Deleted SMS

Tags	File Format Mobile Forensics ROT13 Recovery Database SQLite
Type	Digital Forensics
Season	Gitex-2023
Difficulty	Very Easy
Flag Prefix	CHH
Flag Content	you_caN_reCOVeRy_sQ1i73

Description

We receive information about thieves using text messages to exchange sensitive information. Before leaving, they deleted messages to erase traces. Is deleting this data safe? Please help us restore that secret message.

- **Flag Format: CHH{XXXXX}**

Guided Mode

1. What is the database? > `sqlite`
2. What is the table contains the message? > `sms`
3. How many records in the table contain the message? > `10`
4. What is the command of the strings of printable characters in files? `strings`
5. Who is the author of `sqbrite` tools on Github? > `mattboyer`
6. Which encryption algorithm is similar to Caesar's encryption algorithm? > `ROT13`

Writeup

After downloading the file, we need to check what type of file this is using the `file` command in Linux.

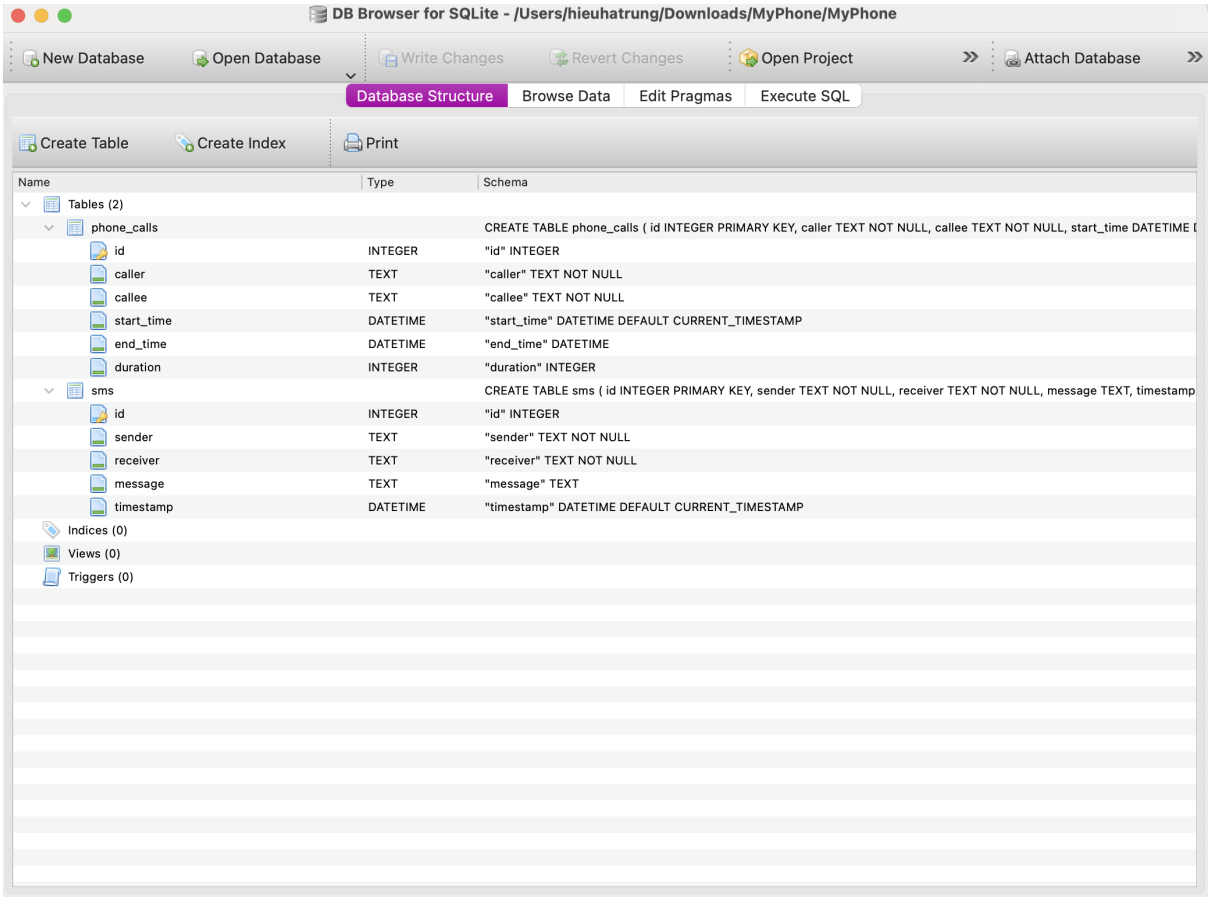
Sau khi tải file chúng ta cần kiểm tra xem đây là loại File gì bằng lệnh `file` trong Linux.

```
└─○ ls -la MyPhone
-rw-r--r--@ 1 hieuhatrung  staff  12288 Oct 11 15:29 MyPhone
```

`MyPhone` is a SQLite 3.x database, then use the `DB Browser` tool `https://sqlitebrowser.org/dl/` to observe the database.

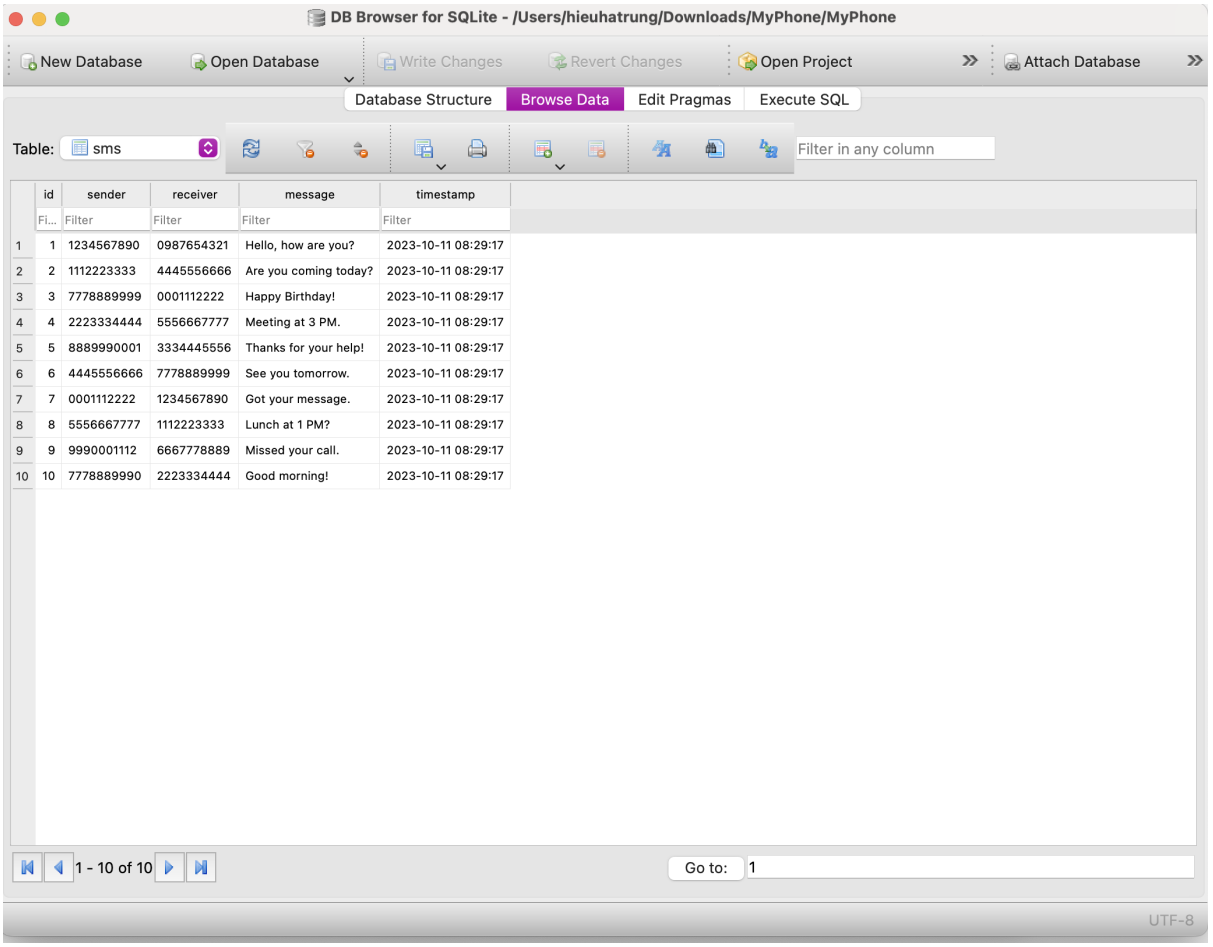
`MyPhone` là cơ sở dữ liệu `SQLite 3.x` , sau đó sử dụng công cụ `DB Browser` https://sqlitebrowser.org/dl/ để quan sát cơ sở dữ liệu.

```
└─○ file MyPhone
MyPhone: SQLite 3.x database, last written using SQLite version 3039002, file counter
6, database pages 3, cookie 0x2, schema 4, UTF-8, version-valid-for 6
```



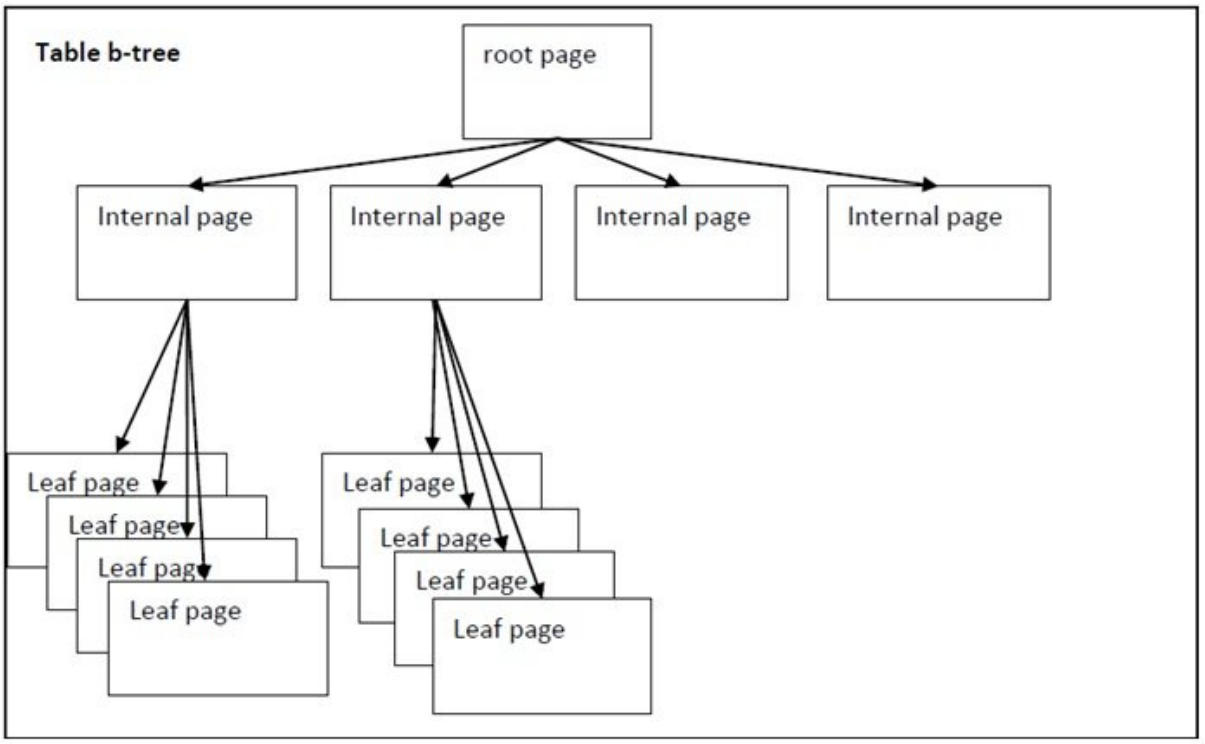
This database includes two tables `phone_calls` and `sms` to store call and message information. Through this tool, we only see 10 message records. We need to find a way to restore messages in this database.

Cơ sở dữ liệu này bao gồm 2 bảng `phone_calls` và `sms` để lưu trữ thông tin cuộc gọi và tin nhắn. Qua công cụ này chúng ta chỉ thấy 10 bản ghi tin nhắn. Chúng ta cần tìm cách khôi phục lại tin nhắn trong sở dữ liệu này.



SQLite uses a B-Tree structure to store table and Index data in the database. The data is stored in the Leaf Page of B-Tree. Deleting a record or executing a DELETE FROM table (...) statement, that Leaf Page will be considered empty. That SQLite will leave that space open for future INSERT operations.

SQLite sử dụng cấu trúc B-Tree để lưu trữ dữ liệu của bảng và Index trong cơ sở dữ liệu. Dữ liệu thực sự được lưu trữ trong Leaf Page của B-Tree. Khi xoá một bản ghi hay thực hiện một câu lệnh `DELETE FROM table (...)`, Leaf Page đó sẽ được đánh giá là trống. SQLite đó sẽ để không gian đó cho các hoạt động INSERT về sau.



However, the data will only be deleted partially; it is located somewhere in the Freespace area of the SQLite file. Therefore, we can still find and restore it by using author Mattboyer's sqbrite tool to perform data recovery.

Tuy nhiên, dữ liệu sẽ không bị xoá bỏ hoàn toàn mà nó nằm đâu đó ở vùng Freespace của file SQLite. Vì thế, chúng ta vẫn có thể tìm và khôi phục lại được. Bằng việc sử dụng công cụ `sqbrite` của tác giả `mattboyer` để thực hiện khôi phục dữ liệu.

```
└─○ pip3 install sqbrite

└─○ sqbrite undelete MyPhone RecoverRecover
2023-10-18 15:34:29,868 INFO: Processing /Users/hieuhatrung/Downloads/MyPhone/MyPhone
2023-10-18 15:34:29,869 INFO: Database: <SQLite DB, page count: 3 | page size: 4096>
2023-10-18 15:34:29,869 WARNING: <SQLite DB, page count: 3 | page size: 4096> does not
have ptrmap pages!
```

Or use the `strings` command to search for records in an SQLite database.

Hoặc sử dụng lệnh `strings` để tìm kiếm các bản ghi trong cơ sở dữ liệu SQLite.

```
└─.⁖ strings MyPhone
SQLite format 3
#tablephone_callsphone_calls
CREATE TABLE phone_calls (
    id INTEGER PRIMARY KEY,
    caller TEXT NOT NULL,
    callee TEXT NOT NULL,
    start_time DATETIME DEFAULT CURRENT_TIMESTAMP,
    end_time DATETIME,
    duration INTEGER
_tablesmssms
CREATE TABLE sms (
    id INTEGER PRIMARY KEY,
    sender TEXT NOT NULL,
    receiver TEXT NOT NULL,
    message TEXT,
    timestamp DATETIME DEFAULT CURRENT_TIMESTAMP
K!!E31234567890098765432 1PUU{lbh_pnA_erPBIrEl_fD1v73}2023-10-11 08:29:18:
!!'377788899902223334444Good morning!2023-10-11 08:29:17>
!!/399900011126667778889Missed your call.2023-10-11 08:29:17;
!!)3555666777711122233333Lunch at 1 PM?2023-10-11 08:29:17>
!!/300011122221234567890Got your message.2023-10-11 08:29:17>
!!/34445556666777888999 9See you tomorrow.2023-10-11 08:29:17B
!!738889990001333444555 6Thanks for your help!2023-10-11 08:29:17=
!!-32223334444555666777 7Meeting at 3 PM.2023-10-11 08:29:17<
!!+377788899990001112222Happy Birthday!2023-10-11 08:29:17B
!!731112223333444555666 6Are you coming today?2023-10-11 08:29:17@
!!331234567890098765432 1Hello, how are you?2023-10-11 08:29:17
!!33
777888999022233344442023-10-10 19:00:002023-10-10 19:10:00
!!33
999000111266677788892023-10-10 18:00:002023-10-10 18:20:00
!!33
555666777711122233332023-10-10 17:00:002023-10-10 17:05:00
!!33
000111222212345678902023-10-10 16:00:002023-10-10 16:05:00
!!33
444555666677788899992023-10-10 15:00:002023-10-10 15:15:00
!!33
888999000133344455562023-10-10 14:00:002023-10-10 14:45:00
!!33
222333444455566677772023-10-10 13:00:002023-10-10 13:03:00
!!33
777888999900011122222023-10-10 12:00:002023-10-10 12:30:00
!!33
111222333344455566662023-10-10 11:00:002023-10-10 11:10:00
!!33
123456789009876543212023-10-10 10:00:002023-10-10 10:05:00
```

Deleted records will be identified by the letter `K!!`

Bản ghi bị xoá sẽ được nhận diện bằng ký tự `K!!`

```
K!!E31234567890098765432 1PUU{lbh_pnA_erPBIrEl_fD1v73}2023-10-11 08:29:18:
```

The string PUU{lbh_pnA_erPBIrEl_fD1v73} has been encoded to a different type. We know that the Prefix of the Flag is CHH. The letter U is repeated and transformed into the letter H.

Chuỗi `PUU{lbh_pnA_erPBIrEl_fD1v73}` đã được Encoding sang một kiểu khác. Mà ta biết từ Prefix của Flag là `CHH` Chữ `U` được lặp lại và biến đổi thành chữ `H` .

So, it can be concluded that they use a rotation code similar to Caesar. In this case, it is ROT13; we can easily decode and see that the FLAG is CHH{you_caN_reCOVeRy_sQ1i73}

Nên có thể kết luận họ sử dụng một mã xoay nào đó tương tự như Caesar. Trong trường hợp này là ROT13, chúng ta dễ dàng decode và thấy được FLAG là `CHH{you_caN_reCOVeRy_sQ1i73}`