



the CENTER for
INTERNET SECURITY

Center for Internet Security Benchmark for MySQL Versions 4.1, 5.0, and 5.1 Community Editions

**Version 1.0
August 3, 2007**

Document Editor: Mike Eddington, Leviathan Security Group

Copyright 2006-2007, The Center for Internet Security
<http://cisecurity.org>

cis-feedback@cisecurity.org

Table of Contents

Agreed Terms of Use	3
Introduction.....	7
MySQL versions prior to 4.1 (3.X, 4.0).....	7
MySQL version 5.1.....	7
1. Operating System Level Configuration	8
2. File System Permissions	12
3. Logging	15
4. General.....	17
5. MySQL Permissions	23
6. MySQL Configuration Options	28
7. SSL Configuration	33
8. Backup and Disaster Recovery	35
References.....	36

Agreed Terms of Use

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:

1. No network, system, device, hardware, software or component can be made fully secure;
2. We are using the Products and the Recommendations solely at our own risk;
3. 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;
4. 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;

5. 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 its 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:

1. 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;
2. 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.

The distribution of the NSA Security Recommendations is subject to the terms of the NSA Legal Notice and the terms contained in the NSA Security Recommendations themselves (<http://nsa2.www.conxion.com/cisco/notice.htm>).

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.

Introduction

This document is derived from research conducted utilizing MySQL on various Windows and Unix platforms. This document provides the necessary settings and procedures for the secure setup, configuration, and operation of a MySQL database system. With the use of the settings and procedures in this document, a MySQL database can be configured to conform to general industry “best practices” regarding secure configuration. Although these settings will improve the security of a MySQL “out of the box” installation, they are by no means a guarantee of overall database and information security.

The Level column indicates the following:

Level 1 settings are generally considered “safe” to apply to most systems. The use of these configuration recommendations is not likely to have a negative impact on performance or functionality unless otherwise notes in the comments.

Level 2 settings provide a higher level of security, but will result in a negative impact to performance, functionality, or cost.

Included in the Level column, as well, are settings which pertain to automated configuration and scoring tools. These settings indicate to what degree a scoring tool can assess a control, if any. They include:

S – To be scored

N – Not to be scored

R – Reportable, but not to be scored

It is extremely important to conduct testing of security configurations on non-production systems prior to implementing them on production systems.

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.

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 vulnerability. A restricted account will be unable to access resources unrelated	ALL	X	X	1 N

			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 R
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 /dev/null or linking .mysql_history to /dev/null	Rationale: All commands run in the MySQL console application are saved to a history file. Disabling the MySQL command history reduces the probability of exposing sensitive	ALL		X	1 S

			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 R
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.	ALL		X	1 S
Auditing Guidance for section 1.9: N/A							
1.10	MySQL User (Windows)	Disable interactive login [Windows 2000]	Deny the account the “Log on locally” right. Rationale: Preventing the MySQL user from logging in interactively may reduce the impact of a compromised MySQL account.	ALL	X		1 S
Auditing Guidance for section 1.10: N/A							
1.11	Windows Network Service Account	MySQL should run as a network service account	Rationale: Executing the MySQL user as the NETWORK_SERVICE account	ALL	X		1 S

		[Windows 2003, Windows XP]	may reduce the impact of a MySQL-born vulnerability because this account has a restricted privilege set.				
Auditing Guidance for section 1.11: N/A							
1.12	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 1.12: N/A							

2. File System Permissions

Item #	Configuration Item	Action / Recommended Parameters	Comments	Version	Windows	Unix	Level
2.1	Data directory	Read and write by MySQL user only.	<p>This is the location of the MySQL databases.</p> <p>Rationale: Limiting the accessibility of these objects will protect the confidentiality, integrity, and availability of the MySQL database.</p>	ALL	X	X	1 S
<p>Auditing Guidance for section 2.1:</p> <ol style="list-style-type: none"> 1. Locating directory: SQL: "show variables like 'datadir';" 2. Verify permissions 							
2.2	Binaries	<p>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 /usr/bin and /usr/sbin folders. For Windows they are located in the installation folder.</p> <p>Can be found by locating the mysqld, mysqladmin, and mysql executables.</p>	<p>Rationale: Limiting the accessibility of these objects will protect the confidentiality, integrity, and availability of the MySQL database.</p>	ALL	X	X	1 S

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[= <i>file_name</i>]	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”							

2. Get log location from configuration/command like item "log_bin"/"log-bin" 3. Verify not located on system partition							
3.3	Logs not on database partition	Logs should be on their own partition	MySQL logs should not be written to the same file system as MySQL databases Rationale: 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.	ALL	X	X	1 S
Auditing Guidance for section 3.3: 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							
3.4	Do not use Update log	Do not use --log-update	Rationale: 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.	ALL	X	X	1 N
Auditing Guidance for section 3.4: Verify that the "--log-update" option is not used on command line or in configuration files.							

4. General

Item #	Configuration Item	Action / Recommended Parameters	Comments	Version	Windows	Unix	Level
4.1	Supported version of MySQL	Migrate to version 4.1 or 5.0	Rationale: 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.	ALL	X	X	2 S
Auditing Guidance for section 4.1: SQL: "show variables like 'version';"							
4.2	Latest security patches	Verify 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. Rationale: Maintaining currency with MySQL patches will help protect the confidentiality, integrity, and availability of the data housed in MySQL.	ALL	X	X	2 N
Auditing Guidance for section 4.2: N/A							
4.3	Upgrade fix privilege tables	When upgrading always fix the privilege tables	MySQL has a script for checking and upgrading the tables.	ALL	X	X	1 S

			<p>mysql_upgrade for v5.0+, mysql_fix_privilege_tables otherwise.</p> <p>Rationale: Some revisions of MySQL have added privileges that did not exist 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.</p>				
<p>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.</p>							
4.4	Remove test database	Remove test database	<p>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.</p> <p>Rationale: Removing unutilized components will eliminate an attacker's ability to leverage them.</p>	ALL	X	X	1 S
<p>Auditing Guidance for section 4.4: "SHOW DATABASES like 'test';"</p>							
4.5	Change admin account name	Change admin account from default ("root") to something	Verify root user no longer exists using following query: "select user	ALL	X	X	1 S

		else	<pre>from mysql.user where user = 'root';"</pre> <p>Rationale: Disabling the root user's ability to interact with MySQL will limit the use of this sensitive account for 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.</p>				
<p>Auditing Guidance for section 4.5:</p> <ol style="list-style-type: none"> SQL: "select user from mysql.user where user = 'root';" Verify no results were returned 							
4.6	Complex Passwords	Minimum 8 characters in length with characters from at least three of the following categories: uppercase, lowercase, numeric, non-alphanumeric	<p>A policy should be in place to require complex passwords on all database accounts.</p> <p>Rationale: Complex passwords help mitigate dictionary, brute forcing, and other password attacks.</p>	ALL	X	X	1 N
<p>Auditing Guidance for section 4.6: N/A</p>							
4.7	Verify Secure Password Hashes	All password hashes should be 41 bytes or longer	<pre>Use "select User, Password from mysql.user where length(password) < 41;" query to verify.</pre> <p>Rationale: Starting in v4.1 a stronger</p>	ALL	X	X	1 S

			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 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 R
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	ALL	X	X	2 R

			wildcards within hostnames will ensure that only trusted principals are capable of interacting with MySQL.				
Auditing Guidance for section 4.9: 1. SQL: "select user from mysql.user where host = '%';" 2. Verify that no results are returned							
4.10	No blank passwords	Verify no blank passwords	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 = null;" query to verify. Rationale: Blank passwords negate the benefits provided by authentication mechanisms.	ALL	X	X	1 S
Auditing Guidance for section 4.10: 1. SQL: "select user, password from mysql.user where length(password) = 0 or password = null;" 2. Verify that no results are returned							
4.11	Anonymous account	Verify and remove anonymous accounts	Anonymous accounts are users with no name (''). They allow for default logins and there permissions can sometimes be used by other users. Check for anonymous users using the query "select user from mysql.user where user =	ALL	X	X	1 S

			<pre>'';".</pre> <p>Rationale: Anonymous accounts are users with no name (''). They allow for default logins and their permissions can sometimes be used by other users. Avoiding the use of anonymous accounts will ensure that only trusted principals are capable of interacting with MySQL.</p>				
<p>Auditing Guidance for section 4.11:</p> <ol style="list-style-type: none"> 1. SQL: "select user from mysql.user where user = '';" 2. Verify that no results are returned 							

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	<p>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')));"</p> <p>Rationale: Limiting the accessibility of the 'mysql' database will protect the confidentiality, integrity, and availability of the data housed within MySQL.</p>	ALL	X	X	1 R

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 R
-----	-----------------------	---------------------------------	--	-----	---	---	-----

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 R
-----	--------------------------	---------------------------------	--	-----	---	---	-----

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 R
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 R

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 R
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 R
Auditing Guidance for section 5.7: 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							
5.8	Global GRANT privilege	Do not grant to non Admin users	<p>Allows changing of permissions. Verify using following query: <pre>"select user, host from mysql.user where Grant_priv = 'Y';"</pre></p> <p>Rationale: The GRANT privilege allows a principal to grant other principals additional privileges. This may be used by an attacker to compromise MySQL.</p>	ALL	X	X	1 R
<p>Auditing Guidance for section 5.8:</p> <ol style="list-style-type: none"> 1. SQL: <pre>"select user, host from mysql.user where Create_user_priv = 'Y';"</pre> 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.

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

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

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

			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"							

6.11	Client password	Do not use password= configuration option	<p>The [Client] section of the MySQL configuration file allows setting a password to be used. Verify this option is not used.</p> <p>Rationale: The use of this parameter may negatively impact the confidentiality of the user's password.</p>	ALL	X	X	2 S
<p>Auditing Guidance for section 6.11: Examine the [Client] section of the MySQL configuration file and ensure this option is not employed.</p>							

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.

Item #	Configuration Item	Action / Recommended Parameters	Comments	Version	Windows	Unix	Level
7.1	Client Verify Server Cert	--ssl-verify-server-cert	Causes the server's common name (CN) to be verified against the server's hostname. Rationale: Verifying the server's certificate will help protect against man in the middle attacks.	5.1	X	X	1 S
Auditing Guidance for section 7.1: In the [client] portion of the MySQL configuration file check for the existence of <code>ssl_verify_server_cert</code>							
7.2	SSL Connection	Must use SSL over untrusted networks (internet) or when restricted PII is transferred	Rationale: SSL will protect the confidentiality and integrity of sensitive information as it traverses untrusted networks.	ALL	X	X	2 S
Auditing Guidance for section 7.2: 1. SQL: "show variables like 'have_openssl';" is "YES" 2. 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 <code>mysql.user.ssl_type</code> field to ANY, X509, or SPECIFIED							

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							

References

The following references are available for further reading:

From the MySQL documentation:

- [MySQL v4.1 General Security Issues](http://dev.mysql.com/doc/refman/4.1/en/security.html)
<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)
<http://dev.mysql.com/doc/refman/5.0/en/security.html>
- [MySQL v5.1 General Security Issues](http://dev.mysql.com/doc/refman/5.1/en/security.html)
<http://dev.mysql.com/doc/refman/5.1/en/security.html>

MySQL Change History:

- [MySQL v4.1 Change History](http://dev.mysql.com/doc/refman/4.1/en/news.html)
<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)
<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)
<http://dev.mysql.com/doc/refman/5.1/en/news.html>

Webinar from MySQL:

- [Best Practices for Securing MySQL 5.0](http://www.mysql.com/news-and-events/on-demand-webinars/security-2006-04-19.php)
<http://www.mysql.com/news-and-events/on-demand-webinars/security-2006-04-19.php>

Security Focus

- [Securing MySQL: step-by-step](http://www.securityfocus.com/infocus/1726)
<http://www.securityfocus.com/infocus/1726>
- [Secure MySQL Database Design](http://www.securityfocus.com/infocus/1667)
<http://www.securityfocus.com/infocus/1667>

Chrooting MySQL on Unix

- [Chrooting MySQL on Debian](http://blog.blackdown.de/2005/03/04/chrooting-mysql-on-debian/)
<http://blog.blackdown.de/2005/03/04/chrooting-mysql-on-debian/>