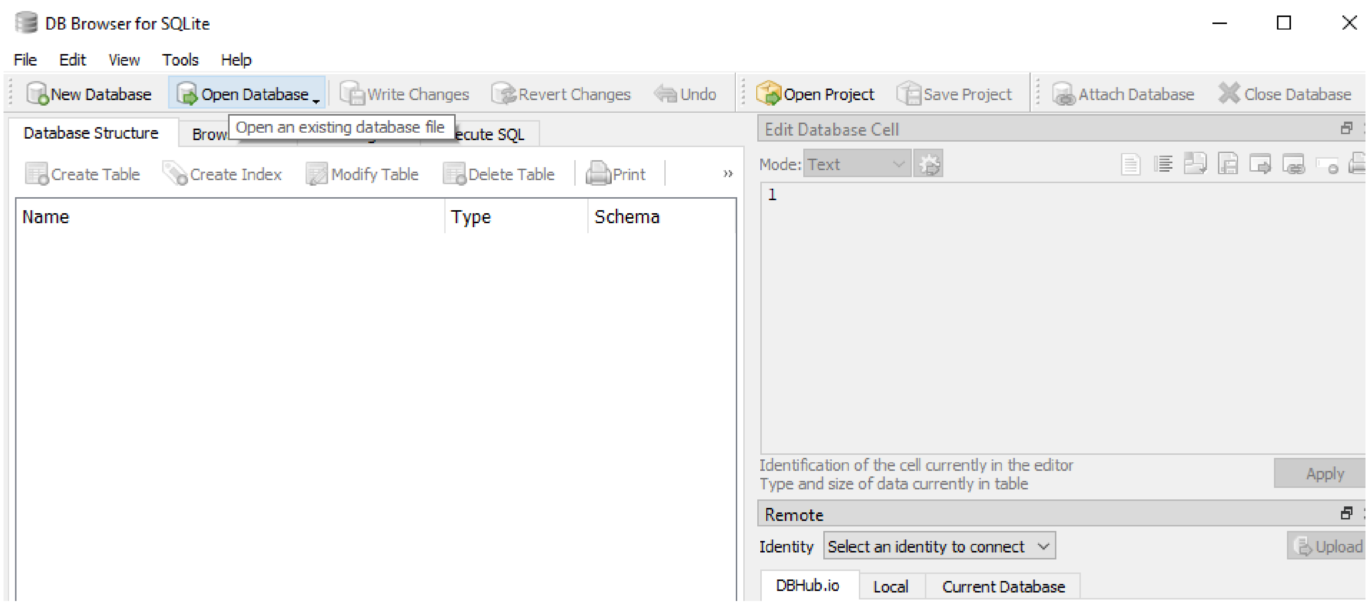
Cyber Forensics - 24CY611
Lab 8 - Database Forensics

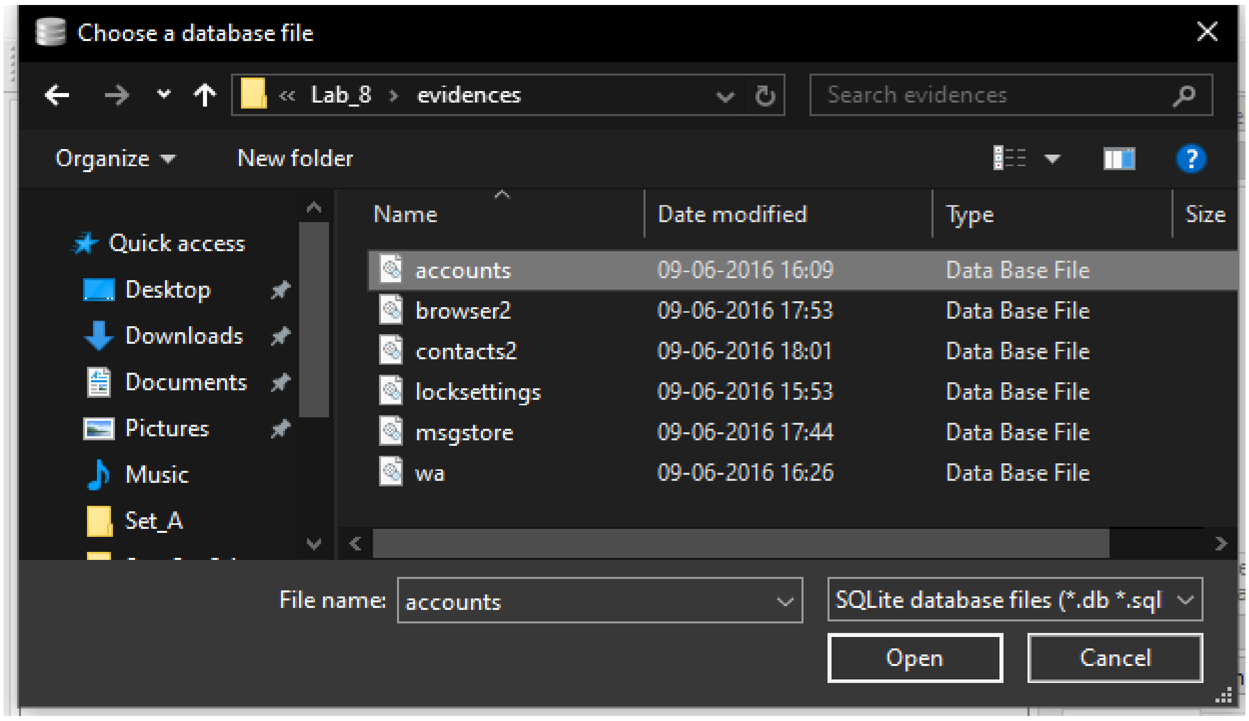
Analyzing SQLite Databases using DB Browser for SQLite

Start the wamp server

Open DB Browser of SQLite. Click open database in the toolbar



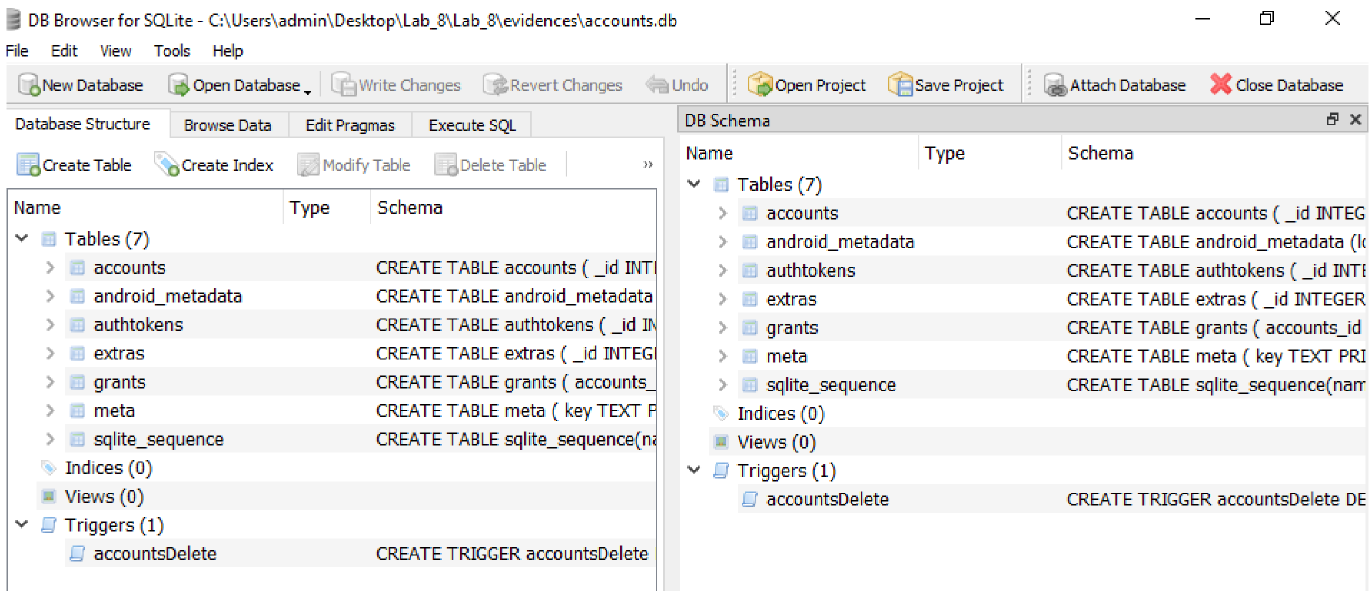
Choose the database file (account.db) and click open



Name	Date modified	Type	Size
accounts	09-06-2016 16:09	Data Base File	
browser2	09-06-2016 17:53	Data Base File	
contacts2	09-06-2016 18:01	Data Base File	
locksettings	09-06-2016 15:53	Data Base File	
msgstore	09-06-2016 17:44	Data Base File	
wa	09-06-2016 16:26	Data Base File	

File name: accounts

The application displays the structure of accounts database under the Database Structure tab as shown in the following screenshot:

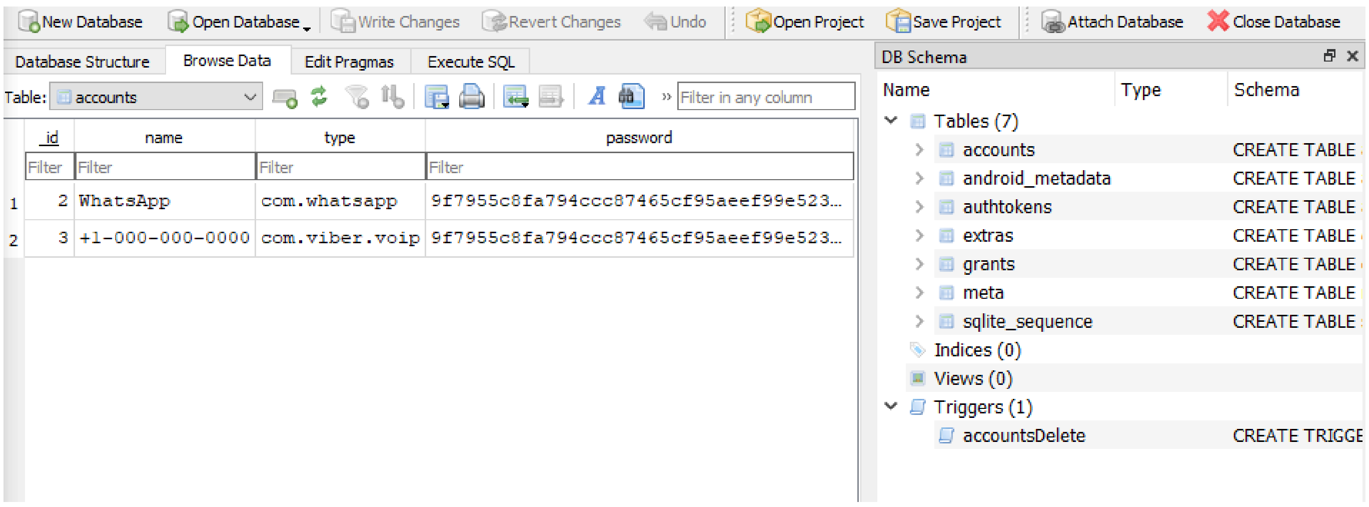


Name	Type	Schema
Tables (7)		
accounts		CREATE TABLE accounts (_id INTI
android_metadata		CREATE TABLE android_metadata
authtokens		CREATE TABLE authtokens (_id IN
extras		CREATE TABLE extras (_id INTEGI
grants		CREATE TABLE grants (accounts_
meta		CREATE TABLE meta (key TEXT P
sqlite_sequence		CREATE TABLE sqlite_sequence(na
Indices (0)		
Views (0)		
Triggers (1)		
accountsDelete		CREATE TRIGGER accountsDelete

Name	Type	Schema
Tables (7)		
accounts		CREATE TABLE accounts (_id INTEG
android_metadata		CREATE TABLE android_metadata (lo
authtokens		CREATE TABLE authtokens (_id INTE
extras		CREATE TABLE extras (_id INTEGER
grants		CREATE TABLE grants (accounts_id
meta		CREATE TABLE meta (key TEXT PRI
sqlite_sequence		CREATE TABLE sqlite_sequence(nam
Indices (0)		
Views (0)		
Triggers (1)		
accountsDelete		CREATE TRIGGER accountsDelete DE

Click Browse Data tab to view the data in the accounts database.

Once you click the tab button, the accounts table will be selected by default and the table contents (the accounts synchronized with the device) will be displayed under the Table section, and the database schema will be displayed in the right pane of the UI as shown in the following screenshot:



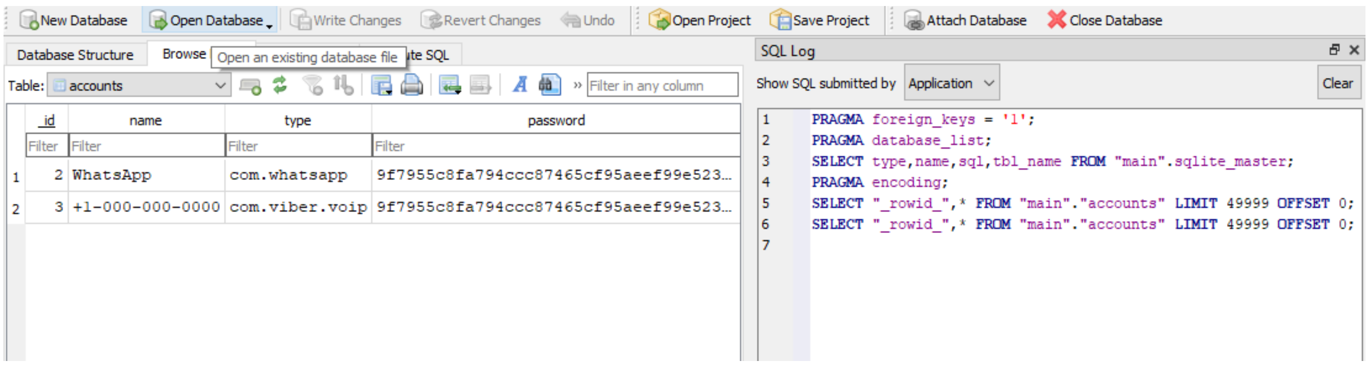
	_id	name	type	password
	Filter	Filter	Filter	Filter
1	2	WhatsApp	com.whatsapp	9f7955c8fa794ccc87465cf95aeef99e523...
2	3	+1-000-000-0000	com.viber.voip	9f7955c8fa794ccc87465cf95aeef99e523...

Name	Type	Schema
Tables (7)		
accounts		CREATE TABLE
android_metadata		CREATE TABLE
authtokens		CREATE TABLE
extras		CREATE TABLE
grants		CREATE TABLE
meta		CREATE TABLE
sqlite_sequence		CREATE TABLE
Indices (0)		
Views (0)		
Triggers (1)		
accountsDelete		CREATE TRIGGE

We can observe that the device was synchronized with two accounts: **WhatsApp and Viber.**

In the same way, you may also view the contents of other tables by selecting them from the Table drop-down list.

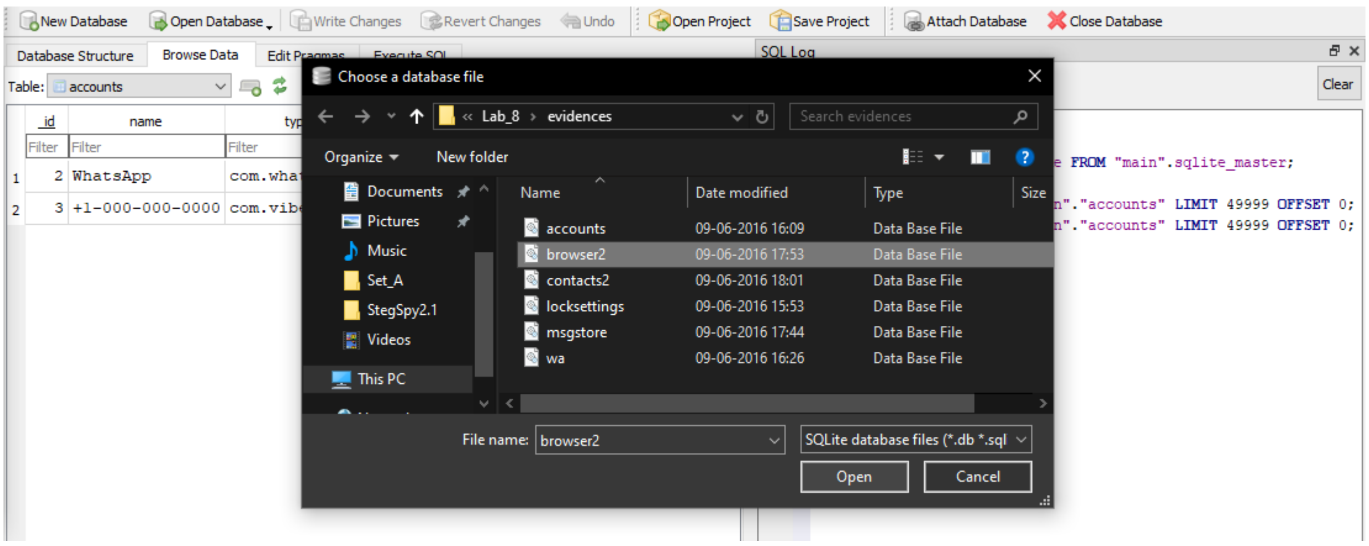
Now, we shall view the information stored in the browser database. To go to the database, click Open Database from the toolbar.



	_id	name	type	password
1	2	WhatsApp	com.whatsapp	9f7955c8fa794ccc87465cf95aeef99e523...
2	3	+1-000-000-0000	com.viber.voip	9f7955c8fa794ccc87465cf95aeef99e523...

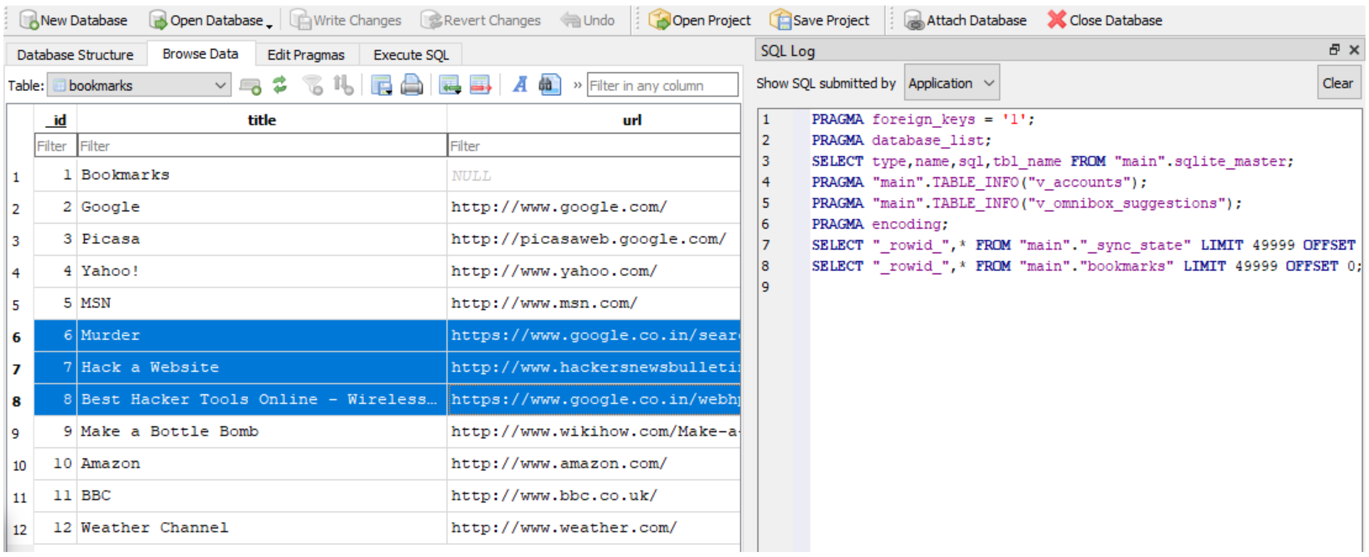
```
PRAGMA foreign_keys = '1';
PRAGMA database_list;
SELECT type,name,sql,tbl_name FROM "main".sqlite_master;
PRAGMA encoding;
SELECT "_rowid_",* FROM "main"."accounts" LIMIT 49999 OFFSET 0;
SELECT "_rowid_",* FROM "main"."accounts" LIMIT 49999 OFFSET 0;
```

Choose the database file (browser2.db) and click open



Name	Date modified	Type
accounts	09-06-2016 16:09	Data Base File
browser2	09-06-2016 17:53	Data Base File
contacts2	09-06-2016 18:01	Data Base File
locksettings	09-06-2016 15:53	Data Base File
msgstore	09-06-2016 17:44	Data Base File
wa	09-06-2016 16:26	Data Base File

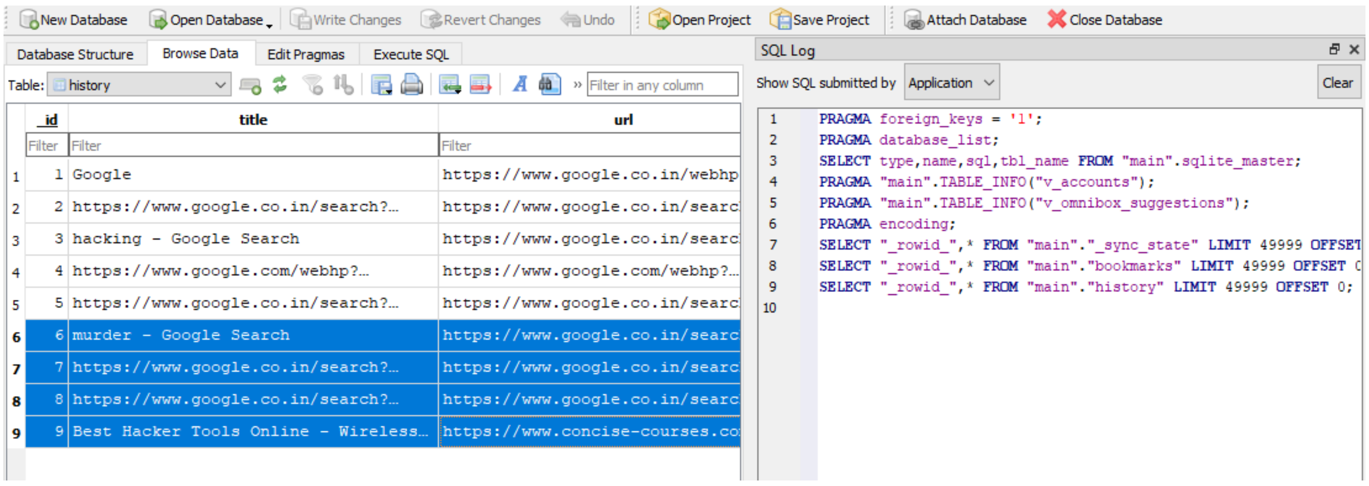
Select Browse Data tab and select bookmarks table from the Table drop-down list. This displays all the URLs that were bookmarked on the device as shown in the following screenshot:



	_id	title	url
1	1	Bookmarks	NULL
2	2	Google	http://www.google.com/
3	3	Picasa	http://picasaweb.google.com/
4	4	Yahoo!	http://www.yahoo.com/
5	5	MSN	http://www.msn.com/
6	6	Murder	https://www.google.co.in/sear
7	7	Hack a Website	http://www.hackersnewsbulleti
8	8	Best Hacker Tools Online - Wireless...	https://www.google.co.in/webh
9	9	Make a Bottle Bomb	http://www.wikihow.com/Make-a
10	10	Amazon	http://www.amazon.com/
11	11	BBC	http://www.bbc.co.uk/
12	12	Weather Channel	http://www.weather.com/

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

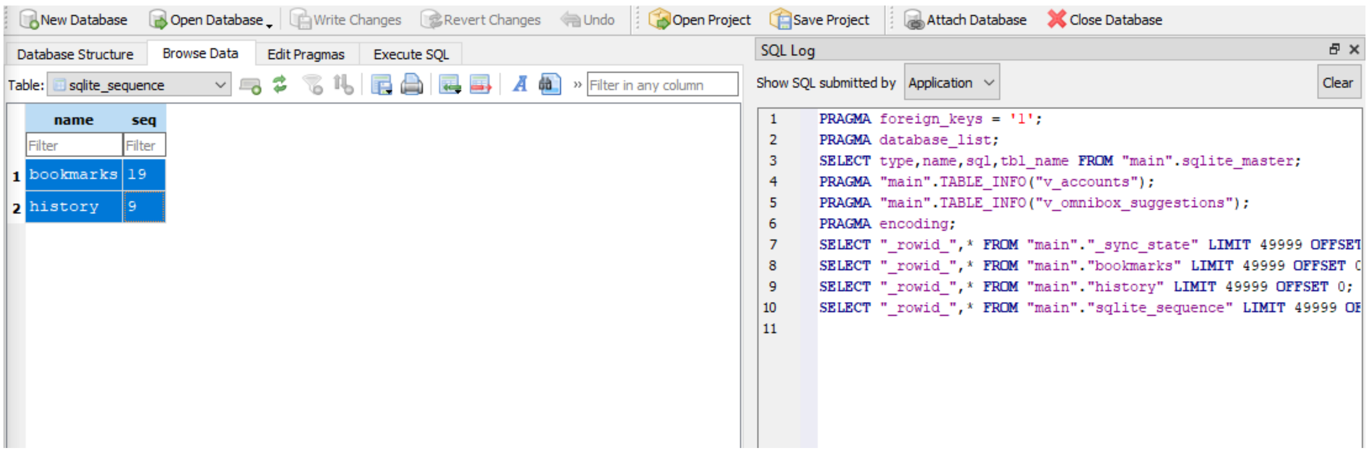
Select history table from the Table drop-down list to view the browser history.



	_id	title	url
1	1	Google	https://www.google.co.in/webhp
2	2	https://www.google.co.in/search?...	https://www.google.co.in/searc
3	3	hacking - Google Search	https://www.google.co.in/searc
4	4	https://www.google.com/webhp?...	https://www.google.com/webhp?...
5	5	https://www.google.co.in/search?...	https://www.google.co.in/searc
6	6	murder - Google Search	https://www.google.co.in/searc
7	7	https://www.google.co.in/search?...	https://www.google.co.in/searc
8	8	https://www.google.co.in/search?...	https://www.google.co.in/searc
9	9	Best Hacker Tools Online - Wireless...	https://www.concise-courses.co

```
PRAGMA foreign_keys = '1';
PRAGMA database_list;
SELECT type,name,sql,tbl_name FROM "main".sqlite_master;
PRAGMA "main".TABLE_INFO("v_accounts");
PRAGMA "main".TABLE_INFO("v_omnibox_suggestions");
PRAGMA encoding;
SELECT "_rowid_",* FROM "main"."_sync_state" LIMIT 49999 OFFSET
SELECT "_rowid_",* FROM "main"."bookmarks" LIMIT 49999 OFFSET 0
SELECT "_rowid_",* FROM "main"."history" LIMIT 49999 OFFSET 0;
```

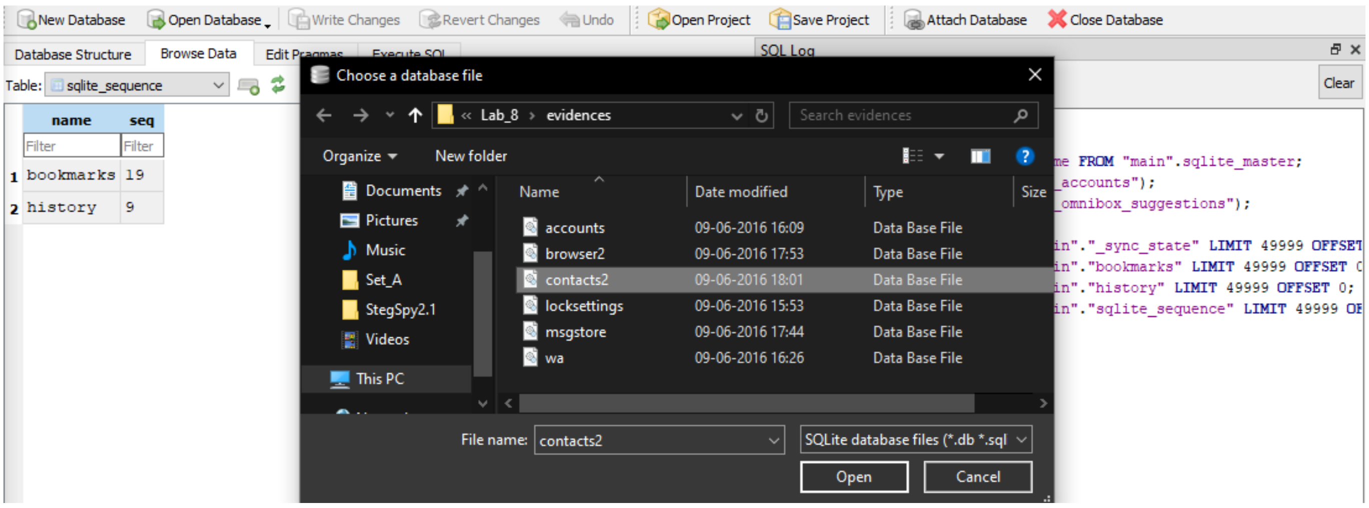
The sqlite_sequence table stores information related to history (number of websites browsed) and bookmarks (number of websites bookmarked). To view this data, select sqlite_sequence table from the Table drop-down list.



	name	seq
1	bookmarks	19
2	history	9

```
PRAGMA foreign_keys = '1';
PRAGMA database_list;
SELECT type,name,sql,tbl_name FROM "main".sqlite_master;
PRAGMA "main".TABLE_INFO("v_accounts");
PRAGMA "main".TABLE_INFO("v_omnibox_suggestions");
PRAGMA encoding;
SELECT "_rowid_",* FROM "main"."_sync_state" LIMIT 49999 OFFSET
SELECT "_rowid_",* FROM "main"."bookmarks" LIMIT 49999 OFFSET 0
SELECT "_rowid_",* FROM "main"."history" LIMIT 49999 OFFSET 0;
SELECT "_rowid_",* FROM "main"."sqlite_sequence" LIMIT 49999 OF
```

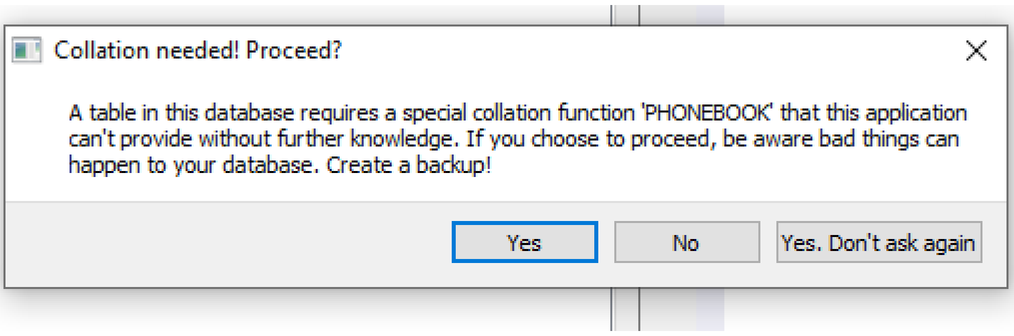
To view the database, click Open Database from the toolbar. Choose a database file window appears. Select and click open



	name	seq
1	bookmarks	19
2	history	9

Name	Date modified	Type	Size
accounts	09-06-2016 16:09	Data Base File	
browser2	09-06-2016 17:53	Data Base File	
contacts2	09-06-2016 18:01	Data Base File	
locksettings	09-06-2016 15:53	Data Base File	
msgstore	09-06-2016 17:44	Data Base File	
wa	09-06-2016 16:26	Data Base File	

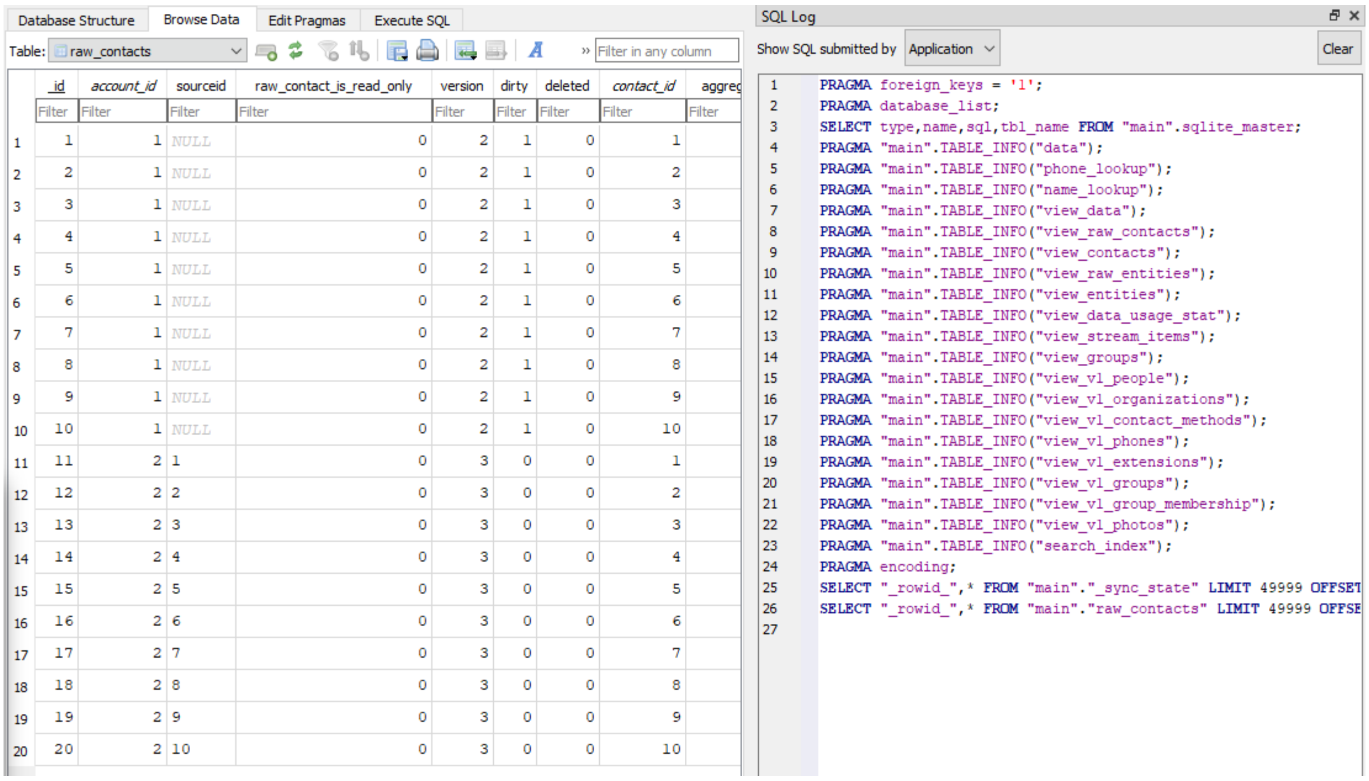
If a dialog-box appears stating that a table in the database requires a special collation function, click Yes to proceed without the collation



Collation needed! Proceed?

A table in this database requires a special collation function 'PHONEBOOK' that this application can't provide without further knowledge. If you choose to proceed, be aware bad things can happen to your database. Create a backup!

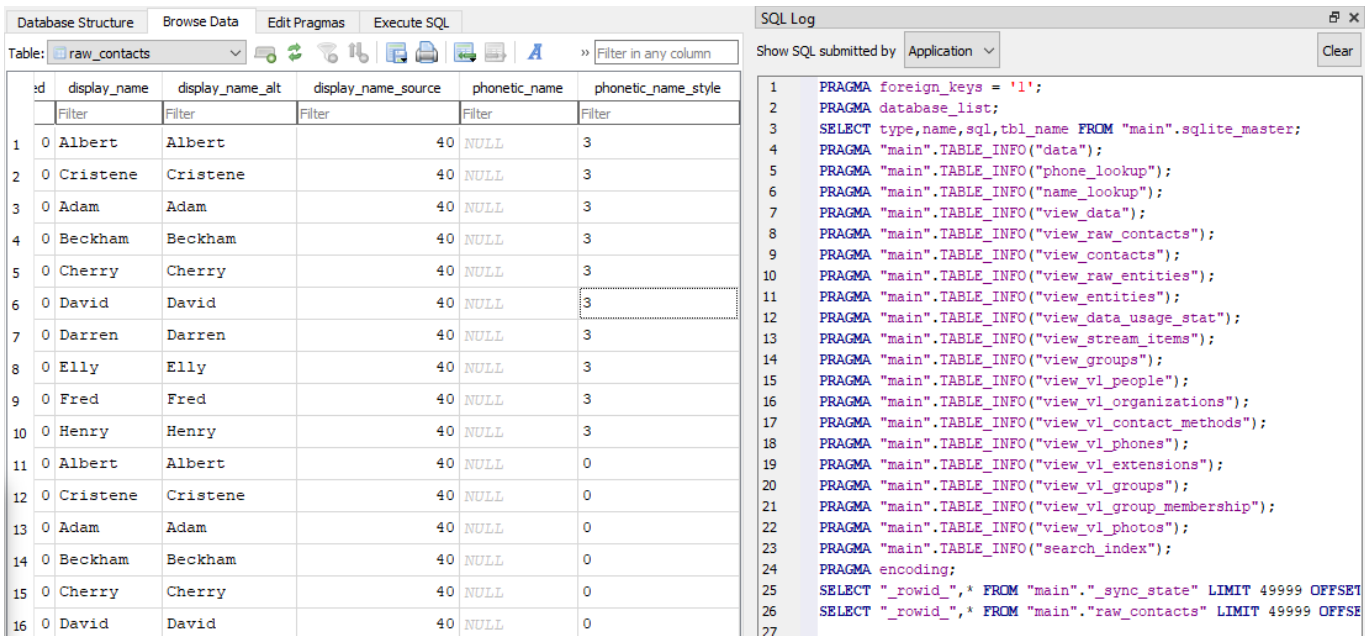
The application displays the _sync_state table by default. To view the contacts stored in the database, select raw_contacts table from the Table drop-down list. The raw_contacts table stores information such as display name, account id, last time contacted, etc.



	_id	account_id	sourceid	raw_contact_is_read_only	version	dirty	deleted	contact_id	aggreg
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	1	NULL	0	2	1	0	1	
2	2	1	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	2	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	

```
PRAGMA foreign_keys = '1';
PRAGMA database_list;
SELECT type,name,sql,tbl_name FROM "main".sqlite_master;
PRAGMA "main".TABLE_INFO("data");
PRAGMA "main".TABLE_INFO("phone_lookup");
PRAGMA "main".TABLE_INFO("name_lookup");
PRAGMA "main".TABLE_INFO("view_data");
PRAGMA "main".TABLE_INFO("view_raw_contacts");
PRAGMA "main".TABLE_INFO("view_contacts");
PRAGMA "main".TABLE_INFO("view_raw_entities");
PRAGMA "main".TABLE_INFO("view_entities");
PRAGMA "main".TABLE_INFO("view_data_usage_stat");
PRAGMA "main".TABLE_INFO("view_stream_items");
PRAGMA "main".TABLE_INFO("view_groups");
PRAGMA "main".TABLE_INFO("view_v1_people");
PRAGMA "main".TABLE_INFO("view_v1_organizations");
PRAGMA "main".TABLE_INFO("view_v1_contact_methods");
PRAGMA "main".TABLE_INFO("view_v1_phones");
PRAGMA "main".TABLE_INFO("view_v1_extensions");
PRAGMA "main".TABLE_INFO("view_v1_groups");
PRAGMA "main".TABLE_INFO("view_v1_group_membership");
PRAGMA "main".TABLE_INFO("view_v1_photos");
PRAGMA "main".TABLE_INFO("search_index");
PRAGMA encoding;
SELECT "_rowid_",* FROM "main"."_sync_state" LIMIT 49999 OFFSET
SELECT "_rowid_",* FROM "main"."raw_contacts" LIMIT 49999 OFFSE
```

You may scroll down and scroll to the right of the table to view the data stored in the table

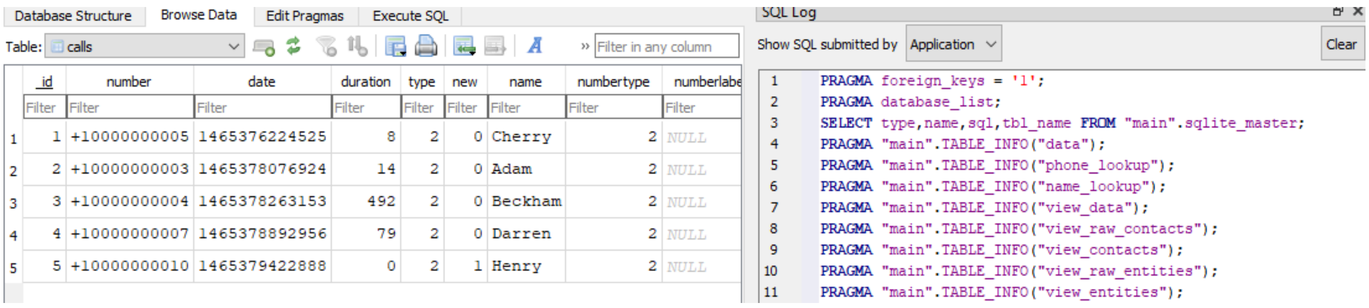


	d	display_name	display_name_alt	display_name_source	phonetic_name	phonetic_name_style
1	0	Albert	Albert	40	NULL	3
2	0	Cristene	Cristene	40	NULL	3
3	0	Adam	Adam	40	NULL	3
4	0	Beckham	Beckham	40	NULL	3
5	0	Cherry	Cherry	40	NULL	3
6	0	David	David	40	NULL	3
7	0	Darren	Darren	40	NULL	3
8	0	Elly	Elly	40	NULL	3
9	0	Fred	Fred	40	NULL	3
10	0	Henry	Henry	40	NULL	3
11	0	Albert	Albert	40	NULL	0
12	0	Cristene	Cristene	40	NULL	0
13	0	Adam	Adam	40	NULL	0
14	0	Beckham	Beckham	40	NULL	0
15	0	Cherry	Cherry	40	NULL	0
16	0	David	David	40	NULL	0

```
PRAGMA foreign_keys = '1';
PRAGMA database_list;
SELECT type,name,sql,tbl_name FROM "main".sqlite_master;
PRAGMA "main".TABLE_INFO("data");
PRAGMA "main".TABLE_INFO("phone_lookup");
PRAGMA "main".TABLE_INFO("name_lookup");
PRAGMA "main".TABLE_INFO("view_data");
PRAGMA "main".TABLE_INFO("view_raw_contacts");
PRAGMA "main".TABLE_INFO("view_contacts");
PRAGMA "main".TABLE_INFO("view_raw_entities");
PRAGMA "main".TABLE_INFO("view_entities");
PRAGMA "main".TABLE_INFO("view_data_usage_stat");
PRAGMA "main".TABLE_INFO("view_stream_items");
PRAGMA "main".TABLE_INFO("view_groups");
PRAGMA "main".TABLE_INFO("view_v1_people");
PRAGMA "main".TABLE_INFO("view_v1_organizations");
PRAGMA "main".TABLE_INFO("view_v1_contact_methods");
PRAGMA "main".TABLE_INFO("view_v1_phones");
PRAGMA "main".TABLE_INFO("view_v1_extensions");
PRAGMA "main".TABLE_INFO("view_v1_groups");
PRAGMA "main".TABLE_INFO("view_v1_group_membership");
PRAGMA "main".TABLE_INFO("view_v1_photos");
PRAGMA "main".TABLE_INFO("search_index");
PRAGMA encoding;
SELECT "_rowid_",* FROM "main"."_sync_state" LIMIT 49999 OFFSET
SELECT "_rowid_",* FROM "main"."raw_contacts" LIMIT 49999 OFFSE
```

The calls table contains the call history associated with the device. This table contains details such as the dialed numbers, dialed contact name, timestamp, call duration, etc.

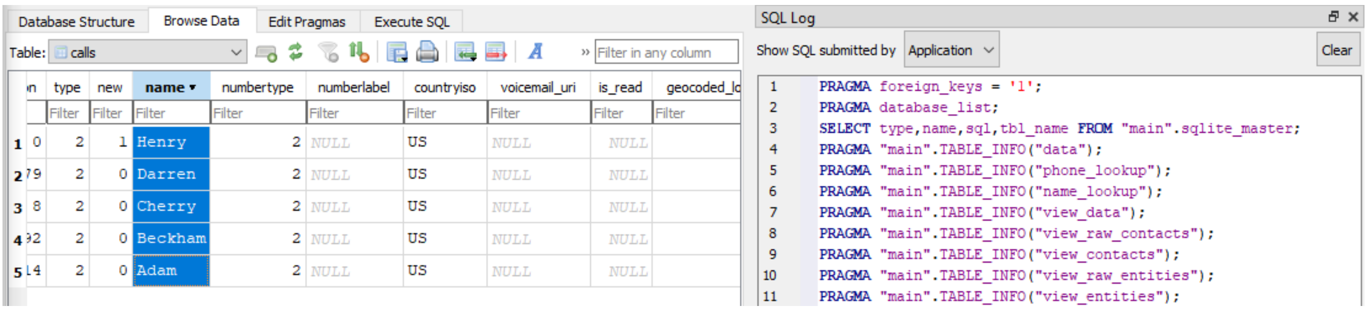
To view this information, select calls from the Table drop-down list.



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

```
PRAGMA foreign_keys = '1';
PRAGMA database_list;
SELECT type,name,sql,tbl_name FROM "main".sqlite_master;
PRAGMA "main".TABLE_INFO("data");
PRAGMA "main".TABLE_INFO("phone_lookup");
PRAGMA "main".TABLE_INFO("name_lookup");
PRAGMA "main".TABLE_INFO("view_data");
PRAGMA "main".TABLE_INFO("view_raw_contacts");
PRAGMA "main".TABLE_INFO("view_contacts");
PRAGMA "main".TABLE_INFO("view_raw_entities");
PRAGMA "main".TABLE_INFO("view_entities");
```

You may scroll down and scroll to the right of the table to view the data stored in the table

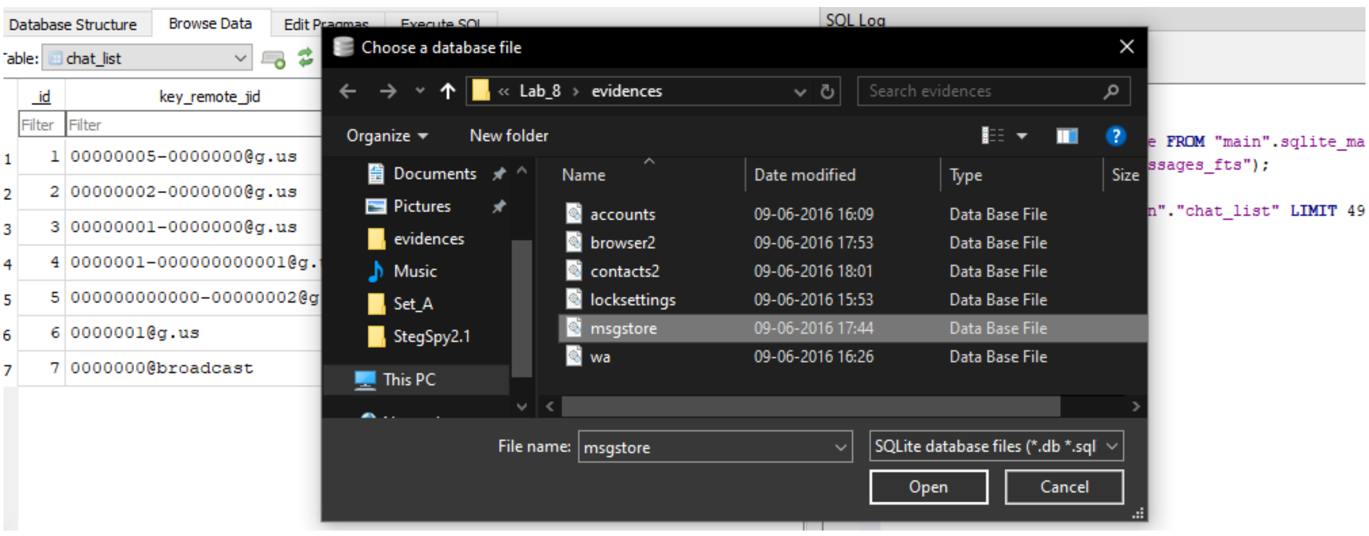


	n	type	new	name ▾	numbertype	numberlabel	countryiso	voicemail_uri	is_read	geocoded_lo
1	0	2	1	Henry	2	NULL	US	NULL	NULL	
2	79	2	0	Darren	2	NULL	US	NULL	NULL	
3	8	2	0	Cherry	2	NULL	US	NULL	NULL	
4	32	2	0	Beckham	2	NULL	US	NULL	NULL	
5	14	2	0	Adam	2	NULL	US	NULL	NULL	

```
PRAGMA foreign_keys = '1';
PRAGMA database_list;
SELECT type,name,sql,tbl_name FROM "main".sqlite_master;
PRAGMA "main".TABLE_INFO("data");
PRAGMA "main".TABLE_INFO("phone_lookup");
PRAGMA "main".TABLE_INFO("name_lookup");
PRAGMA "main".TABLE_INFO("view_data");
PRAGMA "main".TABLE_INFO("view_raw_contacts");
PRAGMA "main".TABLE_INFO("view_contacts");
PRAGMA "main".TABLE_INFO("view_raw_entities");
PRAGMA "main".TABLE_INFO("view_entities");
```

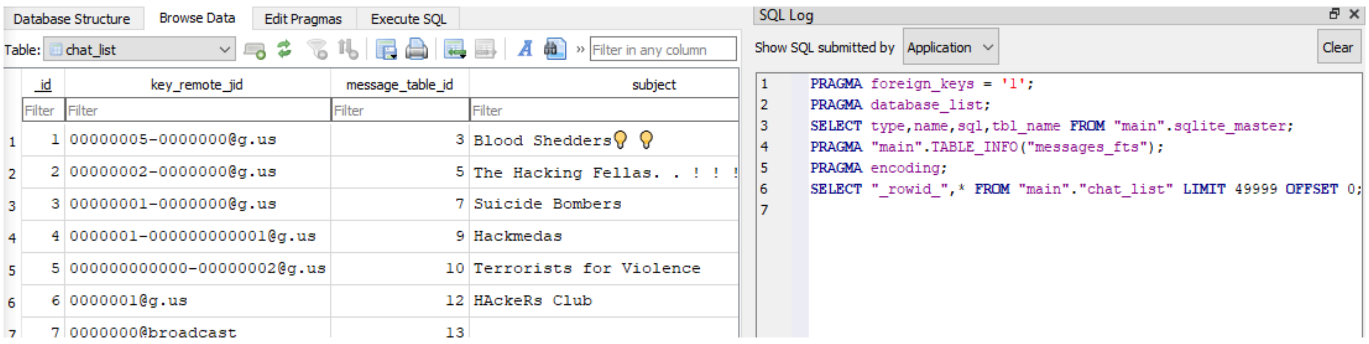
Now, we shall view the data stored in the msgstore database. The msgstore database contains information related to the messages stored on the device, such as timestamps of sent and received messages, subject of the message, etc.

To view this database, click Open Database from the toolbar. In the Choose a database file window, select msgstore.db and click open



Name	Date modified	Type
accounts	09-06-2016 16:09	Data Base File
browser2	09-06-2016 17:53	Data Base File
contacts2	09-06-2016 18:01	Data Base File
locksettings	09-06-2016 15:53	Data Base File
msgstore	09-06-2016 17:44	Data Base File
wa	09-06-2016 16:26	Data Base File

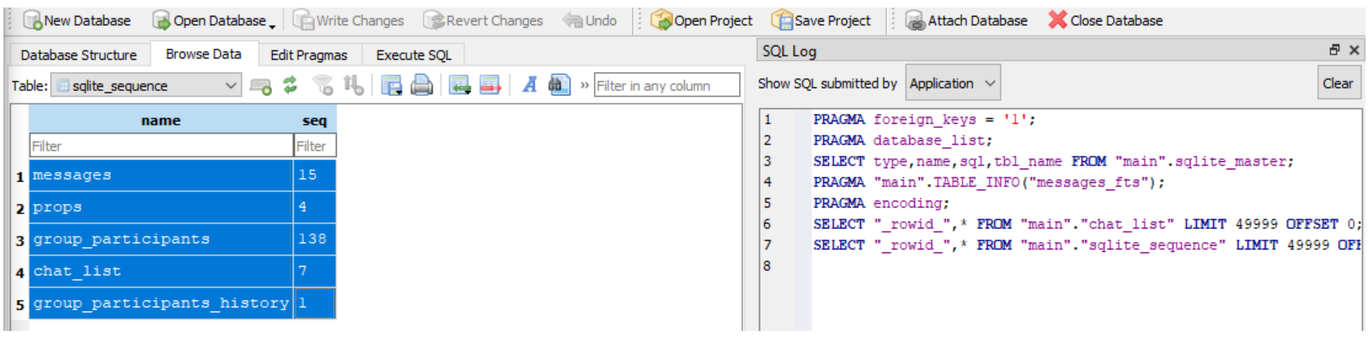
Select chat_list from the Table drop-down list. The chat_list table contains information such as subject of the message, key remote id, message creation time, etc., as shown in the following screenshot



	_id	key_remote_jid	message_table_id	subject
1	1	00000005-0000000@g.us	3	Blood Shedders💡 💡
2	2	00000002-0000000@g.us	5	The Hacking Fellas. . ! ! !
3	3	00000001-0000000@g.us	7	Suicide Bombers
4	4	0000001-000000000001@g.us	9	Hackmedas
5	5	000000000000-00000002@g.us	10	Terrorists for Violence
6	6	0000001@g.us	12	HAcKeRs Club
7	7	0000000@broadcast	13	

```
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;
```

In the same way, you may analyze the other tables in the database in order to find more information associated with the database



	name	seq
1	messages	15
2	props	4
3	group_participants	138
4	chat_list	7
5	group_participants_history	1

```
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"."sqlite_sequence" LIMIT 49999 OFF
```

Now, we shall view the data stored in WhatsApp database. The wa database contains information related to the WhatsApp messages stored on the device, timestamps of the sent and received messages, subject of the message, etc.

To view this information, click Open Database from the toolbar. Choose a database file window appears. Select wa.db and click open.

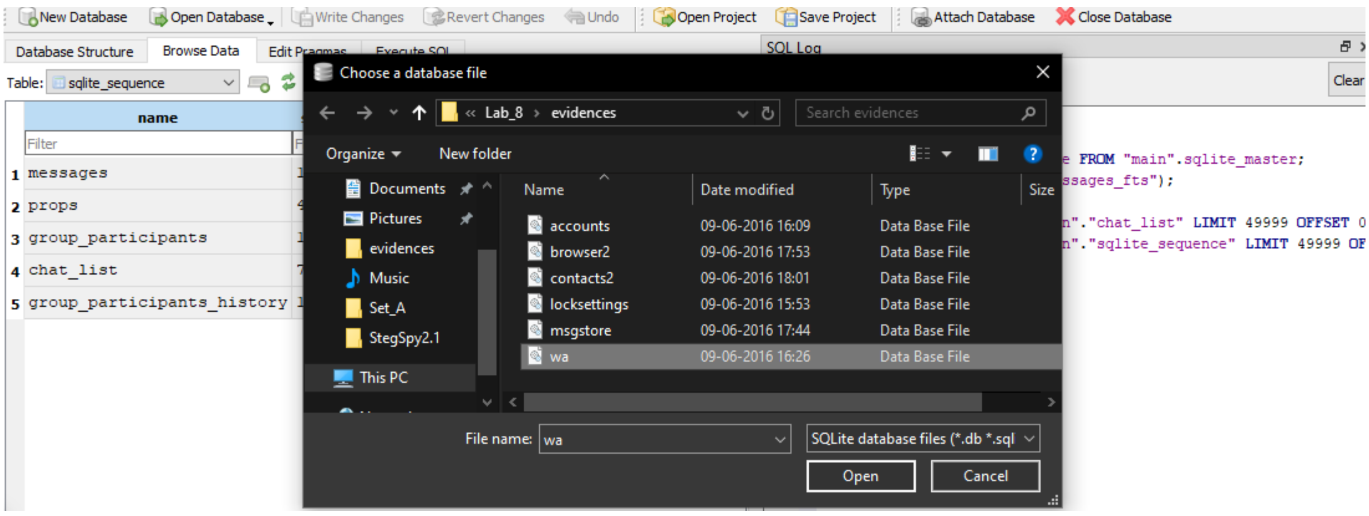


Table: sqlite_sequence

	name
1	messages
2	props
3	group_participants
4	chat_list
5	group_participants_history

Name	Date modified	Type
accounts	09-06-2016 16:09	Data Base File
browser2	09-06-2016 17:53	Data Base File
contacts2	09-06-2016 18:01	Data Base File
locksettings	09-06-2016 15:53	Data Base File
msgstore	09-06-2016 17:44	Data Base File
wa	09-06-2016 16:26	Data Base File

File name: wa

SQLite database files (*.db *.sql

```
e FROM "main".sqlite_master;
ssages_fts");
n"."chat_list" LIMIT 49999 OFFSET 0
n"."sqlite_sequence" LIMIT 49999 OF
```

You may browse various tables in the database to view information such as number of WhatsApp contacts, WhatsApp contacts' names, etc. as shown in the following screenshots.

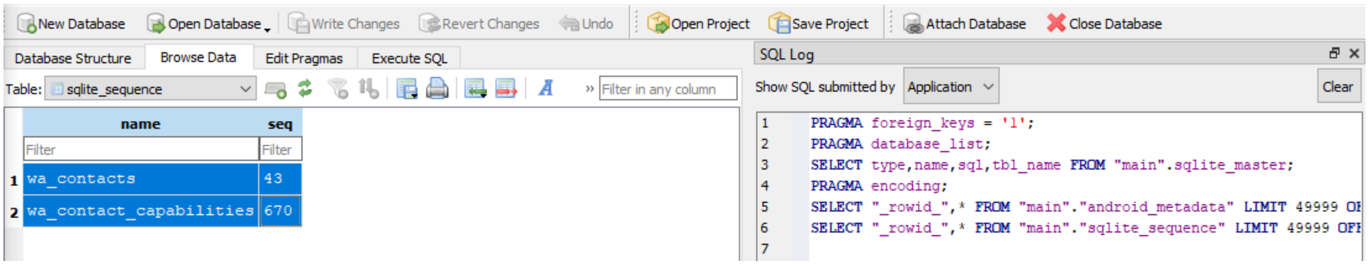


Table: sqlite_sequence

	name	seq
1	wa_contacts	43
2	wa_contact_capabilities	670

Show SQL submitted by: Application

```
PRAGMA foreign_keys = '1';
PRAGMA database_list;
SELECT type,name,sql,tbl_name FROM "main".sqlite_master;
PRAGMA encoding;
SELECT "_rowid_",* FROM "main"."android_metadata" LIMIT 49999 OF
SELECT "_rowid_",* FROM "main"."sqlite_sequence" LIMIT 49999 OFF
```

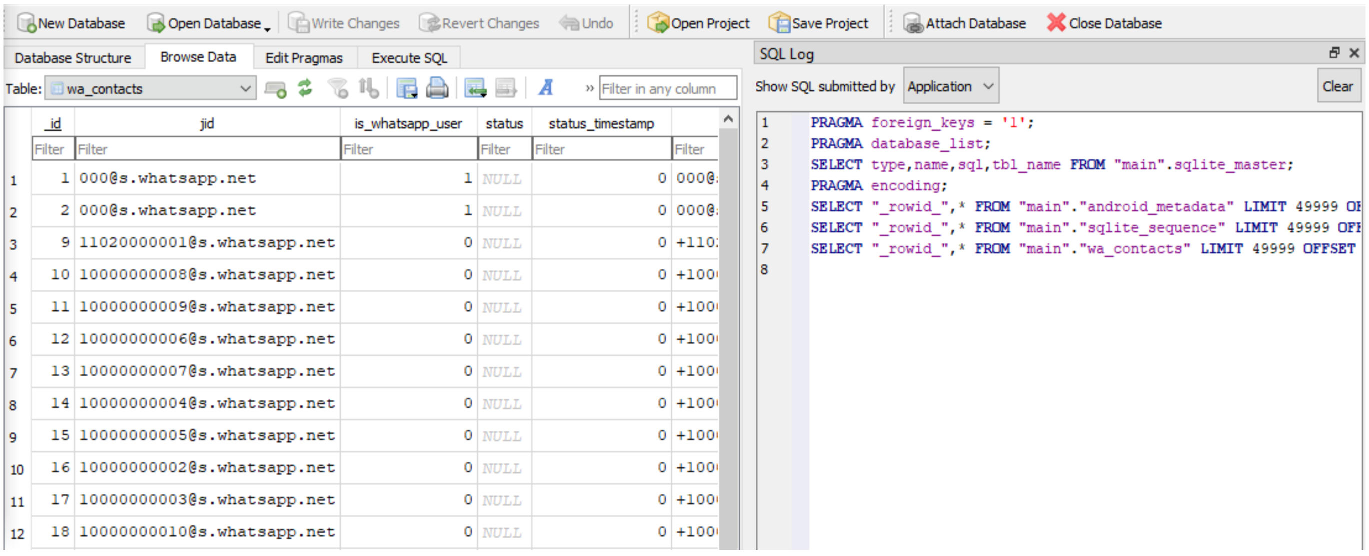


Table: wa_contacts

	_id	jid	is_whatsapp_user	status	status_timestamp	
1	1	000@s.whatsapp.net	1	NULL	0	000@
2	2	000@s.whatsapp.net	1	NULL	0	000@
3	9	11020000001@s.whatsapp.net	0	NULL	0	+110
4	10	10000000008@s.whatsapp.net	0	NULL	0	+100
5	11	10000000009@s.whatsapp.net	0	NULL	0	+100
6	12	10000000006@s.whatsapp.net	0	NULL	0	+100
7	13	10000000007@s.whatsapp.net	0	NULL	0	+100
8	14	10000000004@s.whatsapp.net	0	NULL	0	+100
9	15	10000000005@s.whatsapp.net	0	NULL	0	+100
10	16	10000000002@s.whatsapp.net	0	NULL	0	+100
11	17	10000000003@s.whatsapp.net	0	NULL	0	+100
12	18	10000000010@s.whatsapp.net	0	NULL	0	+100

Show SQL submitted by: Application

```
PRAGMA foreign_keys = '1';
PRAGMA database_list;
SELECT type,name,sql,tbl_name FROM "main".sqlite_master;
PRAGMA encoding;
SELECT "_rowid_",* FROM "main"."android_metadata" LIMIT 49999 OF
SELECT "_rowid_",* FROM "main"."sqlite_sequence" LIMIT 49999 OFF
SELECT "_rowid_",* FROM "main"."wa_contacts" LIMIT 49999 OFFSET
```

The locksettings database contains the settings such as the status of the lock screen, lockscreen password type, status of the lockscreen pattern autolock (enabled or disabled), visibility of the lockscreen pattern, etc.

To view this settings, click Open Database from the toolbar. Choose a database file window appears. Select **locksettings.db** and click open.

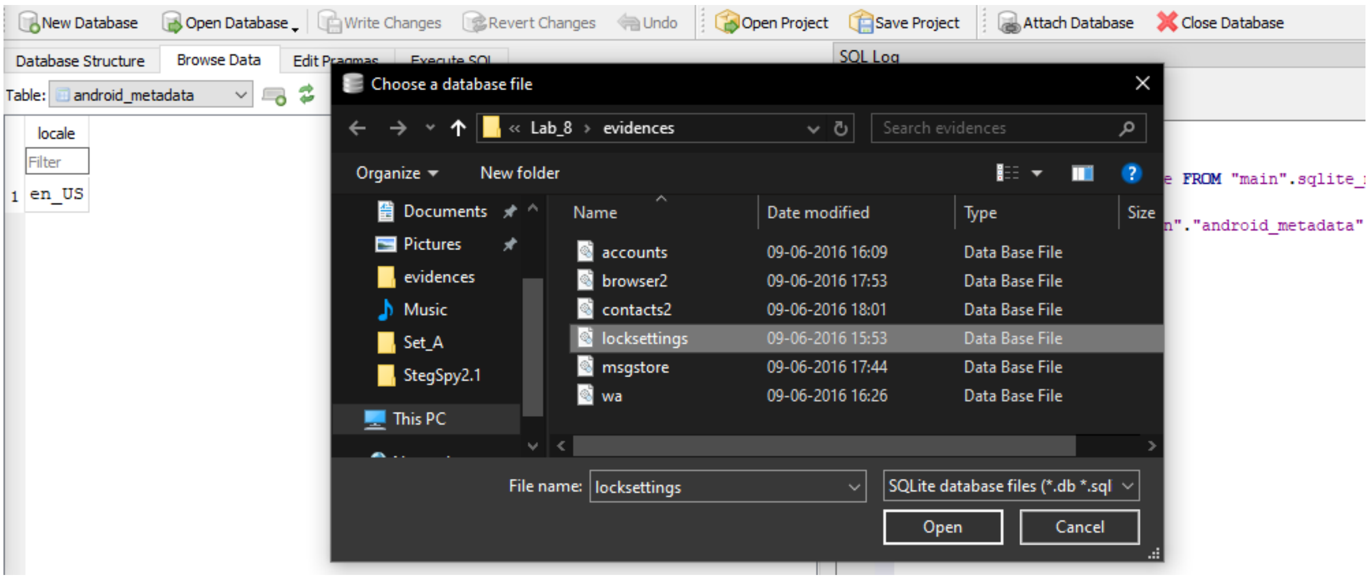


Table: android_metadata

	locale
1	en_US

Lab_8 › evidences

Name	Date modified	Type
accounts	09-06-2016 16:09	Data Base File
browser2	09-06-2016 17:53	Data Base File
contacts2	09-06-2016 18:01	Data Base File
locksettings	09-06-2016 15:53	Data Base File
msgstore	09-06-2016 17:44	Data Base File
wa	09-06-2016 16:26	Data Base File

File name: locksettings   SQLite database files (*.db *.sql

Select locksettings from the Table drop-down list, to view settings associated with the lock screen pattern as shown in the following screenshot:

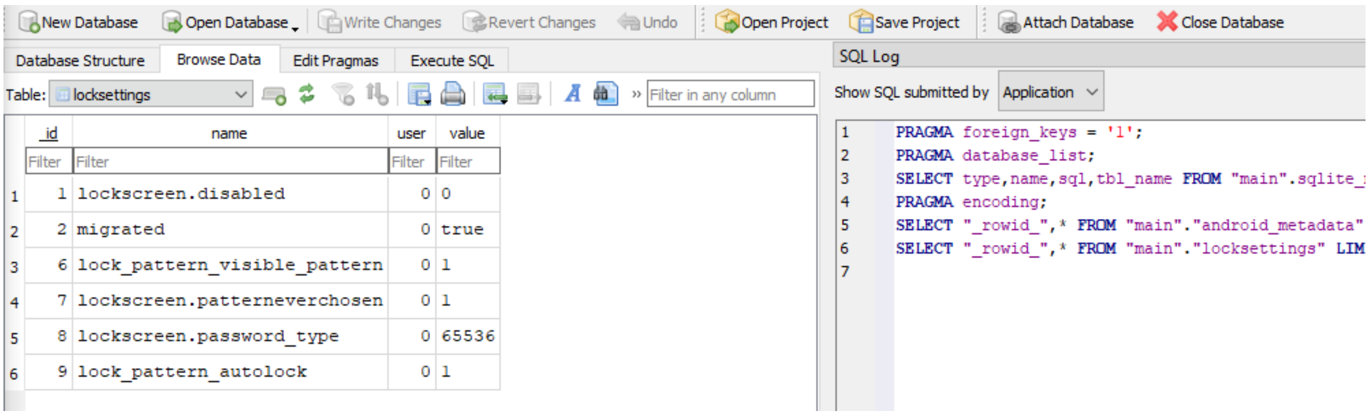


Table: locksettings

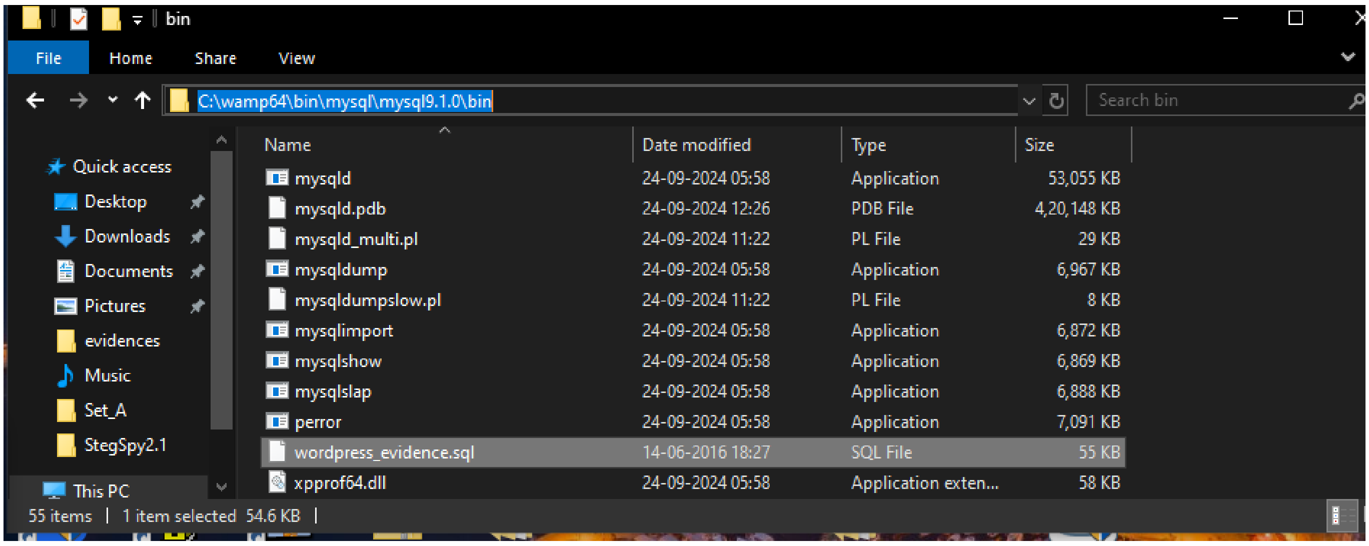
	_id	name	user	value
1	1	lockscreen.disabled	0	0
2	2	migrated	0	true
3	6	lock_pattern_visible_pattern	0	1
4	7	lockscreen.patterneverchosen	0	1
5	8	lockscreen.password_type	0	65536
6	9	lock_pattern_autolock	0	1

Show SQL submitted by Application

```
PRAGMA foreign_keys = '1';
PRAGMA database_list;
SELECT type,name,sql,tbl_name FROM "main".sqlite_
PRAGMA encoding;
SELECT "_rowid_",* FROM "main"."android_metadata"
SELECT "_rowid_",* FROM "main"."locksettings" LIM
```

Performing Forensics Investigation on a MySQL Server Database

Copy wordpress_evidence.sql and paste it in C:\wamp64\bin\mysql\mysql9.1.0\bin



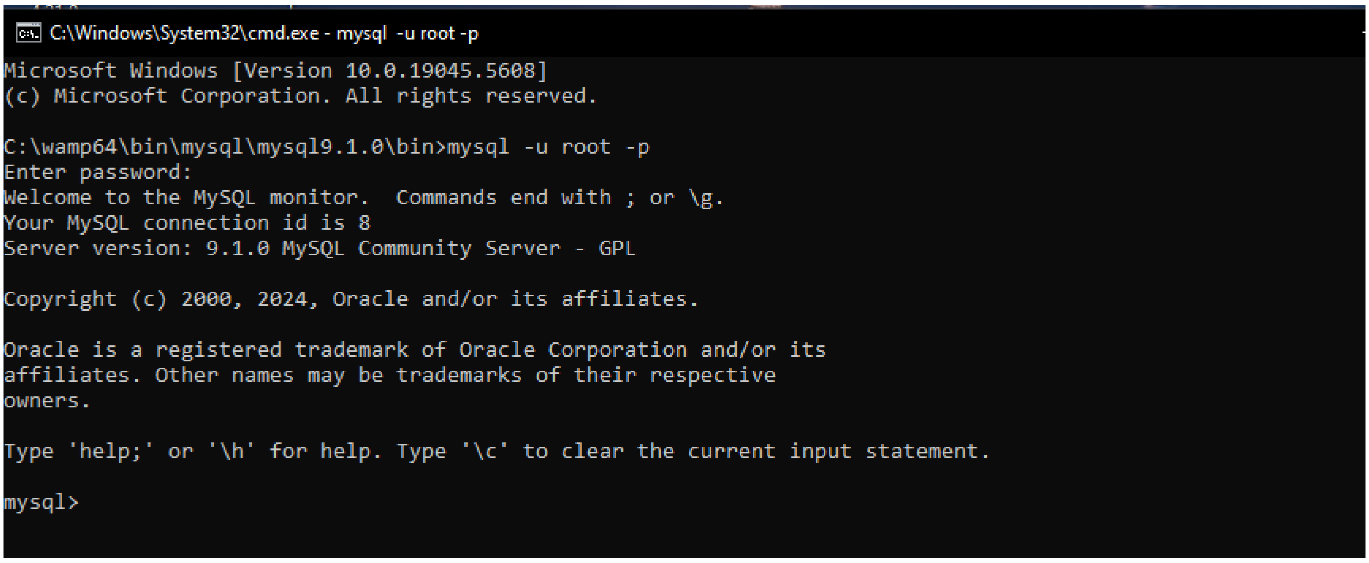
Name	Date modified	Type	Size
mysqld	24-09-2024 05:58	Application	53,055 KB
mysqld.pdb	24-09-2024 12:26	PDB File	4,20,148 KB
mysqld_multi.pl	24-09-2024 11:22	PL File	29 KB
mysqldump	24-09-2024 05:58	Application	6,967 KB
mysqldumpslow.pl	24-09-2024 11:22	PL File	8 KB
mysqlimport	24-09-2024 05:58	Application	6,872 KB
mysqlshow	24-09-2024 05:58	Application	6,869 KB
mysqlslap	24-09-2024 05:58	Application	6,888 KB
perror	24-09-2024 05:58	Application	7,091 KB
wordpress_evidence.sql	14-06-2016 18:27	SQL File	55 KB
xpprof64.dll	24-09-2024 05:58	Application exten...	58 KB

55 items | 1 item selected 54.6 KB |

Now, navigate to C:\wamp64\bin\mysql\mysql9.1.0\bin and open command prompt

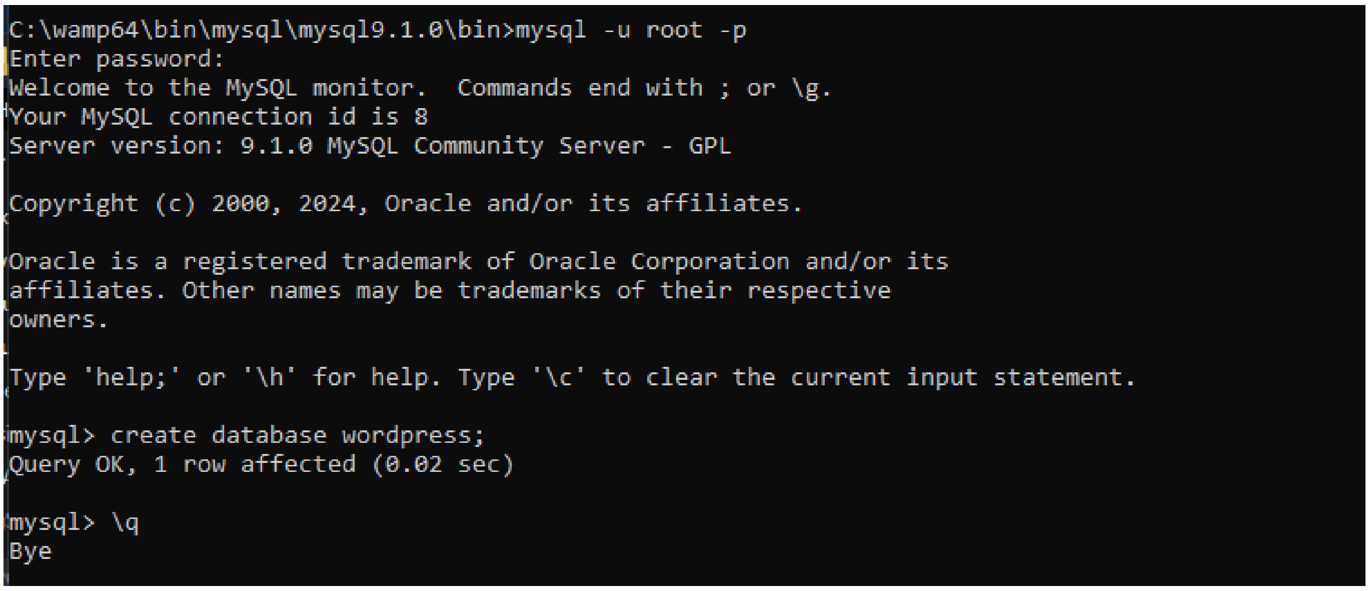
Command prompt appears. Point the location of the bin folder. Type mysql -u root -p and press Enter. You will be asked to enter a password. In the Enter password field, press Enter without issuing any password.

A mysql shell appears as shown in the following screenshot.



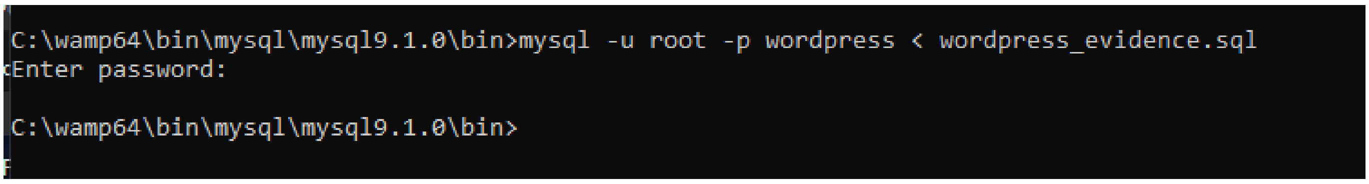
```
Microsoft Windows [Version 10.0.19045.5608]
(c) Microsoft Corporation. All rights reserved.

C:\wamp64\bin\mysql\mysql9.1.0\bin>mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 9.1.0 MySQL Community Server - GPL

Copyright (c) 2000, 2024, 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>
```

Type **create database wordpress;** into the MySQL shell and press Enter. This command will create a database named wordpress. After that, type \q and press Enter to exit the MySQL shell.



```
C:\wamp64\bin\mysql\mysql9.1.0\bin>mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 9.1.0 MySQL Community Server - GPL

Copyright (c) 2000, 2024, 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 (0.02 sec)

mysql> \q
Bye
```

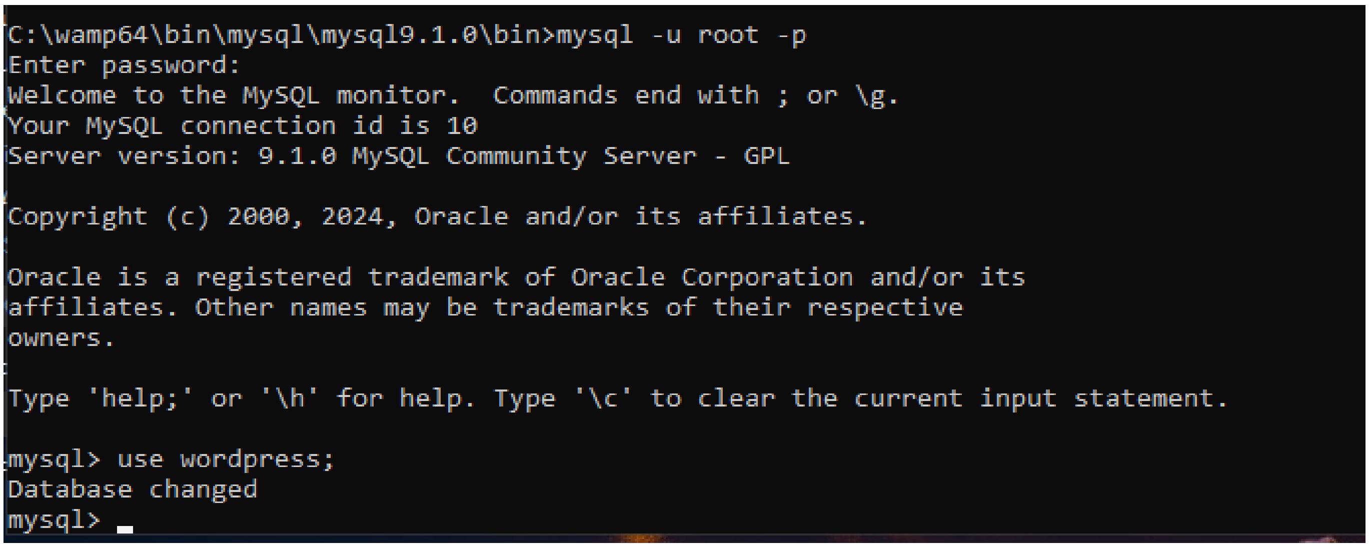
Now, we shall copy all the contents of the dump file to the newly created database. To copy, type **mysql -u root -p wordpress < wordpress_evidence.sql** in the command prompt and press Enter. You will be asked to enter a password. In the Enter password field, press Enter without issuing any password.



```
C:\wamp64\bin\mysql\mysql9.1.0\bin>mysql -u root -p wordpress < wordpress_evidence.sql
Enter password:

C:\wamp64\bin\mysql\mysql9.1.0\bin>
```

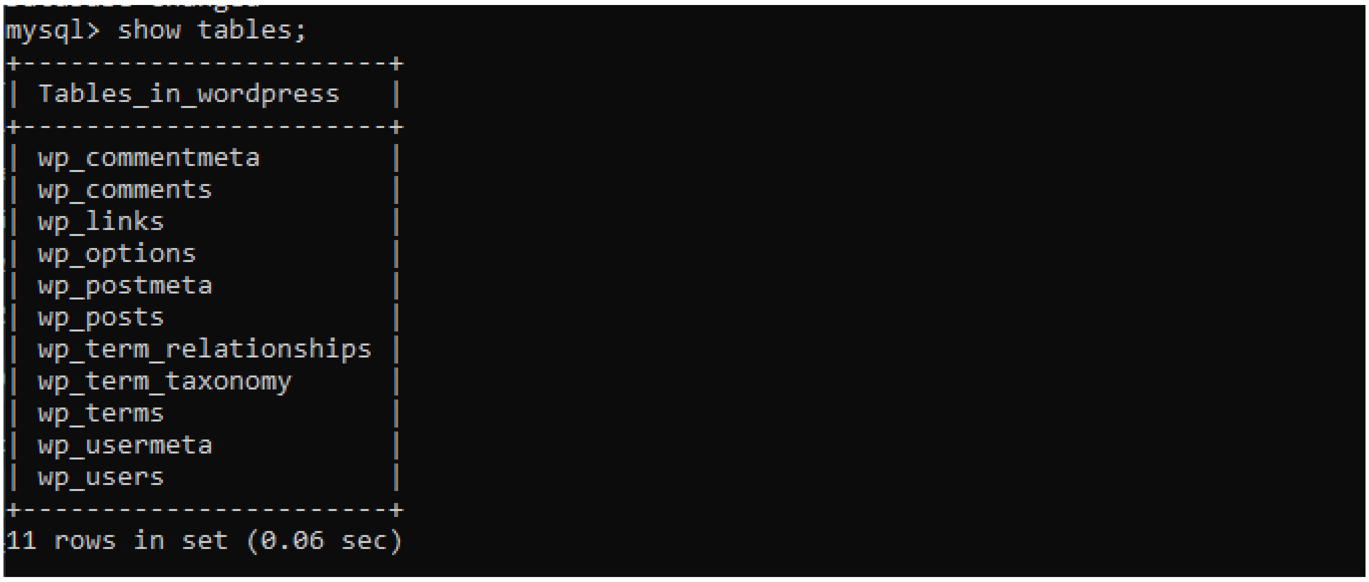
Once the backup is copied to the database, we shall log in to MySQL shell (by entering **mysql -u root -p** and then issuing an empty password) and start examining the database. To examine the database, we need to use the database.

Type **use wordpress;** and press Enter to use the wordpress database.



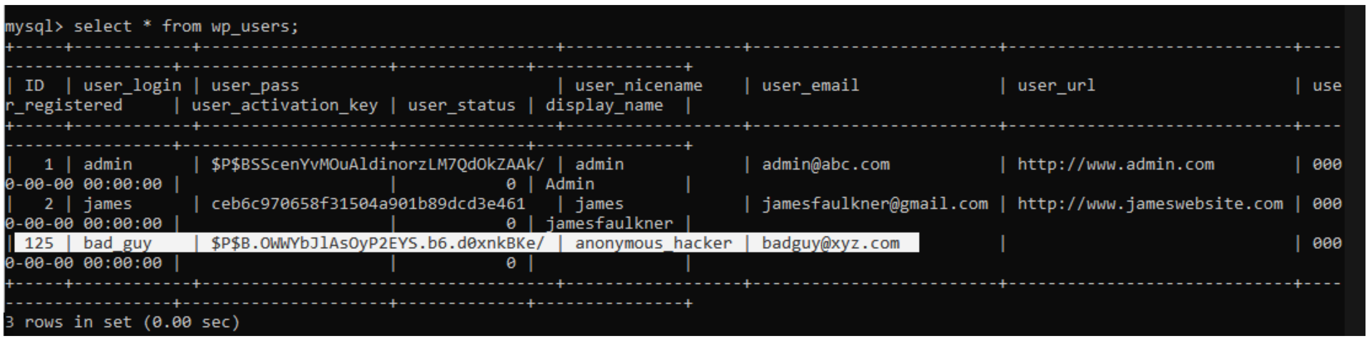
```
C:\wamp64\bin\mysql\mysql9.1.0\bin>mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 9.1.0 MySQL Community Server - GPL

Copyright (c) 2000, 2024, 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> use wordpress;
Database changed
mysql>
```

Now, we shall view the tables in this database. To view, type **show tables;** and press Enter



```
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 (0.06 sec)
```

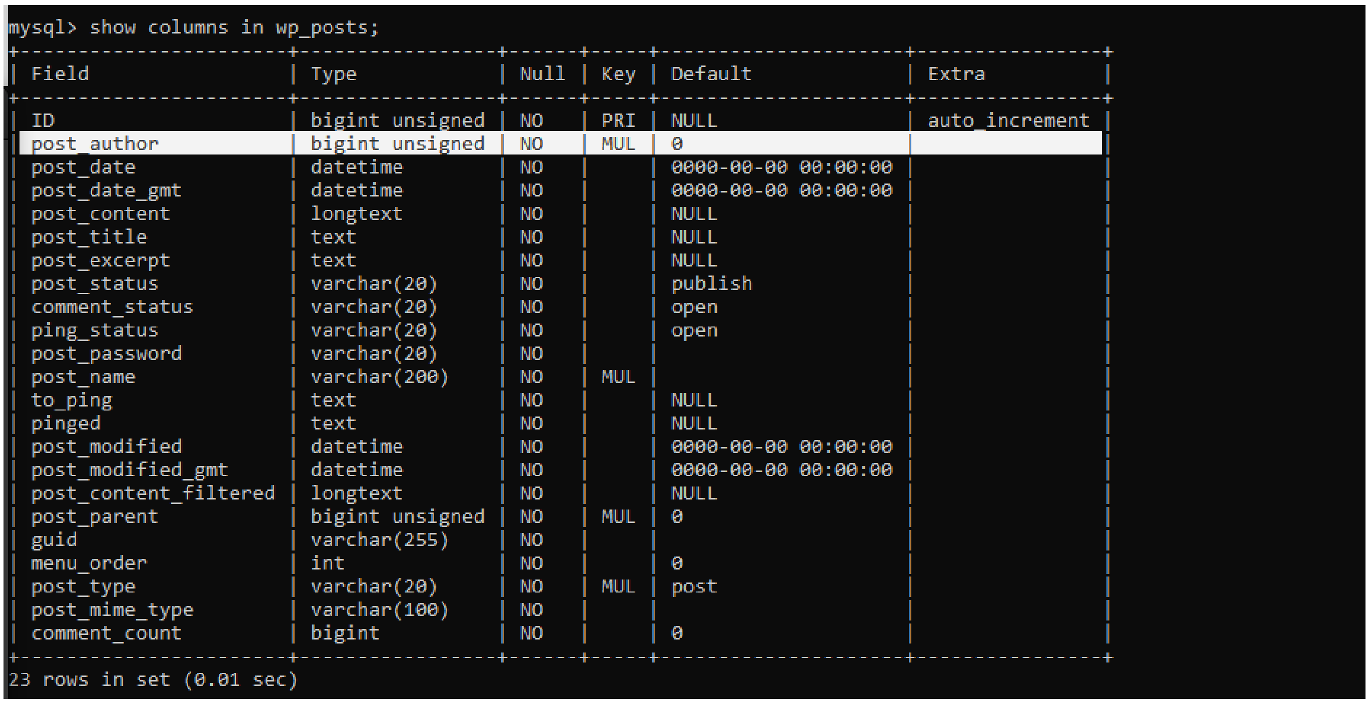
The wp_users table contains all the user accounts associated with the WordPress website. To view the users, type **select * from wp_users;** and press Enter



```
mysql> select * from wp_users;
+-----+------------+------------------------------------+------------------+-------------------------+-----------------------------+----
-----------------+---------------------+-------------+---------------+
| ID  | user_login | user_pass                          | user_nicename    | user_email              | user_url                    | use
r_registered     | user_activation_key | user_status | display_name  |
+-----+------------+------------------------------------+------------------+-------------------------+-----------------------------+----
-----------------+---------------------+-------------+---------------+
|   1 | admin      | $P$BSScenYvMOuAldinorzLM7QdOkZAAk/ | admin            | admin@abc.com           | http://www.admin.com        | 000
0-00-00 00:00:00 |                     |           0 | Admin         |
|   2 | james      | ceb6c970658f31504a901b89dcd3e461   | james            | jamesfaulkner@gmail.com | http://www.jameswebsite.com | 000
0-00-00 00:00:00 |                     |           0 | jamesfaulkner |
| 125 | bad_guy    | $P$B.OWWYbJlAsOyP2EYS.b6.d0xnkBKe/ | anonymous_hacker | badguy@xyz.com          |                             | 000
0-00-00 00:00:00 |                     |           0 |               |
+-----+------------+------------------------------------+------------------+-------------------------+-----------------------------+----
-----------------+---------------------+-------------+---------------+
3 rows in set (0.00 sec)
```

It is observed that a suspicious user account with the **username : bad_guy** is present in the table. Make a note of the **user ID which is 125**

Since the scenario in the beginning of the lab states that a suspicious post was found on the webpage, we shall view the columns in wp_posts table. To view the columns, type **show columns in wp_posts;** and press Enter.



```
mysql> show columns in wp_posts;
+-----------------------+-----------------+------+-----+---------------------+----------------+
| Field                 | Type            | Null | Key | Default             | Extra          |
+-----------------------+-----------------+------+-----+---------------------+----------------+
| ID                    | bigint unsigned | NO   | PRI | NULL                | auto_increment |
| post_author           | bigint unsigned | NO   | MUL | 0                   |                |
| post_date             | datetime        | NO   |     | 0000-00-00 00:00:00 |                |
| post_date_gmt         | datetime        | NO   |     | 0000-00-00 00:00:00 |                |
| post_content          | longtext        | NO   |     | NULL                |                |
| post_title            | text            | NO   |     | NULL                |                |
| post_excerpt          | text            | NO   |     | NULL                |                |
| post_status           | varchar(20)     | NO   |     | publish             |                |
| comment_status        | varchar(20)     | NO   |     | open                |                |
| ping_status           | varchar(20)     | NO   |     | open                |                |
| post_password         | varchar(20)     | NO   |     |                     |                |
| post_name             | varchar(200)    | NO   | MUL |                     |                |
| to_ping               | text            | NO   |     | NULL                |                |
| pinged                | text            | NO   |     | NULL                |                |
| post_modified         | datetime        | NO   |     | 0000-00-00 00:00:00 |                |
| post_modified_gmt     | datetime        | NO   |     | 0000-00-00 00:00:00 |                |
| post_content_filtered | longtext        | NO   |     | NULL                |                |
| post_parent           | bigint unsigned | NO   | MUL | 0                   |                |
| guid                  | varchar(255)    | NO   |     |                     |                |
| menu_order            | int             | NO   |     | 0                   |                |
| post_type             | varchar(20)     | NO   | MUL | post                |                |
| post_mime_type        | varchar(100)    | NO   |     |                     |                |
| comment_count         | bigint          | NO   |     | 0                   |                |
+-----------------------+-----------------+------+-----+---------------------+----------------+
23 rows in set (0.01 sec)
```

You will observe a column named post_author, which corresponds to the posts made by the users.

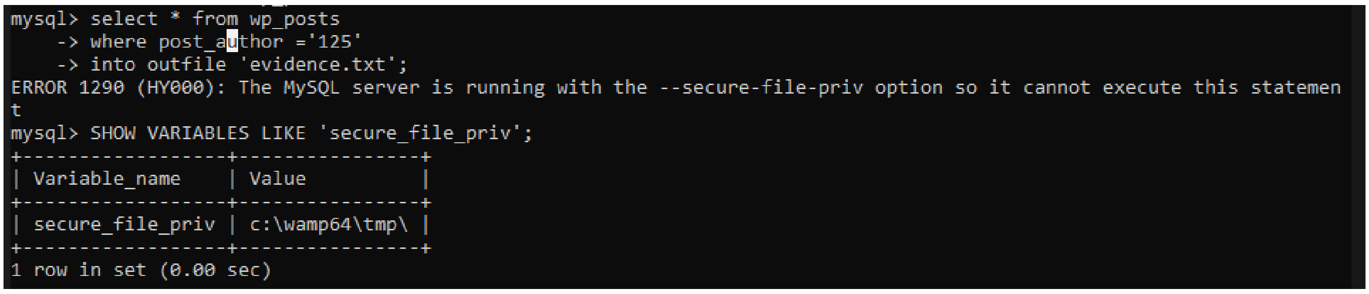
Now, using post_author and the user id of bad_guy, we can collect all the posts made by the suspicious user (bad_guy).

Issue the following commands to collect the posts:

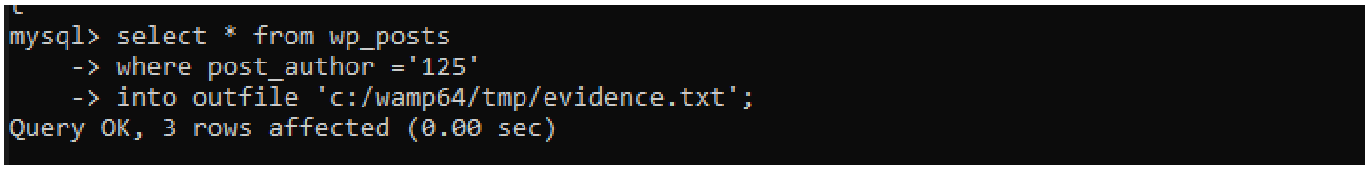
select * from wp_posts

where post_author = '125'

into outfile 'c:/wamp64/tmp/evidence.txt';

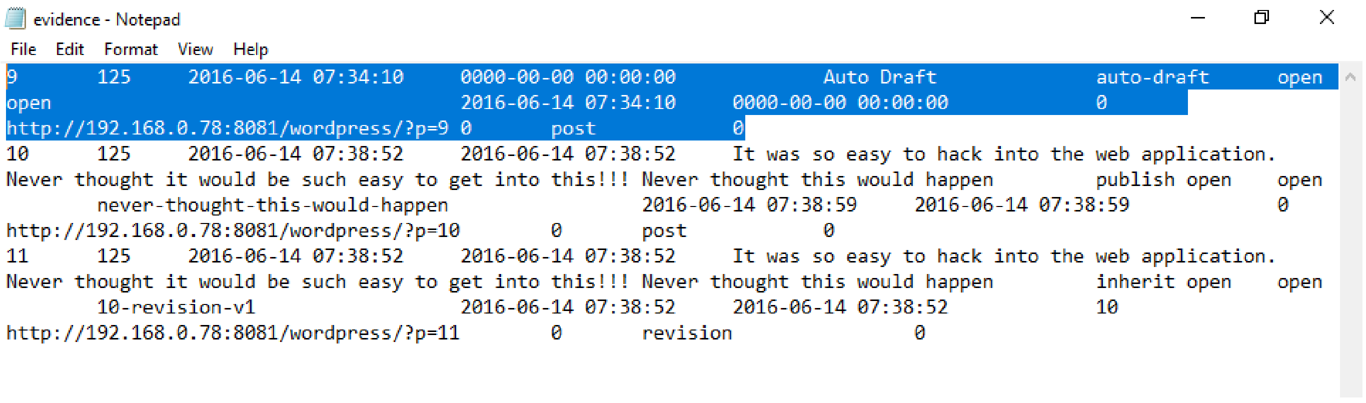


```
mysql> select * from wp_posts
    -> where post_author ='125'
    -> into outfile 'evidence.txt';
ERROR 1290 (HY000): The MySQL server is running with the --secure-file-priv option so it cannot execute this statemen
t
mysql> SHOW VARIABLES LIKE 'secure_file_priv';
+------------------+----------------+
| Variable_name    | Value          |
+------------------+----------------+
| secure_file_priv | c:\wamp64\tmp\ |
+------------------+----------------+
1 row in set (0.00 sec)
```



```
mysql> select * from wp_posts
    -> where post_author ='125'
    -> into outfile 'c:/wamp64/tmp/evidence.txt';
Query OK, 3 rows affected (0.00 sec)
```

By issuing the above commands, the posts made by the user whose ID is 125 are collected and saved to a file named evidence.txt

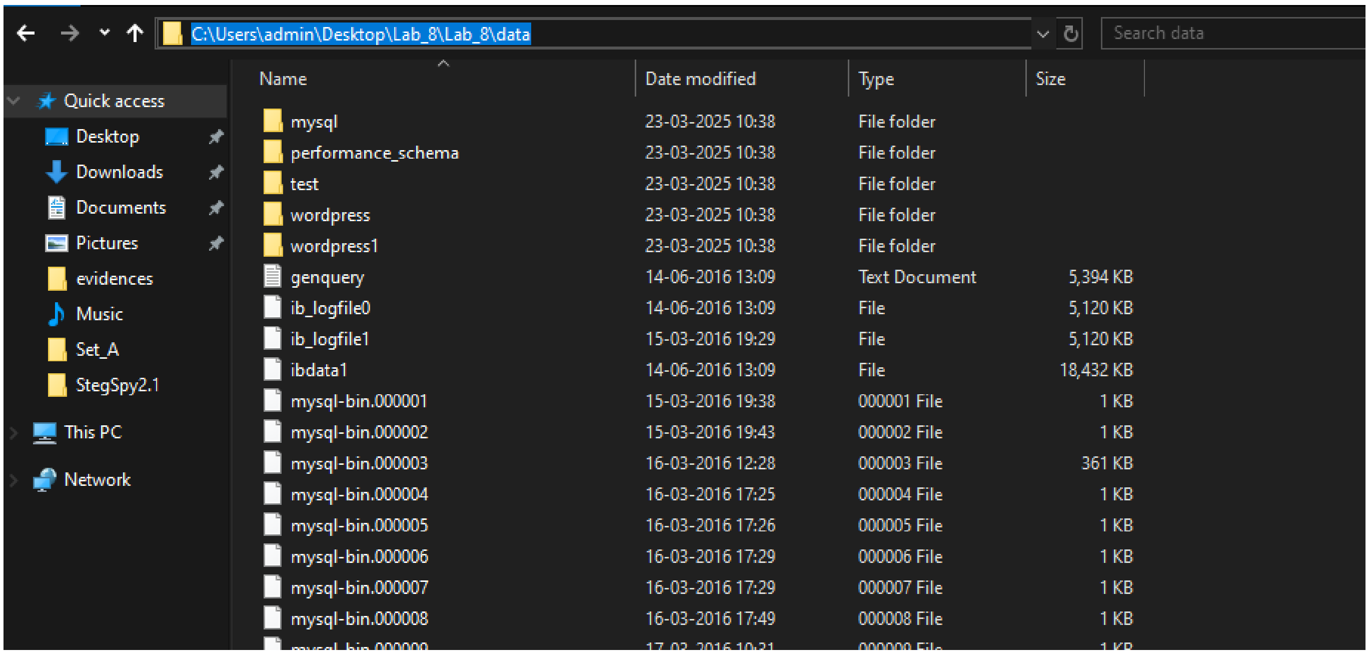


```
9       125     2016-06-14 07:34:10     0000-00-00 00:00:00             Auto Draft              auto-draft      open
open                                    2016-06-14 07:34:10     0000-00-00 00:00:00             0
http://192.168.0.78:8081/wordpress/?p=9 0       post            0
10      125     2016-06-14 07:38:52     2016-06-14 07:38:52     It was so easy to hack into the web application.
Never thought it would be such easy to get into this!!! Never thought this would happen         publish open    open
        never-thought-this-would-happen                 2016-06-14 07:38:59     2016-06-14 07:38:59     0
http://192.168.0.78:8081/wordpress/?p=10        0       post            0
11      125     2016-06-14 07:38:52     2016-06-14 07:38:52     It was so easy to hack into the web application.
Never thought it would be such easy to get into this!!! Never thought this would happen         inherit open    open
        10-revision-v1                  2016-06-14 07:38:52     2016-06-14 07:38:52             10
http://192.168.0.78:8081/wordpress/?p=11        0       revision                0
```

Now, we shall track events performed by the malicious user (MyISAM Storage Engine) and recover the deleted data.

The binary log files store all the transactions occurred on the databases. An investigator can examine these files to track the events performed by a particular user on the target database.

Navigate to C:\Users\admin\Desktop\Lab_8\Lab_8\data. You will find all the logs associated with the database as shown in the following screenshot:

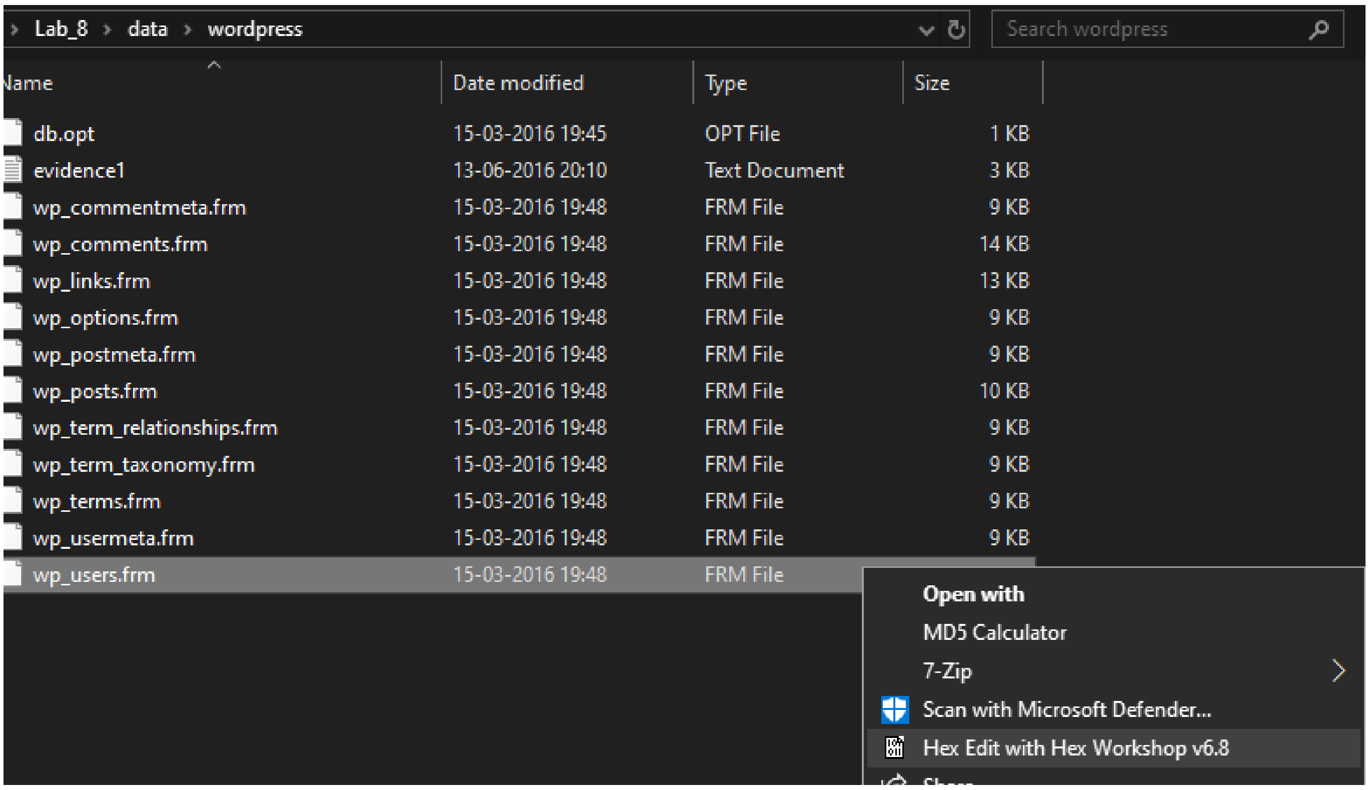


```
Name                  Date modified      Type            Size
mysql                 23-03-2025 10:38   File folder
performance_schema    23-03-2025 10:38   File folder
test                  23-03-2025 10:38   File folder
wordpress             23-03-2025 10:38   File folder
wordpress1            23-03-2025 10:38   File folder
genquery              14-06-2016 13:09   Text Document   5,394 KB
ib_logfile0           14-06-2016 13:09   File            5,120 KB
ib_logfile1           15-03-2016 19:29   File            5,120 KB
ibdata1               14-06-2016 13:09   File            18,432 KB
mysql-bin.000001      15-03-2016 19:38   000001 File     1 KB
mysql-bin.000002      15-03-2016 19:43   000002 File     1 KB
mysql-bin.000003      16-03-2016 12:28   000003 File     361 KB
mysql-bin.000004      16-03-2016 17:25   000004 File     1 KB
mysql-bin.000005      16-03-2016 17:26   000005 File     1 KB
mysql-bin.000006      16-03-2016 17:29   000006 File     1 KB
mysql-bin.000007      16-03-2016 17:29   000007 File     1 KB
mysql-bin.000008      16-03-2016 17:49   000008 File     1 KB
mysql-bin.000009      17-03-2016 10:31   000009 File     1 KB
```

Analyzing the .frm files helps a forensic examiner to understand the table format and the terms related to the table content.

Since the malicious user created a user account for himself with the login name bad_guy, you may analyze the wp_users.frm file with a hex editor to view the column name (along with its hexadecimal equivalent) that contains a list of login names associated with the users.

Now, open the wordpress folder, right-click wp_users.frm, and select Hex Edit with Hex Workshop v6.8 from the context menu.

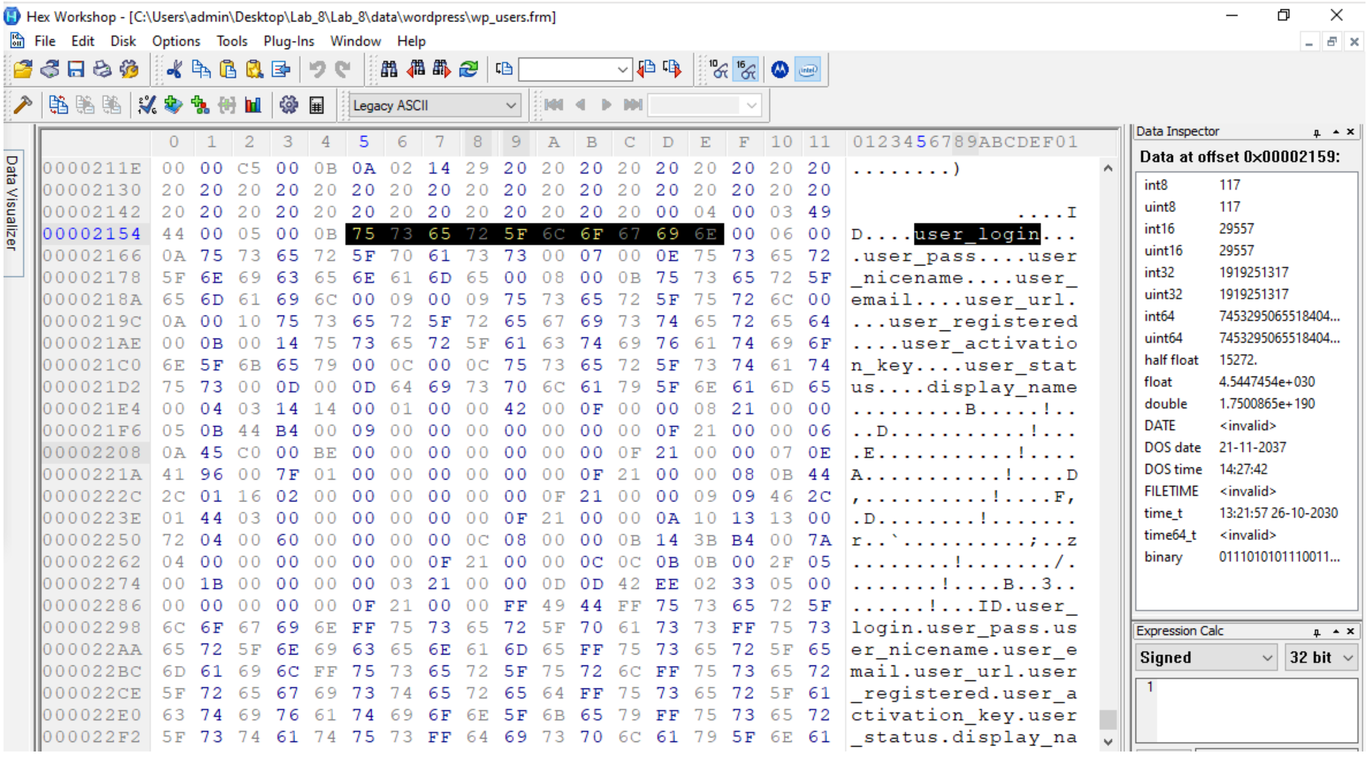


Lab_8 › data › wordpress

Name	Date modified	Type	Size
db.opt	15-03-2016 19:45	OPT File	1 KB
evidence1	13-06-2016 20:10	Text Document	3 KB
wp_commentmeta.frm	15-03-2016 19:48	FRM File	9 KB
wp_comments.frm	15-03-2016 19:48	FRM File	14 KB
wp_links.frm	15-03-2016 19:48	FRM File	13 KB
wp_options.frm	15-03-2016 19:48	FRM File	9 KB
wp_postmeta.frm	15-03-2016 19:48	FRM File	9 KB
wp_posts.frm	15-03-2016 19:48	FRM File	10 KB
wp_term_relationships.frm	15-03-2016 19:48	FRM File	9 KB
wp_term_taxonomy.frm	15-03-2016 19:48	FRM File	9 KB
wp_terms.frm	15-03-2016 19:48	FRM File	9 KB
wp_usermeta.frm	15-03-2016 19:48	FRM File	9 KB
wp_users.frm	15-03-2016 19:48	FRM File	

Search wordpress
Open with
MD5 Calculator
7-Zip
Scan with Microsoft Defender...
Hex Edit with Hex Workshop v6.8

We can observe that the login names are stored under the **user_login** column, whose hexadecimal equivalent is **757365725F6C6F67696E**.



Hex Workshop - [C:\Users\admin\Desktop\Lab_8\Lab_8\data\wordpress\wp_users.frm]

```
          0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  10 11  0123456789ABCDEF01
0000211E  00 00 C5 00 0B 0A 02 14 29 20 20 20 20 20 20 20 20 20  ........)
00002130  20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20
00002142  20 20 20 20 20 20 20 20 20 20 20 20 20 00 04 00 03 49      ....I
00002154  44 00 05 00 0B 75 73 65 72 5F 6C 6F 67 69 6E 00 06 00  D....user_login...
00002166  0A 75 73 65 72 5F 70 61 73 73 00 07 00 0E 75 73 65 72  .user_pass....user
00002178  5F 6E 69 63 65 6E 61 6D 65 00 08 00 0B 75 73 65 72 5F  _nicename....user_
0000218A  65 6D 61 69 6C 00 09 00 09 75 73 65 72 5F 75 72 6C 00  email....user_url.
0000219C  0A 00 10 75 73 65 72 5F 72 65 67 69 73 74 65 72 65 64  ...user_registered
000021AE  00 0B 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 0C 75 73 65 72 5F 73 74 61 74  n_key....user_stat
000021D2  75 73 00 0D 00 0D 64 69 73 70 6C 61 79 5F 6E 61 6D 65  us....display_name
000021E4  00 04 03 14 14 00 01 00 00 42 00 0F 00 00 08 21 00 00  .........B.....!..
000021F6  05 0B 44 B4 00 09 00 00 00 00 00 00 00 0F 21 00 00 06  ..D...........!...
00002208  0A 45 C0 00 BE 00 00 00 00 00 00 00 0F 21 00 00 07 0E  .E............!....
0000221A  41 96 00 7F 01 00 00 00 00 00 00 0F 21 00 00 08 0B 44  A...........!....D
0000222C  2C 01 16 02 00 00 00 00 00 00 0F 21 00 00 09 09 46 2C  ,...........!....F,
0000223E  01 44 03 00 00 00 00 00 00 0F 21 00 00 0A 10 13 13 00  .D........!.......
00002250  72 04 00 60 00 00 00 00 0C 08 00 00 0B 14 3B B4 00 7A  r..`..........;..z
00002262  04 00 00 00 00 00 00 0F 21 00 00 0C 0C 0B 0B 00 2F 05  ........!......./.
00002274  00 1B 00 00 00 00 03 21 00 00 0D 0D 42 EE 02 33 05 00  .......!....B..3..
00002286  00 00 00 00 00 0F 21 00 00 FF 49 44 FF 75 73 65 72 5F  ......!...ID.user_
00002298  6C 6F 67 69 6E FF 75 73 65 72 5F 70 61 73 73 FF 75 73  login.user_pass.us
000022AA  65 72 5F 6E 69 63 65 6E 61 6D 65 FF 75 73 65 72 5F 65  er_nicename.user_e
000022BC  6D 61 69 6C FF 75 73 65 72 5F 75 72 6C FF 75 73 65 72  mail.user_url.user
000022CE  5F 72 65 67 69 73 74 65 72 65 64 FF 75 73 65 72 5F 61  _registered.user_a
000022E0  63 74 69 76 61 74 69 6F 6E 5F 6B 65 79 FF 75 73 65 72  ctivation_key.user
000022F2  5F 73 74 61 74 75 73 FF 64 69 73 70 6C 61 79 5F 6E 61  _status.display_na
```

Data Inspector

Data at offset 0x00002159:

int8	117
uint8	117
int16	29557
uint16	29557
int32	1919251317
uint32	1919251317
int64	7453295065518404...
uint64	7453295065518404...
half float	15272.
float	4.5447454e+030
double	1.7500865e+190
DATE	<invalid>
DOS date	21-11-2037
DOS time	14:27:42
FILETIME	<invalid>
time_t	13:21:57 26-10-2030
time64_t	<invalid>
binary	0111010101110011...

Expression Calc

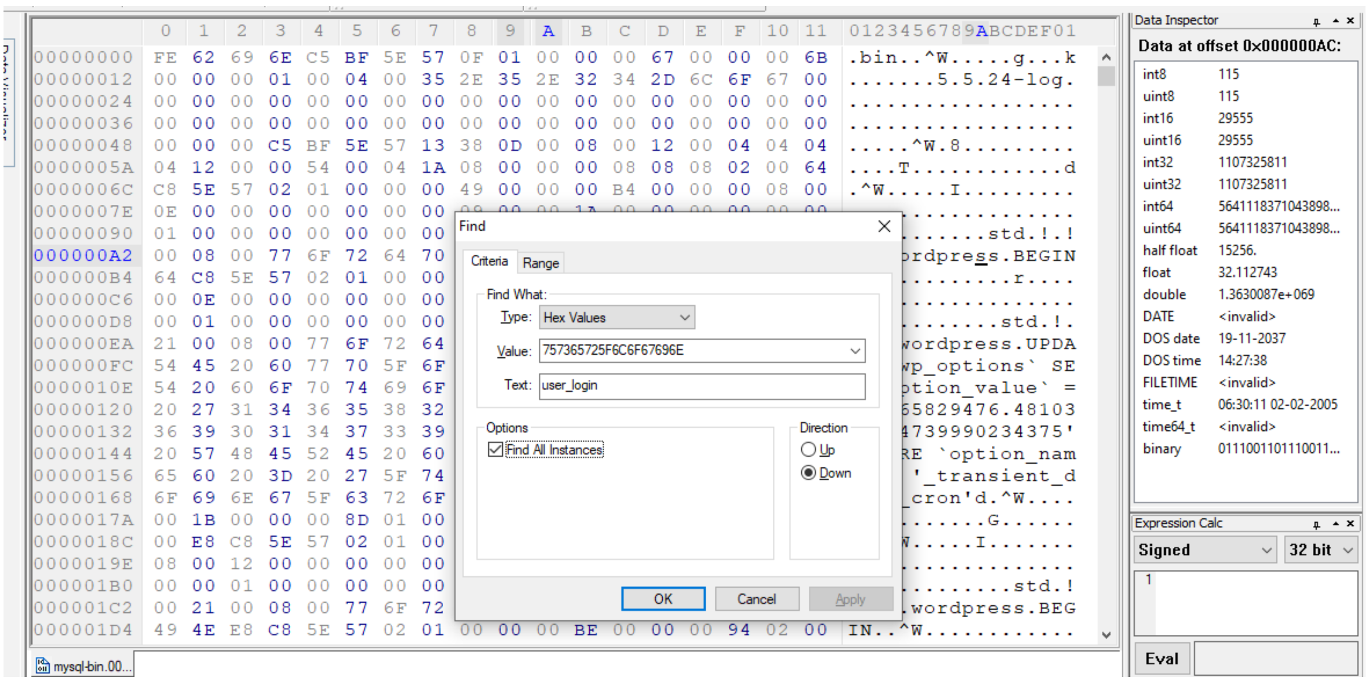
Signed | 32 bit

1

Using this phrase, we shall first find the attacker's login name, that is, bad_guy from the binary logs, and from there on, we shall trace the user activities performed by the malicious user.

In this lab, we shall analyze the **mysql-bin.000034 log file**. Open the file with Hex Workshop.

Examine each binary log for the text string user_login or hex value **757365725F6C6F67696E**.



```
          0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  10 11  0123456789ABCDEF01
00000000  FE 62 69 6E C5 BF 5E 57 0F 01 00 00 00 67 00 00 00 6B  .bin..^W.....g...k
00000012  00 00 00 01 00 04 00 35 2E 35 2E 32 34 2D 6C 6F 67 00  .......5.5.24-log.
00000024  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ..................
00000036  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ..................
00000048  00 00 00 C5 BF 5E 57 13 38 0D 00 08 00 12 00 04 04 04  .....^W.8.........
0000005A  04 12 00 00 54 00 04 1A 08 00 00 00 08 08 08 02 00 64  ....T.............d
0000006C  C8 5E 57 02 01 00 00 00 49 00 00 00 B4 00 00 00 08 00  .^W.....I.........
0000007E  0E 00 00 00 00 00 00 00                                ..................
00000090  01 00 00 00 00 00 00 00                                .......std.!.!
000000A2  00 08 00 77 6F 72 64 70                                ordpress.BEGIN
000000B4  64 C8 5E 57 02 01 00 00                                ..........r....
000000C6  00 0E 00 00 00 00 00 00                                ..............
000000D8  00 01 00 00 00 00 00 00                                .........std.!.
000000EA  21 00 08 00 77 6F 72 64                                wordpress.UPDA
000000FC  54 45 20 60 77 70 5F 6F                                wp_options` SE
0000010E  54 20 60 6F 70 74 69 6F                                ption_value` =
00000120  20 27 31 34 36 35 38 32                                65829476.48103
00000132  36 39 30 31 34 37 33 39                                4739990234375'
00000144  20 57 48 45 52 45 20 60                                RE `option_nam
00000156  65 60 20 3D 20 27 5F 74                                 '_transient_d
00000168  6F 69 6E 67 5F 63 72 6F                                _cron'd.^W....
0000017A  00 1B 00 00 00 8D 01 00                                .......G......
0000018C  00 E8 C8 5E 57 02 01 00                                W.....I.......
0000019E  08 00 12 00 00 00 00 00                                ..............
000001B0  00 00 01 00 00 00 00 00                                .........std.!
000001C2  00 21 00 08 00 77 6F 72                                .wordpress.BEG
000001D4  49 4E E8 C8 5E 57 02 01 00 00 00 BE 00 00 00 94 02 00  IN..^W............
```

mysql-bin.00...

Find

Criteria | Range

Find What:

Type: Hex Values

Value: 757365725F6C6F67696E

Text: user_login

Options: [✓] Find All Instances

Direction: Up [✓ Down]

OK | Cancel | Apply

Data Inspector

Data at offset 0x000000AC:

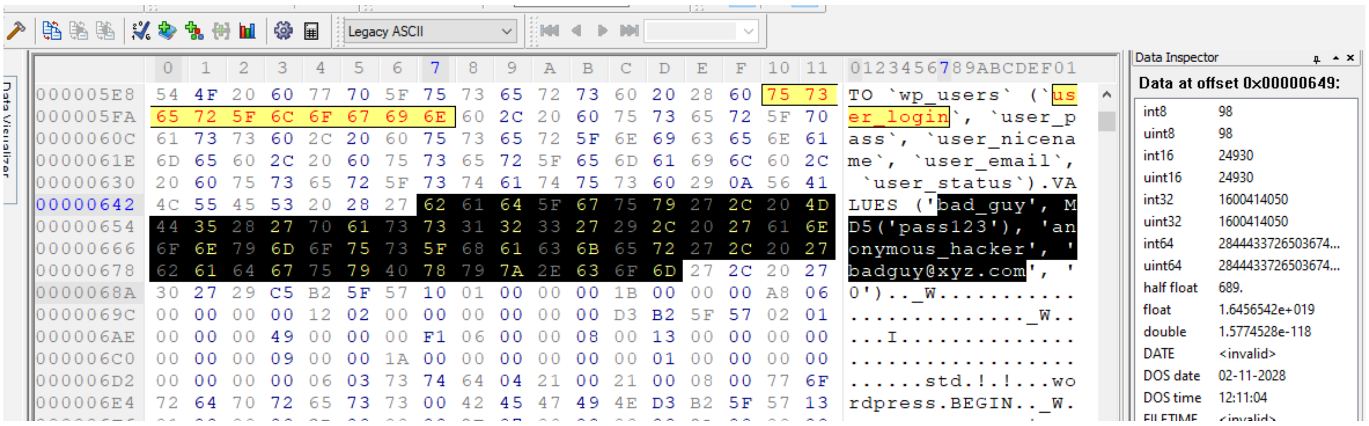
int8	115
uint8	115
int16	29555
uint16	29555
int32	1107325811
uint32	1107325811
int64	5641118371043898...
uint64	5641118371043898...
half float	15256.
float	32.112743
double	1.3630087e+069
DATE	<invalid>
DOS date	19-11-2037
DOS time	14:27:38
FILETIME	<invalid>
time_t	06:30:11 02-02-2005
time64_t	<invalid>
binary	0111001101110011...

Expression Calc

Signed | 32 bit

1

Eval



Legacy ASCII

```
          0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  10 11  0123456789ABCDEF01
000005E8  54 4F 20 60 77 70 5F 75 73 65 72 73 60 20 28 60 75 73  TO `wp_users` (`us
000005FA  65 72 5F 6C 6F 67 69 6E 60 2C 20 60 75 73 65 72 5F 70  er_login`, `user_p
0000060C  61 73 73 60 2C 20 60 75 73 65 72 5F 6E 69 63 65 6E 61  ass`, `user_nicena
0000061E  6D 65 60 2C 20 60 75 73 65 72 5F 65 6D 61 69 6C 60 2C  me`, `user_email`,
00000630  20 60 75 73 65 72 5F 73 74 61 74 75 73 60 29 0A 56 41   `user_status`).VA
00000642  4C 55 45 53 20 28 27 62 61 64 5F 67 75 79 27 2C 20 4D  LUES ('bad_guy', M
00000654  44 35 28 27 70 61 73 73 31 32 33 27 29 2C 20 27 61 6E  D5('pass123'), 'an
00000666  6F 6E 79 6D 6F 75 73 5F 68 61 63 6B 65 72 27 2C 20 27  onymous_hacker', '
00000678  62 61 64 67 75 79 40 78 79 7A 2E 63 6F 6D 27 2C 20 27  badguy@xyz.com', '
0000068A  30 27 29 C5 B2 5F 57 10 01 00 00 00 1B 00 00 00 A8 06  0').._W...........
0000069C  00 00 00 00 12 02 00 00 00 00 00 00 D3 B2 5F 57 02 01  ..............._W..
000006AE  00 00 00 49 00 00 00 F1 06 00 00 08 00 13 00 00 00 00  ...I..............
000006C0  00 00 00 09 00 00 1A 00 00 00 00 00 00 01 00 00 00 00  ..................
000006D2  00 00 00 00 06 03 73 74 64 04 21 00 21 00 08 00 77 6F  ......std.!.!...wo
000006E4  72 64 70 72 65 73 73 00 42 45 47 49 4E D3 B2 5F 57 13  rdpress.BEGIN.._W.
```

Data Inspector

Data at offset 0x00000649:

int8	98
uint8	98
int16	24930
uint16	24930
int32	1600414050
uint32	1600414050
int64	2844433726503674...
uint64	2844433726503674...
half float	689.
float	1.6456542e+019
double	1.5774528e-118
DATE	<invalid>
DOS date	02-11-2028
DOS time	12:11:04
FILETIME	<invalid>

While conducting a detailed examination on the binary files, we can find that one of the binary files recorded an event where a query is executed for creating a user account with the:

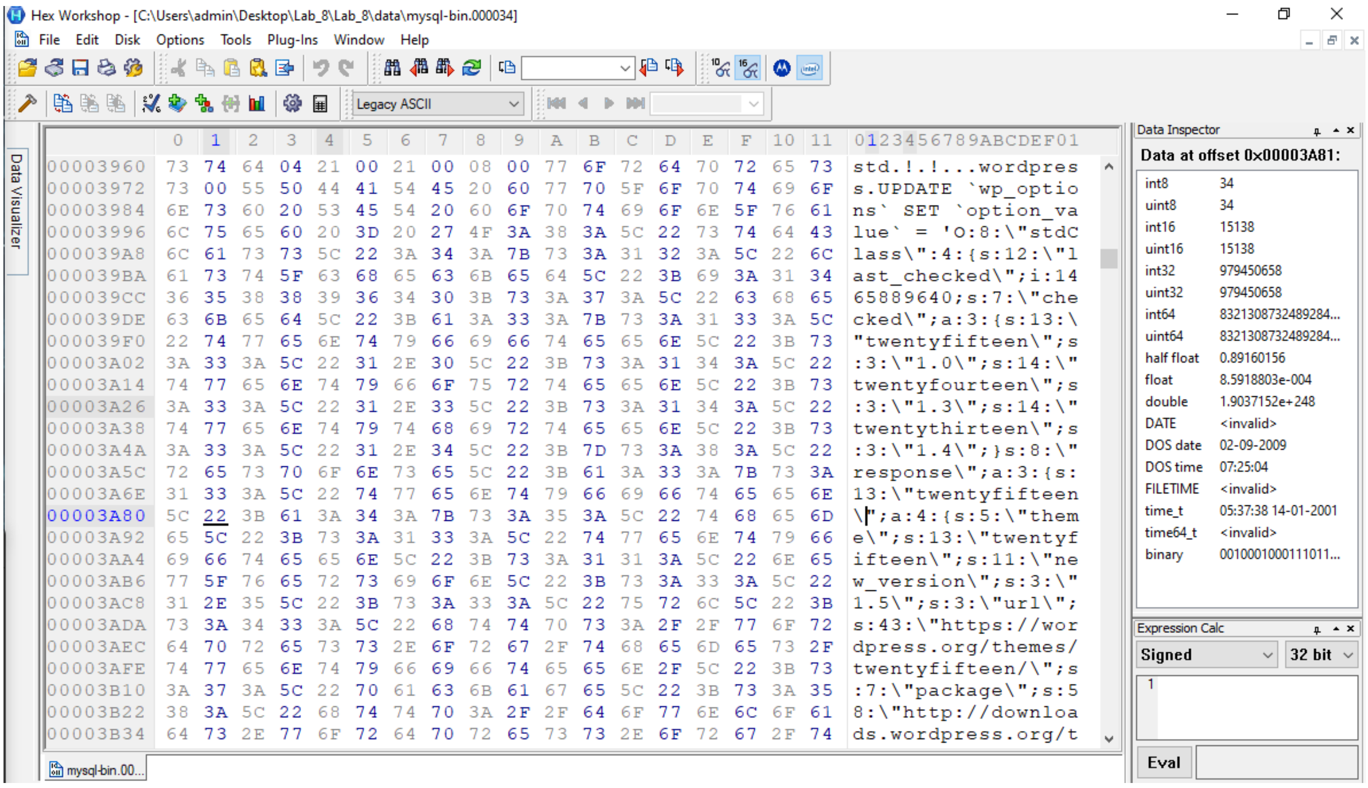
Login name – bad_guy

Password – pass123

Nice name – anonymous_hacker

Email ID – badguy@xyz.com

In the same way, scroll down the binary logs one-by-one to see the logs corresponding to the malicious user's actions



Hex Workshop - [C:\Users\admin\Desktop\Lab_8\Lab_8\data\mysql-bin.000034]

```
          0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  10 11  0123456789ABCDEF01
00003960  73 74 64 04 21 00 21 00 08 00 77 6F 72 64 70 72 65 73  std.!.!...wordpres
00003972  73 00 55 50 44 41 54 45 20 60 77 70 5F 6F 70 74 69 6F  s.UPDATE `wp_optio
00003984  6E 73 60 20 53 45 54 20 60 6F 70 74 69 6F 6E 5F 76 61  ns` SET `option_va
00003996  6C 75 65 60 20 3D 20 27 4F 3A 38 3A 5C 22 73 74 64 43  lue` = 'O:8:\"stdC
000039A8  6C 61 73 73 5C 22 3A 34 3A 7B 73 3A 31 32 3A 5C 22 6C  lass\":4:{s:12:\"l
000039BA  61 73 74 5F 63 68 65 63 6B 65 64 5C 22 3B 69 3A 31 34  ast_checked\";i:14
000039CC  36 35 38 38 39 36 34 30 3B 73 3A 37 3A 5C 22 63 68 65  65889640;s:7:\"che
000039DE  63 6B 65 64 5C 22 3B 61 3A 33 3A 7B 73 3A 31 33 3A 5C  cked\";a:3:{s:13:\
000039F0  22 74 77 65 6E 74 79 66 69 66 74 65 65 6E 5C 22 3B 73  "twentyfifteen\";s
00003A02  3A 33 3A 5C 22 31 2E 30 5C 22 3B 73 3A 31 34 3A 5C 22  :3:\"1.0\";s:14:\"
00003A14  74 77 65 6E 74 79 66 6F 75 72 74 65 65 6E 5C 22 3B 73  twentyfourteen\";s
00003A26  3A 33 3A 5C 22 31 2E 33 5C 22 3B 73 3A 31 34 3A 5C 22  :3:\"1.3\";s:14:\"
00003A38  74 77 65 6E 74 79 74 68 69 72 74 65 65 6E 5C 22 3B 73  twentythirteen\";s
00003A4A  3A 33 3A 5C 22 31 2E 34 5C 22 3B 7D 73 3A 38 3A 5C 22  :3:\"1.4\";}s:8:\"
00003A5C  72 65 73 70 6F 6E 73 65 5C 22 3B 61 3A 33 3A 7B 73 3A  response\";a:3:{s:
00003A6E  31 33 3A 5C 22 74 77 65 6E 74 79 66 69 66 74 65 65 6E  13:\"twentyfifteen
00003A80  5C 22 3B 61 3A 34 3A 7B 73 3A 35 3A 5C 22 74 68 65 6D  \";a:4:{s:5:\"them
00003A92  65 5C 22 3B 73 3A 31 33 3A 5C 22 74 77 65 6E 74 79 66  e\";s:13:\"twentyf
00003AA4  69 66 74 65 65 6E 5C 22 3B 73 3A 31 31 3A 5C 22 6E 65  ifteen\";s:11:\"ne
00003AB6  77 5F 76 65 72 73 69 6F 6E 5C 22 3B 73 3A 33 3A 5C 22  w_version\";s:3:\"
00003AC8  31 2E 35 5C 22 3B 73 3A 33 3A 5C 22 75 72 6C 5C 22 3B  1.5\";s:3:\"url\";
00003ADA  73 3A 34 33 3A 5C 22 68 74 74 70 73 3A 2F 2F 77 6F 72  s:43:\"https://wor
00003AEC  64 70 72 65 73 73 2E 6F 72 67 2F 74 68 65 6D 65 73 2F  dpress.org/themes/
00003AFE  74 77 65 6E 74 79 66 69 66 74 65 65 6E 2F 5C 22 3B 73  twentyfifteen/\";s
00003B10  3A 37 3A 5C 22 70 61 63 6B 61 67 65 5C 22 3B 73 3A 35  :7:\"package\";s:5
00003B22  38 3A 5C 22 68 74 74 70 3A 2F 2F 64 6F 77 6E 6C 6F 61  8:\"http://downloa
00003B34  64 73 2E 77 6F 72 64 70 72 65 73 73 2E 6F 72 67 2F 74  ds.wordpress.org/t
```

Data Inspector

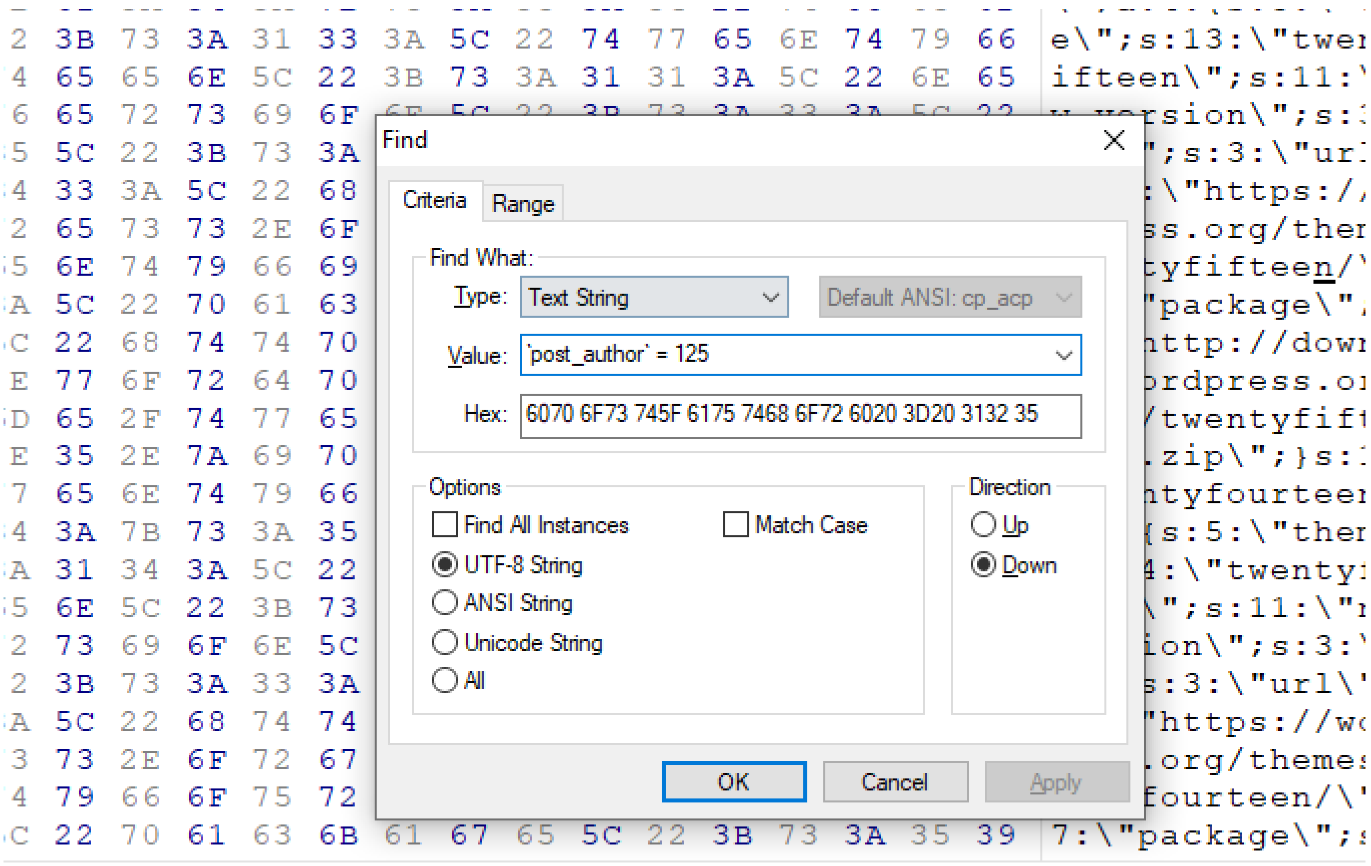
Data at offset 0x00003A81:

int8	34
uint8	34
int16	15138
uint16	15138
int32	979450658
uint32	979450658
int64	8321308732489284...
uint64	8321308732489284...
half float	0.89160156
float	8.5918803e-004
double	1.9037152e+248
DATE	<invalid>
DOS date	02-09-2009
DOS time	07:25:04
FILETIME	<invalid>
time_t	05:37:38 14-01-2001
time64_t	<invalid>
binary	0010001000111011...

We can observe that the attacker made a post (post_author id: 125) on 14th June, 2016, at GMT 07:37:45.

In the same way, you may search for all the actions performed by the attacker on the posts by looking for 'post_author = 125' in the hex editor.

To find the actions performed by the attacker, press Ctrl+F on the keyboard. The Find window appears. Select 'Text String' from the Type drop-down list, enter 'post_author = 125' in the Value text field, select the 'Down' radio button under the Direction section, and click OK



Find

Criteria | Range

Find What:

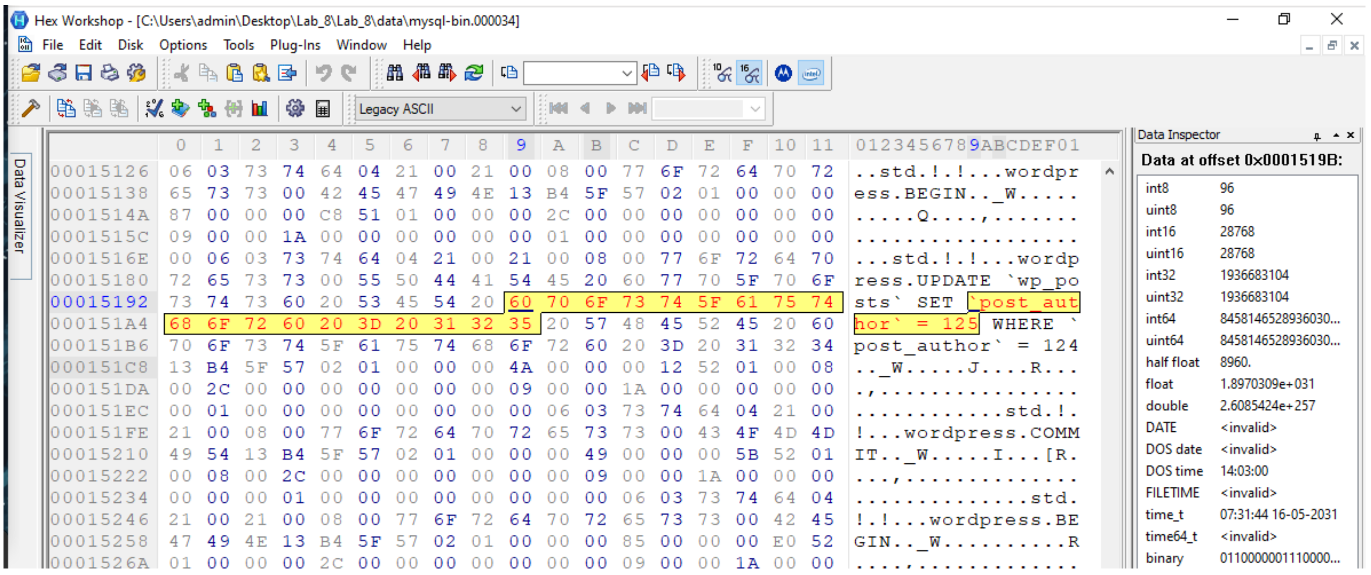
Type: Text String — Default ANSI: cp_acp

Value: `post_author` = 125

Hex: 6070 6F73 745F 6175 7468 6F72 6020 3D20 3132 35

Options: [ ] Find All Instances [ ] Match Case [✓ UTF-8 String] ANSI String Unicode String All

Direction: Up [✓ Down]



Hex Workshop - [C:\Users\admin\Desktop\Lab_8\Lab_8\data\mysql-bin.000034]

```
          0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  10 11  0123456789ABCDEF01
00015126  06 03 73 74 64 04 21 00 21 00 08 00 77 6F 72 64 70 72  ..std.!.!...wordpr
00015138  65 73 73 00 42 45 47 49 4E 13 B4 5F 57 02 01 00 00 00  ess.BEGIN.._W.....
0001514A  87 00 00 00 C8 51 01 00 00 00 2C 00 00 00 00 00 00 00  .....Q....,.......
0001515C  09 00 00 1A 00 00 00 00 00 00 01 00 00 00 00 00 00 00  ..................
0001516E  00 06 03 73 74 64 04 21 00 21 00 08 00 77 6F 72 64 70  ...std.!.!...wordp
00015180  72 65 73 73 00 55 50 44 41 54 45 20 60 77 70 5F 70 6F  ress.UPDATE `wp_po
00015192  73 74 73 60 20 53 45 54 20 60 70 6F 73 74 5F 61 75 74  sts` SET `post_aut
000151A4  68 6F 72 60 20 3D 20 31 32 35 20 57 48 45 52 45 20 60  hor` = 125 WHERE `
000151B6  70 6F 73 74 5F 61 75 74 68 6F 72 60 20 3D 20 31 32 34  post_author` = 124
000151C8  13 B4 5F 57 02 01 00 00 00 4A 00 00 00 12 52 01 00 08  .._W.....J....R...
000151DA  00 2C 00 00 00 00 00 00 00 09 00 00 1A 00 00 00 00 00  .,................
000151EC  00 01 00 00 00 00 00 00 00 00 06 03 73 74 64 04 21 00  ............std.!.
000151FE  21 00 08 00 77 6F 72 64 70 72 65 73 73 00 43 4F 4D 4D  !...wordpress.COMM
00015210  49 54 13 B4 5F 57 02 01 00 00 00 49 00 00 00 5B 52 01  IT.._W.....I...[R.
00015222  00 08 00 2C 00 00 00 00 00 00 00 09 00 00 1A 00 00 00  ...,..............
00015234  00 00 00 01 00 00 00 00 00 00 00 00 06 03 73 74 64 04  ..............std.
00015246  21 00 21 00 08 00 77 6F 72 64 70 72 65 73 73 00 42 45  !.!...wordpress.BE
00015258  47 49 4E 13 B4 5F 57 02 01 00 00 00 85 00 00 00 E0 52  GIN.._W..........R
0001526A  01 00 00 00 2C 00 00 00 00 00 00 00 09 00 00 1A 00 00  ....,.............
```

Data Inspector

Data at offset 0x0001519B:

int8	96
uint8	96
int16	28768
uint16	28768
int32	1936683104
uint32	1936683104
int64	8458146528936030...
uint64	8458146528936030...
half float	8960.
float	1.8970309e+031
double	2.6085424e+257
DATE	<invalid>
DOS date	<invalid>
DOS time	14:03:00
FILETIME	<invalid>
time_t	07:31:44 16-05-2031
time64_t	<invalid>
binary	0110000001110000...

In the above screenshot, you can observe a MySQL query for changing the post_author value from 124 to 125 for all relevant records in the wp_posts table.. In the same way, you may examine all the log files and find the transactions performed by the attacker.