the CENTER for
INTERNET SECURITY

Security Configuration Benchmark For

MySQL 4.1, 5.0, 5.1 Community Editions

Version 1.0.2
April 2009

Copyright 2001-2009, The Center for Internet Security

http://cisecurity.org
feedback@cisecurity.org

Terms of Use Agreement

Background.

CIS provides benchmarks, scoring tools, software, data, information, suggestions, ideas, and other services and materials from the CIS website or elsewhere (“**Products**”) as a public service to Internet users worldwide. Recommendations contained in the Products (“**Recommendations**”) result from a consensus-building process that involves many security experts and are generally generic in nature. The Recommendations are intended to provide helpful information to organizations attempting to evaluate or improve the security of their networks, systems and devices. Proper use of the Recommendations requires careful analysis and adaptation to specific user requirements. The Recommendations are not in any way intended to be a “quick fix” for anyone’s information security needs.

No representations, warranties and covenants.

CIS makes no representations, warranties or covenants whatsoever as to (i) the positive or negative effect of the Products or the Recommendations on the operation or the security of any particular network, computer system, network device, software, hardware, or any component of any of the foregoing or (ii) the accuracy, reliability, timeliness or completeness of any Product or Recommendation. CIS is providing the Products and the Recommendations “as is” and “as available” without representations, warranties or covenants of any kind.

User agreements.

By using the Products and/or the Recommendations, I and/or my organization (“**we**”) agree and acknowledge that:

No network, system, device, hardware, software or component can be made fully secure;
We are using the Products and the Recommendations solely at our own risk;

We are not compensating CIS to assume any liabilities associated with our use of the Products or the Recommendations, even risks that result from CIS’s negligence or failure to perform;

We have the sole responsibility to evaluate the risks and benefits of the Products and Recommendations to us and to adapt the Products and the Recommendations to our particular circumstances and requirements;

Neither CIS, nor any CIS Party (defined below) has any responsibility to make any corrections, updates, upgrades or bug fixes or to notify us if it chooses at it sole option to do so; and

Neither CIS nor any CIS Party has or will have any liability to us whatsoever (whether based in contract, tort, strict liability or otherwise) for any direct, indirect, incidental, consequential, or special damages (including without limitation loss of profits, loss of sales, loss of or damage to reputation, loss of customers, loss of software, data, information or emails, loss of privacy, loss of use of any computer or other equipment, business interruption, wasted management or other staff resources or claims of any kind against us from third parties) arising out of or in any way connected with our use of or our inability to use any of the Products or Recommendations (even if CIS has been advised of the possibility of such damages), including without limitation any liability associated with infringement of intellectual property, defects, bugs, errors, omissions, viruses, worms, backdoors, Trojan horses or other harmful items.

Grant of limited rights.

CIS hereby grants each user the following rights, but only so long as the user complies with all of the terms of these Agreed Terms of Use:

Except to the extent that we may have received additional authorization pursuant to a written agreement with CIS, each user may download, install and use each of the Products on a single computer;

Each user may print one or more copies of any Product or any component of a Product that is in a .txt, .pdf, .doc, .mcw, or .rtf format, provided that all such copies are printed in full and are kept intact, including without limitation the text of this Agreed Terms of Use in its entirety.

Retention of intellectual property rights; limitations on distribution.

The Products are protected by copyright and other intellectual property laws and by international treaties. We acknowledge and agree that we are not acquiring title to any intellectual property rights in the Products and that full title and all ownership rights to the Products will remain the exclusive property of CIS or CIS Parties. CIS reserves all rights not expressly granted to users in the preceding section entitled "Grant of limited rights." Subject to the paragraph entitled "Special Rules" (which includes a waiver, granted to some classes of CIS Members, of certain limitations in this paragraph), and except as we may have otherwise agreed in a written agreement with CIS, we agree that we will not (i) decompile, disassemble, reverse engineer, or otherwise attempt to derive the source code for any software Product that is not already in the form of source code; (ii) distribute, redistribute, encumber, sell, rent, lease, lend, sublicense, or otherwise transfer or exploit rights to any Product or any component of a Product; (iii) post any Product or any component of a Product on any website, bulletin board, ftp server, newsgroup, or other similar mechanism or device, without regard to whether such mechanism or device is internal or external, (iv) remove or alter trademark, logo, copyright or other proprietary notices, legends, symbols or labels in any Product or any component of a Product; (v) remove these Agreed Terms of Use from, or alter these Agreed Terms of Use as they appear in, any Product or any component of a Product; (vi) use any Product or any component of a Product with any derivative works based directly on a Product or any component of a Product; (vii) use any Product or any component of a Product with other products or applications that are directly and specifically dependent on such Product or any component for any part of their functionality, or (viii) represent or claim a particular level of compliance with a CIS Benchmark, scoring tool or other Product. We will not facilitate or otherwise aid other individuals or entities in any of the activities listed in this paragraph.

We hereby agree to indemnify, defend and hold CIS and all of its officers, directors, members, contributors, employees, authors, developers, agents, affiliates, licensors, information and service providers, software suppliers, hardware suppliers, and all other persons who aided CIS in the creation, development or maintenance of the Products or Recommendations ("**CIS Parties**") harmless from and against any and all liability, losses, costs and expenses (including attorneys' fees and court costs) incurred by CIS or any CIS Party in connection with any claim arising out of any violation by us of the preceding paragraph, including without limitation CIS's right, at our expense, to assume the exclusive defense and control of any matter subject to this indemnification, and in such case, we agree to cooperate with CIS in its defense of such claim. We further agree that all CIS Parties are third-party beneficiaries of our undertakings in these Agreed Terms of Use.

Special rules.

CIS has created and will from time to time create special rules for its members and for other persons and organizations with which CIS has a written contractual relationship. Those special rules will override and supersede these Agreed Terms of Use with respect to the users who are covered by the special rules. CIS hereby grants each CIS Security Consulting or Software Vendor Member and each CIS Organizational User Member, but only so long as such Member remains in good standing with CIS and complies with all of the terms of these Agreed Terms of Use, the right to distribute the Products and Recommendations within such Member's own organization, whether by manual or electronic means. Each such Member acknowledges and agrees that the foregoing grant is subject to the terms of such Member's membership arrangement with CIS and may, therefore, be modified or terminated by CIS at any time.

Choice of law; jurisdiction; venue.

We acknowledge and agree that these Agreed Terms of Use will be governed by and construed in accordance with the laws of the State of Maryland, that any action at law or in equity arising out of or relating to these Agreed Terms of Use shall be filed only in the courts located in the State of Maryland, that we hereby consent and submit to the personal jurisdiction of such courts for the purposes of litigating any such action. If any of these Agreed Terms of Use shall be determined to be unlawful, void, or for any reason unenforceable, then such terms shall be deemed severable and shall not affect the validity and enforceability of any remaining provisions. We acknowledge and agree that we have read these Agreed Terms of Use in their entirety, understand them and agree to be bound by them in all respects.

Table of Contents

Overview

This document, *Security Configuration Benchmark for MySQL 4.1, 5.0, 5.1*, provides prescriptive guidance for establishing a secure configuration posture for MySQL versions 4.1, 5.0, and 5.1 running on the Windows Server 2003 and RedHat Enterprise Linux 5 platforms. This guide was tested against MySQL 4.1, 5.0, and 5.1 as installed by MySQL RPM and MSI. To obtain the latest version of this guide, please visit http://cisecurity.org. If you have questions, comments, or have identified ways to improve this guide, please write us at feedback@cisecurity.org.

Consensus Guidance

This guide was created using a consensus review process comprised of volunteer and contract subject matter experts. Consensus participants provide perspective from a diverse set of backgrounds including consulting, software development, audit and compliance, security research, operations, government, and legal.

Intended Audience

This document is intended for system and application administrators, security specialists, auditors, help desk, and platform deployment personnel who plan to develop, deploy, assess, or secure solutions that incorporate MySQL on a Windows or Linux platform.

Acknowledgements

The following individuals have contributed greatly to the creation of this guide:

Authors
Michael Eddington, Leviathan Security Group.

Contributors and Reviewers
Blake Frantz
Steven Piliero
Neil Quiogue
Dave Shackleford

Typographic Conventions

The following typographical conventions are used throughout this guide:

Convention	Meaning
`Stylized Monospace font`	Used for blocks of code, command, and script examples. Text should be interpreted exactly as presented.
`Monospace font`	Used for inline code, commands, or examples. Text should be interpreted exactly as presented.
<italic font in brackets>	Italic texts set in angle brackets denote a variable requiring substitution for a real value.
Italic font	Used to denote the title of a book, article, or other publication.
Note	Additional information or caveats

Configuration Levels

This section defines the configuration levels that are associated with each benchmark recommendation. Configuration levels represent increasing levels of security assurance.

Level-I Benchmark settings/actions

Level-I Benchmark recommendations are intended to:

- be practical and prudent;
- provide a clear security benefit; and
- do not negatively inhibit the utility of the technology beyond acceptable means

Level-II Benchmark settings/actions

Level-II Benchmark recommendations exhibit one or more of the following characteristics:

- are intended for environments or use cases where security is paramount
- acts as defense in depth measure
- may negatively inhibit the utility or performance of the technology

Scoring Status

This section defines the scoring statuses used within this document. The scoring status indicates whether compliance with the given recommendation is discernable in an automated manner.

Scorable

The platform's compliance with the given recommendation can be determined via automated means.

Not Scorable

The platform's compliance with the given recommendation cannot be determined via automated means.

MySQL versions prior to 4.1 (3.X, 4.0)

MySQL versions prior to 3.23 are no longer supported and migration to a supported version of MySQL is highly recommended. For versions 3.23 and 4.0 only critical bugs are being addressed. Additionally, version 4.1 introduced a number of significant security improvements into MySQL. It is recommended that companies form a migration plan to move to currently supported versions of MySQL that contain the latest security improvements. As of this writing those supported versions are v4.1 (since Oct 2004) and v5.0 (since Oct 2005).

MySQL version 5.1

At the time of this writing MySQL version 5.1 is currently beta software and not recommended for production use. This document does include benchmark information for v5.1 based on version 5.1.11-beta and information available at that time.

Recommendations

1. Operating System Level Configuration

Item #	Configuration Item	Action / Recommended Parameters	Comments	Version	Windows	Unix	Level
1.1	**OS Hardening**	Harden OS using appropriate CIS benchmark		ALL	X	X	1 S
Auditing Guidance for section 1.1: N/A							
1.2	**Dedicated Machine**	Machine dedicated to running MySQL	**Rationale**: Limiting the number of services executing on the machine hosting MySQL will reduce the probability of the data within MySQL being compromised.	ALL	X	X	2 N
Auditing Guidance for section 1.2: N/A							
1.3	**Unix Run in Chroot**	Run MySQL in Jail or Chroot	**Rationale**: Running MySQL in a chroot environment may reduce the impact of a MySQL-born vulnerability by making portions of the file system inaccessible to the MySQL instance.	ALL		X	1 N
Auditing Guidance for section 1.3: Configuration setting in my.cnf “chroot=” or startup parameter ‟chroot=”							
1.4	**Dedicated Account**	Dedicated non-administrative account for MySQL daemon/service	**Rationale**: Utilizing a least privilege account for MySQL to execute as may reduce the impact of a MySQL-born	ALL	X	X	1 N

			vulnerability. A restricted account will be unable to access resources unrelated to MySQL, such as operating system configurations.				
Auditing Guidance for section 1.4: N/A							
1.5	**Restrict network access**	Restrict network access using local IP filtering	**Rationale**: Limiting the accessibility of the MySQL network socket may reduce the exposure to a MySQL-born vulnerability by preventing unauthorized hosts from communicating with the service.	ALL	X	X	2 N
Auditing Guidance for section 1.5: N/A							
1.6	**Database not on system partition**	Databases must not be located on system partitions .	For windows where the operating system is installed on `(%SYSTEMDRIVE%).` For UNIX not on the common or root (/) file system **Rationale**: Moving the database off the system partition will reduce the probability of denial of service via the exhaustion of available disk space to the operating system.	ALL	X	X	1 S
Auditing Guidance for section 1.6: 1. Get data folder name "`show variables like 'datadir';`" 2. Verify that the database is not located on the root or system partition							
1.7	**Command history**	Admin and DBA's should disable command history by setting MYSQL_HISTFILE to	**Rationale**: All commands run in the MySQL console application are saved to a history file. Disabling the MySQL	ALL		X	1 S

		/dev/null or linking .mysql_history to /dev/null	command history reduces the probability of exposing sensitive information, such as passwords.				
Auditing Guidance for section 1.7: N/A							
1.8	**MYSQL_PWD**	MySQL can read the database password from an environmental variable called `MYSQL_PWD`. Verify `MYSQL_PWD` environmental variable not used	**Rationale**: The use of the `MYSQL_PWD` environment variable implies the clear text storage of MySQL credentials. Avoiding this may increase assurance that the confidentiality of MySQL credentials is preserved.	ALL	X	X	1 N
Auditing Guidance for section 1.8: N/A							
1.9	**MySQL User**	Disable interactive login	**Rationale**: Preventing the MySQL user from logging in interactively may reduce the impact of a compromised MySQL account. **Remediation:** Unix: Set the user's shell to `/sbin/nologin`, or similar. Windows: Deny the user the "Log on locally" right	ALL	X	X	1 S
Auditing Guidance for section 1.9: N/A							
1.10	**Windows Network Service Account**	MySQL should run as a network service account [Windows 2003, Windows XP]	**Rationale:** Executing the MySQL user as the `NETWORK_SERVICE` account may reduce the impact of a MySQL-born vulnerability because this account	ALL	X		1 S

			has a restricted privilege set.				
Auditing Guidance for section 1.10: N/A							
1.11	**Windows Platform Selection**	Do not install MySQL on a domain controller	**Rationale:** Installing MySQL on a non-domain controller may reduce the impact of a MySQL-born vulnerability.	ALL	X		1 S
Auditing Guidance for section 1.11: N/A							

2. File System Permissions

<table>
<tr><th>Item #</th><th>Configuration Item</th><th>Action / Recommended Parameters</th><th>Comments</th><th>Version</th><th>Windows</th><th>Unix</th><th>Level</th></tr>
<tr><td><b>2.1</b></td><td><b>Data directory</b></td><td>Read and write by MySQL user only.</td><td>This is the location of the MySQL databases.

<b>Rationale:</b> Limiting the accessibility of these objects will protect the confidentiality, integrity, and availability of the MySQL database.</td><td>ALL</td><td>X</td><td>X</td><td>1 S</td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 2.1:</b>
1. Locating directory: SQL: “<code>show variables like 'datadir';</code>”
2. Verify permissions</td></tr>
<tr><td><b>2.2</b></td><td><b>Binaries</b></td><td>Verify and set permissions such that binaries are accessible only by database administrators and database users. Typically these are located on Unix systems in the <code>/usr/bin</code> and <code>/usr/sbin</code> folders. For Windows they are located in the installation folder.
Can be found by locating the <code>mysqld</code>, <code>mysqladmin</code>, and <code>mysql</code> executables.</td><td><b>Rationale:</b> Limiting the accessibility of these objects will protect the confidentiality, integrity, and availability of the MySQL database.</td><td>ALL</td><td>X</td><td>X</td><td>1 S</td></tr>
</table>

Auditing Guidance for section 2.2: 1. Locate base directory: SQL: "`show variables like 'basedir';`" 2. Verify permissions							
2.3	**Configuration File**	Set permissions so that configuration files are readable by database administrators and database users. Typically the MySQL configuration file on Unix systems is located in `/etc/mysql/my.cnf`. On Windows it will be located in the `%SYSTEMDIR%` or install folder.	**Rationale:** Limiting the accessibility of these objects will protect the confidentiality, integrity, and availability of the MySQL database.	ALL	X	X	1 S
Auditing Guidance for section 2.3: Locate the configuration file and assess permissions.							
2.4	**Log files**	Permission log files to be readable and writeable by MySQL user and authorized administrators only.	**Rationale:** Limiting the accessibility of these objects will protect the confidentiality, integrity, and availability of the MySQL logs.	ALL	X	X	1 S
Auditing Guidance for section 2.4: 1. Find `log_bin` entry in configuration file (contains path to logs) 2. Verify permissions							
2.5	**SSL files**	SSL files should be readable by MySQL user. No other read or write permissions.	**Rationale:** Limiting the accessibility of these objects will protect the confidentiality, integrity, and availability of the MySQL database.	ALL	X	X	1 S

Auditing Guidance for section 2.5:

1. Locate files using the following variables: `ssl_ca`, `ssl_cert`, `ssl_key`
2. Include these variables in SQL statements such as "`show variables like 'XXX';`"
3. Verify permissions

3. Logging

Configuration options can be added two ways. First is using the MySQL configuration file *my.cnf* and placing options under the proper section of “[mysqld]”. Options placed in the configuration file should not prefix with a double dash “--“. Options can also be placed on the command line by modifying the MySQL startup script. The startup script is system dependent based on your operating system.

Item #	Configuration Item	Action / Recommended Parameters	Comments	Version	Windows	Unix	Level
3.1	**Error Logging Enabled**	`--log-error[=`*`file_name`*`]`	The error log must be enabled. **Rationale**: Enabling error logging may increase the ability to detect malicious attempts against MySQL.	ALL	X	X	1 S
Auditing Guidance for section 3.1: 1. SQL: “`show variables like ‘log_error’;`” 2. Verify entry							
3.2	**Logs not on system partition**	Logs should be on a non-system partition	For windows where the operating system is installed on (`%SYSTEMDRIVE%`). For UNIX not on the common or root (/) file system. **Rationale**: Moving the MySQL logs off the system partition will reduce the probability of denial of service via the exhaustion of available disk space to the operating system.	ALL	X	X	1 S
Auditing Guidance for section 3.2: 1. Verify “`show variables like ‘log_bin’;`” is “ON”							

<table>
<tr><td colspan="8">2. Get log location from configuration/command like item “log_bin”/”log-bin”
3. Verify not located on system partition</td></tr>
<tr><td><b>3.3</b></td><td><b>Logs not on database partition</b></td><td>Logs should be on their own partition</td><td>MySQL logs should not be written to the same file system as MySQL databases

<b>Rationale</b>: Moving the MySQL logs off the database partition will reduce the probability of denial of service via the exhaustion of available disk space to MySQL.</td><td>ALL</td><td>X</td><td>X</td><td>1 S</td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 3.3:</b>
1. Verify “show variables like ‘log_bin’;” is “ON”
2. Get the log file location from configuration/command like item “log_bin”/”log-bin”
3. Verify whether the logs are located on a separate partition</td></tr>
<tr><td><b>3.4</b></td><td><b>Do not use Update log</b></td><td>Do not use --log-update</td><td><b>Rationale:</b> The update log is now deprecated and the binary log should be used instead. The update log is not transaction safe. Avoiding the --log-update option may increase the integrity and availability of MySQL log files.</td><td>ALL</td><td>X</td><td>X</td><td>1 N</td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 3.4:</b>
Verify that the “--log-update” option is not used on command line or in configuration files.</td></tr>
</table>

4. General

<table>
<tr><th>Item #</th><th>Configuration Item</th><th>Action / Recommended Parameters</th><th>Comments</th><th>Version</th><th>Windows</th><th>Unix</th><th>Level</th></tr>
<tr><td><b>4.1</b></td><td><b>Supported version of MySQL</b></td><td>Migrate to version 4.1 or 5.0</td><td><b>Rationale</b>: 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.</td><td>ALL</td><td>X</td><td>X</td><td>2 S</td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 4.1:</b>
SQL: “<code>show variables like 'version';</code>”</td></tr>
<tr><td><b>4.2</b></td><td><b>Latest security patches</b></td><td>Verify latest security patches.</td><td><b>Rationale</b>: Maintaining currency with MySQL patches will help protect the confidentiality, integrity, and availability of the data housed in MySQL.</td><td>ALL</td><td>X</td><td>X</td><td>2 N</td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 4.2:</b>
<code>mysql -h HOSTNAME -V</code></td></tr>
<tr><td><b>4.3</b></td><td><b>Upgrade fix privilege tables</b></td><td>When upgrading always fix the privilege tables</td><td>MySQL has a script for checking and upgrading the tables. <code>mysql_upgrade</code> for v5.0+, <code>mysql_fix_privilege_tables</code> otherwise.

<b>Rationale</b>: Some revisions of MySQL have added privileges that did not exist</td><td>ALL</td><td>X</td><td>X</td><td>1 S</td></tr>
</table>

			in earlier versions. Ensuring that privileges are appropriately applied to MySQL objects will help ensure the confidentiality, integrity, and availability of the data housed in MySQL.				
Auditing Guidance for section 4.3: Tables that will need to be checked: `mysql.user`, `mysql.host`, `mysql.db`, `mysql.tables_priv`, `mysql.columns_priv`, `mysql.func`, and `mysql.procs_priv`.							
4.4	**Remove test database**	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. **Rationale**: Removing unutilized components will eliminate an attacker's ability to leverage them.	ALL	X	X	1 S
Auditing Guidance for section 4.4: `"SHOW DATABASES like 'test';"`							
4.5	**Change admin account name**	Change admin account from default ("root") to something else	Verify root user no longer exists using following query: `"select user from mysql.user where user = 'root';"` **Rationale:** Disabling the root user's ability to interact with MySQL will limit the use of this sensitive account for	ALL	X	X	1 S

<table>
<tr><td></td><td></td><td></td><td>non-operating system administrative purposes. Additionally, avoiding the ‘root’ account for MySQL interactions will reduce the possibility of compromising the system via a MySQL client-born vulnerability.</td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 4.5:</b>
1. SQL: “select user from mysql.user where user = ‘root’;”
2. Verify no results were returned</td></tr>
<tr><td><b>4.6</b></td><td><b>Complex Passwords</b></td><td>Minimum 8 characters in length with characters from at least three of the following categories: uppercase, lowercase, numeric, non-alphanumeric</td><td>A policy should be in place to require complex passwords on all database accounts.
<b>Rationale:</b> Complex passwords help mitigate dictionary, brute forcing, and other password attacks.</td><td>ALL</td><td>X</td><td>X</td><td>1 N</td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 4.6:</b> N/A</td></tr>
<tr><td><b>4.7</b></td><td><b>Verify Secure Password Hashes</b></td><td>All password hashes should be 41 bytes or longer</td><td>Use “select User, Password from mysql.user where length(password) < 41;” query to verify.
<b>Rationale:</b> Starting in v4.1 a stronger password hash is used that result in hashes 41 bytes long. Older password hashes were only 16 bytes. Utilizing the stronger hashing algorithm will ensure the confidentiality, integrity, and availability of the data housed within</td><td>ALL</td><td>X</td><td>X</td><td>1 S</td></tr>
</table>

			MySQL by protecting the confidentiality of authentication credentials.				
Auditing Guidance for section 4.7: 1. SQL: “select User, Password from mysql.user where length(password) < 41;” 2. Validate that no results are returned							
4.8	**Single use accounts**	Each database user should be used for single purpose/person	Database user accounts should not be reused for multiple applications or users. **Rationale:** Utilizing unique database accounts across applications will reduce the impact of a compromised MySQL account.	ALL	X	X	1 N
Auditing Guidance for section 4.8: N/A							
4.9	**Wildcards in user hostname**	Verify if users have wildcard (‘%’) in hostname	When possible, host parameters for users should not contain wildcards (‘%’). This can be checked using “select user from mysql.user where host = '%';”. **Rationale:** Avoiding the use of wildcards within hostnames will ensure that only trusted principals are capable of interacting with MySQL.	ALL	X	X	2 S
Auditing Guidance for section 4.9: 1. SQL: “select user from mysql.user where host = ‘%’;”							

<table>
<tr><td colspan="8">2. Verify that no results are returned</td></tr>
<tr><td><b>4.10</b></td><td><b>No blank passwords</b></td><td>Verify no blank passwords</td><td>Blank passwords allow a user to login with out using a password. Use the “select User, Password from mysql.user where length(password) = 0 or password is null;” query to verify.

<b>Rationale:</b> Blank passwords negate the benefits provided by authentication mechanisms.</td><td>ALL</td><td>X</td><td>X</td><td>1 S</td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 4.10:</b>
1. SQL: “select user, password from mysql.user where length(password) = 0 or password is null;”
2. Verify that no results are returned</td></tr>
<tr><td><b>4.11</b></td><td><b>Anonymous account</b></td><td>Verify and remove anonymous accounts</td><td>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 = ‘’;”.

<b>Rationale:</b> Anonymous accounts are users with no name (‘’). They allow for default logins and there permissions can sometimes be used by other users.</td><td>ALL</td><td>X</td><td>X</td><td>1 S</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>Avoiding the use of anonymous accounts will ensure that only trusted principals are capable of interacting with MySQL.</td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 4.11:</b>
1. SQL: “<code>select user from mysql.user where user = ‘’;</code>”
2. Verify that no results are returned</td></tr>
</table>

5. MySQL Permissions

Item #	Configuration Item	Action / Recommended Parameters	Comments	Version	Windows	Unix	Level

5.1	**Access to mysql database**	Only admin users should have access to the `mysql` database	Verify access by checking the `user` and `db` tables. Use the following two queries: “`select 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');`” and
			“`select 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'));`
			
			Rationale: Limiting the accessibility of the ‘`mysql`’ database will protect the confidentiality, integrity, and availability of the data housed within MySQL.
				ALL	X	X	1 N

Auditing Guidance for section 5.1: SQL: “`select 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');`” and “`select 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'));`”							
5.2	**`FILE` privilege**	Do not grant to non Admin users	Verify using following query: “`select user, host from mysql.user where File_priv = 'Y';`” **Rationale:** The `FILE` privilege allows `mysql` users to write files to disk. This may be leveraged by an attacker to further compromise MySQL.	ALL	X	X	1 N
Auditing Guidance for section 5.2: 1. SQL: “`select user, host from mysql.user where File_priv = 'Y';`” 2. Ensure proper access controls are in place, and that the principle of least privilege is enforced							
5.3	**`PROCESS` privilege**	Do not grant to non Admin users	Verify using following query: “`select user, host from mysql.user where Process_priv = 'Y';`” **Rationale:** The `PROCESS` privilege allows principals to view currently executing MySQL statements, including statements used to manage passwords. This may be leveraged by an attacker to compromise MySQL.	ALL	X	X	1 N

Auditing Guidance for section 5.3: 1. SQL: “select user, host from mysql.user where Process_priv = 'Y';” 2. Ensure proper access controls are in place, and that the principle of least privilege is enforced							
5.4	**SUPER privilege**	Do not grant to non Admin users	Verify using following query: “select user, host from mysql.user where Super_priv = 'Y';” **Rationale:** The SUPER privilege allows principals to view and terminate currently executing MySQL statements, including statements used to manage passwords. This privilege also provides the ability to configure MySQL. This may be leveraged by an attacker to compromise MySQL.	ALL	X	X	1 N
Auditing Guidance for section 5.4: 1. SQL: “select user, host from mysql.user where Super_priv = 'Y';” 2. Ensure proper access controls are in place, and that the principle of least privilege is enforced							
5.5	**SHUTDOWN privilege**	Do not grant to non Admin users	Verify using following query: “select user, host from mysql.user where Shutdown_priv = 'Y';” **Rationale:** The SHUTDOWN privilege allows principals to shutdown MySQL. This may be leveraged by an attacker to negatively impact the availability of MySQL.	ALL	X	X	1 N

Auditing Guidance for section 5.5: 1. SQL: “`select user, host from mysql.user where Shutdown_priv = 'Y';`” 2. Ensure proper access controls are in place, and that the principle of least privilege is enforced							
5.6	**`CREATE USER` privilege**	Do not grant to non Admin users	Verify using following query: “`select user, host from mysql.user where Create_user_priv = 'Y';`” **Rationale:** The `CREATE USER` privilege allows principals to create MySQL users. This may be leveraged by an attacker to compromise MySQL.	ALL	X	X	1 N
Auditing Guidance for section 5.6: 1. SQL: “`select user, host from mysql.user where Create_user_priv = 'Y';`” 2. Ensure proper access controls are in place, and that the principle of least privilege is enforced							
5.7	**`RELOAD` privilege**	Do not grant to non Admin users	Allows reloading of grant tables (flush-privileges is a synonym). Verify using following query: “`select user, host from mysql.user where Reload_priv = 'Y';`” **Rationale:** The `RELOAD` privilege allows a principal to reload privileges/grants. Non administrative are not capable of modifying grants/privileges and should therefore have no need for this privilege.	ALL	X	X	1 N
Auditing Guidance for section 5.7: 1. SQL: “`select user, host from mysql.user where Reload_user_priv = 'Y';`”							

2. Ensure proper access controls are in place, and that the principle of least privilege is enforced							
5.8	**Global GRANT privilege**	Do not grant to non Admin users	Allows changing of permissions. Verify using following query: ‟select user, host from mysql.user where Grant_priv = 'Y';” **Rationale:** The GRANT privilege allows a principal to grant other principals additional privileges. This may be used by an attacker to compromise MySQL.	ALL	X	X	1 N
Auditing Guidance for section 5.8: 1. SQL: ‟select user, host from mysql.user where Create_user_priv = 'Y';” 2. Ensure proper access controls are in place, and that the principle of least privilege is enforced							

6. MySQL Configuration Options

Configuration options can be added two ways. First is using the MySQL configuration file *my.cnf* and placing options under the proper section of “[mysqld]”. Options placed in the configuration file should not prefix with a double dash “--“. Options can also be placed on the command line by modifying the MySQL startup script. The startup script is system dependent based on your operating system.

<table>
<tr><th>Item #</th><th>Configuration Item</th><th>Action / Recommended Parameters</th><th>Comments</th><th>Version</th><th>Windows</th><th>Unix</th><th>Level</th></tr>
<tr><td><b>6.1</b></td><td><b>Suspicious UDFs</b></td><td>Avoid using the <code>--allow-suspicious-udfs</code> parameter</td><td>This option prevents attaching arbitrary shared library functions as user-defined functions by checking for at least one corresponding method named _init, _deinit, _reset, _clear, or _add.
<b>Rationale:</b> This will help prevent an attacker from executing arbitrary code.</td><td>ALL</td><td>X</td><td>X</td><td>1 S</td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 6.1:</b>
Verify that <code>--allow-suspicious-udfs</code> is not used as a startup parameter</td></tr>
<tr><td><b>6.2</b></td><td><b>Disable Load data local</b></td><td><code>--local-infile=0</code></td><td>Local loading allows loading files from the <i>client</i> machine. This feature is sometimes used to perform data loading from remote machines.
<b>Rationale:</b> In a web environment where clients are connecting from a web server an attacker could use a SQL Injection vulnerability to read files from the web server.</td><td>ALL</td><td>X</td><td>X</td><td>2 S</td></tr>
</table>

<table>
<tr><td colspan="8"><b>Auditing Guidance for section 6.2:</b>
1. SQL: <code>“show variables like ‘local_infile’;”</code>
2. Verify value is “OFF”</td></tr>
<tr><td><b>6.3</b></td><td><b>Old password hashing</b></td><td>Must not use:
<code>--old-passwords</code></td><td>This configuration parameter forces use of older insecure password hashing method.
<b>Rationale:</b> Utilizing stronger hashing algorithms will help protect the confidentiality of authentication credentials.</td><td>ALL</td><td>X</td><td>X</td><td>1 S</td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 6.3:</b>
1. SQL: <code>“show variables like ‘old_passwords’;”</code>
2. Verify value is “OFF”</td></tr>
<tr><td><b>6.4</b></td><td><b>Safe show database</b></td><td><code>--safe-show-database</code></td><td>This option causes the <code>SHOW DATABASES</code> statement to display names of only those databases for which the user has some kind of privilege (default in 5.1)
<b>Rationale:</b> This reinforces the least privilege model by limiting a user’s knowledge of other existing databases.</td><td>4.1, 5.0</td><td>X</td><td>X</td><td>1 S</td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 6.4:</b>
1. SQL: <code>“show variables like ‘safe_show_database’;”</code>
2. Verify value is “ON”</td></tr>
<tr><td><b>6.5</b></td><td><b>Secure auth</b></td><td><code>--secure-auth</code></td><td>Disallow authentication for accounts that have old (pre-4.1) passwords</td><td>ALL</td><td>X</td><td>X</td><td>2 S</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td><b>Rationale:</b> This is an added measure to prevent potentially compromised credentials from being used for authentication.</td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 6.5:</b>
1. SQL: “show variables like ‘secure_auth’;”
2. Verify value is “ON”</td></tr>
<tr><td><b>6.6</b></td><td><b>Grant tables</b></td><td>Must not use: --skip-grant-tables</td><td><b>Rationale:</b> This option causes the server not to use the privilege system at all. This gives anyone with access to the server <i>unrestricted access</i> to <i>all databases</i>.</td><td>ALL</td><td>X</td><td>X</td><td>1 S</td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 6.6:</b>
1. SQL: “show variables like ‘skip_grant_tables’;”
2. Verify value is “OFF” or variable does not exist.</td></tr>
<tr><td><b>6.7</b></td><td><b>Skip merge</b></td><td>--skip-merge</td><td><b>Rationale:</b> Prevent continued table access using a merge table even after permission is revoked. This option will disable use of MERGE tables.</td><td>5.1</td><td>X</td><td>X</td><td>2 S</td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 6.7:</b>
1. SQL: “show variables like ‘have_merge_engine’;”
2. Verify value is “DISABLED”</td></tr>
<tr><td><b>6.8</b></td><td><b>Skip networking</b></td><td>Use --skip-networking startup option</td><td>Do not allow TCP/IP connections; do not bind to a port. Use if no remote access is needed.</td><td>ALL</td><td>X</td><td>X</td><td>2 S</td></tr>
</table>

			Rationale: If remote access is not required, preventing MySQL from binding to a network socket may reduce the exposure of a MySQL-born vulnerability.				
Auditing Guidance for section 6.8: 1. SQL: `“show variables like ‘skip_networking’;”` 2. Verify value is “ON”							
6.9	**Safe user create**	`NO_AUTO_CREATE_USER` or `--safe-user-create`	Prevent `GRANT` from creating a new user unless a non-empty password is also specified **Rationale:** Blank passwords negate the benefits provided by authentication mechanisms.	ALL	X	X	1 S
Auditing Guidance for section 6.9: 1. SQL: `“select @@global.sql_mode;”` must contain `NO_AUTO_CREATE_USER` 2. SQL: `“select @@session.sql_mode;”` must contain `NO_AUTO_CREATE_USER`							
6.10	**Skip Symbolic Links**	`--skip-symbolic-links`	**Rationale:** Prevents sym links being used for data base files. This is especially important when MySQL is executing as root as arbitrary files may be overwritten.	ALL	X	X	2 S
Auditing Guidance for section 6.10: 1. SQL: `“show variables like ‘have_symlink’;”` 2. Verify value is “DISABLED”							

<table>
<tr><td><b>6.11</b></td><td><b>Client password</b></td><td>Do not use <code>password=</code> configuration option</td><td>The <code>[Client]</code> section of the MySQL configuration file allows setting a password to be used. Verify this option is not used.

<b>Rationale:</b> The use of this parameter may negatively impact the confidentiality of the user’s password.</td><td>ALL</td><td>X</td><td>X</td><td>2 S</td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 6.11:</b>
Examine the <code>[Client]</code> section of the MySQL configuration file and ensure this option is not employed.</td></tr>
</table>

7. SSL Configuration

Configuration options can be added two ways. First is using the MySQL configuration file *my.cnf* and placing options under the proper section of “[mysqld]”. Options placed in the configuration file should not prefix with a double dash “--“. Options can also be placed on the command line by modifying the MySQL startup script. The startup script is system dependent based on your operating system.

<table>
<tr><th>Item #</th><th>Configuration Item</th><th>Action / Recommended Parameters</th><th>Comments</th><th>Version</th><th>Windows</th><th>Unix</th><th>Level</th></tr>
<tr><td><b>7.1</b></td><td><b>Client Verify Server Cert</b></td><td><code>--ssl-verify-server-cert</code></td><td>Causes the server’s common name (CN) to be verified against the server’s hostname.

<b>Rationale:</b> Verifying the server’s certificate will help protect against man in the middle attacks.</td><td>5.1</td><td>X</td><td>X</td><td>1 S</td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 7.1:</b>
In the [client] portion of the MySQL configuration file check for the existence of <code>ssl_verify_server_cert</code></td></tr>
<tr><td><b>7.2</b></td><td><b>SSL Connection</b></td><td>Must use SSL over untrusted networks (internet) or when restricted PII is transferred</td><td><b>Rationale:</b> SSL will protect the confidentiality and integrity of sensitive information as it traverses untrusted networks.</td><td>ALL</td><td>X</td><td>X</td><td>2 S</td></tr>
<tr><td colspan="8"><b>Auditing Guidance for section 7.2:</b>
1. SQL: <code>“show variables like ‘have_openssl’;”</code> is “YES”
2. SQL: <code>“show variables like ‘ssl_cert’;”</code> is set (and file exists)
3. SQL: <code>“show variables like ‘ssl_key’;”</code> is set (and file exists)
4. SQL: <code>“show variables like ‘ssl_ca’;”</code> is set (and file exists)
5. Users are forced to use SSL by setting the <code>mysql.user.ssl_type</code> field to <code>ANY</code>, <code>X509</code>, or <code>SPECIFIED</code></td></tr>
</table>

Note: `have_openssl` is an alias for `have_ssl` as of MySQL 5.0.38.							
7.3	**Unique Key/Cert**	Do not use a default or example certificate. Generate a key specifically for MySQL	**Rationale:** Use of default certificates can allow an attacker to impersonate the MySQL server.	ALL	X	X	1 N
Auditing Guidance for section 7.3: N/A							

8. Backup and Disaster Recovery

Item #	Configuration Item	Action / Recommended Parameters	Comments	Version	Windows	Unix	Level
8.1	**Backup of databases**	Regularly occurring backup	**Rationale:** Backing up MySQL databases, including ‘`mysql`’, will help ensure the availability of data in the event of an incident.	ALL	X	X	1 N
Auditing Guidance for section 8.1: N/A							
8.2	**Verify backups**	Verify backups are good	**Rationale:** Verifying that backups are occurring appropriately will help ensure the availability of data in the event of an incident.	ALL	X	X	1 N
Auditing Guidance for section 8.2: N/A							
8.3	**Replication slave backups**	Verify `master.info`, `relay-log.info`, and `SQL_LOAD-*` files.	**Rationale:** Additional files must be backed up for replication slaves. `SQL_LOAD-*` files are in the `slave-load-tmpdir` (defaults to `tmpdir`). Use “`show variables;`”	ALL	X	X	1 N
Auditing Guidance for section 8.3: N/A							

Appendix A: References

Resource	Location
MySQL v4.1 General Security Issues	http://dev.mysql.com/doc/refman/4.1/en/security.html
MySQL v5.0 General Security Issues	http://dev.mysql.com/doc/refman/5.0/en/security.html
MySQL v0.1 General Security Issues	http://dev.mysql.com/doc/refman/5.1/en/security.html
MySQL v4.1 Change History	http://dev.mysql.com/doc/refman/4.1/en/news.html
MySQL v5.0 Change History	http://dev.mysql.com/doc/refman/5.0/en/news.html
MySQL v5.1 Change History	http://dev.mysql.com/doc/refman/5.1/en/news.html
Securing MySQL: step-by-step	http://www.securityfocus.com/infocus/1726
Secure MySQL Database Design	http://www.securityfocus.com/infocus/1667
Chrooting MySQL on Debian	http://blog.blackdown.de/2005/03/04/chrooting-mysql-on-debian/

Appendix B: Change History

Date	Version	Changes for this version
August 3rd, 2007	1.0.0	Initial Public Release
January 13th, 2009	1.0.1	Fixed 4.10 to compare null with "is" vice "=".
April 10th, 2009	1.0.2	• Fixes broken link in 4.2. Moved audit steps into audit section. • Merged 1.9 and 1.10 as both were recommendations to disable logon rights for mysql user. • Fixed spelling errors in 4.11, 5.1, and 6.7 • Fixed erroneous audit guidance in 5.7 • Added note to 7.2 indicating have_openssl is an alias for have_ssl