ABSTRACT

Database security has been an issue by itself. Unauthorized use, malware infections, disclosure of essential data, physical damage, data corruption are some of the complications faced when there is no proper security system involved in protection of the database. The main challenge is the identification of the data that has been corrupted. In these cases if a proper database forensic tool is applied it will help to save the threats and would solve some issues in security.

This paper discusses about the selection of database and the usage of tools which are utilized by most of hackers to attack the database, it portrays the attacks through SQL injection tools, Havij and Sqlmap to take over the database through the application layer and later performing some commonly known attacks on the database. Forensic tools are used to determine the nature of attacks and breaches and also to determine if these tools are able to identify any kind of modifications to the database, through some popularly known forensic tools Pro Discover, FTK, IDEA and Sleuth Kit with Autopsy.
TABLE OF CONTENTS

Abstract ................................................................................................................................. ii

Table of Contents ................................................................................................................ iii

List of Figures ...................................................................................................................... v

List of Tables ....................................................................................................................... vii

1. Introduction .................................................................................................................... 1

2. Background and Rationale ............................................................................................ 2

   2.1 What is a Database? ........................................................................................................ 3

   2.2 Types of Database ......................................................................................................... 5

      2.2.1 Hierarchic Databases ............................................................................................... 6

      2.2.2 Network Database .................................................................................................... 7

      2.2.3 Relational Database ................................................................................................. 7

      2.2.4 Object Oriented Database ......................................................................................... 9

   2.3 Database Security Issues ............................................................................................. 9

   2.4 Overview of Database Forensics .................................................................................. 11

   2.5 Brief Introduction of Forensic Tools ............................................................................ 12

      2.5.1 Forensic Tool Kit (FTK) .......................................................................................... 13

      2.5.2 ProDiscover .............................................................................................................. 13

      2.5.3 IDEA ....................................................................................................................... 13

      2.5.4 ACL ........................................................................................................................ 14

      2.5.5 Arbutus .................................................................................................................... 14

      2.5.6 The Sleuth Kit ......................................................................................................... 15
3. System Design or Research ..................................................................................... 16
   3.1 Selection of Database ......................................................................................... 16
   3.2 Creation of Database ......................................................................................... 18
   3.3 Pro Discover for Forensic Analysis ..................................................................... 59
   3.4 Idea as Forensic tool ......................................................................................... 62
   3.5 AccessData Forensic toolkit .............................................................................. 64
   3.6 The SleuthKit with Autopsy Browser .................................................................. 67
4. Evaluation and Result ............................................................................................. 77
5. Conclusion ................................................................................................................ 78
Acknowledgement ........................................................................................................ 79
Bibliography and References ....................................................................................... 80
**LIST OF FIGURES**

- Figure 1: A logical view of hierarchic database .............................................. 6
- Figure 2: Logical and Tabular view of relational database ................................. 8
- Figure 3: Architecture of Arbutus software ...................................................... 14
- Figure 4: Official Website of MySQL .............................................................. 16
- Figure 5: Table t1 overview ............................................................................. 18
- Figure 6: Insertion of values to table T1 ......................................................... 19
- Figure 7: Executed query ................................................................................ 19
- Figure 8: Saved data in table t1 ..................................................................... 20
- Figure 9: Backup process of MySQL .............................................................. 21
- Figure 10: Overview of two tables .................................................................. 22
- Figure 11: Data of table Info .......................................................................... 23
- Figure 12: Data of table t1 ............................................................................. 24
- Figure 13: HTML query form .......................................................................... 24
- Figure 14: Output of MySQL.Php .................................................................... 25
- Figure 15: PHP code to process query from database ...................................... 26
- Figure 16: Login form with username and password ability .............................. 28
- Figure 17: PHP code to process the data of form shown in Figure 16 ............... 29
- Figure 18: Successful Login Result ................................................................. 30
- Figure 19: Invalid Login Result ..................................................................... 30
- Figure 20: Result page after invalid username ................................................. 30
- Figure 21: The result page for second query .................................................... 31
- Figure 22: Successful Injection ...................................................................... 31
- Figure 23: Form just to check the member description ...................................... 32
- Figure 24: The result page showing the description of john .............................. 33
- Figure 25: Result of Injection URL executions ................................................. 34
- Figure 26: The user interface of Havij .............................................................. 36
- Figure 27: Havij Interface with the simple target database URL ....................... 38
- Figure 28: Havij Results after pressing the analyze button .............................. 39
- Figure 29: Havij Log for both invalid and valid target URL .............................. 40
- Figure 30: Options to extract information from database ................................. 41
- Figure 31: The extraction of other database names ........................................... 42
- Figure 32: Table extraction log ...................................................................... 43
- Figure 33: View of Test database tables .......................................................... 44
- Figure 34: The view of extracted columns ...................................................... 45
- Figure 35: Column Selection .......................................................................... 45
- Figure 36: Extracted Data .............................................................................. 46
- Figure 37: SqlMap in BackTrack 5 R1 .............................................................. 47
- Figure 38: Checking for Injection .................................................................... 48
- Figure 39: The extraction of database names ................................................... 49
- Figure 40: The extraction of database names ................................................... 50
- Figure 41: Execution of Sqlmap with --column switch .................................... 51
- Figure 42: Extraction of Columns from Database Test .................................... 52
- Figure 43: Extraction of data from database columns ..................................... 53
Figure 44: Data of Table t1 from database........................................54
Figure 45: Extraction of data from second table info of database test........55
Figure 46: Server Log File..........................................................57
Figure 47: Interface of Toad.......................................................58
Figure 48: Data of Info Table.....................................................59
Figure 49: Backup Image before database intrusion............................60
Figure 50: Both Images and their Checksum...................................60
Figure 51: Comparison of Hash values after Deletion Attack.................61
Figure 52: Hash comparison after alteration attack............................62
Figure 53: Original Database Loaded in Idea....................................63
Figure 54: Database after deletion attack......................................63
Figure 55: Database after alteration attack....................................64
Figure 56: Imaging Verification of database prior to attacks with FTK image..65
Figure 57: Imaging and Hash verification of database after stealing attack....65
Figure 58: Hash Calculation after deletion attack on database image.........66
Figure 59: Hash Calculation after alteration attack on database image.......66
Figure 60: Autopsy server running with the instructions to open the browser..68
Figure 61: Autopsy interface in Browser.......................................68
Figure 62: Creation of Database images........................................69
Figure 63: First step to case creation............................................70
Figure 64: Case is created along with the directories for Case..................70
Figure 65: Addition of Host for the communication over network..............71
Figure 66: The host and case is created: adding an image.....................71
Figure 67: Adding new image with location, type and method..................72
Figure 68: Details of image.......................................................73
Figure 69: Calculation of MD5 Hash of Original backup image of database...73
Figure 70: Hash Calculation of database image after stealing attacks.........74
Figure 71: Hash calculation of database image after deletion attack..........75
Figure 72: Hash calculation of database image after alteration attacks.......76
LIST OF TABLES

Table 1: Comparison of Different Databases .......................................................... 5
Table 2: Hash Values Obtained from Forensic Tools .............................................. 77
1. INTRODUCTION

Database is considered most sensitive part of an organization where the sensitive information is stored; the database is involved nearly in every sector. Currently, internet is a great medium where databases are involved at greater level. Such as if a user signs up with Amazon, he or she would provide his or her relevant information including the credit or debit card numbers which would be stored in Amazon database. Suppose, the Amazon database is hacked, it would cause a great loss to both Amazon and its customers. The hacker may easily do shopping online after stealing sensitive information of credit cards without letting the owners of Amazon, or it may get too late for owners to detect the intrusion.

This paper is intended to carry out a survey over the database attacks and forensic tools which being utilized to prove the attacks and breaches in court of law. Due to the usage of MySQL database at wider level, it has been selected as a simple database and the most common threat to these types of database is SQL Injection. Various types of SQL injection attacks are carried out in this survey with the help of Havij and SqlMap and afterwards FTK, Pro Discover, IDEA and Sleuth Kit with Autopsy have been utilized for forensic purposes.
1. BACKGROUND AND RATIONALE

Database is the heart of any company or organization. Like any other important protected assets, the information that is stored in the database of a company should also be provided with security and protection. Due to the growing change and an increase in complexity and expansion of the company information systems, AAA (Authentication, Authorization, and Access) measures have been used.

Today’s technology is strongly concerned with the security measures to prevent data theft online. Sensitive information is preserved by companies, various technologies and programs are implemented to curb the attacks and breaches of a database. It can be comprehended that a student is intending to alter his grades, for this purpose, he hacks into the school database, a hacker may also intend to change the program behavior on a company’s server to steal information or money, a past employee or disgruntled one might be intending to change the behavior or function of a program to cause incorrect output and functions so that it may delete the records of customers or transaction history etc. In 2007, five students of California State University, Fresno allegedly broke into the system and were charged with identity theft, change in grades and conspiracy, unauthorized computer entrance [1].

Another case for unauthorized access was witnessed in San Diego health clinic, when the former manager damaged the company’s software and caused the data loss of schedules and appointments; he was sentenced to 10 years in prison. The other form of hacking has evolved in the form of web site damage, such as changing the information and contents of a website without knowledge or permission of its owner. This process is
carried out by individuals with grudges or hackers who want to draw information towards themselves [2].

To affect the website, the database of that website it attacked by the hackers to steal information regarding purchases, credit card numbers, passwords of users and administrators, to drop tables, to destroy database or to include the malicious code to change the face of website. After collecting the sensitive information, a hacker might break into the database again and again, what if he or she doesn’t change or alter information from database? It is for sure that the website owner would not be able to Figure out that his or her website is under attack or sensitive information is being stolen.

Various types of stealth attacks are initiated by hackers to break into the database of a website, forensic tools are utilized to unhide the aftereffects of those attacks. A hacker may break into the database of Amazon website, he or she would be capable of placing unauthorized product orders or he or she may be intending to steal the credit card numbers of their clients. Another form of database attack can also be referred to as a DoS attack which is intended to slow down the website so that it may not respond to legitimate users [2].

2.1 What is a Database?

A database can be referred to as a collection of relevant information which is organized in a proper manner and stored in tables so that it may be accessed effortlessly along with proper management [5]. Telephone directory can be referred as a suitable example of a database. A telephone directory can be considered a single file which contains the data of citizens with phone numbers. A telephone directory normally consists of name, address and telephone number. The telephone directory database which
is maintained physically brings many drawbacks such as the errors chances due to mistyping, data security such as the allotment of access to different people, dynamic updating and information sharing as it cannot be accessed by countless people simultaneously. The database management system was introduced to cope with this dilemma as it can be referred as a computerized system for record keeping which aids the users to create, modify, and production and deletion of records from database [6].

DBMS can further be referred to as a collection of data which is inter-related along with a set of programs for data access. The access does the retrieving, storing and manipulating of data in tabular form. There are various advantages which are interconnected with the implementation of DBMS such as it provides the ease of searching like to search the desired phone number, address or name within seconds without exploring the physical telephone directory. The most important advantage is the management, with DBMS, a user is provided with the ability to update, change, deletion of data from directory. The sharing of information has become effortless with DBMS as the same file can be accessed by several users without distribution of copies to each user.

DBMS provides an elegant way to prevent the data duplication as the physical directory may observe the duplication of entries, but such errors are easily detected with DBMS. This is considered secured in DBMS as it requires the legitimate entry into the system with the provision of user identity and password. Database managements system also provides the data independence such as the tables in database can be manipulated by known languages such as C, C++, SQL and JAVA for data retrieval, storage and manipulation. Some of the popular systems which are integrated with database include library management, railway reservation, hospital management etc. Relational database
management system (RDBMS) is one of the types of DBMS. With this database management system, relationships are defined between different tables in various forms. Table 1 processes several RDBMS systems that are available in markets: SQL from Microsoft, DB2 from IBM, and Oracle from Oracle Corporation, Sybase, MS Access, Ingress, PostgreSQL and MySQL. The database which is selected for survey in this paper regarding breaches and attacks is MySQL.

<table>
<thead>
<tr>
<th>Product</th>
<th>Number of Users</th>
<th>Data Location</th>
<th>Data usage</th>
<th>XML</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Single user</td>
<td>Multiuser</td>
<td>Centralized</td>
<td></td>
</tr>
<tr>
<td>Ms Access</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Ms SQL Server</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>IBM DB2</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>My SQL</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Oracle RDBMS</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Table 1. Different RDBMS Systems

2.2 Types of Database

The database management system in large organizations has been appreciated and there are various models have been developed and adopted for database systems implementation. There are four approaches that have been adopted to design database systems: network, hierarchic, relational and object oriented [7]. The approaches network, hierarchic and network are considered as historical instead of technical. In this contemporary era, the majority of developed systems are utilizing the relational approach for database design and implementation. There are some shortcomings in relational database which are also being tackled by the object oriented approach. Various commercial products are utilizing the object oriented database approach but they are very few in number.
2.2.1 Hierarchic Databases

The hierarchic database was introduced in 1968 as file management system for database. In this system, records are arranged in the form of sets which are integrated with each other with a special link called ‘ownership’. Hierarchic databases are ideal for those systems which can be modeled naturally by a hierarchical model; there exists countless models which are based on the IMS system. With many other systems, it becomes vague to implement the hierarchy system as it may cause the waste of resources due to the severe duplication of data.

![Hierarchic Database Diagram](image)

**Figure 1. Hierarchic Database**

Figure 1 shows the information regarding orders where the duplication of order can be seen effortlessly. Furthermore it does not show any information regarding those products which are not placed in orders. It means two types of losses are being faced i.e. loss of data and data duplication.
2.2.2 Network Database

Conference on Data Systems Languages (CODASYL) established the Database Task Group in the 1970s for the production of guidelines set for database file management approach. The new model which was developed by CODASYL was named the network model. This model works on links and records which can also be referred as the basic constructs [8]. This can be referred to as a set of physical pointers which normally initiate ownership between a set of records. Thus, it can be comprehended that the customer is the owner of orders and orders own stock items same as in the hierarchic model. The direction of ownership is not one directional, there may exist countless ownerships.

2.2.3 Relational Database

In 1970’s, Dr.E.F.Codd proposed the relational databases originally. It was afterwards marked as a stable model in 1980s and has been adopted by the coherent models of DBMS. The relational database approach depicts two dimensional tables which consist of columns and rows. Tables are called relations, rows are called tuples and columns are referred to as attributes.
Figure 2. Relational Database
Figure 2 shows four relations which are ORDERS, CUSTOMERS, STOCK and ORDERLINES. The CUSTOMER relation further possess three attribute CUSTNO, CUST-NAME and city along with three tuples. Similarly, STOCK relation also posses’ three attributes along with the tuples.

Besides this, ORDERS relation possesses two attributes along with five tuples which further depicts five orders in database. The ORDERLINES table posses’ eight tuples along with three attributes. The relational database works by associating the data with the values of attributes of different tables instead of using explicit pointers to establish links between record sets with network approach.

2.2.4 Object Oriented Database

All approaches which are described previously are also called classic approaches for the implementation of Database Management Systems which are solely based on passive set of data. Object oriented database approached consists of collection of classes. A class consists of structured objects which operate in same manner as other object do in same class.

2.3 Database Security Issues

Database is not just storage of data, it grants information to users. This is why database security is strongly concerned for the protection of sensitive data; it is not only concerned with the security but with the provision of data to its users in a safe environment [9]. The database security is associated with the controlled access to information instead of data. After the design and implementation of a database, it is strongly required to analyze the security issues. The proper mechanism should be adopted
to unhide the number of ways through which the data view can access in controlled relational database environment.

To ensure the security of a database, it is essential to utilize the database security manager for security and maintenance within the organization. The database security issue solution requires various tasks to be handled properly and in a timely manner. It is mandatory for the business entrepreneur to understand the database security issues and flaws to make it stable and hack proof. This can only be done by understanding the where, how and why of a database for future safety [10]. The very first thing which should be adapted is to check the database security issues regarding the daily maintenance. This process involves the audit of logs on a daily basis to ensure that no fraud or misuse of data occurred in database. This strongly requires the supervision of privileges of database along with the consistent updating of the user accounts access levels.

The database security management is also capable of allotting access levels to both the users and the programs which are actually interacting with the database. The users which are causing threats to the database, with this daily checking they can be curbed in a timely fashion. It is mandatory to deploy the database with the proper controls for access for the security of the database or the sensitive data will be at on risk. The database when upgraded strongly requires the post-upgrade evaluation by admin which ensures the consistent data security amongst all programs interacting with database. If this process is not performed, the database is open to be attacked by hackers.

Another database security issue is application spoofing. Hackers develop such applications that behave like genuine application. These applications interact with the
database allowing further unauthorized access to sensitive information. One of the main reasons which cause the eruption of database security issues is the split of positions. IT administrators are often assigned the duties of database security managers; this is done to save company’s expenses.

Another major database security issue is the usage of SQL injections which is being utilized as an invasive procedure and cannot be easily tracked [11]. These attacking procedures act like a piece of cake for hackers. During these attacks, incorrect queries are crafted by the attackers which are not filtered by SQL; this helps the attacker to pull down information from the database. With this method, a hacker can easily release all the information from database without letting the victim know about the requests. The information may include the credit card number, social security number, customer purchasing patterns, company products etc.

The very interesting phenomenon is that such types of attacks are rarely reported due the fact that most of companies who use special firewalls don’t admit that they are using flawed code. Another database security threat is known as buffer overflow which is being utilized to interrupt the business by shutting down a database.

2.4 Overview of Database Forensics

Database forensics can be referred to as a special branch of forensics which deals with the study of digital science of database along with the metadata [3]. This branch is not much different from computer forensics as the investigation process involves the same techniques applied to metadata and the contents of database. The information hence cached may also exist in server RAM. The database forensic are normally associated with the timestamps which are tested to ensure the integrity of data in database. These
timestamps are applied to the rows in relational tables during the inspection to test and ensure the integrity of the database contents.

The forensic examination may further be carried out with the identification of transactions in the database which ensures fraud detection. “Database forensics can be further referred to as the application of computer analysis and investigation procedures to accumulate the evidences of database to be presented in court of law”[4]. Various benefits are coupled with these techniques so as it becomes easy to trace user data definition and data manipulation language operations. It also helps the comparison of post and pre transactions.

Forensic analysis and techniques not only help to determine the source of an attack, but also helps to recover the data lost during the course of attacks, such as data rows or tables dropped by an intruder during the course of hacking. The main purpose of database forensic is to analysis of any security breach over database. Forensic tools are not only utilized to determine the security breaches in the operating system environment but also utilized to unhide the facts and Figures related to the attacker and their damage to system.

2.5 Brief Introduction of Forensic Tools

There are numerous tools which are being utilized to track the database attacks for the provision of proof in the court of law. Database forensic tools being utilized for forensics are ACL, Arbutus, Idea, ProDiscover, Sleuth Kit, Forensic Tool Kit etc. These tools provide several features which aid to examine the attacks and threats to database and also determine that to which extent the database is vulnerable.
2.5.1 Forensic Toolkit (FTK)

This tool is utilized for computer forensics to determine the digital theft from computer. This software has been validated by court for digital investigations on different platforms to ensure accurate computer forensic analysis. It also supports the analysis of decryption and cracking of software and database. The latest forensic tool kit is provided with the customizable interface along with the analytics, speed and scalability for enterprise class.

2.5.2 ProDiscover Forensics

ProDiscover Forensics tool is an efficient tool which has been developed as a security tool which enables the investigators to discover the data on computer disk for the provision of evidences in the legal proceedings of court. This tool comes with various benefits and features such as it creates the bit-stream copy for disk analysis which also includes the hidden section of HPA in non-intrusive method. For complete disk analysis, it searches through the whole disk including slack spaces. The most attractive feature is the preview feature which enables the investigator to view the deleted or hidden files without alteration of data over disk which also includes the metadata.

2.5.3 IDEA – Caseware International

IDEA is efficient data analysis software which has been designed to detect frauds and provides ample data analysis. With its integrated functions and features, it enables the investigator to save time during the investigation of data. This tool is widely adopted by the IT auditors and financial auditors at large scale. It is embedded with the remarkable features such as it enables investigator to detect the changes made to database
along with the maintenance of audit or operation logs. It is also capable of importing the test which is carried out over database. The import and export of data, enables in various format which are also accepted by other accounting software.

2.5.4 ACL

ACL provides software globally for continuous monitoring and audit analysis and further handles the financial management community. ACL not only provides audit analytic software but also provides professional services and expertise. The ACL technology aids the decision makers in financial institutions to detect the fraud, along with the assurance of risk reduction and profitability enhancements.

2.5.5 Arbutus Software

Arbutus is a company which is providing various solutions for data security and fraud detection software. This company currently provides the suite of products for business solutions such as fraud detection, analytics, migrate, query, instant warehouse and connect etc.

Figure 3. Arbutus Software [4]
Figure 3 shows the technology components which are utilized by core software for the provision of business solutions. Arbutus technology components consist of arbutus legacy link and arbutus analyzer.

2.5.6 The Sleuth Kit

This forensic tool is based on command line files and a C library which is utilized for the forensic analysis of a computer disk. This tool aids to investigate victim computer without interfering with its environment. This is because this tool is not actually based on any operating systems. This is why it can run on both the Windows and Unix platforms to process deleted and hidden files.

Due to the complications with command line interface of this tool, it has been integrated with the graphical interface allowing the easiness of investigation. It not only provides the image integrity, but also provides the case management, keyword searching and many other operations automatically.
3. SYSTEM DESIGN OR RESEARCH

3.1 Selection of Database

To perform the survey regarding breaches and attacks, MySQL database has been chosen. The very reason for the selection of this database is its common usage over internet. Major website owners are utilizing MySQL at backend for data acquisition and to pursue information. MySQL is a relational database managements system. In 2003, according to the information published by AB (MySQL developers), the number of installations reached more than 4 million [13]. MySQL has been utilized to power the website sites, data warehouses, search engines along with systems and applications. The major companies such as HP, Sony, Xerox and NASA benefit from MySQL.

Figure 4. Official Website of MySQL
MySQL is an open source database which has been developed by AB Company; giving strict competition to Oracle and Microsoft’s SQL servers. AB Company is located in Sweden (www.mysql.com, Figure 4). The open source term is mentioned to clear that MySQL can be modified according to needs. MySQL has been integrated with the multi-threading architecture which makes the concurrent communication robust and light speed. The core engine of MySQL allows the simultaneous transactions from clients along with the fast performance.

The earlier versions of MySQL were not included with many features such as the transactions, competing systems, referential integrity along with stored procedures, the very reason for not including these functions was to enhance the performance gain as these features produce server complexity and the performance is hit as result. Due to the rising demands from users, MySQL versions later than 3.23.34a were incorporated with the transaction support enabling the users to make choice between enabling and disabling them. MySQL 4.0 afterwards was integrated with the unique feature of query cache. This feature enabled the database to cache those queries which are being hit most; this phenomenon boosted the speed up to 200 percent fast without any specific programming on the part of users.

MySQL has been incorporated with the multi user support system. It means that numerous clients can establish simultaneous connections with the database for the acquisition of information and data. MySQL provides an efficient way to handle large and complex databases without performance drop. Tables having data in gigabytes can easily be contained in database. MySQL has been integrated the portability options such as it supports both non-Unix and Unix operating systems, which also includes Solaris,
Linux, OS/2, FreeBSD and all versions of Windows. It also supports Alpha, Intel x86, PowerPC, SPARC and IA64 along with countless hardware configurations. The team who developed MySQL has followed the standards and rules of SQL. The latest versions of MySQL support the ANSI SQL-99 standards.

3.2 Creation of Database

To create the test environment for database, XAMPP has been downloaded under the operating system Windows XP. The XAMPP is open source software which provides effortless ways for the configuration and creation of database through phpMyAdmin.

Figure 5. Table t1 overview

In Figure 5, a test database has been created with the initial table t1. These tables are inserted with the attributes of ID, Username, Password and description. The storage engine that has been adopted for this database is MyISAM. XAMPP provides efficient and effective ways for creations, manipulation, insertion and deletion of data through
graphical interface, however, the command line method is not necessary with XAMPP MySQL configuration. The Figure shows the overall view of columns and their attributes. Now the next step is the insertion of data into created database.

![Figure 6. Insertion of values to table T1](image)

In Figure 6, the insert option is selection for insertion of data into database. The column ID is set as primary key as index of database, it cannot remain null and should be unique in order to maintain the uniqueness of records in database.

![Figure 7. Executed query](image)

The insertion command in phpMyAdmin automatically crafts the command for the insertion of data as shown in Figure 7. When this command is further executed, the data is inserted into database.
In Figure 8, this is the overall view of database test with the table view, this Figure shows the values which are inserted into database through phpMyAdmin. Now a simple database is created which is also protected with the password protection. This sample database would be dealt with the server side script.

After the creation and insertion of data, the next step is the backup of database. The backup ensures the integrity of database such as if the database is hacked and the contents are changed, the backup can be utilized to restore the database to its previous state. However, the creation of database backup is essential on daily basis which ensures the safety and security of data.

<table>
<thead>
<tr>
<th>ID</th>
<th>Username</th>
<th>Password</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>John</td>
<td>123usa</td>
<td>John is a Database Programmer</td>
</tr>
<tr>
<td>2</td>
<td>David</td>
<td>usa123</td>
<td>David is a web developer.</td>
</tr>
<tr>
<td>3</td>
<td>Sam</td>
<td>sam123</td>
<td>Sam is Project Manager.</td>
</tr>
</tbody>
</table>
Figure 9. Backup process of MySQL

Figure 9 shows the option in phpMyAdmin to create the backup of MySQL database. With this feature, it becomes easy to include the required pieces of information. With export function the complete database in the form of compressed zip file can be exported. So the backup of the database is created so that it may be helpful to distinguish the changes before and after database attacks.

The language that was selected for database interaction is PHP. The server side script PHP refers to the preprocessed hypertext markup language, the PHP coding is utilized in web development, it is embedded into the HTML code which is executed at server and the user is presented with the compiled output of the embedded code. HTML
itself cannot perform the functions, for this purpose, server side and client side scripting is utilized to mold the data according to the needs.

There are various other languages which are being utilized for web development such as ASP.NET, JAVA, CGI and Pearl etc. People have developed content management systems which are running successfully over internet and they are all based over PHP. One of the most famous content management systems is Joomla. However, wordpress blog is also based on PHP and MySQL.

Now the next phase is the creation of server side scripting which can be utilized through browser so that the information may be accessed online. Hence for this purpose, the PHP code is induced for the safe and secured communication with the database. However, this communication is password protected and no unauthorized access can be granted without username and password.

Figure 10. Overview of two tables

Figure 10 shows a database named test, this database further consists of two more tables. These tables are used to save information of the employees working in an organization, the database is just simple and the concept is also kept simple, to test the vulnerability of database.
Figure 11. Data of table info

The Figure 11 shows that the data which is stored in table “info” of database test. This table consists of id, fname, age and password columns and has been assigned with values.
Figure 12 shows the table t1 of database test. This table consists of id, username, password and description. The ID is marked as primary index. An important thing to be noted here is that there is no relation defined between two tables in test database.

Figure 13. HTML query form

In Figure 13, a simple form to check member with just the username has been designed. This form takes the text in input box and after pressing the submit button, the information is fetched from database. Suppose, the username john is typed in the box, after pressing the submit button, description of john would be displayed.
Figure 14 shows the description regarding username input John. It is pertinent to note the address bar which is showing a string along with the address, this string is “username=John check=submit”. The values which are parsed by form with GET methods are showed in the address bar on next page; however, values which are passed through post method are hidden and cannot be viewed [14]. The URL which is acquired through GET can be used for reference. To acquire direct access to contents a simple link can be created such as the URL in address bar of Figure 14 can be used to access the information directly i.e. `<a href="http://localhost/mysql.php?username=john">Click here to know about John</a>`, this is an example to create a link and just by selecting the link, the user would be directed to information page.
Figure 15. PHP code to process query from database

```php
<?php
$username=$_GET['username'];
if($username)
{
    $conn=mysql_connect("127.0.0.1","root","") or die('Error: ' . mysql_error());
    mysql_select_db("test");
    $SQL="select * from t1 where Username='".$username."'";
    //echo "$SQL";
    $result=mysql_query($SQL) or die('Error: ' . mysql_error());
    $row=mysql_fetch_array($result);
    if($row['Username'])
    {
        echo 'Username:"'.$row['Username'].'"<br>
        echo 'Description:"'.$row['Description'].'"<br>
    }
    else echo "No Record!";
    mysql_free_result($result);
    mysql_close();
}
?>
```

Figure 15 depicts the code utilized to take the input from form through get methods. The condition ensures that the username is not null, if it is not null, a database connection is opened. After opening, a prepared statement of SQL is executed by PHP code to fetch data from database; the data from database is pulled row by row. The code is also integrated with the condition to check whether the database consists of some values or it is empty. If the table of database is empty, it would print that there is no record found in database.

However, if the record is found, it would print according to the pattern specified through echo. It should be noted here that there are two tables in the database, however the script which is crafted to communicate with database only communicates with one table t1. The script and database are kept simple, and hence there is no admin panel, so to check the integrity and accuracy of attacks, the second table in database is added.

Attacks which are intended to attack the company’s website is called as SQL injection. It is a technique which is used to exploit the vulnerability. There are three types of software vulnerabilities such as Cross Site Scripting (XSS), Injection Flaws, and
Malicious File Execution etc [15]. SQL injection technique is utilized to check the vulnerability. The injection weakness lies in the database layer of an application [16]. This vulnerability is found in the incorrect filtration of user input such as escape characters which are further embedded in SQL prepared statements or these statements get executed unexpectedly. It can be referred as vulnerability in general which happens by embedding one script or programming inside other.

A great number of websites are being hacked through SQL injections. It is very easy for attackers to find the SQL injection to get into the database of website. A website is integrated with the admin panel for accounts administration. To acquire the access to admin panels, admin username and password is required through which the database structure and its data can be easily altered by hackers. This technique has been utilized by hackers to steal passwords, secret information such as credit card number etc.

Many online advertising companies are also utilizing this technique to accumulate the information regarding online users such as their email address and phone numbers for the purpose of bulk advertisement. The most common platform which has been adopted On March, 28, 2011, an unexpected information was posted on SC Magazine which revealed that the official website of Oracle’s MySQL was hacked through MySQL injection and hacker have published the list of their username and password online [17].

The SQL injection occurs due to the poor programming and database handling. With SQL injection technique, the database server is interacted along with different commands to extract sensitive information from database. SQL injection enables the unauthorized user to steal all sensitive information from database such as private data. The process of injection is intended to execute invalid commands to database server to
find a backdoor. It is also utilized to change the contents of a website without owner’s permission.

There are several SQL injection attacks which are going to be discussed here. The very first thing that needs to be checked is the vulnerability of website to be attacked with the SQL injection. Countless websites over internet which are integrated with the login and password form so that registered members may acquire access to the protected contents. For this purpose, each user is assigned with unique identity and password, but hackers have invented a way through which they can bypass the login form to get access to the password protected area. If the website is not sanitized from programming perspective, it would be very easy to break into website.

![Login form with username and password ability](image.png)

Figure 16. Login form with username and password ability

The form shown in Figure 16 is integrated with the password function to check and ensure that the username and password which are saved in database are correct and the genuine user is trying to access the membership area of website. But without knowing the actual username and password, it should be easy to bypass the login process and would be effortless to gain access to the next page. Let’s try to understand the things going on in the background when the submit button is hit after submitting the username and password. Suppose that the username that will be tried is John and the password which is set for john is 123usa. When submit button is pressed, a query “SELECT user
from database where username='John' AND password='123usa'" is generated. If this command returns true, the user will be given access to the membership area, but the code is not sanitized in a proper way by the programmer, it would be the piece of cake for programmer to break into website and a backdoor would be opened for hackers to access the website illegally.

```php
<?php
$Username=$GET['username'];
$Password=$GET['password'];
if($Username) {
    $con=mysql_connect("127.0.0.1","root","") or die('Error: '.mysql_error());
    mysql_select_db("test");
    $SQL="select * from tj where Username="#Username," and Password="#Password.";"
    //echo "$SQL."<br>";
    $result=mysql_query($SQL) or die('Error: '.mysql_error());
    if($row["Username"])
    {
        echo "Username":"$row['Username']."<br>";
        echo "Description":"$row['Description']."<br>";
        echo "Record Synchronized!"<br>";
    }
    else echo "No Record!";
    mysql_free_result($result);
    mysql_close();
}?
```

Figure 17.PHP code to process the data of form shown in Figure 16

The php code that is shown in Figure 17 gives an idea that the username and password is stored in two variables: username and password. After that a connection is opened to database, after checking that the username has been put by the user, the code sends the query to database either the username and password are correct, if the username and password are correct then the result is printed out, however if one of them is missing, “No Records!” can be observed on the next page.
Figure 18. Successful Login Result

Figure 18 shows the result of input John and its password, the result shown in figure contains the username, description and status of record. It shows that the username and password were found in database accordingly and user is authorized to view this page.

Figure 19. Invalid Login Result

To check that the code is working in standard or not, a username smith with random password has been typed into the form, after pressing the submit button, the result page showed No Record in Figure 19. It means that there is no such user exists in the database, so there is no description available regarding that person.

The very first attack to the database would be to bypass the login process. For this purpose, a string is entered into username so that it may be executed with the database query command by the server side script. To bypass the login process, “a or 1=1--” would be entered in username field and the password field would be left blank.

Figure 20. Result page after invalid username
Figure 20 shows the result of the above crafted query; it means that the query did not work. However another query would be used again as an attack to bypass the login. Now, ‘a’ would be used as username and (‘‘) would be placed in password as an attempt to bypass the login process.

![Image](localhost/sqli.png)

Error: You have an error in your SQL statement, check the manual that corresponds to your MySQL server version for the right syntax to use near 'a' and Password=''' at line 1

Figure 21. The result page for second query

Figure 21 shows that the second query produced an error which depicts that the script is vulnerable, it is better to keep trying, for this purpose. One thing that is notable is the addition of and, or, 1=1--. The logic behind adding “1=1” is to execute the SQL query true, no matter either the entered values are valid or invalid, the result would be always true. Various combinations are described below:

- `username:' or 1='1 password:' or 1='1`
- `username:' or '1'='1' password:' or '1'='1`
- `username:or 1=1 password:or 1=1`

![Image](localhost/sqli2.png)

Figure 22. Successful Injection
After entering the code (“ or 1 = 1) into both the username and password field, the login page bypassed successfully, which depicted that the membership area can be access by the hackers by injecting invalid strings to both the username and password fields. The creation and application of above input string successfully bypassed the login page and password protected area was successfully accessed. The database has been successfully attacked to acquire unauthorized access to password protected area.

Now, the second attack comprises of an SQL injection which would reveal the sensitive information underlying in database. The SQL injection would first test that either the database is vulnerable or not, after detection of vulnerability, different SQL injections are applied to get access to sensitive information stored in database such as username, passwords, and other tables and their contents. This would work in special manner and the process contains the procedures which strongly need to be followed one by one.

![Form](image)

Figure 23.Form just to check the member description

Figures 23 shows the first form that has been created earlier but now it contains the value “john”, this form is intended to check the information such as description related to john on next page. Now, the injection technique is used to retrieve information regarding other tables and their contents.
Figure 24. The result page showing the description of john

The most important thing that is notable in Figure 24 is the URL, to attack the database it is necessary to mould the string in order to check that either database gives error or not. This is the first step to check the database vulnerability, if database gives error; it is very easy to extract information through addition database queries. The URL that is obtained after the submission of “John” in the form is the string “http://localhost/mysql.php?username=john&check=submit”.

This shows that the username which was submitted to fetch the information from database was through get method, this is because the username is visible in URL. Now the next step is to include something fishy in this URL in order to make the database erroneous. For this purpose, an apostrophe would be added anywhere in the URL, to ensure the erroneous output, the apostrophe would be placed after the name “john”, so the modified URL would be:


Now, the next step is the execution of the above crafted string.
Figure 25. Result of Injection URL executions

Figure 25 shows that the URL which was crafted for the purpose of injection worked and it produced a database error. An important thing that is worth to be noted here is that the database server name is visible through this injection. The error page shows that the database server is running “MySQL server” and the URL entered produced an error.

There are various other options which can be tried if this option fails to fetch the desired results. Such as the variable username can be assigned the value “john union select 1—”, this can be done only if the first method of intrusion fails. Since, it has been checked that the simple database is vulnerable and can be easily hacked through SQL injection, the next step is the extraction of information from database. However, for this, different combination strings in URL would be utilized to execute database commands.

For this, an SQL command would be included in the string to check the number of columns; this strategy is simple as when the query would be executed with the increment of 1, nothing would happen. But at a certain point when the database gives error, the number of columns contained in that table of the database can be extracted.

The above described way of attack is very basic, but in the current era of technology, SQL attacking software provides their services with the surety that it is easy to break in. Software over internet which are used for the purpose are also available for free, software which are developed for the purpose to check the vulnerability are also
being utilized by hackers to exploit the security of those websites which are programmed with weak codes.

There are various commands which actually work in background and the pre made software run those command through graphical interface, this saves the attacker from overhead of creating and typing the injection URL again and again in the address bar. The attacking software carries out all process itself and produces random strings from injection again and again until the website database is hacked. The software which I am about to use to attack my simple database is named as “Havij”.
Figure 26. The user interface of Havij

Figure 26 shows that this software program provides effortless services as it is visible that a text box is provided to input the target URL. The text box for URL already shows the syntax or format of URL which needs to be placed here. The next box is regarding keyword, if the keyword is known then enter it in the target; it can be simply written here, otherwise leave it unchecked so that it may get checked automatically.
The third box is syntax, if a hacker is intention is to attack the website with special type of query; the box may be checked along with filling of syntax in the text box. There are several options provided for the database which can be selected manually or let the software choose automatically. The method option box provides two options i.e. get and post. The next box tells about the type of injection, the options let the attacker select the integer or string option. But, for the attack, “auto detect” is utilized for all of options so that the software may elegantly select all correct options. However, these options can be manipulated for advance injections.

Now, the URL of script which is communicated with database through username and password to show information on next page such as description would be utilized in Havij to attack and dominate it completely.

When this tool is opened, it shows that example of URL written in Target box, so to attack the database; URL with same pattern would be input in the target box. The sample URL is “http://www.target.com/index.asp?id=123”. This URL clearly indicates that the variable “id” is assigned with the value through get method. So the URL of same pattern is entered to attack the simple database.
Now, the next step would be the execution of this software which would process all types of SQL injections on this database that is shown in Figure 27. The most important benefit with this tool is that hackers don’t need the special type of expertise to attack any website with SQL injection.
Figure 28. Havij Results after pressing the analyze button

The URL which is depicted in Figure 28 in the target box is not in accordance with the sample which was already provided in the target box, an additional string “&check=submit” is included in the string, so it should be excluded to make it work.
Figure 29. Havij Log for both invalid and valid target URL

In Figure 29, after pressing the analyze button, it first resolved the host and after that it ran a script in background to unveil the Web Server which further showed all information regarding server in log box.

The next step was to determine the keyword, so this tool found “database” as keyword. The very next step after that is to determine the type of injection such as it would be integer or string, however, the “auto detect” option was already selected so it discovered the type automatically and found the injection type as string. After that, the log shows that this tool tried to find the MySQL version and found that version is greater or equal than 5.
The next step after that was to determine the number of columns of current database; the log showed that the current database consists of 4 columns. The next line shows that this tool tried to find the column of that string which was passed through the username variable and discovered it in column 2.

After accumulating the bundle of this information, this tool concluded that the target is vulnerable. So the next step would be the extraction of database names from the MySQL servers, which is away from just a single click. Now, the other components of Havij would be utilized to access the information pertaining to the database.

Figure 30. Options to extract information from database
It has been shown in Figure 30 that options such as tables, get databases, get tables, get columns and get data can be easily utilized to accumulate information from database in unauthorized manner. The most important thing which should be noted here is the database “test”. The process of analyzing revealed that the current database is test and can be easily hacked, let’s see that either it would be able to know about all other databases present on MySQL server.

Figure 31. The extraction of other database names
This tool is great to attack and it successfully attacked the simple database. As seen in Figure 31 it did not only extracted the name of the current database, but also acquired the names of all the other databases present on MySQL. Now, to fully attack the database, the name of tables is essential. For this purpose, the button “get table” would extract all information regarding tables present in all databases just with one click.

Havij log shows that it executed the injection code in background and discovered databases one by one. It discovered total 6 databases just in seconds; the core functionality of Havij also depends upon the speed of internet connection, the higher the speed, the stronger would be the attack.

Figure 32. Table extraction log
In Figure 32, the log has been taken from Havij, this log occurred when the button of get table was pressed, however, this attacking tool provides the ability to choose the specific database for data extraction, however, simply one can be selected, or all databases can be selected to steal information from there. Now, the real part is to extract information from database.

![Figure 33. View of Test database tables](image)

The most important thing which is pertinent to note in Figure 33 is the number of tables in database. Since, the PHP server side script was only communicating with the table “t1” in “test” database and there was not linked with the other table, Havij tool exploited the security and also extracted information regarding second table in current database. Now the next step would be the extraction of columns in test databases.
After checking the table, this tool has revealed the columns which are shown in Figure 34, customizing the columns selection can be easily done. So here, all columns are selected to extract information from tables. However, the table “info” is selected from the database test. 

Figure 35. Column Selection
Now, the next step is the selection of columns, so columns which are selected are appeared in the adjacent window. By selecting the Get Data button all the information is retrieved.

Figure 36. Extracted Data

The data has been extracted and can be saved with the save data or save tables option. The Figure 36 shows that the selected table which was not interaction with the PHP server side script is also synchronized with this attacking tool. Hence the database is attacked and got hacked effortlessly with Havij free tool.

Now, the most famous operating system which is being utilized over internet for hacking and exploiting is Back Track, the new version which is released recently is Back Track 5 R1. This operating system is based on Linux and contains all those tools which can be utilized for hacking and exploiting. However, an efficient tool has been integrated in Back Track to take over the database which is based on command line. This tool is
called “sqlmap” and it works in the same manner as Havij. This tool is based on injection techniques, various injection techniques such as error based, blind, string and integer based are tried with this tool. It also provides the wizard option where a hacker can simply put the website name and with other configuration. Sqlmap can also be acquired from its website through Google, but to run this, a hacker needs the python engine with basic understanding to run python scripts.

Figure 37. Sqlmap in BackTrack 5 R1
Figure 37 shows that opened terminal of sqlmap in Back Track which shows the very basic commands which are utilized while executing the commands. Now, the next step would be the checking of URL either it is vulnerable or not.

Figure 38. Checking for Injection

Figure 38 shows that the URL contains 192.168.1.3; the very reason behind this is that the operating system Back Track has been run on Virtual machine, the XAMPP server was executed on XP operating system with local area network IP address 192168.1.3. Sqlmap is checking for the connection, parameter type, injection and blind injection including AND, WHERE, HAVING etc. The log in Figure is portraying that the target is injectable, so the next step would be the extraction of database names with the help of customized injection. For this purpose, the switch --dbs would be utilized; the
inclusion of this switch uses the injections with the combination of string to pull the data out of database. To extract database through Sqlmap from target, “./Sqlmap –u http://192.168.1.3/mysql.php?username=john --dbs” would be executed through this tool.

Figure 39. The extraction of database names

Figure 39 shows the execution of Sqlmap with the switch --dbs from database. The plus point with Sqlmap is that it always creates a session file and takes the relevant information for the execution of other switches. The timestamp with info shows that the relevant information regarding this URL is being taken from session file, the log shows that it has discovered that the version of SQL server is 5, the log file indicated the injection type which already has been tested with --u switch.
Figure 40. Extraction of Database Names

The result which is shown in the Figure 40 has been acquired after the execution of --dbs switch with the URL. This result showed all underlying database which are currently present on the database. However, the current target to which the attack is planned is test. So, the tables, columns and data of the current database are needed to which the MySql. Php is interacting. The above Figure shows that the tool has extracted 6 databases in total and saved the logged in the file which is named over the URL.
The usage of --column switch with Sqlmap has been shown in Figure 41, this shows that the command is executed from start, again the data regarding URL has been accumulated from log file and the string based injection process is started.
Figure 42. Extraction of Columns from Database Test

As shown in previous figures of database creation, there were two tables created in the database test, however the PHP script was only interacting with the table t1 of database, but the other table “info” which was also created in same database test has also been synchronized by Sqlmap. As seen in Figure 42, the structure has been extracted with the switch –column. The next step is the extraction of data underlying in database tables.
In this step, another switch has been added with the complete command i.e. –dump, with this switch all of data in database is extracted from database and it is further saved in CSV file as shown in Figure 43.
Figure 44. Data of Table t1 from database

After executing the log Figure 44 shows the data and in rows and columns form which has been extracted from MySQL database. The data is also saved in dump file which is named as t1.csv. Further, the log shows the synchronization of data from other table.
The data has been shown in rows and columns which have been extracted from table “info” of database “test”. This data is further saved in a file named info.csv as shown in Figure 45.

The database is attacked with manually with sql injection script, after that it has been attacked with a tool named Havij which is also available both in free and paid version. However, to attack the simple database, the free version has been utilized which successfully found the backdoor to database and extracted all databases, tables, columns and rows of data from MySQL server.

Secondly, the most famous tool Sqlmap has been utilized under Linux environment. This tool is solely based on command line, however, the commands were
executed and the database is successfully exploited through the injection techniques with SQL map. All data has been saved in files named after tables.

Since, all information regarding MySQL database has been extracted by the attacker, he also has the access to database, and he can easily attack the database and add, delete and modify the information lying in database. However, the log shown in Figure 46 still shows that the database has been attacked; now it can be analyzed if either alteration, modification of deletion of data has been carried out or not.

The next part of this survey is the forensic study of the database to reveal that either the database is safe or not. The goal of forensic study is the detection of attack and collection real-time data. The forensics of database involves the following goals [18]:

- Detection of Security Breach
- Determination of Scope of Intrusion
- Retracing user DML and DDL operations
- Identification of Pre and Post data transactions
- Recovery of deleted databases

According to the first point, it is essential to know the security breach, since the scope of MySQL database is from application layer to database layer, there no special forensic tools developed for this purpose. However, to ensure the safety and security of database, Intruder Detection Systems from various companies are introduced. The traditional forensic tools such as sleuth kit, windows forensic tool kit and ProDiscover cannot be utilized to for the database forensics; this is because these tools work in the scope of windows at cluster level. However, the scope and structure of MySQL database is different and involves the usage of an Intrusion Detection System.
The intrusion detection system is utilized to monitor the activities of users with database, it logs all kind of acts which are performed by both legal and illegal users, during the forensics, these logs and information can be utilized as proof. There is no specific forensic tool however, the forensic regarding MySQL don’t involve any tool as it can be carried out by simple tools.

For the detection of security breach, it is necessary to check the server logs continuously to detect that either an attempt for has been tried to acquire the unauthorized access or not. For this purpose, the server log in Figure 46 is checked.

```
127.0.0.1 - - [11/Nov/2011:08:59:46 -0800] "GET /mysql.php?username=999.9%20UNION%20ALL%20SELECT%200x31303235343830303536,(SELECT%20concat(0x7e,0x27,Hex(cast(t1.Username%20as%20char)),0x27,0x7e)%20FROM%20'tes t'.t1%20LIMIT%202,1)%20,0x31303235343830303536,0x31303235343830303536%20and%20x='x HTTP/1.1" 200 65 "." "Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; SV1; .NET CLR 2.0.50727) Havij"
```

Figure 46. Server Log File

The above Figure shows the server log file, this file clearly indicates the attack which has been carried out over the database. It shows that the union and select statements were utilized to attack the server. It clearly indicates that two tools utilized to attack the database are Sqlmap and Havij. The excess of queries indicate that the
injection was successful and the information has been extracted from the database by the attacker. But, to ensure that the database is not been altered, a tool named toad would be utilized to Figure out either the database was same or the contents or schema was changed by the intruder.

![Figure 47. Interface of Toad](image)

The Figure 47 shows the interface of toad which is connected to the local database with authorize admin privileges. However, this tool would be utilized to ensure that the data is safe, the intrusion happened or not.
In Figure 48, it is clear that the data is unchanged and there is not addition of any database, table, column or tuple on sql server. However, log shows that the database has been read and extracted by the attacker has not harmed the structure of database by including any information. The tool which is used to observe the transactions in database shows that there is no change in data. Now, after carrying out the attacks on database for stealing, modification and alteration of data the forensic tools are utilized.

3.3 Usage of Pro Discover for Forensic Analysis

To analyze the change occurred in the database, Pro Discover would be utilized to check the database integrity. For this purpose, the original database backup image created with Pro Discover forensic tool. First, the hash checking would be carried out through ProDiscover Forensic tool to check the stealing of data from database.
Figure 49. Backup Image before database intrusion

Figure 49 shows the image which is created through Pro Discover, now the second step would be MD5 hash check for stealing attack from database.

Figure 50. Both Images and their Checksum
It is evident in Figure 50 that the MD5 hash checksum is same for images, the image before stealing intrusion and the image after stealing intrusion. ProDiscover shows no change for SQL injection attack for data stealing.

![Figure 51. Comparison of Hash values after Deletion Attack](image)

It is evident from Figure 51, that the attack caused the change in database which has been captured by hash values. The difference between original backup and backup after deletion clearly indicates the deletion attack over database.
In Figure 52, the database which had been altered during the attack with the addition of extra queries is compared with the original database. It is evident from hash values of original and altered database that database has been attacked.

### 3.4 IDEA as Forensic Tool

IDEA is another tool which is used for data analysis, but it can also be used as forensic studies such as with this tool it analysis can be carried out through graphs, tabular comparisons and many other options. But, for survey, the intention to utilize this software to unearth any change occurred in the simple database. For this purpose, database, after observing attacks, is loaded in IDEA for to check the database integrity.
Figure 53. Original Database Loaded in Idea

Figure 53. Shows that for using the IDEA tool, first the database is loaded in the tool.

Figure 54. Database after deletion attack
Figure 54 is the image of the database after the deletion attack has been performed on the database.

![Database Image after Deletion Attack](image)

**Figure 54. Database after deletion attack**

Figure 55 is the image of the database after the alteration attack has been performed on the database.

![Database Image after Alteration Attack](image)

**Figure 55. Database after alteration attack**

IDEA is an authentic auditing tool and above Figures showed the change in original database after deletion, alteration attack. A log has been saved using this tool. This tool will be helpful in identifying the types of attacks performed. Even if any kind of information is stolen form the database, then this tool easily identifies that, as the log consists of information regarding each transaction.

### 3.5 AccessData Forensic Toolkit

AccessData forensic tool kit is considered another best and handy tool for forensic investigation; this is the very reason to include it in the survey. After downloading the
demo version, the FTK imager has been downloaded to create the image of USB drive. The image for both database files i.e. before and after attacks, are created accordingly. With this forensic tool, not only the MD5 hash is obtained but also the SHA1 hash values, in addition to this, these hashes are also verified.

Figure 56. Imaging and Hash Verification of database prior to attacks with FTK imager

Figure 56 shows the MD5 and SHA1 values of the database before any attacks have been performed on the database.

Figure 57. Imaging and Hash verification of database after stealing attack
Figure 57 shows the MD5 and SHA1 values of the database after the stealing attacks performed on the database.

Figure 58. Hash Calculation after deletion attack on database image

Figure 58 shows the MD5 and SHA1 values of the database after the deletion attacks performed on the database.

Figure 59. Hash Calculation after alteration attack on database image

Figure 59 shows the MD5 and SHA1 values of the database after the alteration attacks performed on the database.
It is evident from above shown Figures that the Hash values obtained from databases prior and after stealing attacks are same however, the change in hash values after deletion and alteration attacks can be witnessed through this efficient forensic tool.

3.6 The SleuthKit with Autopsy Browser

The Sleuth kit is a library of C files or it be referred as the command line file collection and Forensic analysis tool for system. It enables the investigator to peer into the files system of suspect computer without changing or affecting the underlying data. The core function of this forensic tools is it independence of operating system which enables it to process hidden and deleted files. To integrate the graphical user interface, Sleuth kit is integrated with Autopsy forensic browser which has simplified the process of investigation through Sleuth Kit forensic tool. With Autopsy case management, keyword search along with image integrity and other automated operations are simplified.

The Sleuth kit has been downloaded from its official website, to run this under Unix OS environment, a special tool Cygwin has been downloaded and after downloading the source code, it was first compiled in Cygwin, the autopsy browser source code was also downloaded and after configuration of both the Sleuth Kit and Autopsy browser, the autopsy server is launched.
Figure 60. Autopsy server running with the instructions to open the browser

It is clearly shown in the Figure 60, that both the SleuthKit and Autopsy browser are successfully configured and now are running OK. The next step is the browsing of URL instructed in the Figure to continue towards investigation.

Figure 61. Autopsy interface in Browser
After configuration and launching the server, the Figure 61 shows the interface with three options; Open case: this option is utilized to open the previous or already created case. News Case is for opening a new case, in case of any difficulty, Help button can be clicked for instruction and information. Now, the next step is the creation of new case and calculation of MD5 hash to ensure the integrity of both the images. The image backups of the database are strongly needed.

![Image of interface with options]

Figure 62. Creation of Database images

The process of creation of images has been carried out by using the command DD in UNIX environment as shown in Figure 62. The images for both databases; before attack and after attack databases, has been created in the images folder in \texttt{usr/local/images} directory in Cygwin folder. These images are named badckup1.img and backup2.img.
Figure 63. First step to case creation

Figure 63 shows the creation of the case with the SleuthKit tool.

Figure 64. Case is created along with the directories for Case

Figure 64 shows the case is created with other directories, and named as test.
In Figure 65 an addition of the host for communication is shown.

Figure 66. The host and case is created, adding an image
In Figure 66, Host and the case are created and the image for verification can now be added.

![Image of adding a new image with location, type and method parameters](image)

**Figure 67.** A new image with location, type and method parameters is added.

In Figure 67, addition of a new image with parameters like type, location and method is shown.
Figure 68. Details of the image

Figure 68 shows the details of the added image.

Figure 69. Calculation of MD5 Hash of Original backup image of database
Figure 69 shows the MD5 values of the database before any attacks have been performed on the database.

Figure 70 shows the MD5 values of the database after the stealing attacks performed on the database.
Figure 71. Hash calculation of database image after deletion attack

Figure 71 shows the MD5 values of the database after the deletion attacks performed on the database.
Figure 72. Hash calculation of database image after alteration attacks

Figure 72 shows the MD5 values of the database after the alteration attacks performed on the database.

Above figures show no change before and after stealing attacks, however, through SleuthKit with Autopsy browser hash calculation, the difference of hash values for database after deletion and alteration attack can be witnessed in above Figures.
4. EVALUATION AND RESULT

To carry out the forensic process through forensic tools Pro Discover, IDEA, AccessDATA forensic tool and Sleuth kit has been utilized. The forensic process in Pro Discover, FTK and Sleuth kit is different. The integrity of database has been checked through images, the MD5 hashes i.e. Pro Discover, FTK and Sleuth Kit can easily detect the attack over database, however, IDEA tool is utilized to check analyze and manipulate the database contents.

<table>
<thead>
<tr>
<th>Tool</th>
<th>Hash value of original database backup</th>
<th>Hash value after steal attack</th>
<th>Hash value after deletion attack</th>
<th>Hash value after alteration attack</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pro Discover</td>
<td>Baa213ff2f9e8f1f8f8ac6c89307c1be8</td>
<td>Baa213ff2f9e8f1f8f8ac6c89307c1be8</td>
<td>BS6b768f91cecc8fa44da3f3a42b90a114</td>
<td>F7b179d9c214c3424641f533909e70</td>
</tr>
<tr>
<td>FTK</td>
<td>7ba63143ca5dde2069a5456a44a3e1f18</td>
<td>7ba63143ca5dde2069a5456a44a3e1f18</td>
<td>5d5b9be3bd13b53e10b2de2a7ca1b60</td>
<td>16a77e236821692a57decc0c4c5a801</td>
</tr>
<tr>
<td>Sleuth Kit</td>
<td>7ba63143ca5dde2069a5456a44a3e1f18</td>
<td>7ba63143ca5dde2069a5456a44a3e1f18</td>
<td>5d5b9be3bd13b53e10b2de2a7ca1b60</td>
<td>16a77e236821692a57decc0c4c5a801</td>
</tr>
</tbody>
</table>

Table 2. Hash Values Obtained from Forensic Tools

It is evident in Table 2 that the data remained unchanged after the steal attack. However, hash values showed difference after the deletion and alteration attack. The hash values of FTK and SleuthKit are same due the fact that these tools utilize the DD raw image format. IDEA tool can detect any modifications to the database through analyzing the log file.
5. CONCLUSION

Database is an important part of any organization. The information that the company has is very secure and may include the credit card number, social security number, customer purchasing patterns, company products etc. With loss of such important data there is a great threat for the loss of personal information. Some of the database security issues seen are post-upgrade evaluation by admin, application spoofing, split of positions, SQL injections and etc. These database security issue solution requires various tasks to be handled properly and in a timely manner.

Database forensics can be referred to as the application of computer analysis and investigation procedures to accumulate the evidences of database to be presented in court of law. Forensic tools like FTK, ACL, Arbutus and Sleuth kit are used in this paper.

To conclude, Pro Discover, FTK and Sleuth Kit remain unable to detect the injection attack to steal the information, however these tools successfully detected the alteration and deletion attacks over database. The biggest drawback of these forensic tools is that it is unable to detect the intrusion unless the hacker portrays any change in database. But the Database forensics tool IDEA is capable of identifying any kind of intrusion when compared to the other tools discussed. As IDEA has a log of all the transaction, it makes it easy to identify any modification or stealing of data.
ACKNOWLEDGEMENT

The preparation of this report and completion of the project was successful because of the never ending support and guidance of Dr. Mario Garcia, Professor of the Department of Computing Sciences, Texas A&M University – Corpus Christi.

I would like to express my sincere thanks to Dr. Longzhuang Li, Associate Professor of Department of Computing Sciences at Texas A&M University – Corpus Christi for his suggestions, comments and guidance throughout the project. His support has tremendously helped to ensure to the success of the project.

I would like to express my sincere thanks to Dr. David Thomas, Associate Professor of Computing Sciences, Texas A&M University – Corpus Christi, for his unending support and warm wishes that helped me to concentrate on completing my project.

My sincere heartfelt thanks to all the faculty, and staff of the Department of Computing Sciences for their outstanding support.

Last but not least, I would like to thank my parents, my family and my friends who provided the much needed moral support and boosted me in reaching the successful completion of the project.


[21] Peter Frühwirt, Markus Huber; Martin Mulazzani, Edgar R. Weippl; “*InnoDB Database Forensics*” IEEE 2010.