Starting to Secure MySQL (Part 1 - mysql_secure_installation)

Author	josh
Date	2017-09-21 01:40:11

Overview

Databases should always secured, regardless of the environment. However, database security is *particularly* important for web-facing applications. MySQL is a very popular database especially for web development, as a part of the LAMP (Linux-Apache-MySQL-PHP) stack. Database security can be a complex topic, but there are a few basic security measures everyone can and should use when building applications that use MySQL.

The `mysql_secure_installation` script

Fortunately, the developers of MySQL wanted to make security accessible. They've added a great security script to the package called `mysql_secure_installation`. This script allows the user to quickly attack some basic security flaws in the default MySQL installation.

What it does

This script addresses several issues with the default installation:

Sets a password for the `root` user
Disallows `root` login from remote hosts
Removes anonymous users
Removes the `test` database

Why these measures improve security

- Sets a password for the `root` user

For any system with an "administrative" user, it is paramount that not just anyone can access that account and the privileges associated with it. Giving the root user a secure password prevents an unauthorized person from modifying databases, users, or user privileges

- Disallows `root` login from remote hosts

Similar to setting a secure password for root, preventing root login from remote hosts is an extra measure that prevents an unauthorized user from having access to do *anything they want* to your database installation.

Removes anonymous users

By default, MySQL allows anyone without a user account to connect to the database engine for testing purposes. This is undesirable because even though anonymous users may not have the same privileges as root, it could allow an unauthorized user to snoop around your MySQL installation and see how databases are defined, what users are available, etc. Removing anonymous users means that only an authorized user with a password can connect to MySQL.

Removes the `test` database

Another default feature of MySQL is a test database that anyone can connect to. Since anyone can connect to the database, this gives an attacker a starting point for access to your database if there are any vulnerabilities. We don't want any databases that allow unauthorized users to connect.

How to use this feature

Since this application comes bundled with MySQL, it is fairly straightforward to run. At a terminal window, execute `mysql_secure_installtion`. You'll see several prompts for each security enhancement: `Change the root password? [Y/n] Remove anonymous users? [Y/n] Disallow root login remotely? [Y/n] Remove test database and access to it? [Y/n] Reload privilege tables now? [Y/n]` The last item ensures that the changes made by the script take effect immediately. It is best practice to answer Y (yes) to all of the prompts!

Next Steps

All of the features provided by `mysql_secure_installation` are a great starting point for securing your MySQL installations. In another blog, I'll address how to set user privileges for your database following the "Principle of Least Privilege".