# Forensic Data Recovery

**S Greeshma[1], Karthika Ravi [2], Joyal Johny[3], Akhil S[4], Akhija Lakshmi R[5]**

[1,2,3,4]B. Tech Students, [5]Assistant Professor

[1,2,3,4,5]Computer Science and Engineering,

[1,2,3,4,5] Kottayam Institute of Technology and Science, Kottayam, India

**Abstract** - With the rising usage of information systems, database forensics is becoming more relevant for investigators. Despite the fact that numerous database forensic methods such as log analysis and investigation model building have been investigated, retrieving deleted data is a key approach in database investigation for DB tampering and anti-forensics. Cyber forensics is extremely important in today's technology-driven generation. Technology paired with forensic forensics allows for faster investigations and more accurate results.

The prior method recovers destroyed data by using journal logs, however this is not practicable when the log set is deleted on a regular basis or contains vital evidence. As a solution, a method for analyzing data at an early stage is introduced. There has been research into recovering small-sized databases such as SQLite and EDB, but there has been no prior work outlining the structure of data files and technology for recovering deleted data from huge databases used by corporations or large organizations. In this context, we look at SQL and MySQL, two of the most popular databases. Our approach focuses on SQL storage engines such as MySQL. We designed and built a method for recovering deleted records as an open-source tool. As a result, when compared to commercial recovery tools, the method's effectiveness improves.

**Index Terms** - Database (DB), Structured Query Language (SQL), Enterprise Database (EDB).

## I. INTRODUCTION

A subset of digital forensics called database forensics studies the contents and structure of databases. It is a subfield of digital forensic science that deals with the forensic analysis of databases and the metadata that goes with them. The discipline uses investigative techniques on database contents and metadata while adhering to the standard forensic process, much as computer forensics. Databases that manage significant personal or business data have inspired forensic researchers. Database forensics is becoming more crucial for a forensic investigator as Internet of Things (IoT) devices are increasingly used to access a wide range of apps and store user data and logs. To locate crucial evidence in databases, a number of forensic techniques, including log analysis and investigative model construction, have been examined. Because it can recover data that was deleted on purpose or in the past for anti-forensics purposes, restoring deleted records among database forensic techniques is crucial to locating the evidence. An investigator can identify and address these behaviors by retrieving deleted records.

Attackers frequently attempt to alter databases in order to steal sensitive information or attempt to destroy information. In forensic accounting, the recovery of deleted documents is also employed to resolve financial fraud cases. The log-based method and the engine-based method are the two basic categories of strategies for recovering deleted records.

The log-based methods that examine the transaction log or journal log were widely proposed by scholars. There are records in the log in order. Each transaction's log is kept in a reliable storage location so that it can be restored from there in the event of a failure. Log is nothing more than a file with a series of records in it, each of which relates to a write operation. The log file contains step-by-step records of every log entry. Log files, so to speak, record the history of all update activities. The transaction's start, transaction number, record number, old value, new value, and end are all included in the log. small statements, like those found at bank ATMs. Using log files, we can go back to the prior state as if nothing had happened to the database if the system breaks during an ongoing operation. The log is stored on disc so that failures other than disc and failures won't influence it. The log will be updated with any database operations that are carried out.

The logs are kept by the administrator's security settings; therefore, it is difficult to retrieve erased data by looking at the logs if the logs are set to be periodically removed or to be of a limited fixed size. Engine-based techniques that first recover erased data after analyzing the data file's structure were investigated as a result. The approaches use a direct approach to raw data, which can help with the log-based method's disadvantage. Previous research mostly focused on endpoint devices (desktop computers, smartphones, and tablets) that employ tiny databases like SQLite and EDB. However, there is no research outlining an engine-based recovery solution for large databases, such as SQL and MYSQL, which are typically used in companies or large organizations. It is challenging to determine the structure of the data file and storage mechanism since the Database Management Systems (DBMS) structure of large databases is complex and the DBMS's source code is not available to the public.

## II. LITERATURE SURVEY

**Jongseong Yoon, Sangjin Lee [1]** They investigate a widely used MongoDB recovery method for deleted data in this article. The Wired Tiger and MMAPv1 storage engines, which are the disk-based storage engines for MongoDB, have had their internal structures examined. Both a recovery strategy based on the signature of deleted records and a recovery method utilizing the namespace file's metadata information were given**. James Wagner, Alexander Rasin [2]** In this work, they suggest and assess a method for identifying direct database file alterations that have previously eluded the DBMS and its internal security controls. In this method, database indexes are first validated using forensic analysis, and then the index state is compared to the information in the DBMS tables. In this paper, they provide DBStorageAuditor, a system that can directly audit internal database structures to find storage discrepancies that can be used to detect database file tampering.

**Joshua Sablatura, Bing Zhou [3]** While this paper identified four key areas of research within forensic database reconstruction and identified prospective future work for each, they believe that the use of the internal redo logs of the database server can provide some promising research opportunities. Unlike other log files, the contents of these files are not susceptible to the logging preferences established by the database administrator. Therefore, they will usually contain the required information to retrace the steps a user took on the database. Furthermore, since these are internal logs, they are extremely difficult for an intruder to modify without detection.
**Christian Meng, Harald Baier [4]** In this study, the authors concentrate on digital remnants of deleted database data that fall under the purview of SQLite. They advise using a structural approach and investigate SQLite's deletion behavior in relation to several database parameters that have an impact on the deletion of database data. They looked into digital remnants of deleted database data in the context of SQLite. We used a structural method to examine the deletion behavior of SQLite in relation to several database settings that influence the erasure of database data in order to increase the recovery rate of deleted SQLite data.

## III. METHODOLOGY

Among database forensic procedures, restoring deleted records is critical in identifying evidence since it can recover material that existed previously or was purposely removed for anti-forensics. Attackers, in particular, try database tampering in order to steal or destroy sensitive information; by retrieving deleted records, an investigator can detect and address these activities. The recovery of lost documents is frequently used in forensic accounting to address financial fraud instances.
The advised remedy is to retrieve the lost data using a forensic recovery programme. It was largely about MYSQL and SQL databases. There are three modules for this project. The first module is the user registration module, followed by the database connectivity module, and finally by the recovery module.

*(1) Architecture of the Proposed System*

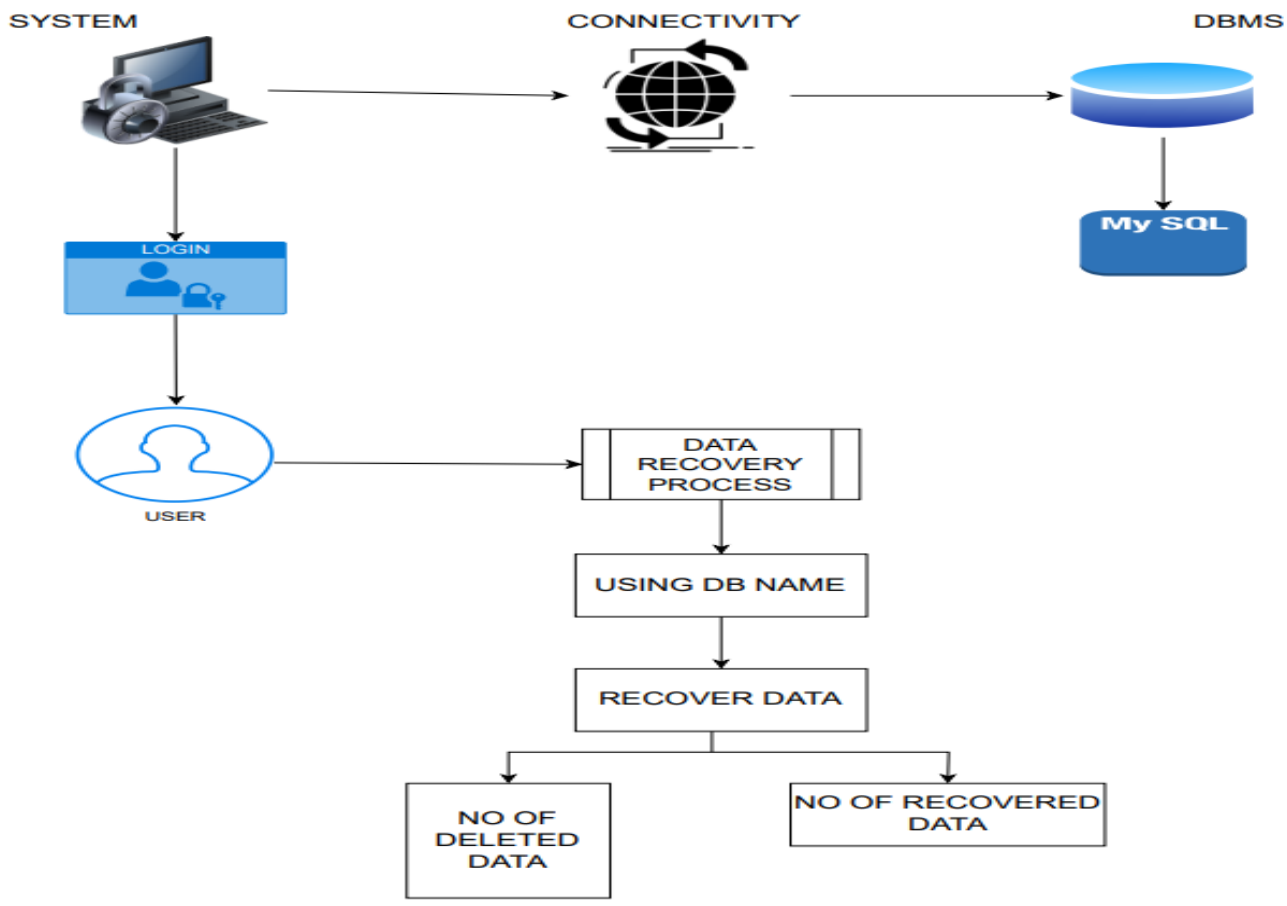


Fig.1 Architecture of proposed system

The user logs into the system in this system. The user can then access the database. MySQL is the database used in this case. The user can then perform the recovery procedure. To recover deleted data, use the file name. Following the recovery of deleted data, it displays the number of deleted data, the number of recovered data, and the file type of deleted data.

*(2) Use-case of the proposed System*

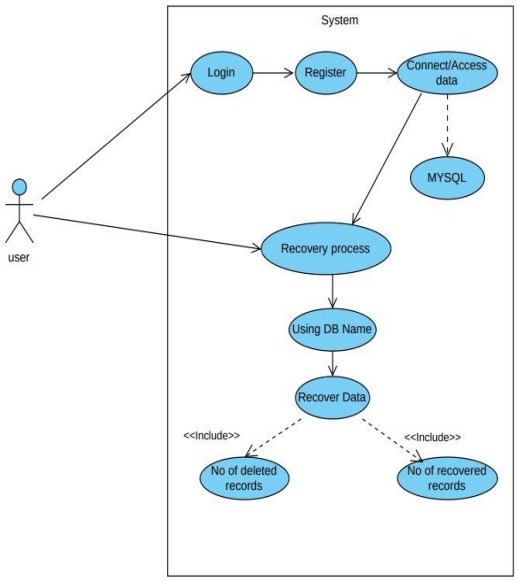


Fig.2 Use-case of proposed system

The proposed system consist of a user and the user is mainly perform three operations. Firstly, the user is able to login the system and from this database is accessed. Here the data is stored in MySQL database. Next the user is able to find the location of the data. Next the user is able to perform the recovery process, in this process the deleted data is recovered.it shows the number of data deleted, number of data recovered and also show the type of file recovered.

*(2) Activity diagram of the proposed System*

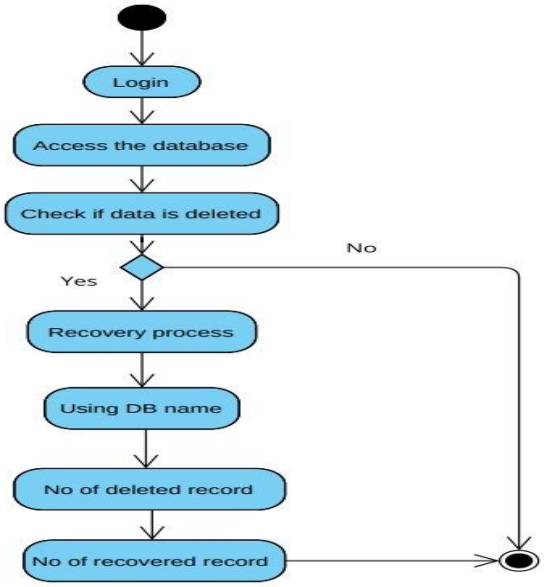


Fig.3 Activity Diagram of proposed system

Firstly, the user is login the system and then access the database where the data is stored. And check whether the data is deleted or not. If the data is not deleted then there is no recovery process to be performed. When the data is deleted then the location of the data is to be find and perform the recovery process. After recovery process the number of deleted data, number of recovered data and the type of file deleted is displayed.

## IV. IMPLEMENTATION

The user registration module is the first module in the suggested technique. The registration page is contained in this module. The user's name, email, phone number, and address are used to register. The user can also specify a username and password. After completing these steps, registration is complete. The user can then log in with the username and password they created.

The user can then check whether or not the data in this module has been erased in the following step. No additional action is necessary if the data is not re-moved following the screening step. If the data is destroyed, a notification will be displayed indicating that the data has been deleted. After that, the database must be connected. For example, MYSQL to MYSQL. The database's connectivity to the other database is then checked. If the connection is used to connect to a deleted database, the user may receive a notification that the data has been erased. After the joining step, the user can begin the recovery process. It is critical to check for data connectivity.

Instead of the original format, the recovered file is in.dat format. Following the recovery process, a.dat file is delivered for the restoration process. It must be restored because the file must be restored in its original format. Following these two procedures, it examines the connectivity to see if the process was successful.

## V. RESULT

The initial stage in the suggested strategy is user registration. Personal information such as the user's name, address, and e-mail address are used for this. After registering, the user can log in with his or her username and password.

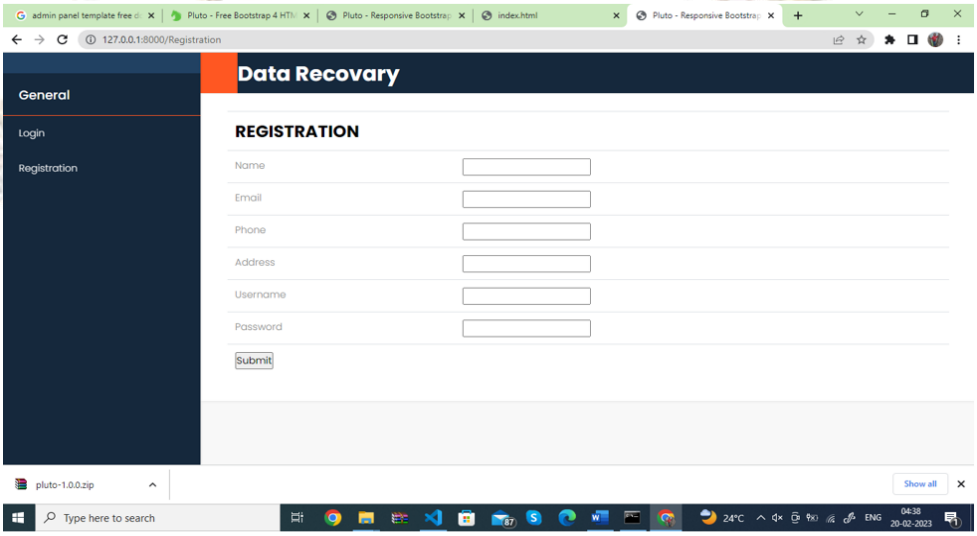


Fig 4. Registration Page

Then there is a recovery process where the server, localhost, username, password and database. Then it makes the connection. After connecting the database then the user can perform the recovery process.

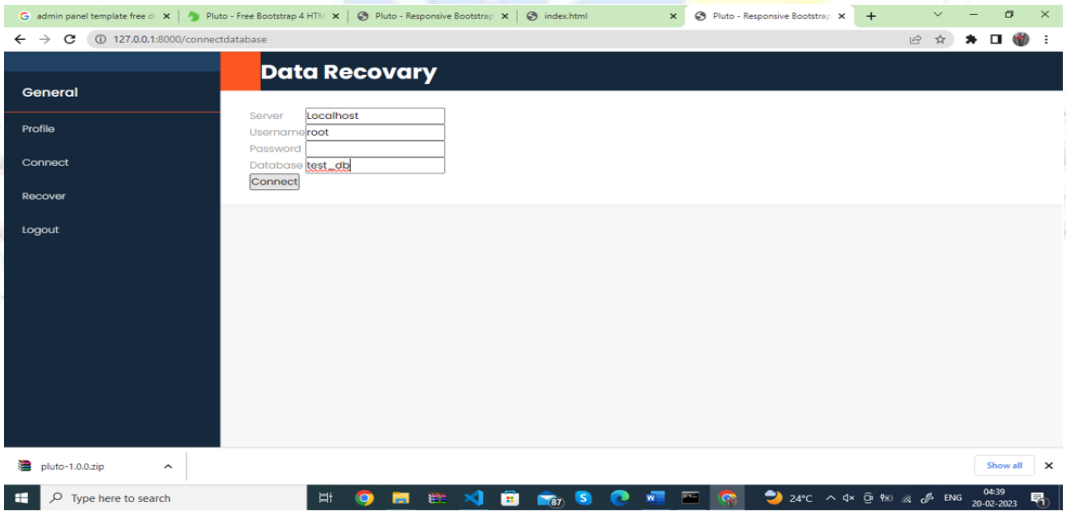


Fig 5. Recovery Process

## VI. CONCLUSIONS

Many people utilize databases to store and manage sensitive personal or business data. Furthermore, as the use of Internet of Things devices increases, databases allow innumerable users to access a range of apps and store the users' data and record; database forensics is becoming increasingly crucial for a forensic investigator. Although some prior studies provided a universal investigation strategy for DBMSes, there is a lack of practical knowledge on how to investigate DBMS and recover lost records. In the suggested method, the MySQL database is analyzed, which is an open-source relational database management system (RDBMS). Several system tables have been examined in order to recover deleted records independent of MySQL version. Here, we suggest a strategy for recovering lost records and implement it as an open-source programme.

In the future, we will conclude that the proposed strategy is also applicable to other industries that hold massive amounts of data

## IV. REFERENCES

[1] J. Yoon and S. Lee, "A method and tool to recover data deleted from a MongoDB," Digit. Invest., vol. 24, pp. 106–120, Mar. 2018.
[2] J. Wagner, A. Rasin, K. Heart, T. Malik, J. Furst, and J. Grier, "Detecting database file tampering through page carving," in Proc. 21st Int. Conf.Extending Database Technol.,2018, pp. 1–12.
[3] J. Sablatura and B. Zhou, "Forensic database reconstruction," in Proc.IEEE Int. Conf.Big Data (Big Data), Dec. 2017, pp. 3700–3704.
[4] C. Meng and H. Baier, "Bring2Lite: A structural concept and tool for forensic data analysis and recovery of deleted SQLite records," Digit. Invest., vol. 29, pp. S31–S41, Jul.2019
[5] S. K. Jung, J. W. Jang, D. W. Jeong, and S. J. Lee, "A study on the improvement method of deleted record recovery in MySQL InnoDB," KIPS Trans. Comput. Commun. Syst., vol. 6, no. 12, pp. 487–496, 2017.
[6] M. Xu, X. Xu, J. Xu, Y. Ren, H. Zhang, and N. Zheng, "A forensic analysis method for Redis database based on RDB and AOF file," J. Comput., vol. 9, no. 11, pp. 2538–2544, Nov. 2014
[7] R. Kumbhare, S. Nimbalkar, R. Chopade, and V. Pachghare, "Tamper detection in MongoDB and CouchDB database," in Proc. Int. Conf. Comput. Sci. Appl. Singapore: Springer, 2020, pp. 109–117.
[8] D. Litchfield, "Oracle forensics part 1: Dissecting the redo logs," NGSSoftware Insight Secur. Res. (NISR), Next Gener. Secur. Softw. Ltd., Sutton, U.K., Tech. Rep., 2007.
[9] M. D´ıaz, C. Mart´ın, and B. Rubio, "State-of-the-art, challenges, and openissues in the integration of Internet of Things and cloud computing," J. Netw. Comput. Appl., vol. 67, pp. 99–117, May 2016.
[10] G. Horsman, "Tool testing and reliability issues in the field of digital forensics," Digit. Invest., vol. 28, pp. 163–175, Mar. 2019.