

MySQL Database

Table of Contents

1. [This is a test post. please check](#)
2. [Implement strong password verify function](#)
3. [Encrypt data stored in the database using MySQL built-in functions](#)
4. [Restrict access to MySQL database](#)
5. [Use SSL connection to transfer sensitive data](#)
6. [Remove test database](#)
7. [Backup database](#)
8. [Apply latest security patches](#)
9. [Migrate to version 4.1 or 5.0 or higher](#)
10. [Harden Configuration](#)
11. [No Anonymous account](#)
12. [No Wildcards in user hostname](#)
13. [Use dedicated non-administrative account for MySQL daemon/service](#)
14. [Restrict administrative privileges](#)
15. [Setup MySQL environment in chroot](#)

1. This is a test post. please check

testing dljfsl sckljfs skdlfjlkdsdafjksldfjksdjfksdfjsa

REFERENCE

sdfsdf sdfsfsd sf sdfsdfsdfsdf sdf sdfsdfs fdssdfsdfs

2. Implement strong password verify function

MAINOPTIONS

Versions: MySQL 4.0, MySQL 4.1, MySQL 5.0, MySQL 5.1

3. Encrypt data stored in the database using MySQL built-in functions

If the data being stored in the database is sensitive then it should ideally be encrypted. MySQL provides inbuilt SQL functions to encrypt and decrypt data using the AES encryption protocol. The only problem with this method of encryption is that the password used to encrypt and decrypt the data must be hard coded into the SQL statements in the application. While this is a problem it does however keep the data safe if the database is backed up to a remote server. It also provides some degree of obfuscation in the event of an attacker gaining access to the operating system.

MAINOPTIONS

Versions: MySQL 4.0, MySQL 4.1, MySQL 5.0, MySQL 5.1, MySQL 6.0

4. Restrict access to MySQL database

Only admin users should have access to the mysql database Verify access by checking the user and db tables.

REFERENCE

Use the following two queries:

```
ýselect user, host from mysql.user where  
(Select_priv = 'Y') or (Insert_priv = 'Y') or (Update_priv = 'Y') or (Delete_priv = 'Y') or (Create_priv = 'Y') or  
(Drop_priv = 'Y');ý and ýselect user, host from mysql.db where db = 'mysql'  
and ((Select_priv = 'Y') or (Insert_priv = 'Y') or (Update_priv = 'Y') or (Delete_priv = 'Y') or (Create_priv = 'Y')  
or  
(Drop_priv = 'Y'));
```

MAINOPTIONS

Versions: MySQL 4.0, MySQL 4.1, MySQL 5.0, MySQL 5.1, MySQL 6.0

5. Use SSL connection to transfer sensitive data

To make it easier to use secure connections, MySQL is bundled with yaSSL as of MySQL 5.0.10. (MySQL and yaSSL employ the same licensing model, whereas OpenSSL uses an Apache-style license.). SSL trusted network connection is recommended when restricted Personal Identifiable Information(PII) or confidential information like (Creditcard etc.) is transferred over the untrusted networks(Internet)

REFERENCE

Auditing Guidance for section:1. SQL: show variables like have_openssl; is YES2. SQL: show variables like ssl_cert; is set (and file exists)3. SQL: show variables like ssl_key; is set (and file exists)4. SQL: show variables like ssl_ca; is set (and file exists)5. Users are forced to use SSL by setting the mysql.user.ssl_type field to ANY, X509, or SPECIFIED <http://dev.mysql.com/doc/refman/5.0/en/ssl-options.html>

MAINOPTIONS

Versions: MySQL 4.1, MySQL 5.0, MySQL 5.1

6. Remove test database

The default MySQL installation comes with a database called “test”. Databases can be viewed using the “SHOW DATABASES;” command. Databases can be dropped using the “DROP DATABASE xxx;” syntax.Removing unutilized components will eliminate an attacker’s ability to leverage them.“SHOW DATABASES like ‘test’;”

MAINOPTIONS

Versions: MySQL 4.0, MySQL 4.1, MySQL 5.0, MySQL 5.1

7. Backup database

Backup and recovery is one of the most important aspects of database administration. If a database crashed and there was no way to recover it, the devastating results to a business could include lost data, lost revenue and customer dissatisfaction. Whether companies operate a single database or multiple databases storing hundreds of gigabytes or even terabytes of data, they share one common factor - the need to back up important data and protect themselves from disaster by developing a backup and recovery plan.Backup and recovery of your MySQL database is important to protecting data from corruptions, hardware failures, and data failures. While MySQL provides many features to protect your data, nothing can replace a backup.

MAINOPTIONS

Versions: MySQL 4.0, MySQL 4.1, MySQL 5.0, MySQL 5.1

8. Apply latest security patches

Determine current version of MySQL using “mysql –h HOSTNAME –V”. Review changes in each revision greater than that running for security changes. See References for links to change history.

MAINOPTIONS

Versions: MySQL 4.0, MySQL 4.1, MySQL 5.0, MySQL 5.1

9. Migrate to version 4.1 or 5.0 or higher

Versions 4.0 and 3.23 only receive critical fixes. Utilizing a supported version of MySQL will help ensure the remediation of identified MySQL vulnerabilities.

MAOPTIONS

Versions: MySQL 4.0, MySQL 4.1

10. Harden Configuration

MAOPTIONS

Versions: MySQL 4.0, MySQL 4.1, MySQL 5.0, MySQL 5.1

11. No Anonymous account

Anonymous accounts are users with no name (""). They allow for default logins and their permissions can sometimes be used by other users. Check for anonymous users using the query "select user from mysql.user where user = '';" . They allow for default logins and their permissions can sometimes be used by other users.

REFERENCE

1.SQL: "select user from mysql.user where user = '';"2.Verify that no results are returned

MAOPTIONS

Versions: MySQL 4.0, MySQL 4.1, MySQL 5.0, MySQL 5.1

12. No Wildcards in user hostname

When possible, host parameters for users should not contain wildcards (%). This can be checked using "select user from mysql.user where host = '%';". Avoiding the use of wildcards within hostnames will ensure that only trusted principals are capable of interacting with MySQL.

REFERENCE

1.SQL: "select user from mysql.user where host = '%';"2.Verify that no results are returned

MAOPTIONS

Versions: MySQL 4.0, MySQL 4.1, MySQL 5.0, MySQL 5.1

13. Use dedicated non-administrative account for MySQL daemon/service

Utilizing a least privilege account for MySQL daemon and services to execute may reduce the impact of a MySQL-born vulnerability. A non-administrative OS account will be unable to access resources unrelated to MySQL, such as operating system configurations. MySQL should always be run as an unprivileged OS user in order to reduce the potential damage to the operating system and other processes in the event of a successful attack against MySQL.

MAINOPTIONS

Versions: MySQL 4.0, MySQL 4.1, MySQL 5.0, MySQL 5.1

14. Restrict administrative privileges

MAINOPTIONS

Versions: MySQL 4.0, MySQL 4.1, MySQL 5.0, MySQL 5.1

15. Setup MySQL environment in chroot

A chroot on UNIX operating systems is an operation that changes the apparent disk root directory for the MySQL process and its children. All of MySQL processes are re-rooted to another directory and cannot access or name files outside that directory. Running MySQL in a chroot environment reduces the impact of a MySQL-born vulnerability by making portions of the file system inaccessible to the MySQL instance. This is a good first measure to tighten system security, but it is not perfect.

REFERENCE

Configuration setting in my.cnf “chroot=” or startup parameter
“chroot=”http://articles.techrepublic.com.com/5100-22_11-5287638.html

MAINOPTIONS

Versions: MySQL 4.0, MySQL 4.1, MySQL 5.0, MySQL 5.1
