

Đánh giá độ bền của kỹ thuật giấu tin LSB trong âm thanh

Mục đích

Giúp sinh viên làm quen với kỹ thuật giấu tin (steganography) trong dữ liệu âm thanh bằng phương pháp LSB (Least Significant Bit). Sinh viên sẽ thực hiện các phép biến đổi âm thanh và đánh giá tính bền vững của thông điệp ẩn thông qua các chỉ số như Bit Error Rate (BER) và Normalized Correlation (NC).

Yêu cầu đối với sinh viên

Có kiến thức cơ bản về hệ điều hành Linux. Hiểu về xử lý tín hiệu âm thanh và kỹ thuật giấu tin LSB.

Nội dung thực hành

Khởi động bài lab:

Mở terminal và gõ lệnh sau để khởi động môi trường bài lab:

```
labtainer -r audiostream-lsb-gid
```

(chú ý: sinh viên sử dụng mã sinh viên của mình để nhập thông tin email người thực hiện bài lab khi có yêu cầu, để sử dụng khi chấm điểm)

Sau khi khởi động, một terminal ảo sẽ xuất hiện, sẵn sàng cho các bước thực hành.

Nhúng thông điệp vào tệp âm thanh

Tạo tệp chứa thông điệp:

```
echo "<MSV>" > data.txt
```

Sinh viên tiến hành giấu tin:

```
python3 lsb_emb.py -s <input.wav> -f data.txt -n <LSBs>
```

Kiểm tra kết quả: Tệp đầu ra output.wav được tạo thành công

```
ls -l
```

Áp dụng các phép biến đổi âm thanh

Sử dụng công cụ SoX để thực hiện các biến đổi sau trên tệp output.wav:

Tăng tốc độ (tempo 1.2x):

```
sox output.wav fast.wav tempo 1.2
```

Giảm tốc độ (tempo 0.8x):

```
sox output.wav slow.wav tempo 0.8
```

Đảo ngược âm thanh:

```
sox output.wav reversed.wav reverse
```

Áp dụng hiệu ứng fade (3s fade-in, 3s fade-out):

```
sox output.wav faded.wav fade t 3 0 3
```

Thay đổi cao độ (+200 cents):

```
sox output.wav pitch.wav pitch 200
```

Cắt tệp (lấy 3 giây đầu tiên):

```
sox output.wav cut.wav trim 0 3
```

Tăng âm lượng (1.5x):

```
sox output.wav vol.wav vol 1.5
```

Trích xuất và đánh giá thông điệp trên các file wav được biến đổi:

```
python3 lsb_extract.py -s <fast.wav> -o data.txt -n 4 -b 10 -e <output_file>
```

Lặp lại lệnh trên cho tất cả các tệp biến đổi.

Sử dụng lệnh cat để xem file <output_file> bị biến đổi như thế nào.

Quan sát các chỉ số BER và NC từ đầu ra của mỗi lệnh để đánh giá mức độ sai lệch của thông điệp trích xuất so với thông điệp gốc.

Kết thúc bài lab

Trên terminal đầu tiên, sử dụng lệnh sau để kết thúc bài lab:

```
stoplab audiostream-lsb-gid
```

Khi bài lab kết thúc, một tệp zip chứa kết quả sẽ được tạo và lưu tại vị trí được hiển thị trong thông báo của lệnh stoplab.

Khởi động lại bài lab

Nếu cần thực hiện lại bài lab, sử dụng lệnh:

```
labtainer -r audiostream-lsb-gid
```