International Journal of Computer and Information System (IJCIS)
Peer Reviewed – International Journal
Vol : Vol. 01, Issue 01, May 2020
e-ISSN : 2745-9659
https://ijcis.net/index.php/ijcis/index

Binary Log Analysis on MySQL to Help Investigation Process Against Database Privillege Attacks

1st Siti Rokhmah, 2nd Ihsan Cahyo Utomo, 3rd Muqorobin
[1,3]Institut Teknologi Bisnis AAS Indonesia Surakarta
[2]Universitas Muhammadiyah Surakarta
[1,3]Jl. Slamet Riyadi No. 361 Windan, Makamhaji, Kartasura, Sukoharjo, Indonesia
[2]Jl. Ahmad Yani, Pabelan, Kartasura, Sukoharjo, Central Java, Indonesia 57162
[1] elfathiey@gmail.com, [2]lcu886@ums.ac.id, [3]robbyaullah@gmail.com

***Abstract—Database is an important part in managing information, because a* database is a collection of data that is processed to produce information. because of the importance of the database, many crimes are directed to attack the database, both attacks against access rights or attacks against the data itself. My SQL is a Database Management System (DBMS) that provides several facilities, one of which is the logging facility. Binary Log is a type of database log in the form of binary digits that contains some information including the record of the time of the transaction, the user who made the transaction and the order in the transaction. With the Binary Log, it can be seen when the transaction occurred, who made the transaction and what transaction occurred in the database. The recording of transactions in the Binary Log can be used as one way to carry out an investigation process in the event of an attack on the database. In this study the focus is on analyzing transaction records in binary logs, namely when, who, dam and what information can be taken from the Binary Log. The output of this research is a table of binary log investigation results and its relation to database attacks.**

Keywords— Binary Log, MySql, Database, Database Attack

I. INTRODUCTION

The database is an important part in managing information systems, it is because the database manages a lot of important company data that is accessed by many users. Therefore, various attacks are aimed at databases. In one of the studies conducted by one of the largest Cyber Security organizations namely impreva with the theme of ten ten database attacks, there are 10 attacks most often aimed at databases, 3 attacks ranked top are attacks on access rights, attacks on unmanaged sensitive data and attacks on database transactions.[1]

However, many database crimes cannot be traced due to lack of investigation into the attack, so many attacks on the database are not handled properly. In addition there is not much research that addresses the process of investigating database attacks. there are several studies relating to database attacks, among others, research relating to data reconstruction techniques using the redo technique on inno db storage machines [2] Other research is research related to the forensic database framework that discusses the forensic database inquiry framework [3].

Therefore, we need a way to help the process of investigating database attacks, one of which is by analyzing the database log. MySql is a DBMS (Database Management System) that provides many features including the Log feature. There are several types of database logs, including Binary Logs that contain records of when the transaction occurred, who made the transaction and what the transaction contained. so by analyzing the binary log database records will be obtained that will help the investigation of database attacks.

II. RESEARCH METHOD

The data used in this study is data from the academic information system at STIE AAS Surakarta, where the data comes from the tables in the academic information system database of STIE AAS Surakarta.

In this research a transaction simulation will be carried out on the STIE AAS Surakarta academic information system, a database system that uses MySql as its database management system, the transaction includes input data transactions, data update transactions, data delete transactions and query transactions.

After conducting a transaction simulation, an analysis of the database log will be performed, the log being analyzed is a binary log, so that a database access time record will be obtained, the user accessing the database and other records related to the database transaction [4].

2.1 *Binary Log*

MySql Server is a very popular open source based Database Management System (DBMS). Here is the architecture of Mysql Server [5]

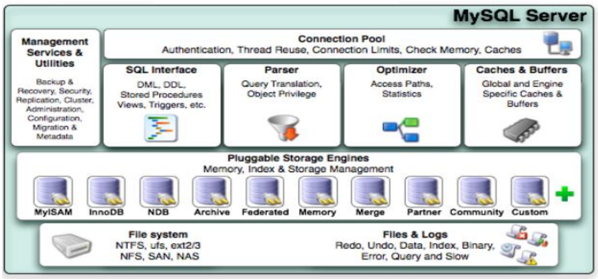

MySQL Server
Management Services & Utilities
Backup & Recovery, Security, Replication, Cluster, Administration, Configuration, Migration & Metadata
Connection Pool
Authentication, Thread Reuse, Connection Limits, Check Memory, Caches
SQL Interface
DML, DDL, Stored Procedures Views, Triggers, etc.
Parser
Query Translation, Object Privilege
Optimizer
Access Paths, Statistics
Caches & Buffers
Global and Engine Specific Caches & Buffers
Pluggable Storage Engines
Memory, Index & Storage Management
MyISAM
InnoDB
NDB
Archive
Federated
Memory
Merge
Partner
Community
Custom
File system
NTFS, ufs, ext2/3
NFS, SAN, NAS
Files & Logs
Redo, Undo, Data, Index, Binary, Error, Query and Slow

Figure 1. Architecture of MySQL

The components in the Mysql architecture must be well understood for the purposes of database transaction investigations. In investigating database transactions, the log files and directories of mysql server are very important to analyze.

2.2 Binary Log

Log files in the database contain important information related to transactions that occur on the database. On Mysql servers that use the InnoDB storage engine generally use two types of log files namely ib_logfiole0 and ib_logfile1 with a capacity of 5 Mega Bytes.

The Binary Log contains files that record statements for database memoification, such as delete, insert, replace, create table, drop table, grant and revoke commands. The contents of the binary log are written in SQL with the binary format[6]

2.3 Research Stages

The stages in this study consisted of several sequences, following the sequence in this study

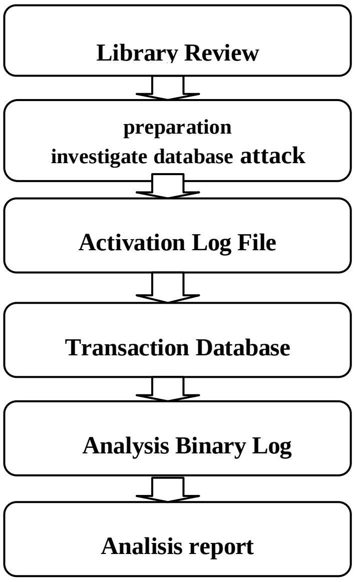


Figure 2. Reseaarch Stages

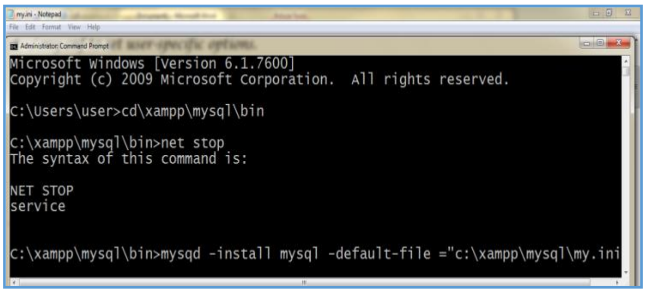
III.RESULT AND ANALYSIS

3.1 Privilleges attack investigation preparation

At this stage hardware and software preparations are made for the forensic database analysis process, at this stage also determining the use of a Database Management system (DBMS). in this study the DBMS used was MySql Server with My ISAM Storage Engine, while the observed environment was the STIE AAS Academic Information System.

3.2 Activation Log database

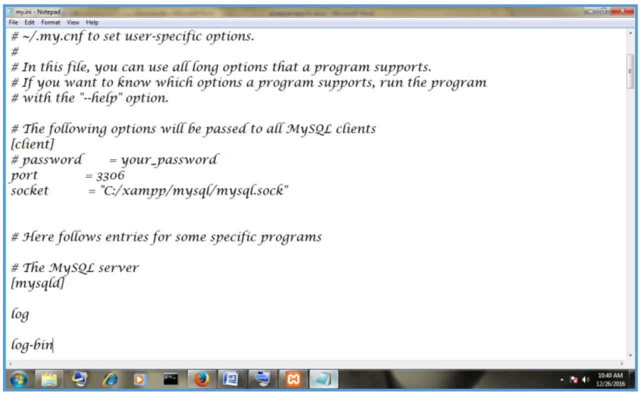
To activate the Log File in the database, first install mysqld in the MY.INI file, after MYSqld is active, add the Log function to the MY.INI file [7], like the picture below.



```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\user>cd\xampp\mysql\bin

C:\xampp\mysql\bin>net stop
The syntax of this command is:

NET STOP
service


C:\xampp\mysql\bin>mysqd -install mysql -default-file ="c:\xampp\mysql\my.ini
```

Figure 3. Activate Log File

My INI is a place to store log files, while to activate the binary log, the bin log code is inserted in My INI, as shown in the following image.



```
# ~/.my.cnf to set user-specific options.
#
# In this file, you can use all long options that a program supports.
# If you want to know which options a program supports, run the program
# with the "--help" option.

# The following options will be passed to all MySQL clients
[client]
# password       = your_password
port             = 3306
socket           = "C:/xampp/mysql/mysql.sock"


# Here follows entries for some specific programs

# The MySQL server
[mysqld]

log

log-bin
```

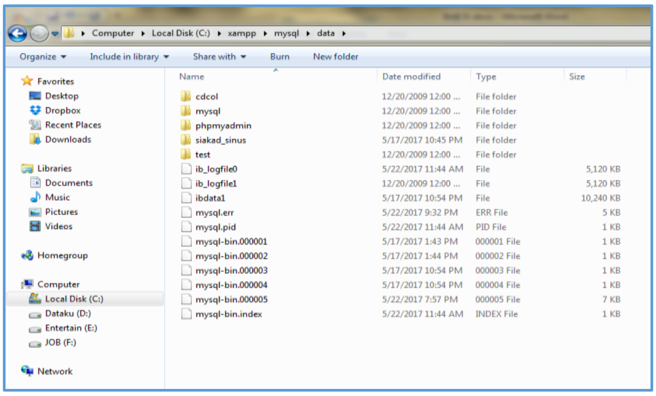
Figure 4. Insert binary log code

3.3 Database transaction simulation

At this stage the data simulation is performed to support the Binary log analysis process, the data simulation uses academic data with many users, where the user consists of students, the academic section, the financial section, the administration section, lecturers and leaders. This data simulation will simulate the transaction process of requesting data to an academic database, with various query requests from users who have different accesses, besides that there is also a simulation of attacks on access rights, where illegal access occurs by changing the contents of the database. From this simulation, data can be analyzed using the access rights of each user in making transactions to the academic database.

3.4 Analisa Binary Log

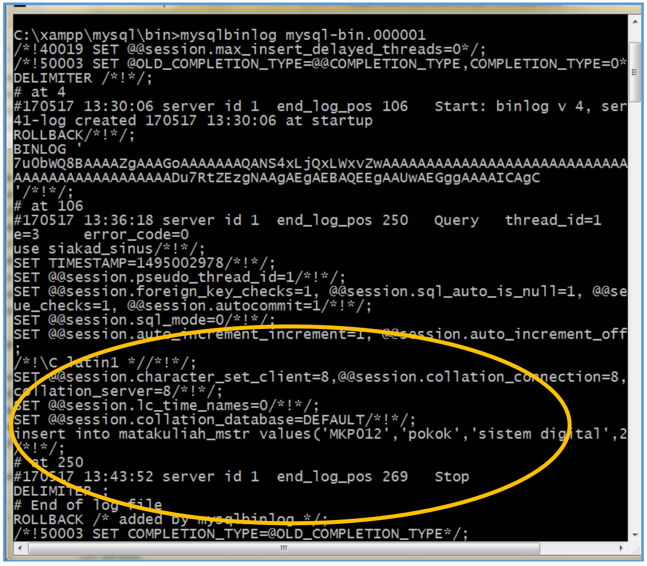
At this stage an analysis will be conducted related to the Binary Log. After activating the Binary Log, each database transaction will be recorded in the C: / xampp / mysql / data directory, while the format of the log is in binary format, with the following binary log file in the mysl / data directory



Name	Date modified	Type	Size
cdcol	12/20/2009 12:00 ...	File folder	
mysql	12/20/2009 12:00 ...	File folder	
phpmyadmin	12/20/2009 12:00 ...	File folder	
siakad_sinus	5/17/2017 10:45 PM	File folder	
test	12/20/2009 12:00 ...	File folder	
ib_logfile0	5/22/2017 11:44 AM	File	5,120 KB
ib_logfile1	12/20/2009 12:00 ...	File	5,120 KB
ibdata1	5/17/2017 10:54 PM	File	10,240 KB
mysql.err	5/22/2017 9:32 PM	ERR File	5 KB
mysql.pid	5/22/2017 11:44 AM	PID File	1 KB
mysql-bin.000001	5/17/2017 1:43 PM	000001 File	1 KB
mysql-bin.000002	5/17/2017 1:44 PM	000002 File	1 KB
mysql-bin.000003	5/17/2017 10:54 PM	000003 File	1 KB
mysql-bin.000004	5/17/2017 10:54 PM	000004 File	1 KB
mysql-bin.000005	5/22/2017 7:57 PM	000005 File	7 KB
mysql-bin.index	5/22/2017 11:44 AM	INDEX File	1 KB

Figure 5. Binary Log Directory

Binary Log is a binary log file format, to open the Binary Log using a special tool that has been provided by MySql, the tool is **MySqlBinlog**, where to open the binary log file the Mysqlbilog code is inputted with the name_file_log following how to access the Binary Log

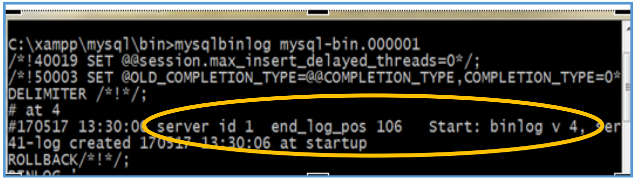


```
C:\xampp\mysql\bin>mysqlbinlog mysql-bin.000001
/*!40019 SET @@session.max_insert_delayed_threads=0*/;
/*!50003 SET @OLD_COMPLETION_TYPE=@@COMPLETION_TYPE,COMPLETION_TYPE=0*
DELIMITER /*!*/;
# at 4
#170517 13:30:06 server id 1  end_log_pos 106   Start: binlog v 4, ser
41-log created 170517 13:30:06 at startup
ROLLBACK/*!*/;
BINLOG '
7u0bWQ8BAAAAZgAAAGoAAAAAAAAQANS4xLjQxLWxvZwAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAADu7RtZEzgNAAgAEgAEBAQEEgAAUwAEGggAAAAICAgC
'/*!*/;
# at 106
#170517 13:36:18 server id 1  end_log_pos 250   Query   thread_id=1
e=3     error_code=0
use siakad_sinus/*!*/;
SET TIMESTAMP=1495002978/*!*/;
SET @@session.pseudo_thread_id=1/*!*/;
SET @@session.foreign_key_checks=1, @@session.sql_auto_is_null=1, @@se
ue_checks=1, @@session.autocommit=1/*!*/;
SET @@session.sql_mode=0/*!*/;
SET @@session.auto_increment_increment=1, @@session.auto_increment_off
;
/*!\C latin1 *//*!*/;
SET @@session.character_set_client=8,@@session.collation_connection=8,
collation_server=8/*!*/;
SET @@session.lc_time_names=0/*!*/;
SET @@session.collation_database=DEFAULT/*!*/;
insert into matakuliah_mstr values('MKP012','pokok','sistem digital',2
/*!*/;
# at 250
#170517 13:43:52 server id 1  end_log_pos 269   Stop
DELIMITER ;
# End of log file
ROLLBACK /* added by mysqlbinlog */;
/*!50003 SET COMPLETION_TYPE=@OLD_COMPLETION_TYPE*/;
```

Figure 6. Open Binary Log with Mysqllogbin tool

From the results of the Binary Log translation, the following information can be obtained:

- Start Datetimedan Stop Date Time



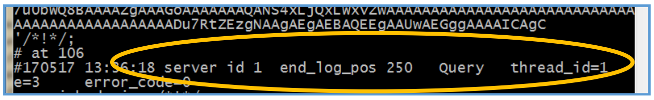
```
C:\xampp\mysql\bin>mysqlbinlog mysql-bin.000001
/*!40019 SET @@session.max_insert_delayed_threads=0*/;
/*!50003 SET @OLD_COMPLETION_TYPE=@@COMPLETION_TYPE,COMPLETION_TYPE=0*
DELIMITER /*!*/;
# at 4
#170517 13:30:06 server id 1  end_log_pos 106   Start: binlog v 4, ser
41-log created 170517 13:30:06 at startup
ROLLBACK/*!*/;
```

Figure 7. transactiondatetime

Shows the date and time when the transaction was recorded in a log, the date and time that was recorded adjusted to the date and time where the log was recorded

Server ID

- **Id server**server_id value (numbering server identity) from the server where the transaction originated.
- **End_log_pos**indicates where a subsequent transaction event begins (i.e., the final position of the current event + 1).**Thread ID**

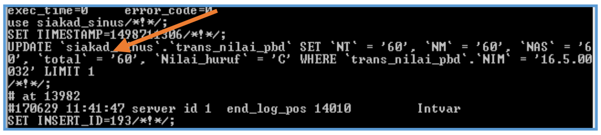


```
7u0bWQ8BAAAAZgAAAGoAAAAAAAAQANS4xLjQxLWxvZwAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAADu7RtZEzgNAAgAEgAEBAQEEgAAUwAEGggAAAAICAgC
'/*!*/;
# at 106
#170517 13:36:18 server id 1  end_log_pos 250   Query   thread_id=1
e=3     error_code=0
```

Figure 8. End Log pos on binary log

Thread_id menunjukkan thread mana yang mengeksekusi event.

- **Exec_time** istime spent running the event, on the server. On the client side the time difference is that the final execution on the client minus the initial execution time on the server. The difference becomes an indicator of how much replication is left from the master.**Error Code**
- Error_code shows the results of conducting the Transaction Event. Zero means there is no error. For a more detailed explanation of the error code in the transaction event, an error log should be investigated
- Database Name

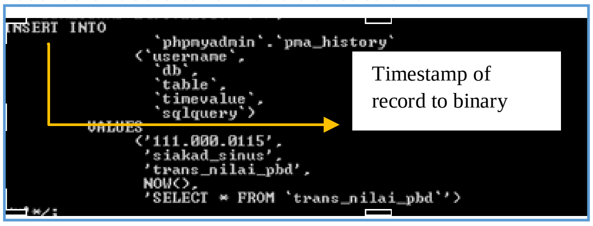


```
exec_time=0     error_code=0
use siakad_sinus/*!*/;
SET TIMESTAMP=1498711306/*!*/;
UPDATE `siakad_sinus`.`trans_nilai_pbd` SET `NT` = '60', `NM` = '60', `NAS` = '6
0', `total` = '60', `Nilai_huruf` = 'C' WHERE `trans_nilai_pbd`.`NIM` = '16.5.00
032' LIMIT 1
/*!*/;
# at 13982
#170629 11:41:47 server id 1  end_log_pos 14010         Intvar
SET INSERT_ID=193/*!*/;
```

Figure 9. Database name on binary log

It is a database that is accessed by the user and also shows a record of transactions made by the user and the value that was inputted in the transaction

- User Name
  records of users who make transactions

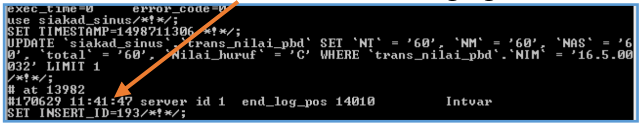


```
INSERT INTO
                `phpmyadmin`.`pma_history`
             (`username`,
                `db`,
                `table`,
                `timevalue`,
                `sqlquery`)
        VALUES
             ('111.000.0115',
              'siakad_sinus',
              'trans_nilai_pbd',
              NOW(),
              'SELECT * FROM `trans_nilai_pbd`')
/*!*/;
```

Figure 10. User name on binary log

3.5 Analysis Report

From the simulation results of database transactions can be obtained the following analysisID_server In this study, the transaction simulation uses 1 server and 2 clients, so that the ID_server recorded in the binary log is only one server, i.e. ID_server = 1, as shown in the following figure:

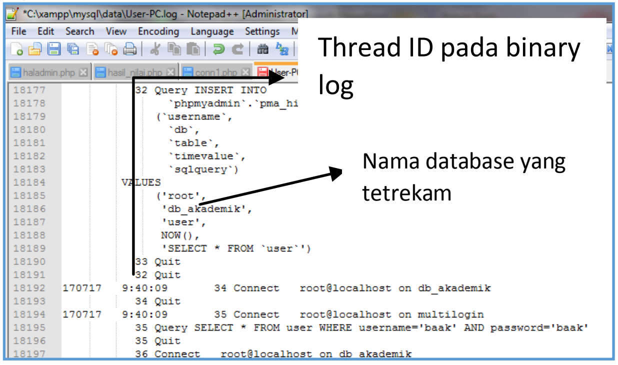


```
exec_time=0     error_code=0
use siakad_sinus/*!*/;
SET TIMESTAMP=1498711306/*!*/;
UPDATE `siakad_sinus`.`trans_nilai_pbd` SET `NT` = '60', `NM` = '60', `NAS` = '6
0', `total` = '60', `Nilai_huruf` = 'C' WHERE `trans_nilai_pbd`.`NIM` = '16.5.00
032' LIMIT 1
/*!*/;
# at 13982
#170629 11:41:47 server id 1  end_log_pos 14010         Intvar
SET INSERT_ID=193/*!*/;
```

Figure 11. Server ID on Binary Log

1. End_Log_Process
   End Log Process is a marker of the beginning and end of a log, the end of a log when the transaction is completed. end log process is recorded after ID_server

2. TimeStamp
   Each transaction to the database will be recorded server timestamp, timestamp is the time recorded by the system when, for more details can be seen in the following figure



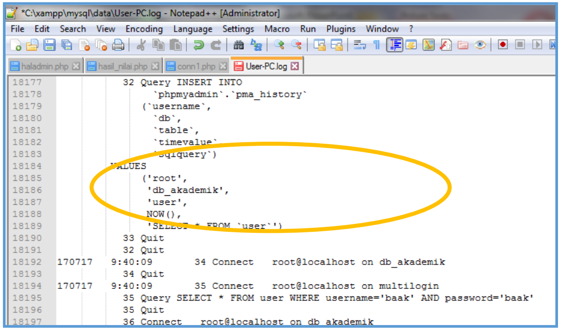
*C:\xampp\mysql\data\User-PC.log - Notepad++ [Administrator]
File Edit Search View Encoding Language Settings M
haladmin.php
hasil_nilai.php
conn1.php
User-P
Thread ID pada binary log
Nama database yang tetrekam

```
                 32 Query INSERT INTO
                    `phpmyadmin`.`pma_hi
                  (`username`,
                    `db`,
                    `table`,
                    `timevalue`,
                    `sqlquery`)
                VALUES
                  ('root',
                   'db_akademik',
                   'user',
                   NOW(),
                   'SELECT * FROM `user`')
                 33 Quit
                 32 Quit
170717   9:40:09       34 Connect   root@localhost on db_akademik
                 34 Quit
170717   9:40:09       35 Connect   root@localhost on multilogin
                 35 Query SELECT * FROM user WHERE username='baak' AND password='baak'
                 35 Quit
                 36 Connect   root@localhost on db_akademik
```

Figure 12. Database Name on Binary Log

from the picture above shows the time record on each transaction, the format of the timestamp is YYYYMMDD - HH-MM-SS. With the timestamp it can be seen when the transaction occurs, so that if an attack occurs it can be seen the time of the attack. In the timestamp there is also an exect time. Exec time is the time needed to process a transaction.

3. User Connection
   In the binary log the user connections are active and the user who is conducting a transaction is recorded, as shown in the following figure

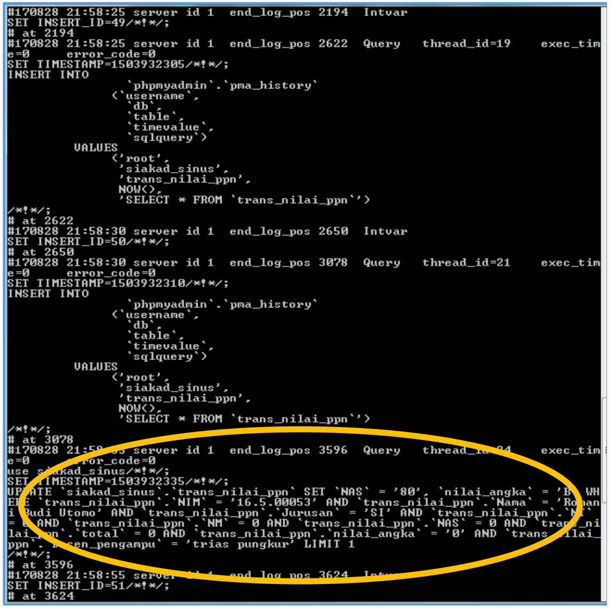

*C:\xampp\mysql\data\User-PC.log - Notepad++ [Administrator]
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
haladmin.php
hasil_nilai.php
conn1.php
User-PC.log

```
                32 Query INSERT INTO
                    `phpmyadmin`.`pma_history`
                   (`username`,
                    `db`,
                    `table`,
                    `timevalue`,
                    `sqlquery`)
               VALUES
                   ('root',
                    'db_akademik',
                    'user',
                    NOW(),
                    'SELECT * FROM `user`')
                33 Quit
                32 Quit
170717  9:40:09         34 Connect   root@localhost on db_akademik
                34 Quit
170717  9:40:09         35 Connect   root@localhost on multilogin
                35 Query SELECT * FROM user WHERE username='baak' AND password='baak'
                35 Quit
                36 Connect   root@localhost on db_akademik
```

Figure 13. User Connection

From the picture above you can see the user connection = root while username = Baak. This user record is very necessary in the investigation process, because from that note it can be seen which user is conducting the transaction, and if an attack occurs it will be seen which user made the attack.

4. Transaction Notes
   Other records recorded in the binary log are records of transactions that occur on the database server, to see the transaction records can be seen in the following figure



```
#170828 21:58:25 server id 1  end_log_pos 2194  Intvar
SET INSERT_ID=49/*!*/;
# at 2194
#170828 21:58:25 server id 1  end_log_pos 2622  Query   thread_id=19    exec_tim
e=0     error_code=0
SET TIMESTAMP=1503932305/*!*/;
INSERT INTO
                `phpmyadmin`.`pma_history`
              (`username`,
                `db`,
                `table`,
                `timevalue`,
                `sqlquery`)
         VALUES
              ('root',
               'siakad_sinus',
               'trans_nilai_ppn',
               NOW(),
               'SELECT * FROM `trans_nilai_ppn`')
/*!*/;
# at 2622
#170828 21:58:30 server id 1  end_log_pos 2650  Intvar
SET INSERT_ID=50/*!*/;
# at 2650
#170828 21:58:30 server id 1  end_log_pos 3078  Query   thread_id=21    exec_tim
e=0     error_code=0
SET TIMESTAMP=1503932310/*!*/;
INSERT INTO
                `phpmyadmin`.`pma_history`
              (`username`,
                `db`,
                `table`,
                `timevalue`,
                `sqlquery`)
         VALUES
              ('root',
               'siakad_sinus',
               'trans_nilai_ppn',
               NOW(),
               'SELECT * FROM `trans_nilai_ppn`')
/*!*/;
# at 3078
#170828 21:58:55 server id 1  end_log_pos 3596  Query   thread_id=24    exec_tim
e=0     error_code=0
use siakad_sinus/*!*/;
SET TIMESTAMP=1503932335/*!*/;
UPDATE `siakad_sinus`.`trans_nilai_ppn` SET `NAS` = '80', `nilai_angka` = 'B' WH
ERE `trans_nilai_ppn`.`NIM` = '16.5.00053' AND `trans_nilai_ppn`.`Nama` = 'Roman
i Budi Utomo' AND `trans_nilai_ppn`.`Jurusan` = 'SI' AND `trans_nilai_ppn`.`NT`
= 0 AND `trans_nilai_ppn`.`NM` = 0 AND `trans_nilai_ppn`.`NAS` = 0 AND `trans_ni
lai_ppn`.`total` = 0 AND `trans_nilai_ppn`.`nilai_angka` = '0' AND `trans_nilai_
ppn`.`dosen_pengampu` = 'trias pungkur' LIMIT 1
/*!*/;
# at 3596
#170828 21:58:55 server id 1  end_log_pos 3624  Intvar
SET INSERT_ID=51/*!*/;
# at 3624
```

Figure 14. Record Transaction on Binary Log

From the picture, it can be seen that the user named root accesses the academic database and updates the transaction value table and changes the UAS value.

5. Binary Log repot analysis
   After simulating the transaction and analyzing the binary log the records are recorded as follows

Table 1. Report Analysis

Time stamp	User	User mysql connection	Privilleges	Transactiom
18/12/19	Baak	Root	All table and transaction	Insert on tablemahasiswa _master
18/12/19	Baak	Root	All table and transaction	Insert on tablematakuliah _master
18/12/19	16.5.00053	mahasiswa	Select trans_nilai, select jadwal	Display/select trans_nilai
18/12/19	16.5.00012	Mahasiswa	Select trans_nilai, select jadwal	Select tabeltrans_nilai
28/12/19 21:58:34	18.1.000.56	Dosen	Select trans_nilai, update trans_nilai	Update tabeltrans_nilai _ppn
26/12/19	16.5.000.53	mahasiswa	Select trans_nilai, select jadwal	Select tabeltrans_nilai
26/12/19	16.5.00012	Mahasiswa	Select trans_nilai, select jadwal	Select tabeltrans_nilai
28/81219	16.5.00053	Root	All table and transaction	Update table trans_nilai

From the results of the binary log analysis obtained Analia that there is an attack of access rights, such as one example of a user connection students with access rights only see the value and schedule but on 18/12/2019 can conduct value update transactions.

IV. CONCLUSION

From these studies it can be concluded that the binary log is a binary database log containing database transaction records namely time records, user connection records and transaction records. so with binary log analysis can help the investigation process in the event of a database access rights attack. This detection technique is done by anomaly technique, namely by analyzing the user's behavior and comparing the user's access rights with the transactions made.

The research can be developed by analyzing binary logs to detect other database attacks, or developing research to analyze other types of logs provided by MySql such as Query Log or error log.

REFERENCES

[1] M. Shen, M. Chen, M. Li, and L. Liu, “Least Privilege for Database Administrators,” vol. 26, pp. 50–55, 2013.

[2] P. Fr, P. Kieseberg, S. Schrittwieser, M. Huber, and E. Weippl, “InnoDB Database Forensics : Reconstructing Data Manipulation Queries from Redo Logs,” 2010.

[3] H. K. Khanuja and D. S. Adane, "Ramework for database forensic analysis," vol. 2, no. 3, pp. 27–41, 2012.

[4] A. Rosenthal and E. Sciore, "Extending SQL ' s Grant and Revoke Operations ," pp. 1–16, 2000.

[5] Chavan Jitendra R. and H. K. Khanuja, "Database Forensic Analysis Using Log Files," *Int. J. Eng. Res. Appl.*, no. April, pp. 6–9, 2014.

[6] P. Frühwirt, P. Kieseberg, S. Schrittwieser, M. Huber, and E. Weippl, "InnoDB database forensics: Enhanced reconstruction of data manipulation queries from redo logs," *Inf. Secur. Tech. Rep.*, vol. 17, no. 4, pp. 227–238, 2013.

[7] K. Fowler and G. Gold, "SQL Server Database Forensics," *Memory*, 2007.

[8] Abdullah, Robi W., et al. "Keamanan Basis Data pada Perancangan Sistem Kepakaran Prestasi Sman Dikota Surakarta." Creative Communication and Innovative Technology Journal, vol. 12, no. 1, 2019, pp. 13-21.

[9] Muqorobin, M., Apriliyani, A., & Kusrini, K. (2019). Sistem Pendukung Keputusan Penerimaan Beasiswa dengan Metode SAW. Respati, 14(1).

[10] Muqorobin, M., Hisyam, Z., Mashuri, M., Hanafi, H., & Setiyantara, Y. (2019). Implementasi Network Intrusion Detection System (NIDS) Dalam Sistem Keamanan Open Cloud Computing. Majalah Ilmiah Bahari Jogja, 17(2), 1-9.

[11] K. Kusrini, E. T. Luthfi, M. Muqorobin and R. W. Abdullah, "Comparison of Naive Bayes and K-NN Method on Tuition Fee Payment Overdue Prediction," 2019 4th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE), Yogyakarta, Indonesia, 2019, pp. 125-130, doi: 10.1109/ICITISEE48480.2019.9003782.

[12] Muqorobin, M., Kusrini, K., Rokhmah, S., & Muslihah, I. (2020). Comparison of Naive Bayes and K-NN method on Tuition Fee Payment Overdue Prediction. International Journal of Computer and Information System (IJCIS), 1(1).

[13] Muqorobin, Muqorobin, Siti Rokhmah, Isnawati Muslihah, and Nendy Akbar Rozaq Rais. "Classification of Community Complaints Against Public Services on Twitter." International Journal of Computer and Information System (IJCIS) 1, no. 1 (2020).