**OBJECTIVE**

- Identify MySQL security issues
- Improve MySQL installation process
- Identify data issues
- Improve data integrity and security
- Auditing options

**ABOUT AUTHOR**

Ronald Bradford

- 12 years with MySQL / 22 years with RDBMS
- Senior Consultant at MySQL Inc (06-08)
- Consultant for Oracle Corporation (96-99)
- 7 years presenting MySQL content
- All time top MySQL blogger
- Published author
- Oracle ACE Director

Available NOW for consulting

http://RonalBradford.com
MySQL Security

Default is terrible
Minimum
$ mysql_secure_installation
Recommended
Operating System
Permissions & Privileges

Hacking MySQL

For a default MySQL installation, there is no password for super user account

$ mysql -uroot
$ mysql

mysql> USE test

For a default MySQL installation, there is anonymous access. More on this soon!

$ sudo su -
$ mysql
$ cat ~/.my.cnf
[client]
user=root
password=XXX

# Other users with logins
$ find / -name ".my.cnf"

Many host providers have the superuser password in the default user config file

$ grep mysql $HOME/.*history
$ grep mysql /home/*/.*history

Shell commands leave an audit history. If password is specified on command line, it is in history

Can you view MySQL privileges to find a user with more access?

$ mysql -u<some not privileged user>
mysql> SELECT host,user,password FROM mysql.user;
+--------------+------+----------+
| host         | user | password |
|--------------+------+----------|
| localhost    | root |          |
| mactaz.local | root |          |
| 127.0.0.1    | root |          |
| ::1          | root |          |
| localhost    |      |          |
| mactaz.local |      |          |
+--------------+------+----------+

Can you view MySQL privileges to find a user with more access?
$ cd [mysql-data-dir] # e.g. /var/lib/mysql
$ mysql-bin.001536
20160154
10.0.0.1
repl
XXX
3306
60
$ mysql -urepl -pXXX -h10.0.0.1
mysql> SHOW GRANTS;

Are datadir permissions secure?
Does replication connection have global permissions?

mysql> CREATE SCHEMA IF NOT EXISTS hack;
mysql> use hack;
mysql> CREATE TABLE words(word VARCHAR(100));
mysql> LOAD DATA LOCAL INFILE '/tmp/passwords'
-> INTO TABLE words(word);

mysql> UPDATE mysql.user
-> SET password=PASSWORD('hacked');
mysql> FLUSH PRIVILEGES;

If access to start and stop mysqld process, you can reset passwords.

$ mysql --skip-grant-tables
$ mysql -uroot
mysql> UPDATE mysql.user
-> SET password=PASSWORD('hacked');
mysql> FLUSH PRIVILEGES;

If access to start and stop mysqld process, you can reset passwords.
$ cat wordpress/wp-config.php
$ cat drupal/sites/default/settings.php

# Insert your application here

Applications with cleartext passwords and user may have excessive privileges

Denial of Service (DOS)

9

Anonymous access.

Breaking MySQL

$ mysql
mysql> USE test
mysql> CREATE TABLE filldisk(c VARCHAR(1000));
mysql> INSERT INTO filldisk VALUES(REPEAT('x',1000)),(REPEAT('y',1000)),
(REPEAT('z',1000));
mysql> INSERT INTO filldisk SELECT a.c FROM filldisk a, filldisk b,
filldisk c, filldisk d, filldisk e;
Query OK, 243 rows affected (0.05 sec)
mysql> INSERT INTO filldisk SELECT a.c FROM filldisk a, filldisk b,
filldisk c, filldisk d, filldisk e;
Will it end?
mysql> SELECT POW(3,5);
+----------+
| pow(3,5) |
+----------+
|      243 |
+----------+
mysql> SELECT POW(243,5);
+-----------------+
| pow(243,5)      |
+-----------------+
| 847,288,609,443 |

# Steal CPU cycles
mysql> SELECT MD5(a.c), MD5(b.c)
-> FROM filldisk a, filldisk b ORDER BY RAND();
# Force disk I/O
```
mysql> SET SESSION tmp_table_size=1024*4; # 4K
```

mysql> SELECT ...

---

**INSTALL SECURITY**

- $ mysql_secure_installation
- Remove anonymous users
- Remove test database
- Remove non 'localhost' root users
- Set root password

---

**OS SECURITY**

- Defaults are not always secure
- Separate Data/Binary Logs/Logs/Configuration/Backups
- Individual directory permissions
- Never run mysqld as 'root' user

- Minimize security risk
- Better auditability
Installation

- Best Practice

- Single partition per MySQL Instance

- Global files as symlinks from partition

Installation

Software installed by root

- Separate MySQL permissions for directories

$ chown -R root:root /mysql
$ chown -R mysql:mysql /mysql/{data,log,binlog,etc}
$ chmod 700 /mysql/{data,binlog}
$ chmod 750 /mysql/{etc,log}

Network

MySQL listens on one TCP/IP port

- Default port is 3306
- --skip-networking disables TCP/IP

- Database does not require physical web access generally
- Firewall management

Data Security
User Privileges

**Best Practice**

```sql
CREATE USER goodguy@localhost IDENTIFIED BY 'sakila';
GRANT CREATE, SELECT, INSERT, UPDATE, DELETE ON `db`.`*` TO
goodguy@localhost;
```

**Normal Practice**

```sql
CREATE USER superman@'%';
GRANT ALL ON `*`.* TO superman@'%';
```

---

**GRANT ALL ON *.* TO user@'%'**

- `*.*` gives you access to all tables in all schemas
- `@'%'` give you access from any external location
- `ALL` gives you
  - ALTER, ALTER ROUTINE, CREATE, CREATE ROUTINE, CREATE TEMPORARY TABLES, CREATE USER, CREATE VIEW, DELETE, DROP EVENT, EXECUTE, FILE, INDEX, INSERT, LOCK TABLES, PROCESS, REFERENCES, RELOAD, REPLICATION CLIENT, REPLICATION SLAVE, SELECT, SHOW DATABASES, SHOW VIEW, SHUTDOWN, `SUPER`, TRIGGER, UPDATE, USAGE

---

**SUPER**

- Bypasses read_only
- Bypasses init_connect
- Can Disable binary logging
- Change configuration dynamically
- No reserved connection

---

$ mysql -ugoodguy -psakila db
mysql> insert into test1(id) values(1);
ERROR 1290 (HY000): The MySQL server is running with the --read-only option so it cannot execute this statement

$ mysql -usuperman db
mysql> insert into test1(id) values(1);
Query OK, 1 row affected (0.01 sec)

#my.cnf
[client]
init_connect=SET NAMES utf8

This specifies to use UTF8 for communication with client and data

$ mysql -usuperman db
mysql> SHOW SESSION VARIABLES LIKE 'character%';
+--------------------------+----------+
| Variable_name            | Value    |
+--------------------------+----------+
| character_set_client     | latin1   |
| character_set_connection | latin1   |
| character_set_database   | latin1   |
| character_set_filesystem | binary   |
| character_set_results    | latin1   |
| character_set_server     | latin1   |
| character_set_system     | utf8     |
+--------------------------+----------+

$ mysql -ugoodguy -psakila db
mysql> SHOW SESSION VARIABLES LIKE 'ch%';
+--------------------------+----------+
| Variable_name            | Value    |
+--------------------------+----------+
| character_set_client     | utf8     |
| character_set_connection | utf8     |
| character_set_database   | latin1   |
| character_set_filesystem | binary   |
| character_set_results    | utf8     |
| character_set_server     | latin1   |
| character_set_system     | utf8     |
+--------------------------+----------+

mysql> SHOW MASTER STATUS;
+-------------------+----------+--------------+------------------+
<table>
<thead>
<tr>
<th>File</th>
<th>Position</th>
<th>Binlog_Do_DB</th>
<th>Binlog_Ignore_DB</th>
</tr>
</thead>
<tbody>
<tr>
<td>binary-log.000001</td>
<td>354</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
+-------------------+----------+--------------+------------------+

mysql> INSERT INTO account VALUES (9,'New',100);
mysql> SET SQL_LOG_BIN=0;
mysql> UPDATE account SET balance=1,000 WHERE id=9;
mysql> SET SQL_LOG_BIN=1;
mysql> UPDATE account SET balance = balance - 50 WHERE id=9;
mysql> SHOW MASTER STATUS;
+-------------------+----------+--------------+------------------+
<table>
<thead>
<tr>
<th>File</th>
<th>Position</th>
<th>Binlog_Do_DB</th>
<th>Binlog_Ignore_DB</th>
</tr>
</thead>
<tbody>
<tr>
<td>binary-log.000001</td>
<td>674</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
+-------------------+----------+--------------+------------------+

1 2 3
SUPER / BINARY LOG

$ mysqlbinlog binary-log.000001 --start-position=354 --stop-position=674

# at 354
#100604 18:00:08 server id 1  end_log_pos 450 Query thread_id=1 exec_time=0
error_code=0
use mysql/*!*/;
INSERT INTO account VALUES (9,'New',100);
/*!*/;

# at 579
#100604 18:04:31 server id 1  end_log_pos 674 Query thread_id=2 exec_time=0
error_code=0
use mysql/*!*/;
SET TIMESTAMP=1275689071/*!*/;
mysql> UPDATE balance SET balance = balance - 50 WHERE id=9;
/*!*/;
DELIMITER ;
# End of log file
ROLLBACK /* added by mysqlbinlog */;

SUPER / CONNECTION

$ mysql -uroot
mysql> show global variables like 'max_connections';
+-----------------+-------+
| Variable_name   | Value |
+-----------------+-------+
| max_connections | 3     |
+-----------------+-------+
1 row in set (0.07 sec)

mysql> show global status like 'threads_connected';
+-------------------+-------+
| Variable_name     | Value |
+-------------------+-------+
| Threads_connected | 4     |
+-------------------+-------+

$ mysql -uroot
ERROR 1040 (HY000): Too many connections

SUPER / CONNECTION

$ mysql -uroot
mysql> show global variables like 'max_connections';
+-----------------+-------+
| Variable_name   | Value |
+-----------------+-------+
| max_connections | 3     |
+-----------------+-------+
1 row in set (0.07 sec)

mysql> show global status like 'threads_connected';
+-------------------+-------+
| Variable_name     | Value |
+-------------------+-------+
| Threads_connected | 4     |
+-------------------+-------+

APPLICATION USERS

• Application Viewer (Read Only Access)
  SELECT
• Application User (Read/Write Access)
  INSERT, UPDATE, DELETE, SELECT
• Application DBA (Schema Access Only)
  CREATE, DROP, CREATE ROUTINE, SELECT, INSERT, UPDATE, DELETE, ...

• Track Data Security
• Separation of responsibilities
SECURICH ACL

- User privilege via roles
- http://securich.com

Open source third party package

EffectiveMySQL.com - It's all about Performance and Scalability

NEW FEATURES

- User authentication interface (5.5)
- PAM/LDAP Authentication (5.6 Enterprise)
  http://blogs.oracle.com/MySQL/entry/new_commercial_extensions_for_mysql
  http://blogs.oracle.com/mysql_joro/entry/mysql_55_brings_in_new_ways_to_authenticate_users

EffectiveMySQL.com - It's all about Performance and Scalability

DATA INTEGRITY

Data Integrity

CREATE TABLE sample_data (  
i TINYINT UNSIGNED NOT NULL,  
c CHAR(2) NULL,  
t TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP) 
ENGINE=InnoDB DEFAULT CHARSET latin1;

INSERT INTO sample_data(i) VALUES (0), (100), (255);

Query OK, 3 rows affected (0.00 sec)

SELECT * FROM sample_data;

+-----+------+---------------------+  
| i   | c    | t                   |  
|-----+------|---------------------|  
| 0   | NULL | 2010-06-06 13:28:44 |  
| 100 | NULL | 2010-06-06 13:28:44 |  
| 255 | NULL | 2010-06-06 13:28:44 |  

3 rows in set (0.00 sec)

EffectiveMySQL.com - It's all about Performance and Scalability
**SQL_MODE**

```
SQL_MODE =
  STRICT_ALL_TABLES,
  NO_ZERO_DATE,
  NO_ZERO_IN_DATE,
  NO_ENGINE_SUBSTITUTION;
```

Minimum Recommended Configuration


**Installation**

- Current version is 5.5
- RedHat current version is
  - 5.0.77 - 28 JAN 2009
- Ubuntu version is
  - 5.1.44 - 04 FEB 2010
MySQL is open source
- More frequent updates
- MySQL enterprise monthly updates
- MySQL community ~ 6 months updates
- Release notes


MySQL 5.5 via distro
- Oracle has no yum repository
- No immediate plans for community
- Canonical (Ubuntu) has no ETA

Recommendations
- Use MySQL created rpms
- Use MySQL Tar Binary

Issues
- Dependencies
  - PHP/Perl etc
AUDITING OPTIONS

- Status variables
- Binary log
- Schema compare
- Audit plugin (5.5)
- Oracle Audit Vault

mysql> SELECT variable_name, variable_value
                   -> FROM INFORMATION_SCHEMA.GLOBAL_STATUS
                   -> WHERE variable_name LIKE 'COM_ALTER%'
                   -> OR variable_name LIKE 'COM.Drop%'
                   -> OR variable_name LIKE 'COM.Create%';

+----------------------+----------------+
| variable_name        | variable_value |
+----------------------+----------------+
| COM_ALTER_DB         | 0              |
| COM_ALTER_DB_UPGRADE | 0              |
| COM_ALTER_EVENT      | 0              |
| COM_ALTER_FUNCTION   | 0              |
| COM_ALTER_PROCEDURE  | 0              |
| COM_ALTER_SERVER     | 0              |
| COM_ALTER_TABLE      | 4              |
| COM_ALTER_TABLESPACE | 0              |

STATUS VARIABLES

$ mysqlbinlog /path/to/file > binlog.txt
$ grep -ie ALTER -e CREATE -e DROP binlog.txt
ALTER TABLE example ADD COLUMN i INT;
ALTER TABLE example ADD INDEX (i);
ALTER TABLE example DROP INDEX y;
ALTER TABLE example DROP COLUMN y;

$ mysqldump -uroot -p 
--no-data --all-databases --skip-dump-date 
| sed -e "s/AUTO_INCREMENT=[^ ] //" > schema.`date +%Y%m%d.%H%M`sql

# Compare with a previous version
$ diff schema.20111018.sql schema.20111017.sql
Audit Plugin

- Auditing interface (5.5)

Oracle Audit Vault

MySQL is not secure by default
- Responsibility on you (DBA/SA)
- Starts with OS security
- Use appropriate privileges

Replication
- SSL
- Backups
- SQL Injection
  - Sanitize your database inputs
  - http://xkcd.com/327/
REFERENCES

- Securing MySQL by Sheeri Cabral
  http://sheeri.com/content/securing-mysql-and-how-be-rock-star-dba-pr

- OurSQL podcast
  - Episode 59: Security Blankets 1
  - Episode 61: Security Blankets 2
  - Episode 55: MySQL Data encryption
  http://technocation.org/

Available NOW

Coming Soon
$E_m = p s^n$

Ronald Bradford
http://effectiveMySQL.com