

So sánh phương thức giấu tin LSB cơ bản và LSB nâng cao

Tải bài lab:

imodule https://github.com/Quyen-TT/Audio_Steganography/raw/main/audiosteg-lsb-compare/imodule.tar

Mục đích

Giúp sinh viên hiểu và so sánh hai phương pháp giấu tin (steganography) trong âm thanh: LSB cơ bản (Least Significant Bit) và LSB nâng cao.

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:

Vào terminal, gõ:

```
labtainer -r audiosteg-lsb-compare
```

(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)

Sinh viên tạo file thông tin cần giấu với nội dung là msv của bản thân:

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

Sinh viên thực phân tích khả năng giấu tin của 2 phương pháp:

- Giấu tin qua 8 bit MSB và 3 bit LSB:

```
python3 analyze1.py -s <input.wav>
```

- Giấu tin bằng LSB cơ bản:

```
python3 analyze2.py -s <input.wav>
```

Quan sát và nêu nhận xét về khả năng (dung lượng) giấu tin của 2 phương pháp.

Sinh viên thực hiện giấu tin bằng 2 phương pháp:

- Giấu tin bằng LSB cơ bản:

```
python3 basic_lsb_steg.py -s <input.wav> -f secret.txt -o <output1.wav> -n <LSBs>
```

- Giấu tin qua 8 bit MSB và 3 bit LSB:

```
python3 advan_lsb_steg.py -s <input.wav> -f secret.txt -o <output2.wav>
```

Sinh viên so sánh đặc điểm của các file wav để tìm điểm khác biệt:

So sánh dung lượng của các file wav trước và sau khi giấu tin:

```
ls -l <input.wav> <output1.wav> <output2.wav>
```

Phân tích waveform bằng FFmpeg:

```
ffmpeg -i <file.wav> -filter_complex  
"aformat=channel_layouts=mono,showwavespic=s=800x200" -frames:v 1  
<output.png>
```

- Thay thế <file.wav> và <output.png> thành các file wav đầu vào (file gốc và 2 file sau khi đã giấu tin) và các ảnh đầu ra.
- Sử dụng công cụ feh để xem các ảnh đã tạo và đưa ra nhận xét về waveform của 3 file âm thanh

Phân tích spectrum bằng FFmpeg:

```
ffmpeg -i <file.wav> -lavfi showspectrum=s=hd480 <output.png>
```

- Thay thế <file.wav> và <output.png> thành các file wav đầu vào (file gốc và 2 file sau khi đã giấu tin) và các ảnh đầu ra.
- Sử dụng công cụ feh để xem các ảnh đã tạo và đưa ra nhận xét về đặc điểm spectrum của 3 file âm thanh

Phân tích mảng byte:

```
python3 convert_to_bytes.py -s <file.wav>
```

Kiểm tra phần header của file:

```
xxd -l 320 <file.wav>
```

Quan sát sự biến đổi của mảng byte 2 file wav sau khi giấu tin so với bản gốc rồi nêu nhận xét.

Sinh viên thực hiện tách tin và kiểm tra thông điệp ban đầu:

- Tách tin LSB cơ bản:

```
python3 basic_lsb_rec.py -s <output1.wav> -o <output.txt> -n <LSBs> -b <bytes>
```

- Tách tin LSB nâng cao:

```
python3 advan_lsb_rec.py -s <output2.wav>
```

Kết thúc bài lab

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

stoplab audiostream-lsb-compare

Khi bài lab kết thúc, một tệp zip chứa kết quả sẽ được tạo và lưu tại vị trí hiển thị bên dưới lệnh `stoplab`.

Trong quá trình làm bài sinh viên cần thực hiện lại bài lab, dùng câu lệnh:

labtainer -r audiostream-lsb-compare