloads
$ cat LoveLetter.txt
I went to the park today, saw a lot of fish. Fish are cool, but they aren't my favorite animal!! The monkey is a good animal, so is the Blue-Tongued Skink, but I rarely get to see those at the park! All of this makes me sad, but just encourages me to travel more. I'll start researching where in the world I can see these animals in their natural habitat and start visiting them! Sounds like a good time, I'll update here with my plans. It might be a long while though, because I get so busy with work and never have time to do the actual things I want to do! Oh to be me, and to never go out for working. Well, at least the people at my company are nice! Working there is fun, and I do get to do some things with friends through work, but I still wish I could make friends with those monkeys and skinks! Well, I guess it's official: I shall travel! Not just the rant from this blog post, but an actual thing I will do. Well, I'll show you guys all the pictures anyway. Did you know that a monkey is either going to be a Cercopithecoid or a Platyrrhine? It's true! and there are 26 species of monkey that are known. Sure, is a lot of them! But skinks are also cool, there are over 1200 different species of skink! Skins are lizards, but they look more like snakes with legs to me! But I guess since skinks have a tail and snakes don't... Oh I don't know! I love animals of all kinds, can't even pick favorites. I'm sorry fish, you guys are good animals too. Ha ha, alright, I'll stop my ranting.

--End journal entry

(hahien@kali) ~/Downloads
$
```

- ⇒ Ta nhận thấy thông tin về dấu cách không bình thường. Có thể có thứ chúng ta cần khai thác ở những dấu cách bình thường và những dấu cách đặc biệt trên.
- ⇒ Dưới đây là một chương trình Python để giải mã một chuỗi ký tự được đánh dấu, trong đó dấu cách các ký tự đặc biệt được đánh dấu là 1 và các dấu cách thông thường được đánh dấu là 0. Sau đó, chương trình thực hiện chuyển đổi chuỗi nhị phân thành dạng decimal và sử dụng hàm unhexlify từ thư viện binascii để chuyển đổi lại thành chuỗi ban đầu.

```

1 import binascii # Import thư viện binascii để sử dụng hàm unhexlify()
2
3 binary_string = "" # Khởi tạo chuỗi rỗng để lưu trữ chuỗi nhị phân
4
5 with open("LoveLetter.txt", "r", encoding="ISO-8859-1") as file: # Mở file "LoveLetter.txt" ở chế độ
6     for character in file.read(): # Đọc từng ký tự trong file
7         if character == ' ': # Nếu ký tự là dấu cách
8             binary_string = binary_string + "0" # Thêm "0" vào chuỗi nhị phân
9         if ord(character) == 160: # Nếu ký tự có mã Unicode là 160 (khoảng trắng không phải ASCII)
10            binary_string = binary_string + "1" # Thêm "1" vào chuỗi nhị phân
11
12 bin_format = '0b' + binary_string # Chuỗi nhị phân định dạng
13
14 dec_format = int(bin_format, 2) # Chuyển đổi chuỗi nhị phân thành giá trị số nguyên
15
16 result = binascii.unhexlify('%x' % dec_format) # Giải mã giá trị số nguyên thành chuỗi và lưu vào biến
17
18 print(result) # In ra kết quả giải mã

```

- Chạy code và xem kết quả:

```

(hahien@kali)-[~/Downloads]
$ code .
(Message from Kali developers)
code is not the binary you may be expecting.
You are looking for \"code-oss\"
Starting code-oss for you...

[main 2024-04-13T17:16:58.316Z] update#setState disabled
[main 2024-04-13T17:16:58.318Z] update#ctor - updates are disabled as there is no update UR
[13491:0414/011700.408876:ERROR:command_buffer_proxy_impl.cc(128)] ContextResult::kTransient
^C[main 2024-04-13T17:17:37.850Z] Extension host with pid 13597 exited with code: 0, signal
[main 2024-04-13T17:17:37.875Z] [UtilityProcess id: 1, type: fileWatcher, pid: 13577]: cras

(hahien@kali)-[~/Downloads]
$ ls
exploit-LoveLetter.py  LoveLetter.txt  thecatreturns.mp4.crdownload  transmission.png

(hahien@kali)-[~/Downloads]
$ python exploit-LoveLetter.py
b'FLAG-3b6f70fcf070009561f5276fe98fc9c6'

(hahien@kali)-[~/Downloads]
$

```

⇒ FLAG: b'FLAG-3b6f70fcf070009561f5276fe98fc9c6'

## 5. Kịch bản 09: Thực hiện phân tích, tìm thông tin ẩn giấu:

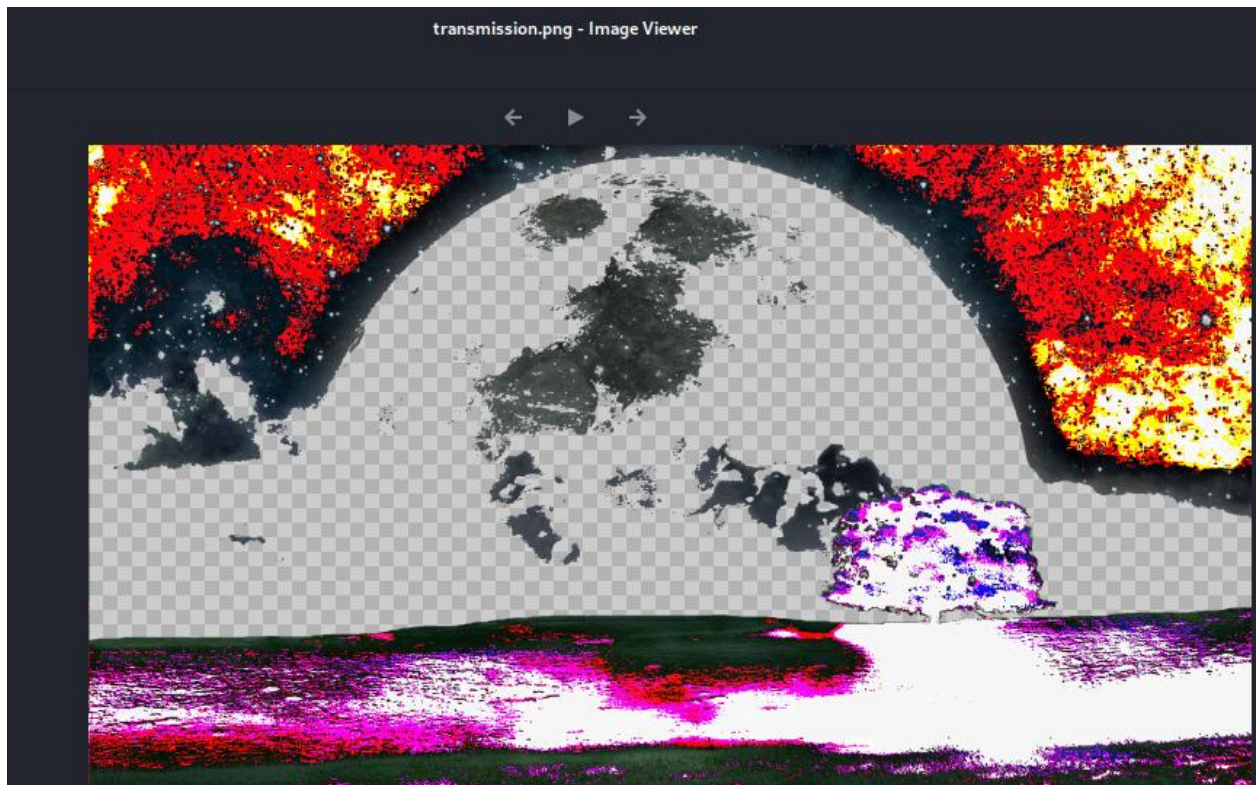
### Kịch bản 09. Thực hiện phân tích, tìm thông tin ẩn giấu:

- Tài nguyên: transmission.png
- Yêu cầu – Gợi ý: Tìm thông điệp được ẩn giấu bằng các công cụ đã học trong buổi này.

**Đáp án:**

- Đầu tiên, mở xem tài nguyên ảnh transmission xem sao:





- ⇒ Ta nhận thấy rằng, màu sắc của ảnh này chưa chính xác là màu ảnh bình thường được biểu diễn theo 3 màu RGB (Red, Green, Blue).
- ⇒ Bây giờ ta sẽ thực hiện chuyển đổi ảnh sang định dạng màu RGB bằng code python sau đây:

```
convert_image.py - Downloads - Code - OSS
File Edit Selection View Go Run Terminal Help
EXPLORER
  DOWNLOADS
    convert_image.py
    exploit-LoveLetter.py
    LoveLetter.txt
    thecatreturns.mp4.crdownload
    transmission.png
  Welcome
  exploit-LoveLetter.py
  convert_image.py
  convert_image.py
    1 from PIL import Image
    2
    3 image = Image.open('transmission.png').convert('RGB')
    4
    5 image.save('transmission-RGB.png')
```

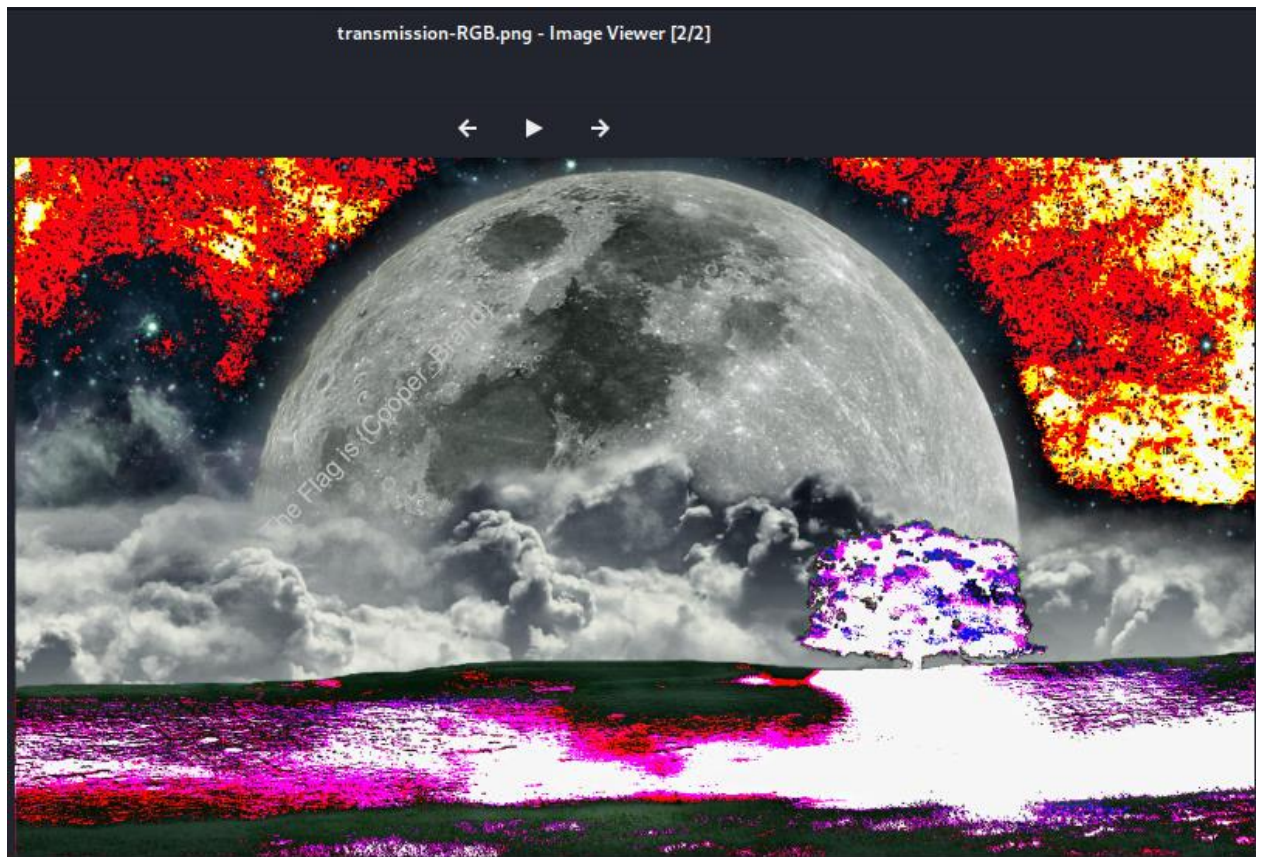
```
(hahien@kali) - [~/Downloads]
$ ls
convert_image.py  exploit-LoveLetter.py  LoveLetter.txt  thecatreturns.mp4.crdownload  transmission.png

(hahien@kali) - [~/Downloads]
$ python convert_image.py

(hahien@kali) - [~/Downloads]
$ ls
convert_image.py  exploit-LoveLetter.py  LoveLetter.txt  thecatreturns.mp4.crdownload  transmission.png  transmission-RGB.png

(hahien@kali) - [~/Downloads]
$
```

- Bây giờ, hãy cùng xem ảnh mới được convert xem có gì khác biệt:



⇒ FLAG: {Cooper\_Brand}

6. Kịch bản 10: Thực hiện phân tích, tìm thông tin ẩn giấu:

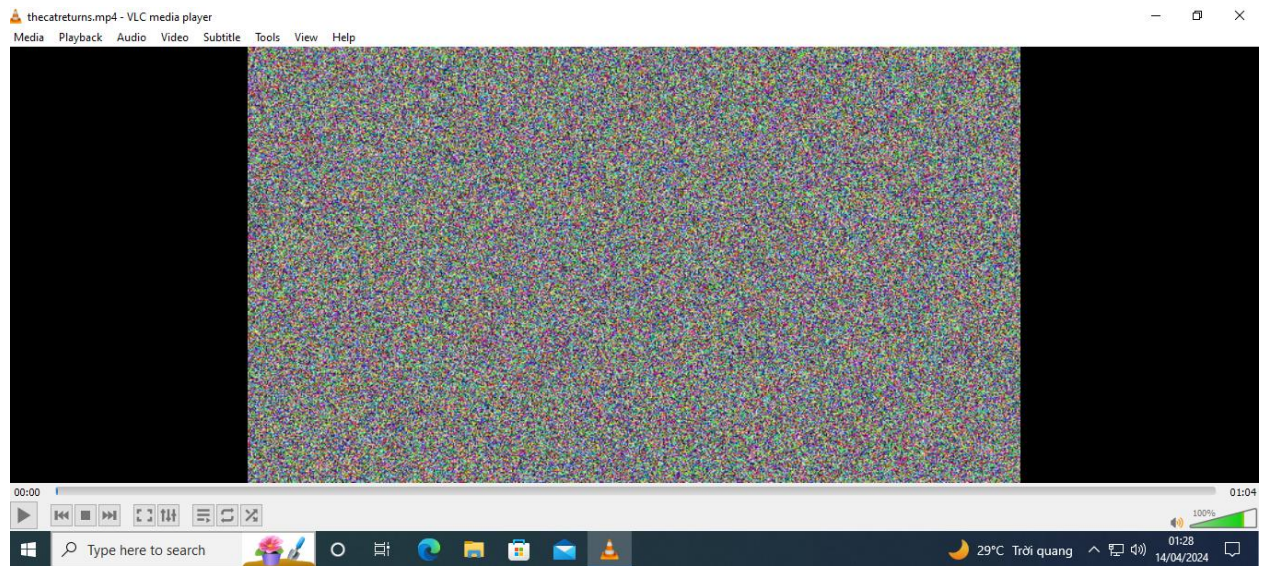
**Kịch bản 10. Thực hiện phân tích, tìm thông tin ẩn giấu:**

- Tài nguyên: thecatreturns.mp4
- Yêu cầu – Gợi ý: Tìm sự khác biệt giữa các khung hình (frame) trong đoạn phim đã cho. Chuyển nội dung đoạn phim thành các khung hình để phân tích. Công cụ ffmpeg, Image].

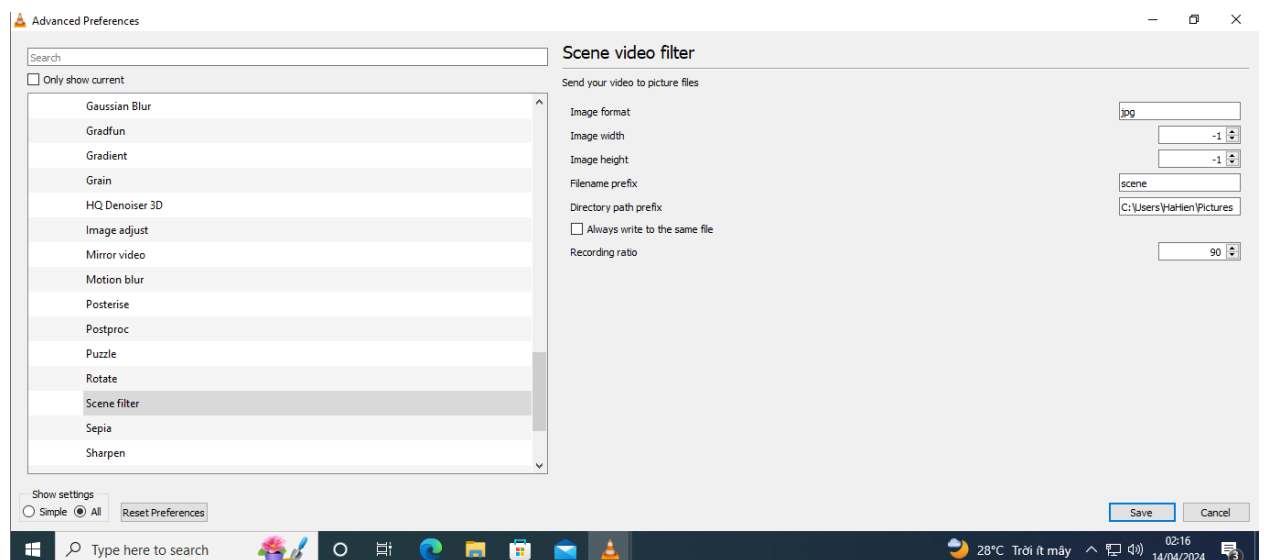
*Đáp án:*

- Đầu tiên ta sẽ sử dụng VLC tạo ra các frame cho video

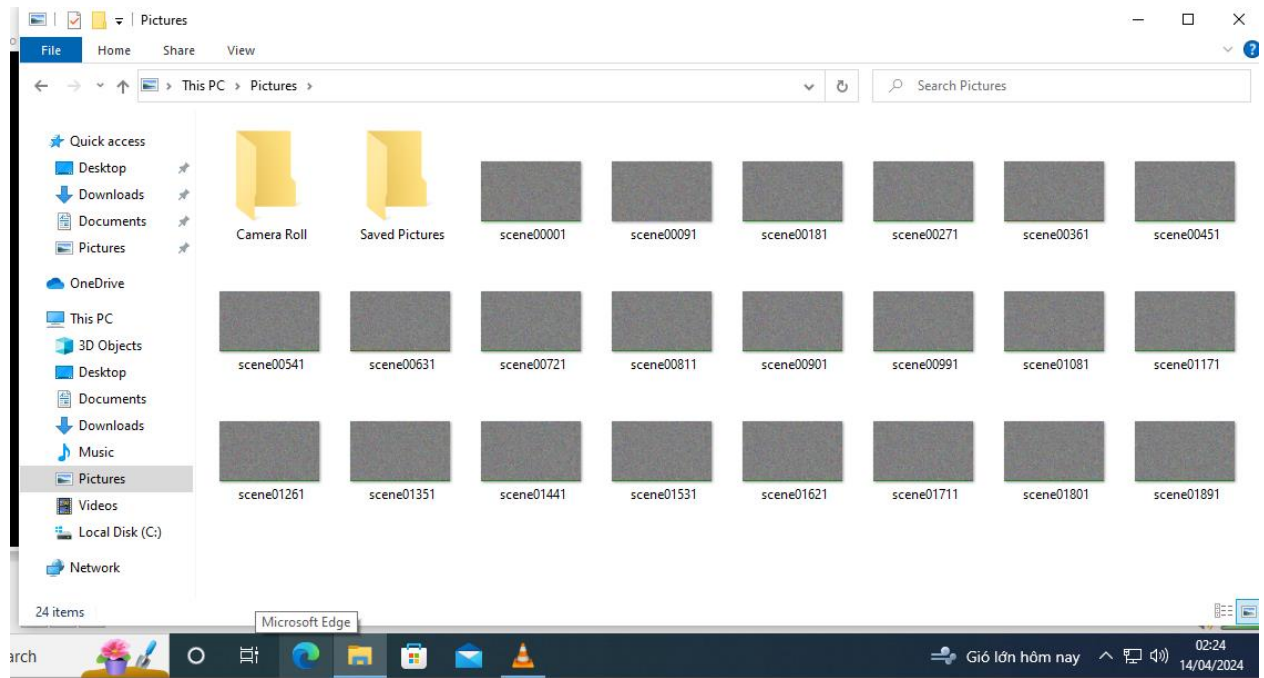




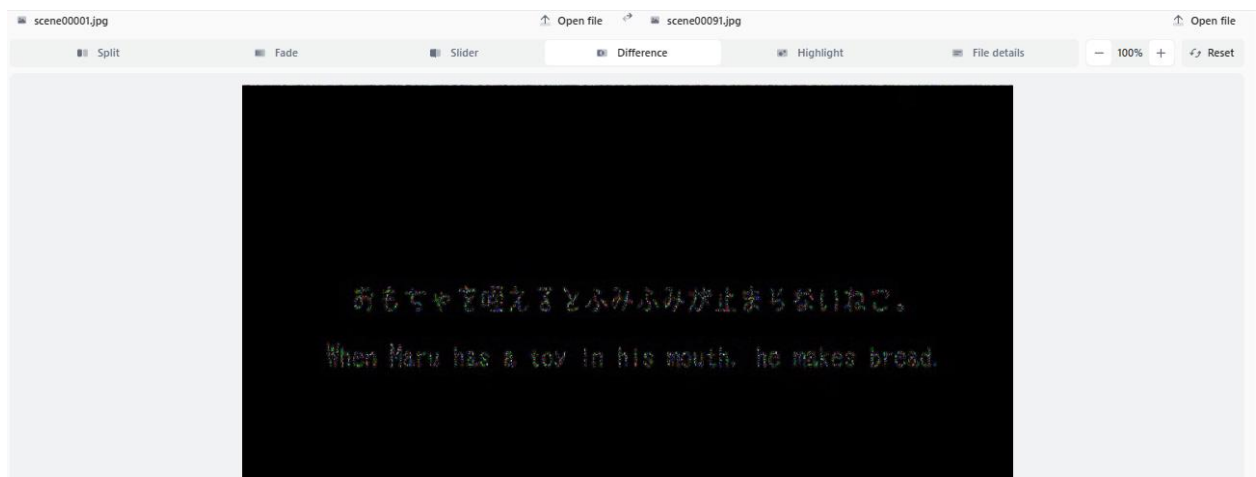
- Ta sẽ thực hiện cấu hình như bên dưới, sau khi cấu hình xong ta lưu lại và tắt VLC:



- Bật lại video và xem, sau đó kiểm tra đường dẫn xem đã có ảnh chưa

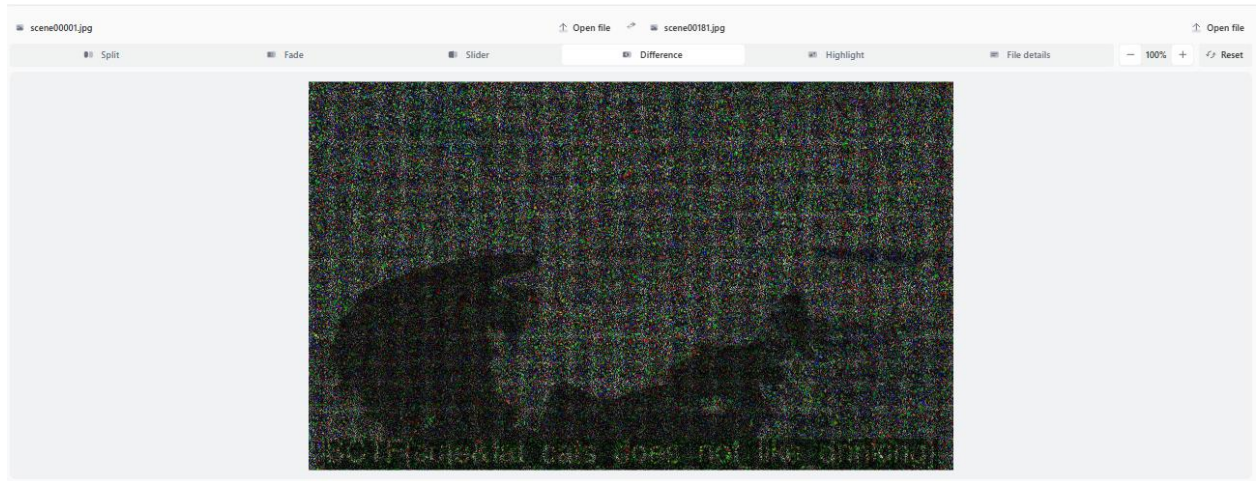


- Dùng tool <https://www.diffchecker.com/> để thực hiện kiểm tra
- Giữa ảnh 1 và 91



- Giữa ảnh 1 và 81





⇒ FLAG: BCTF{cute&fat\_cats\_does\_not\_like\_drinking}

---

*Sinh viên đọc kỹ yêu cầu trình bày bên dưới trang này*

## YÊU CẦU CHUNG

- Sinh viên tìm hiểu và thực hiện bài tập theo yêu cầu, hướng dẫn.
- Nộp báo cáo kết quả chi tiết những việc (**Report**) bạn đã thực hiện, quan sát thấy và kèm ảnh chụp màn hình kết quả (nếu có); giải thích cho quan sát (nếu có).
- Sinh viên báo cáo kết quả thực hiện và nộp bài.

### Báo cáo:

- File **.DOCX và .PDF**. Tập trung vào nội dung, không mô tả lý thuyết.
- Nội dung trình bày bằng **Font chữ Times New Romans/ hoặc font chữ của mẫu báo cáo này (UTM Neo Sans Intel/UTM Viet Sach) – cỡ chữ 13. Canh đều (Justify) cho văn bản. Canh giữa (Center) cho ảnh chụp.**
- Đặt tên theo định dạng: [Mã lớp]-ExeX\_GroupY. (trong đó X là Thứ tự Bài tập, Y là mã số thứ tự nhóm trong danh sách mà GV phụ trách công bố).
- Ví dụ: [NT101.K11.ANTT]-Exe01\_Group03.*
- Nếu báo cáo có nhiều file, nén tất cả file vào file .ZIP với cùng tên file báo cáo.
- **Không đặt tên đúng định dạng – yêu cầu, sẽ KHÔNG chấm điểm bài nộp.**
- Nộp file báo cáo trên theo thời gian đã thống nhất tại [courses.uit.edu.vn](https://courses.uit.edu.vn).

### Đánh giá:

- Hoàn thành tốt yêu cầu được giao.
- Có nội dung mở rộng, ứng dụng.

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

**HẾT**