Don’t Drop that Table: A Case Study in MySQL Forensics

PRESENTED BY: Jeff Hamm

8 JUNE 2014

Objectives

- Introduction
- Rebuilding a Database
  - Case Scenario
  - Identification of Artifacts
  - Rebuilding the Tables
- Conclusions
- What's Next
  - MSSQL Forensics
  - MySQL Forensics
- Review & Questions

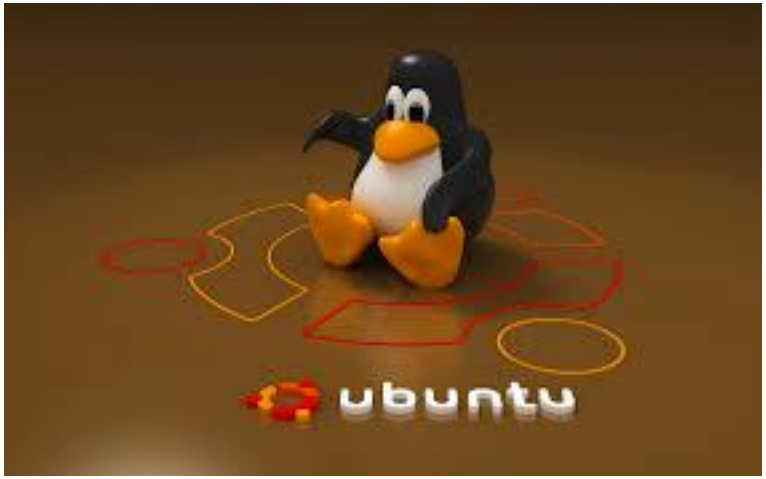
Introduction

Introduction

- Case Background
- Seized MySQL database
  - Messaging Board
- No file system artifacts
- No logs
- Needed to identify one individual
  - All activities and posts
  - Profile information
  - Contacts

Introduction

- Approach
  - Forensic Linux VM
    - MySQL Server
    - Import database tables into “pristine” server
    - BASH scripts to reference relevant data



Rebuilding a Database

Case Scenario

- PII data was targeted
- Criminals use “carder” forums to communicate
  - Sell PII
  - Purchase PII
- Third party database tables
  - No file system artifacts
  - No log data

Identification of Artifacts

- 484 files
  - FRM
  - MYD
  - MYI

- MyISAM

```
.frm   table format
.MYD   table data
.MYI   table indices
```

- “db.opt”

```
default-character-set=utf8
default-collation=utf8_general_ci
```

Identification of Artifacts

- Strings of FRM files

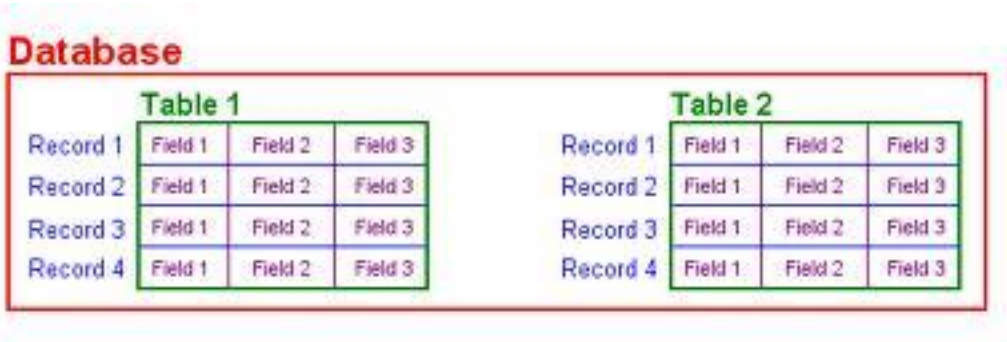
```
PRIMARY
MyISAM
)
userid
forumid
accessmask
userid
forumid
accessmask
```

Identification of Artifacts

- Imported to
  - Clean Ubuntu VM
    - MySQL Server package
    - /var/lib/mysql/evidence_database/
    - Permissions:
      - mysql/mysql

Identification of Artifacts

- Table “evidence_user” columns:
  - username
  - email
  - lastvisit
  - posts
  - ipaddress



Rebuilding the Tables

- mysql-server
  - create empty database
    - evidence_database
  - Copy database files and “db.opt”
    - /var/lib/mysql/evidence_database
  - Log into database
    - mysql -uroot -p{password}

Rebuilding the Tables

```
mysql> SHOW DATABASES;
+-----------------------+
|Database               +
+-----------------------+
|information_schema     +
|evidence_database      +
|mysql                  +
|performance_schmea     +
|test                   +
+-----------------------+
```

Rebuilding the Tables

```
mysql> USE evidence_database;

mysql> SHOW TABLES;
+-----------------------------+
|Tables in evidence database +
+-----------------------------+
|evidence_user                +
|messages                     +
|private_message              +
+-----------------------------+
```

Rebuilding the Tables

```
mysql> SHOW COLUMNS IN evidence_user;
+------------------+----------------------+------+-----+------------+----------------+
| Field            | Type                 | Null | Key | Default    | Extra          |
+------------------+----------------------+------+-----+------------+----------------+
| userid           | int(10) unsigned     | NO   | PRI | NULL       | auto_increment |
| usergroupid      | smallint(5) unsigned | NO   | MUL | 0          |                |
| membergroupids   | varchar(250)         | NO   |     |            |                |
| displaygroupid   | smallint(5) unsigned | NO   |     | 0          |                |
| username         | varchar(100)         | NO   | MUL |            |                |
| password         | varchar(32)          | NO   |     |            |                |
| passworddate     | date                 | NO   |     | 0000-00-00 |                |
| email            | varchar(100)         | NO   |     |            |                |
| ipaddress        | varchar(15)          | NO   |     |            |                |
+------------------+----------------------+------+-----+------------+----------------+
```

Rebuilding the Tables

```
mysql> SELECT userid FROM evidence_user
  WHERE username = 'evil_badguy';

+--------+
| userid |
+--------+
|  12345 |
+--------+
```

Rebuilding the Tables

```
mysql> SELECT * FROM messages
WHERE userid = '12345'
INTO OUTFILE '/tmp/evidence.txt'
FIELDS TERMINATED by '|'
ENCLOSED by '"'
LINES TERMINATED by '\n';
```

Rebuilding the Tables

```
#!/bin/bash

N=0
cat threadid.list | while read LINE ; do
        N=$((N+1))
        echo "Adding $LINE..."
        mysql -uroot ñp{password} evidence_database -e
"SELECT postid,threadid,dateline,username,pagetext FROM
evidence_post
                WHERE threadid = '$LINE'
                INTO OUTFILE '/tmp/junk/$LINE.txt'
                FIELDS TERMINATED by \"|\"
                ENCLOSED BY '\"' LINES TERMINATED BY
'\\n'"
        cat junk/$LINE.txt >> threads
done
```

Rebuilding the Tables

```
#!/bin/bash

N=0
cat pmtextid.list | while read LINE ; do
        N=$((N+1))
        echo "Line $N = $LINE"
        mysql -uroot -p{password} evidence_database -e
"SELECT * FROM evidence_pmtext
                WHERE pmtextid = '$LINE'
                INTO OUTFILE '/tmp/$LINE.txt'
                FIELDS TERMINATED by \"|\"
                ENCLOSED BY '\"' LINES TERMINATED BY
'\\n'"
        cat $LINE.txt >> private_messages
done
```

Conclusions

Conclusions

- Sufficient for a third party system
- If target of attack many other aspects that need to be explored:
  - Deleted items
  - SQL logs
  - System logs
  - Unallocated space
  - Application
    - Logs
    - File System
    - Artifacts

What’s Next

MSSQL Forensics

- Database Configuration
  - Two tier
  - Three tier
- Application Framework Overview
  - ColdFusion
  - ASP.NET
- File System Artifacts
  - Intelligence Gathering Tools
  - Webshells
  - Binaries
- Settings and Configuration Files
  - Windows Registry
  - DBMS Settings
- System and Database Application Logs
  - Windows Event Logs
  - SQL Server Setup Log
  - SQL Server Profiler Log
  - SQL Server Agent Log
  - SQL Server Error Log
  - Transaction Logs
- Web Access and Error Logs
  - IIS Logs
- Viewing a Database Externally

MySQL Forensics

- Database Configuration
  - Two tier
  - Three tier
- Application Framework Overview
  - TomCat
  - ColdFusion
  - Drupal
- File System Artifacts
  - Intelligence Gathering Tools
  - Webshells
  - Binaries
- Settings and Configuration Files
  - init.d
  - rc.d
  - DBMS Settings
- System and Database Application Logs
  - messages/system
  - auth/secure
  - Transaction Logs
- Web Access and Error Logs
  - httpd/apache access
  - httpd/apache error
- Viewing a Database Externally

Review and Questions

Review

- Introduction
- Rebuilding a Database
  - Case Scenario
  - Identification of Artifacts
  - Rebuilding the Tables
- Conclusions
- What's Next
  - MSSQL Forensics
  - MySQL Forensics
- Review & Questions

Questions