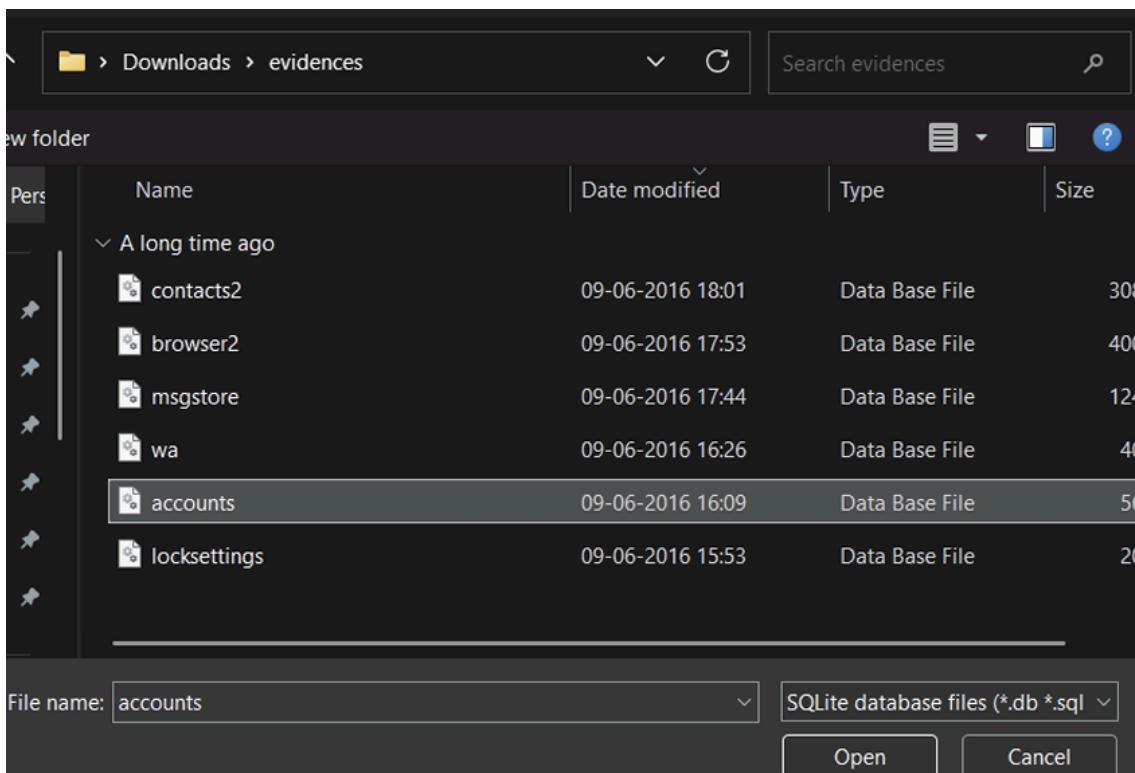
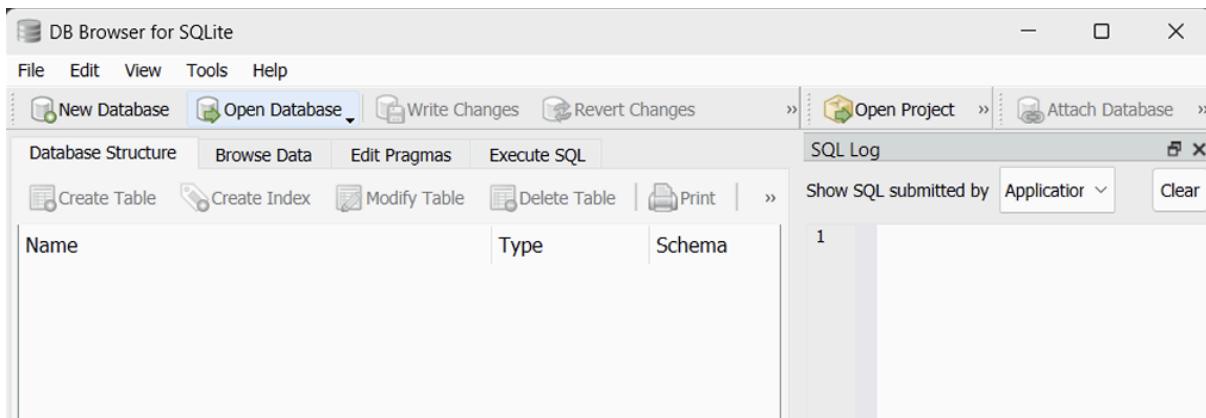


Praneesh R V  
CB.SC.U4CYS23036

## Cyber Forensics Lab - 7 - Database Forensics

TASK 1: Exploring evidence in DB Browser for SQLite. Loading the accounts.db evidence file. Loading the accounts.db evidence file:



## Loading the wa.db evidence file:

The screenshot shows the SQLite Database Browser interface. The top menu bar includes File, Edit, View, Tools, Help, New Database, Open Database, Write Changes, Revert Changes, Undo, Open Project, Save Project, Attach Database, and Close Database. The main window has tabs for Database Structure, Browse Data, Edit Pragmas, and Execute SQL. The Database Structure tab is selected, displaying the DB Schema. The schema includes:

- Tables (7):**
  - accounts: CREATE TABLE accounts (\_id INTEGER PRIMARY KEY AUTOINCREMENT, name TEXT NOT NULL, type TEXT NOT NULL, password TEXT)
  - android\_metadata: CREATE TABLE android\_metadata (locale TEXT)
  - authtokens: CREATE TABLE authtokens (\_id INTEGER PRIMARY KEY AUTOINCREMENT, accounts\_id INTEGER NOT NULL, type TEXT NOT NULL, key TEXT NOT NULL, value TEXT)
  - extras: CREATE TABLE extras (\_id INTEGER PRIMARY KEY AUTOINCREMENT, accounts\_id INTEGER, key TEXT NOT NULL, value TEXT)
  - grants: CREATE TABLE grants (accounts\_id INTEGER NOT NULL, auth\_token\_type STRING NOT NULL, uid INTEGER NOT NULL)
  - meta: CREATE TABLE meta (key TEXT PRIMARY KEY NOT NULL, value TEXT)
  - sqlite\_sequence: CREATE TABLE sqlite\_sequence(name,seq)
- Indices (0):**
- Views (0):**
- Triggers (1):**
  - accountsDelete: CREATE TRIGGER accountsDelete DELETE ON accounts BEGIN DELETE FROM authtokens WHERE accounts\_id=OLD.\_id;

The screenshot shows the SQLite Database Browser interface with the Database Structure tab selected. The left pane shows the tables, and the right pane shows the schema. The accounts table is selected, displaying the following data:

	_id	name	type	password
1	2	WhatsApp	com.whatsapp	9f7955c8fa794ccc87465cf95aeeef99e523
2	3	+1-000-000-0000	com.viber.voip	9f7955c8fa794ccc87465cf95aeeef99e523

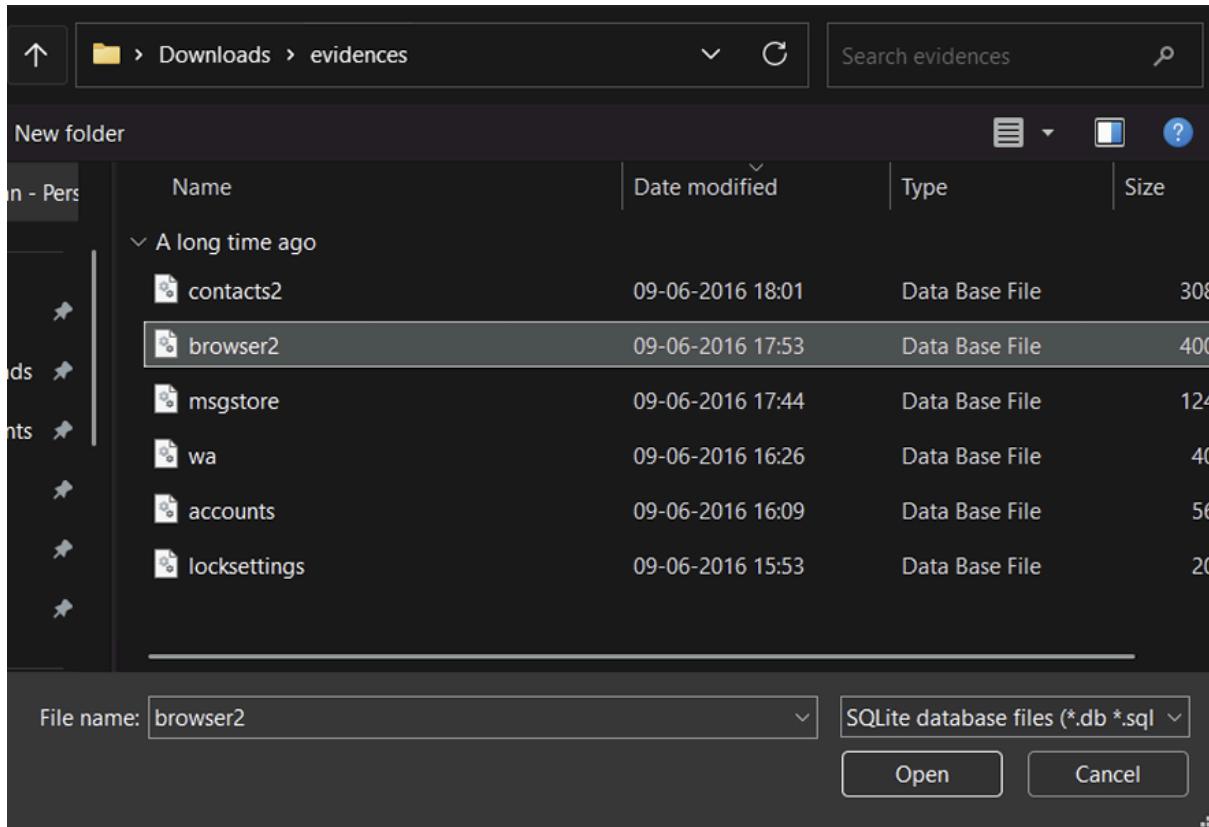
The screenshot shows the SQLite Database Browser interface with the Database Structure tab selected. The left pane shows the tables, and the right pane shows the SQL Log. The log contains the following SQL statements:

```

1 PRAGMA foreign_keys = '1';
2 PRAGMA database_list;
3 SELECT type, name, sql, tbl_name FROM "main".sqlite_master;
4 PRAGMA encoding;
5 SELECT "_rowid_","* FROM "main"."accounts" LIMIT 49999 OFFSET 0;
6

```

## browser2.db evidence file:



The screenshot shows a SQLite browser application with the following interface elements:

- Toolbar:** File, Edit, View, Tools, Help, New Database, Open Database, Write Changes, Revert Changes, Undo, Open Project, Save Project, Attach Database, Close Database.
- Database Structure:** Shows the 'bookmarks' table with columns: id, title, url.
- Data View:** A table listing 12 bookmarks entries.
- SQL Log:** A text area showing the following SQL code:

```

1 PRAGMA foreign_keys = '1';
2 PRAGMA database_list;
3 SELECT type, name, sql, tbl_name FROM "main".sqlite_master;
4 PRAGMA "main".TABLE_INFO("v_accounts");
5 PRAGMA "main".TABLE_INFO("v_omnibox_suggestions");
6 PRAGMA enable_load_extension;
7 SELECT "_rowid_" * FROM "main"."_sync_state" LIMIT 49999 OFFSET 0;
8 SELECT "_rowid_" * FROM "main"."_sync_state" LIMIT 49999 OFFSET 0;
9 SELECT "_rowid_" * FROM "main"."Bookmarks" LIMIT 49999 OFFSET 0;
10

```

Bookmarks tab in browse data:

The screenshot shows the SQLite Database Browser interface. The top menu bar includes File, Edit, View, Tools, Help, New Database, Open Database, Write Changes, Revert Changes, Undo, Open Project, Save Project, Attach Database, and Close Database. Below the menu is a toolbar with icons for New Database, Open Database, Write Changes, Revert Changes, Undo, Open Project, Save Project, Attach Database, and Close Database. The main window displays a table named 'history' with columns '\_id', 'title', and 'url'. The table contains 9 rows of data. An SQL log window on the right shows the following SQL statements:

```

1 PRAGMA foreign_keys = '1';
2 PRAGMA database_list;
3 SELECT type, name, sql, tbl_name FROM "main".sqlite_master;
4 PRAGMA "main".TABLE_INFO("v_accounts");
5 PRAGMA "main".TABLE_INFO("v_omnibox_suggestions");
6 PRAGMA encoding;
7 SELECT "_rowid_," FROM "main"."sync_state" LIMIT 49999 OFFSET 0;
8 SELECT "_rowid_," FROM "main"."sync_state" LIMIT 49999 OFFSET 0;
9 SELECT "_rowid_," FROM "main".bookmarks" LIMIT 49999 OFFSET 0;
10 SELECT "_rowid_," FROM "main".history" LIMIT 49999 OFFSET 0;
11 SAVEPOINT "UNDOPOINT";
12
13 SELECT "_rowid_," FROM "main".thumbnails" LIMIT 49999 OFFSET 0;
14 SELECT "_rowid_," FROM "main".sqlite_sequence" LIMIT 49999 OFFSET 0;

```

## History tab in browse data:

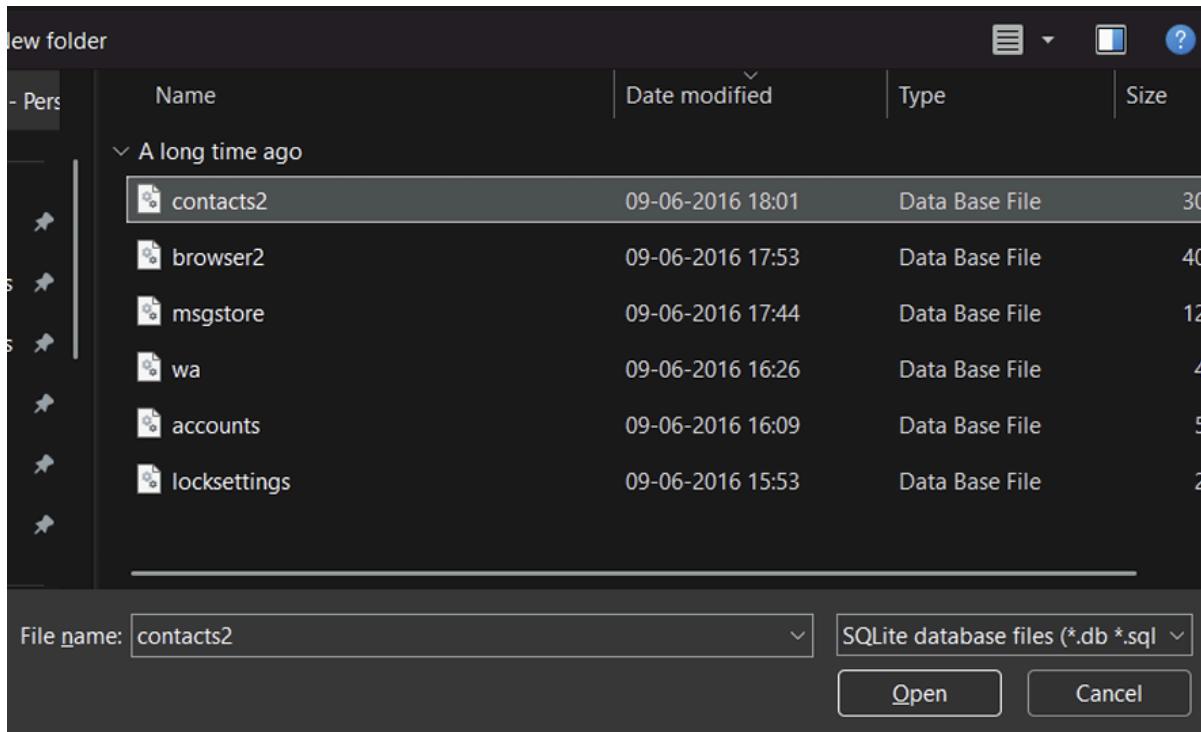
The screenshot shows the SQLite Database Browser interface. The top menu bar includes File, Edit, View, Tools, Help, New Database, Open Database, Write Changes, Revert Changes, Undo, Open Project, Save Project, Attach Database, and Close Database. Below the menu is a toolbar with icons for New Database, Open Database, Write Changes, Revert Changes, Undo, Open Project, Save Project, Attach Database, and Close Database. The main window displays a table named 'sqlite\_sequence' with columns 'name' and 'seq'. The table contains 2 rows of data. An SQL log window on the right shows the following SQL statements:

```

1 history

```

## contacts2.db evidence file:



File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Undo Open Project Save Project Attach Database Close Database

Create Table Create Index Modify Table Delete Table Print Refresh

Tables (35)

```

CREATE TABLE _sync_s
CREATE TABLE _sync_s
CREATE TABLE account
CREATE TABLE agg_ex
CREATE TABLE android
CREATE TABLE contact
CREATE TABLE data
CREATE TABLE data_us
CREATE TABLE default_
CREATE TABLE director
CREATE TABLE groups
CREATE TABLE mimetypes
CREATE TABLE name_k
CREATE TABLE nickname
CREATE TABLE package
CREATE TABLE phone_
CREATE TABLE photo_f
CREATE TABLE propri
CREATE TABLE raw_cor
CREATE VIRTUAL TABL
CREATE TABLE search_
CREATE TABLE settings
CREATE TABLE sqlite_s
CREATE TABLE sqlite_s
CREATE TABLE status_l
CREATE TABLE stream_
CREATE TABLE stream_

```

No cell active.

Type: NULL; Size: 0 bytes

SQL Log

Show SQL submitted by Application

```

1  ROLLBACK TO SAVEPOINT "UNDOPOINT";
2  PRAGMA database_list;
3  SELECT type,name,sql,tbl_name FROM "main".sqlite_master;
4  PRAGMA "main",TABLE_INFO("accounts");
5  PRAGMA "main",TABLE_INFO("omnibox_suggestions");
6  RELEASE "UNDOPOINT";
7  PRAGMA foreign_keys = '1';
8  PRAGMA database_list;
9  SELECT type,name,sql,tbl_name FROM "main".sqlite_master;
10 PRAGMA "main",TABLE_INFO("data");
11 PRAGMA "main",TABLE_INFO("phone_lookup");
12 PRAGMA "main",TABLE_INFO("name_lookup");
13 PRAGMA "main",TABLE_INFO("view_data");
14 PRAGMA "main",TABLE_INFO("view_raw_contacts");
15 PRAGMA "main",TABLE_INFO("view_contacts");
16 PRAGMA "main",TABLE_INFO("view_entities");
17 PRAGMA "main",TABLE_INFO("view_entities");
18 PRAGMA "main",TABLE_INFO("view_data_usage_stat");
19 PRAGMA "main",TABLE_INFO("view_stream_items");
20 PRAGMA "main",TABLE_INFO("view_groups");
21 PRAGMA "main",TABLE_INFO("view_v1_people");
22 PRAGMA "main",TABLE_INFO("view_v1_organizations");
23 PRAGMA "main",TABLE_INFO("view_v1_methods");
24 PRAGMA "main",TABLE_INFO("view_v1_phones");
25 PRAGMA "main",TABLE_INFO("view_v1_extensions");
26 PRAGMA "main",TABLE_INFO("view_v1_groups");
27 PRAGMA "main",TABLE_INFO("view_v1_group_membership");
28 PRAGMA "main",TABLE_INFO("view_v1_photos");
29 PRAGMA "main",TABLE_INFO("search_index");
30 PRAGMA encoding;
31 SELECT "_rowid_.* FROM "main"."_sync_state" LIMIT 49999 OFFSET 0;
32

```

UTF-8

Screenshot of the SQLite Database Browser showing the raw\_contacts table and its SQL log.

**Table: raw\_contacts**

#	account_id	sourceid	raw_contact_is_read_only	version	dirty	deleted	contact_id	age
1	1	NULL		0	2	1	0	1
2	2	NULL		0	2	1	0	2
3	3	1	NULL	0	2	1	0	3
4	4	1	NULL	0	2	1	0	4
5	5	1	NULL	0	2	1	0	5
6	6	1	NULL	0	2	1	0	6
7	7	1	NULL	0	2	1	0	7
8	8	1	NULL	0	2	1	0	8
9	9	1	NULL	0	2	1	0	9
10	10	1	NULL	0	2	1	0	10
11	11	2	1	0	3	0	0	1
12	12	2	1	0	3	0	0	2
13	13	2	3	0	3	0	0	3
14	14	2	4	0	3	0	0	4
15	15	2	5	0	3	0	0	5
16	16	2	6	0	3	0	0	6
17	17	2	7	0	3	0	0	7
18	18	2	8	0	3	0	0	8
19	19	2	9	0	3	0	0	9
20	20	2	10	0	3	0	0	10

**SQL Log**

```

1
2
3
4 PRAGMA "main".TABLE_INFO("v_accounts");
5 PRAGMA "main".TABLE_INFO("v_omnibox_suggestions");
6 RELEASE "UNDOPOINT";
7 PRAGMA foreign_keys = '1';
8 PRAGMA database_list;
9 SELECT type,name,tbl_name FROM "main".sqlite_master;
10 PRAGMA "main".TABLE_INFO("v_data");
11 PRAGMA "main".TABLE_INFO("v_data_lookup");
12 PRAGMA "main".TABLE_INFO("name_lookup");
13 PRAGMA "main".TABLE_INFO("view_data");
14 PRAGMA "main".TABLE_INFO("view_raw_contacts");
15 PRAGMA "main".TABLE_INFO("view_contacts");
16 PRAGMA "main".TABLE_INFO("view_raw_entities");
17 PRAGMA "main".TABLE_INFO("view_entities");
18 PRAGMA "main".TABLE_INFO("view_data_usage_stat");
19 PRAGMA "main".TABLE_INFO("view_stream_items");
20 PRAGMA "main".TABLE_INFO("view_group");
21 PRAGMA "main".TABLE_INFO("view_v1_people");
22 PRAGMA "main".TABLE_INFO("view_v1_organizations");
23 PRAGMA "main".TABLE_INFO("view_v1_contact_methods");
24 PRAGMA "main".TABLE_INFO("view_v1_phones");
25 PRAGMA "main".TABLE_INFO("view_v1_extensions");
26 PRAGMA "main".TABLE_INFO("view_v1_group");
27 PRAGMA "main".TABLE_INFO("view_v1_group_membership");
28 PRAGMA "main".TABLE_INFO("view_v1_photos");
29 PRAGMA "main".TABLE_INFO("search_index");
30
31 SELECT "_rowid_.*" FROM "main"._sync_state" LIMIT 49999 OFFSET 0;
32 SELECT "_rowid_.*" FROM "main"._sync_state" LIMIT 49999 OFFSET 0;
33 SELECT "_rowid_.*" FROM "main".data" LIMIT 49999 OFFSET 0;
34 SELECT "_rowid_.*" FROM "main".raw_contacts" LIMIT 49999 OFFSET 0;
35

```

Screenshot of the SQLite Database Browser showing the calls table and its SQL log.

**Table: calls**

#	number	date	duration	type	new	name	numbertype	number
1	+10000000005	1465376224525	8	2	0	Cherry	2	NULL
2	+10000000003	1465378076924	14	2	0	Adam	2	NULL
3	+10000000004	1465378263153	492	2	0	Beckham	2	NULL
4	+10000000007	1465378892956	79	2	0	Darren	2	NULL
5	+10000000010	1465379422888	0	2	1	Henry	2	NULL

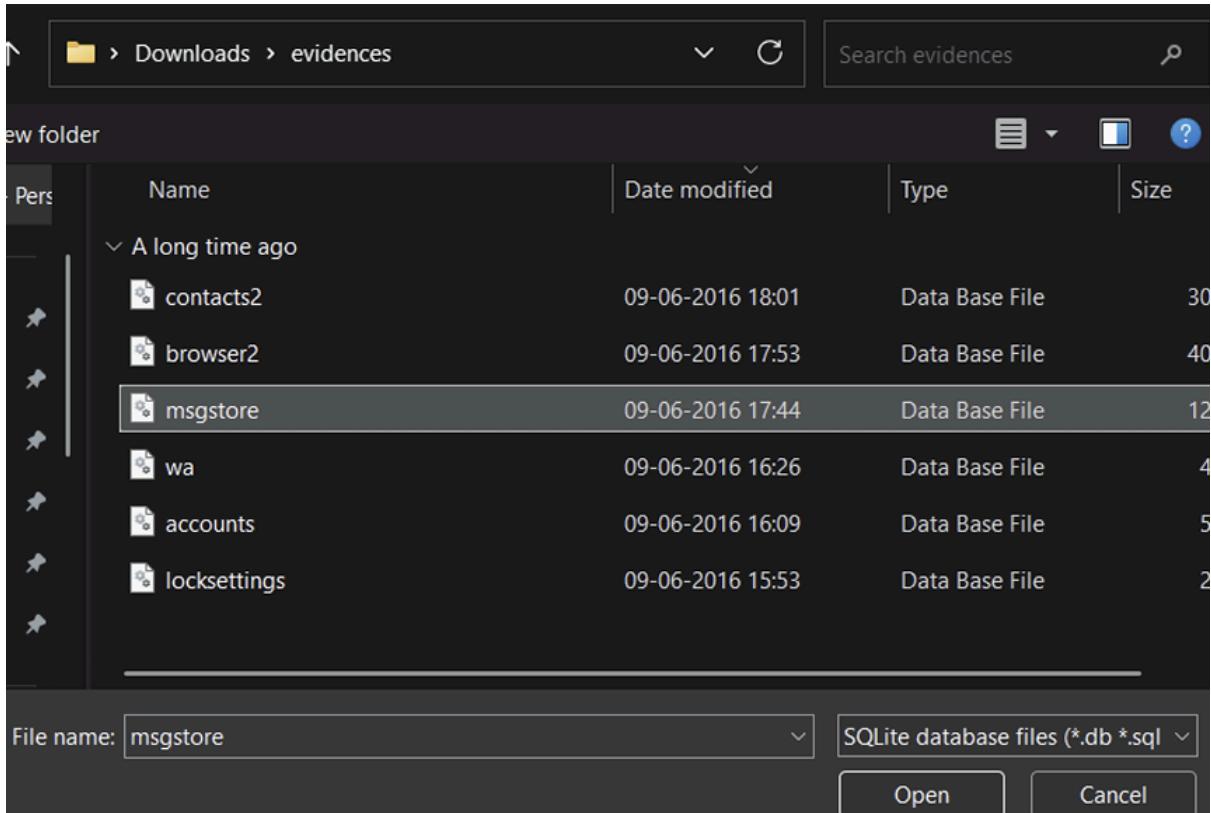
**SQL Log**

```

1
2
3
4
5 PRAGMA "main".TABLE_INFO("v_omnibox_suggestions");
6 RELEASE "UNDOPOINT";
7 PRAGMA foreign_keys = '1';
8 PRAGMA database_list;
9 SELECT type,name,tbl_name FROM "main".sqlite_master;
10 PRAGMA "main".TABLE_INFO("data");
11 PRAGMA "main".TABLE_INFO("phone_lookup");
12 PRAGMA "main".TABLE_INFO("name_lookup");
13 PRAGMA "main".TABLE_INFO("view_data");
14 PRAGMA "main".TABLE_INFO("view_raw_contacts");
15 PRAGMA "main".TABLE_INFO("view_entities");
16 PRAGMA "main".TABLE_INFO("view_raw_entities");
17 PRAGMA "main".TABLE_INFO("view_entities");
18 PRAGMA "main".TABLE_INFO("view_data_usage_stat");
19 PRAGMA "main".TABLE_INFO("view_stream_items");
20 PRAGMA "main".TABLE_INFO("view_group");
21 PRAGMA "main".TABLE_INFO("view_v1_people");
22 PRAGMA "main".TABLE_INFO("view_v1_organizations");
23 PRAGMA "main".TABLE_INFO("view_v1_contact_methods");
24 PRAGMA "main".TABLE_INFO("view_v1_phones");
25 PRAGMA "main".TABLE_INFO("view_v1_extensions");
26 PRAGMA "main".TABLE_INFO("view_v1_group");
27 PRAGMA "main".TABLE_INFO("view_v1_group_membership");
28 PRAGMA "main".TABLE_INFO("view_v1_photos");
29 PRAGMA "main".TABLE_INFO("search_index");
30
31 SELECT "_rowid_.*" FROM "main"._sync_state" LIMIT 49999 OFFSET 0;
32 SELECT "_rowid_.*" FROM "main"._sync_state" LIMIT 49999 OFFSET 0;
33 SELECT "_rowid_.*" FROM "main".data" LIMIT 49999 OFFSET 0;
34 SELECT "_rowid_.*" FROM "main".raw_contacts" LIMIT 49999 OFFSET 0;
35 SELECT "_rowid_.*" FROM "main".calls" LIMIT 49999 OFFSET 0;

```

## msgstore.db evidence file:



The screenshot shows a database management interface with the following details:

- File menu: File, Edit, View, Tools, Help
- Toolbar: New Database, Open Database, Write Changes, Revert Changes, Undo, Open Project, Save Project, Attach Database, Close Database
- Database Structure tab is selected.
- Tables (14) section:

  - chat\_list
  - group\_participants
  - group\_participants\_history
  - media\_refs
  - messages
  - messages\_fts
  - messages\_fts\_content
  - messages\_fts\_segdir
  - messages\_fts\_segments
  - messages\_links
  - messages\_quotes
  - props
  - receipts
  - sqlite\_sequence

- Indices (8) section:

  - group\_participants\_history\_index
  - group\_participants\_index
  - media\_hash\_index
  - media\_type\_index
  - media\_type\_jid\_index
  - messages\_key\_index
  - receipts\_key\_index
  - starred\_index

- Views (0)
- Triggers (4) section:

  - messages\_bd\_for\_links\_trigger
  - messages\_bd\_for\_quotes\_trigger
  - messages\_bd\_for\_receipts\_trigger
  - messages\_bd\_trigger

- SQL Editor:

```
PRAGMA foreign_keys = 1;
PRAGMA database_list;
SELECT type, name, sql, tbl_name FROM "main".sqlite_master;
PRAGMA "main".TABLE_INFO("messages_fts");
PRAGMA encoding;
SELECT "_rowid_","* " FROM "main"."chat_list" LIMIT 49999 OFFSET 0;
```

The screenshot shows the SQLite Database Browser interface. At the top, there's a menu bar with File, Edit, View, Tools, Help, and several database-related buttons like New Database, Open Database, Write Changes, Revert Changes, Undo, Open Project, Save Project, Attach Database, and Close Database. Below the menu is a toolbar with icons for Database Structure, Browse Data, Edit Pragmas, Execute SQL, and other database operations.

The main area displays a table named 'chat\_list'. The columns are: \_id, key\_remote\_jid, message\_table\_id, subject, and creation\_time. The data rows are:

	_id	key_remote_jid	message_table_id	subject	creation_time
1	1	00000005-00000000@g.us	3	Blood Shredders 🎉 🎉	14610507
2	2	00000002-00000000@g.us	5	The Hacking Fellas... ! ! ! 🚨 🚨	14617742
3	3	00000001-00000000@g.us	7	Suicide Bombers	14519280
4	4	00000001-000000000001@g.us	9	Hackmeds	14616784
5	5	00000000-00000000-00000002@g.us	10	Terrorists for Violence	13622011
6	6	00000001@g.us	12	HackeRs Club	14318069
7	7	00000000@broadcast	13		14654548

To the right of the table, there's an 'Edit Database Cell' panel with a mode set to 'Text'. A single character '1' is selected. Below the table, a message says 'Editing row=1, column=0'. The SQL Log panel shows the following SQL code:

```
PRAGMA foreign_keys = 1;
PRAGMA database_list;
SELECT type, name, sql, tbl_name FROM "main".sqlite_master;
PRAGMA "main".TABLE_INFO("messages_fts");
PRAGMA encoding;
SELECT "_rowid_," FROM "main"."chat_list" LIMIT 49999 OFFSET 0;
SELECT "_rowid_," FROM "main"."chat_list" LIMIT 49999 OFFSET 0;
```

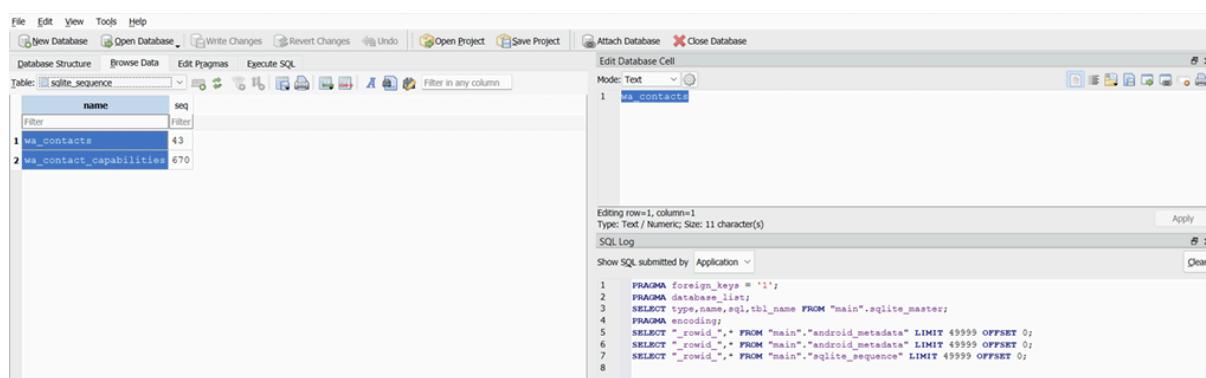
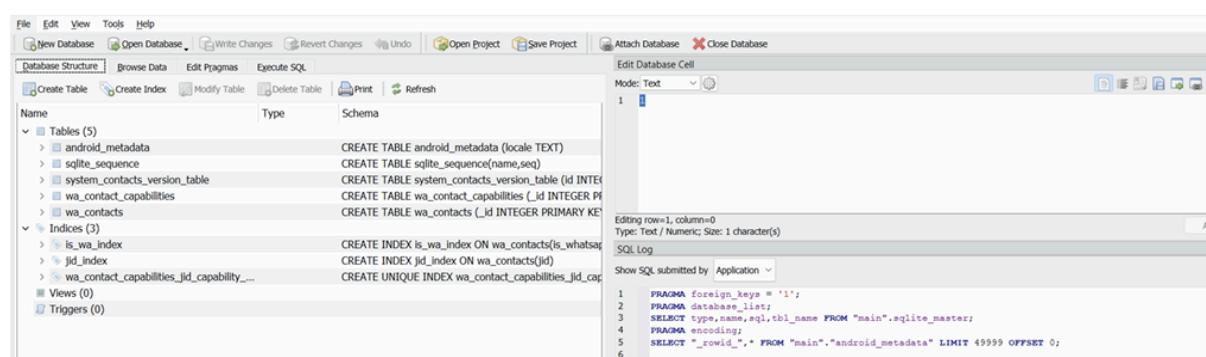
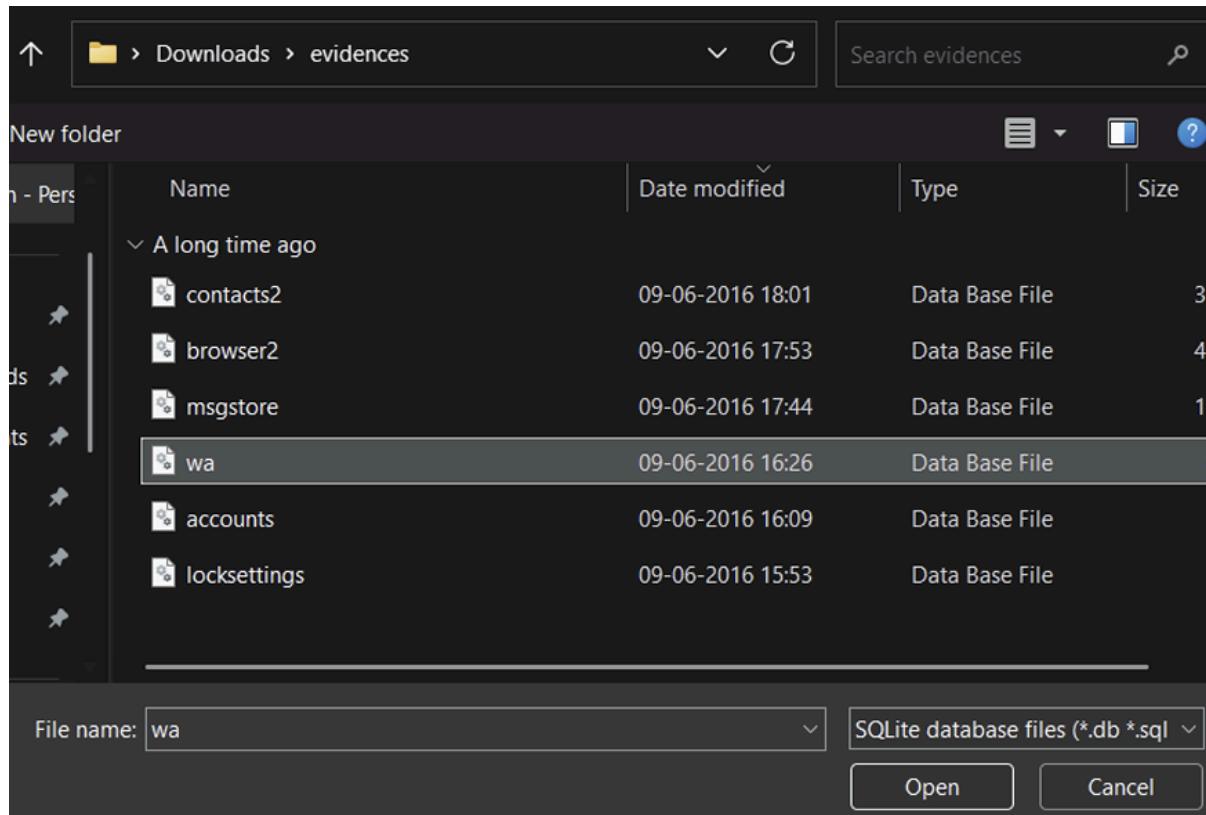
The screenshot shows the DB Browser for SQLite interface. On the left, there's a tree view of the database structure under 'Database Structure'. The main area displays a table named 'sqlite\_sequence' with columns 'name' and 'seq'. The table contains five rows: messages (seq 15), props (seq 4), group\_participants (seq 138), chat\_list (seq 7), and group\_participants\_history (seq 1). Below the table, there's a toolbar with icons for New Database, Open Database, Write Changes, Revert Changes, Undo, Open Project, Save Project, Attach Database, and Close Database.

On the right, an 'Edit Database Cell' dialog is open for the row with name 'group\_participants\_history' and seq '1'. The text input field contains the SQL command:

```
1 group_participants_history
```

Below the input field, it says 'Editing row=5, column=1'. Underneath that, there are buttons for Type: Text / Numeric; Size: 26 character(s); SQL Log; and Show SQL submitted by Application. The application dropdown shows 'Application'. The SQL log pane at the bottom shows the following log entries:

```
1 PRAGMA foreign_keys = '1';
2 PRAGMA database_list;
3 SELECT type, name, sql, tbl_name FROM "main".sqlite_master;
4 PRAGMA "main".TABLE_INFO("messages_fts");
5 PRAGMA table_info("messages_fts");
6 SELECT "_rowid_" FROM "main"."chat_list" LIMIT 49999 OFFSET 0;
7 SELECT "_rowid_" FROM "main"."chat_list" LIMIT 49999 OFFSET 0;
8 SELECT "_rowid_" FROM "main"."sqlite_sequence" LIMIT 49999 OFFSET 0;
9
```



## locksettings.db evidence file:

The screenshot shows the SQLite Database Browser interface. On the left, the 'wa\_contacts' table is displayed with columns: \_id, jid, is\_whatsapp\_user, status, status\_timestamp, number, and raw\_oo. The table contains 31 rows of data, mostly showing '000@s.whatsapp.net' as the jid and various phone numbers. On the right, an 'Edit Database Cell' window is open for the first row, showing the value '1' in the '\_id' column. Below it, the 'SQL Log' window displays the following SQL query:

```

1 PRAGMA foreign_keys = '1';
2 PRAGMA database_list;
3 SELECT name,sql,tbl_name FROM "main".sqlite_master;
4 PRAGMA encoding;
5 SELECT "_rowid_,"* FROM "main"."android_metadata" LIMIT 49999 OFFSET 0;
6 SELECT "_rowid_,"* FROM "main"."android_metadata" LIMIT 49999 OFFSET 0;
7 SELECT "_rowid_,"* FROM "main"."sqlite_sequence" LIMIT 49999 OFFSET 0;
8 SELECT "_rowid_,"* FROM "main"."wa_contacts" LIMIT 49999 OFFSET 0;
9

```

The screenshot shows a file selection dialog box. The title bar says 'Downloads > evidences'. The main area is a file list with the following details:

Name	Date modified	Type	Size
contacts2	09-06-2016 18:01	Data Base File	308
browser2	09-06-2016 17:53	Data Base File	400
msgstore	09-06-2016 17:44	Data Base File	124
wa	09-06-2016 16:26	Data Base File	40
accounts	09-06-2016 16:09	Data Base File	56
<b>locksettings</b>	09-06-2016 15:53	Data Base File	20

At the bottom, there are fields for 'File name:' containing 'locksettings' and a dropdown for 'SQLite database files (\*.db \*.sql)'. There are also 'Open' and 'Cancel' buttons.

The screenshot shows the SQLite Database Browser interface. On the left, the 'Database Structure' tab is selected, displaying a tree view of tables and their metadata. Under 'Tables (3)', the 'locksettings' table is expanded, showing its schema: 'CREATE TABLE locksettings (\_id INTEGER PRIMARY KEY, name TEXT, value INTEGER)'. Below the schema, the table data is shown:

_id	name	value
1	lockscreen.disabled	0
2	migrated	0
3	lock_pattern_visible_pattern	0
4	lockscreen.patternneverchosen	0
5	lockscreen.password_type	0
6	lock_pattern_autolock	0

The right panel contains the SQL log with the following commands:

```

1 PRAGMA foreign_keys = '1';
2 PRAGMA database_list;
3 SELECT type, name, sql, tbl_name FROM "main".sqlite_master;
4 PRAGMA encoding;
5 SELECT "_rowid_.*" FROM "main"."android_metadata" LIMIT 49999 OFFSET 0;
6 SELECT "_rowid_.*" FROM "main"."android_metadata" LIMIT 49999 OFFSET 0;
7 SELECT "_rowid_.*" FROM "main".locksettings" LIMIT 49999 OFFSET 0;

```

This screenshot is identical to the one above, showing the SQLite Database Browser interface. The 'locksettings' table data is displayed in a table:

_id	name	value
1	lockscreen.disabled	0
2	migrated	0
3	lock_pattern_visible_pattern	0
4	lockscreen.patternneverchosen	0
5	lockscreen.password_type	0
6	lock_pattern_autolock	0

The right panel shows the same SQL log as the previous screenshot.

## Task 2: Performing Forensics Investigation on a MySQL Server Database

Creating database:

The screenshot shows the MySQL command-line client interface. It starts with the MySQL monitor welcome message and then displays the following commands and responses:

```

Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.4.7 MySQL Community Server - GPL

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database wordpress
      -> ;
Query OK, 1 row affected (0.01 sec)

```

Copying contents of wordpress\_evidence.sql:

```
C:\Users>cd ..
C:\>cd wamp64
C:\wamp64>cd bin
C:\wamp64\bin>cd mysql
C:\wamp64\bin\mysql>cd mysql8.4.7
C:\wamp64\bin\mysql\mysql8.4.7>cd bin
C:\wamp64\bin\mysql\mysql8.4.7\bin>mysql -u root -p wordpress < wordpress_evidence.sql
Enter password:
C:\wamp64\bin\mysql\mysql8.4.7\bin>
```

Content in wordpress:

```
mysql> show tables;
+-----+
| Tables_in_wordpress |
+-----+
| wp_commentmeta
| wp_comments
| wp_links
| wp_options
| wp_postmeta
| wp_posts
| wp_term_relationships
| wp_term_taxonomy
| wp_terms
| wp_usermeta
| wp_users
+-----+
11 rows in set (0.02 sec)
```

Wp\_users content;

```

mysql> select * from wp_users;
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| ID | user_login | user_pass           | user_nicename | user_email      | user_url          | user_registered | user_activation_key | user_status | display_name |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1  | admin     | $P$0SScenYMDuAldinorzuM7QdK2AAk/ | admin        | admin@abc.com   | http://www.admin.com | 0000-00-00 00:00:00 |                | 0            | Admin       |
| 2  | james     | cebbc970658F31504a901b9dcd4e61 | james        | jamesfaulkner@gmail.com | http://www.jameswebsite.com | 0000-00-00 00:00:00 |                | 0            | jamesfaulkner |
| 125 | bad_guy   | $P$0.OMYbJJAsOyP2EYS.b6.d0xnk8Ke/ | anonymous_hacker | badguy@xyz.com | http://www.badguyxyz.com | 0000-00-00 00:00:00 |                | 0            | bad_guy    |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
1 rows in set (0.00 sec)

mysql>

```

Saved evidence in evidence.txt:

```

mysql> select * from wp_posts
      -> where post_author = '125'
      -> into outfile 'c:/wamp64/tmp/evidence.txt';
Query OK, 3 rows affected (0.01 sec)

```

Hex editor:

Offset(h)	00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F	Decoded text
000020C0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000020D0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000020E0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
000020F0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
00002100	01 00 00 C5 00 36 08 00 22 08 79 00 00 00	....Ã...y...
00002110	00 00 00 00 00 50 00 16 00 00 00 00 00 00 00 00	.....P.....
00002120	C5 00 0B 0A 02 14 29 20 20 20 20 20 20 20 20 20	Ã....)
00002130	20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20	.....
00002140	20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20	.....
00002150	04 00 03 49 44 00 05 00 0B 75 73 65 72 5F 6C 63	...ID...user lo
00002160	67 69 C8 00 06 00 0A 75 73 65 72 5F 70 61 73 73	...user_pass
00002170	00 07 00 0E 75 73 65 72 5F 6E 69 63 65 6E 61 6D	....user_nicename
00002180	65 00 08 00 0B 75 73 65 72 5F 65 6D 61 69 6C 00	....user_email.
00002190	09 00 09 75 73 65 72 5F 75 72 6C 00 0A 00 10 75	....user_url....u
000021A0	73 65 72 5F 72 65 67 69 73 74 65 72 65 64 00 0B	ser_registered..
000021B0	00 14 75 73 65 72 5F 61 63 74 69 76 61 74 69 6F	..user_activatio
000021C0	6E 5F 6B 65 79 00 0C 00 75 73 65 72 5F 73 74	n_key....user_st
000021D0	61 74 75 73 00 0D 00 64 69 73 70 6C 61 79 5F	atus....display_
000021E0	6E 61 6D 65 00 04 03 14 14 00 01 00 00 42 00 0F	name.....B..
000021F0	00 00 08 21 00 00 05 0B 44 84 00 09 00 00 00 00	!....D.....
00002200	00 00 00 0F 21 00 00 06 0A 45 C0 00 BE 00 00 00	!....EA....
00002210	00 00 00 00 0F 21 00 00 07 0E 41 96 00 7F 01 00	....!....A....
00002220	00 00 00 00 00 0F 21 00 00 08 0B 44 2C 01 16 02	....!....D,...
00002230	00 00 00 00 00 00 0F 21 00 00 09 09 46 2C 01 44	....!....F,...D
00002240	03 00 00 00 00 00 00 0F 21 00 00 0A 10 13 13 00	....!....
00002250	72 04 00 60 00 00 00 00 0C 08 00 00 0B 14 3B B4	r.'....;

Got details from mysqlbin.000034

Offset	Value	Description
000005B0	00 00 00 00 00 00 00 00 00 00 1A 00 00 00 00 00	.....
000005C0	01 00 00 00 00 00 00 00 00 06 03 73 74 64 04 21	.....wordpress.std!
000005D0	00 21 00 00 00 77 6F 72 64 70 72 65 73 00 49	!....wordpress.I
000005E0	4E 53 45 52 54 20 49 4E 54 4F 20 60 77 70 57 75	INSERT INTO `wp_u
000005F0	73 65 72 73 60 20 28 60 75 73 65 72 5F 67 67	users` ('user_log
00000600	69 68 60 20 20 60 75 73 65 72 5F 70 61 73 60	in', 'user_pass'
00000610	2C 20 60 75 73 65 72 5F 6E 69 63 65 6E 61 6D 65	, 'user_nicename
00000620	60 20 60 75 73 65 72 5F 65 6D 61 69 6C 60 2C	, 'user_email',
00000630	20 60 75 73 65 72 5F 73 74 61 74 75 73 69 29 0A	'user_status').
00000640	56 41 4C 55 45 53 20 28 27 62 61 64 5F 67 75 73	VALUES ('bad_guy
00000650	27 20 20 49 44 35 28 27 70 61 73 73 31 32 33 27	, MD5('panel23')
00000660	29 20 20 27 61 6E 6F 79 6D 68 75 73 5F 68 61	), 'anonymous_ha
00000670	63 68 65 72 27 2C 20 27 62 61 64 67 75 79 40 78	cker', 'badguy@x
00000680	79 7A 2E 63 6F 6D 27 2C 20 27 30 27 29 C5 B2 5F	yz.com', '0')Ã_
00000690	57 10 01 00 00 00 1B 00 00 00 A8 06 00 00 00 00	W.....
000006A0	12 02 00 00 00 00 00 D0 B2 5F 57 02 01 00 00	....!....W....
000006B0	00 49 00 00 00 F4 06 00 00 08 00 13 00 00 00 00	.I....
000006C0	00 00 00 00 00 00 1A 00 00 00 00 00 01 00 00	....
000006D0	00 00 00 00 00 00 00 00 03 73 74 64 04 21 00 21 00	.....std.!..
000006E0	08 00 77 6F 72 64 70 72 65 73 73 00 42 45 47 49	..wordpress.BEGIN
000006F0	4E D3 B2 5F 57 13 01 00 00 00 3D 00 00 00 2E 07	NO! W.....
00000700	00 00 00 00 0A 00 00 00 00 01 00 00 09 77 6F 72	....!

Username - bad\_guy

Password – pass123

Nice name – anonymous\_hacker

Email ID – badguy@xyz.com

## Additional info

mn.nv	mn.tu	mn.ac
Uln16	go,to	28528
Int24	go,to	7565168
Uln24	go,to	7565168
Int32	go,to	1953722224
Uln32	go,to	1953722224
Int32	go,to	8391720543278296944
Uln64	go,to	8391720543278296944
Uln64	go,to	LEB128
LEB128	go,to	-16
ULEB128	go,to	112
AnsChar / char8_t	p	
WideChar / char16_t	#f	
UTF-8 code point	p (U-0070)	
Single (float32)	7.71477269069928E31	
Double (float64)	9.7996924845332E252	
OLETIME	Invalid	
FILETIME	Invalid	
DOS date	16-11-2035	
DOS time	13:59:32	
DOS time & date	19-03-2038 13:59:32	
time_t (32 bit)	29-11-2031 12:37:04	
time_t (64 bit)	Invalid	
GUID	Invalid	
Disassembly (x86-16)	jo \$00000071	
Disassembly (x86-32)	jo \$00000071	
Disassembly (x86-64)	jo \$00000071	
Byte order		
<input checked="" type="radio"/> Little endian	<input type="radio"/> Big endian	
<input type="checkbox"/> Hexadecimal basis (for integral numbers)		