MySQL Unsuccessful Logon Attempts Control Policy (NIST AC-7)

1. MySQL Unsuccessful Logon Attempts Configuration (next section) shall be implemented for each identified target system.

2. A limit of 3 consecutive invalid logon attempts will be enforced.

3. The account will be automatically locked for a period of 1 day when the maximum number of unsuccessful logon attempts is exceeded.

MySQL Unsuccessful Logon Attempts Configuration (NIST AC-7)

1. In a connected and logged-in mysql prompt, run the following, where <user> is the actual name of the user account, and <password> is the actual password:

   ```
   CREATE USER <user>@localhost IDENTIFIED BY '<password>'
   FAILED_LOGIN_ATTEMPTS 3 PASSWORD_LOCK_TIME 1;
   ```

   Expected response:

   ```
   Query OK, 0 rows affected (0.02 sec)
   ```

2. Existing users can be altered by running the following, where <user> is the actual name of the user account:

   ```
   ALTER USER <user>@localhost FAILED_LOGIN_ATTEMPTS 3
   PASSWORD_LOCK_TIME 1;
   ```

   Expected response:

   ```
   Query OK, 0 rows affected (0.02 sec)
   ```

MySQL Audit Control Policy (NIST AU-2)

1. MySQL Audit Function Install/Configure (next section) shall be implemented for each identified target system.

2. All events will be audited (logged) for each occurrence of the event.

3. Coordinates the security audit function with other organizational entities requiring audit-related information to enhance mutual support and to help guide the selection of auditable events.

4. Auditable events are deemed to be adequate to support after-the-fact investigations of security incidents since each and every event is logged.

5. All events shall be enabled as being audited and logged. Audit logs shall be reviewed once per week by the security operations team. Any event which requires investigation shall kick-off an audit log review by the aforementioned team.

MySQL Audit Function Install/Configure (NIST AU-2)

6. Copy audit_log.so to /usr/lib/mysql/plugin

7. In a connected and logged-in mysql prompt, run the following:
   ```
   install plugin audit_log soname 'audit_log.so';
   ```

   Expected response:
   ```
   Query OK, 0 rows affected (0.03 sec)
   ```

8. In a connected and logged-in mysql prompt, run the following:
   ```
   Show plugins;
   ```

   Expected response (near the bottom):
   ```
   | audit_log | ACTIVE | AUDIT | audit_log.so | GPL |
   46 rows in set (0.00 sec)
   ```

9. Add the below line to /etc/mysql/mysql.conf.d/mysql.cnf and save the file:
   ```
   audit_log_file = /var/log/mysql/audit.log
   ```

10. Restart MySQL:
    ```
    sudo systemctl restart mysql
    ```

11. In a connected and logged-in mysql prompt, run the following:
    ```
    mysql> show global variables like 'audit%';
    ```

    Expected result (or similar):
    ```
    +--------------------------+-------------------------+
    | Variable_name            | Value                   |
    +--------------------------+-------------------------+
    | audit_log_buffer_size    | 1048576                 |
    | audit_log_file           | /var/log/mysql/audit.log |
    ```

```
| audit_log_flush            | OFF                      |
| audit_log_format           | OLD                      |
| audit_log_handler          | FILE                     |
| audit_log_policy           | ALL                      |
| audit_log_rotate_on_size   | 0                        |
| audit_log_rotations        | 0                        |
| audit_log_strategy         | ASYNCHRONOUS             |
| audit_log_syslog_facility | LOG_USER                 |
| audit_log_syslog_ident     | percona-audit            |
| audit_log_syslog_priority | LOG_INFO                 |
+----------------------------+--------------------------+
12 rows in set (0.00 sec)
```

12. Connect to the database and perform a query to test the audit.log function:

```
mysql> use GroupProj;
Database changed
mysql>select * from GroupProj.Angola;
------------------------
7 rows in set (0.00 sec)
```

13. Check /var/log/mysql/audit.log for capture:

```
# sudo cat /var/log/mysql/audit.log
```

Result:

```
<AUDIT_RECORD
  NAME="Query"
  RECORD="69_2020-11-21T01:59:59"
  TIMESTAMP="2020-11-21T02:12:05Z"
  COMMAND_CLASS="select"
  CONNECTION_ID="8"
  STATUS="0"
  SQLTEXT="select * from GroupProj.Angola"
  USER="root[root] @ localhost []"
  HOST="localhost"
  OS_USER=""
  IP=""
  DB=""
/>
```

Software Usage Restrictions (NIST CM-10)

1. Software and associated documentation shall be used in accordance with contract agreements and copyright laws.
2. Software and associated documentation is manually tracked via software inventory, and is protected by tracking quantity licenses to control copying and distribution.
3. Peer-to-peer file sharing shall be restricted to specific roles which are required for identified business processes. Only authorized personnel and user roles shall utilize peer-to-peer file sharing for the purpose of performing identified business functions. The use of peer-to-peer file sharing is documented and controlled to ensure that it is not used for unauthorized distribution, display, performance, or reproduction of copyrighted work.
4. The installation of Open Source Software is restricted to software which must:
    a. Be legally licensed
    b. Adhere to the secure configuration baseline

Software Inventory

Vendor	Product	Version	License	Server Count
Ubuntu	Desktop	20.04.01 LTS	GNU General Public License	1
Oracle	MySQL Community Server	8.0.22	GNU General Public License	1
Oracle	MySQL Workbench	8.0.22	GNU General Public License	1

Peer-to-Peer Tracking

Server	Technology	User or Role	Reason
none	none	none	n/a

Authenticator Feedback (NIST IA-6)

1. The information system shall obscure feedback of authentication information during the authentication process to protect the information from possible exploitation/use by unauthorized individuals.

2. MySQL versions 8 and higher inherently obscure feedback of authentication information.

3. MySQL Workbench versions 8 and higher inherently obscure feedback of authentication information.

4. Ubuntu Desktop versions 20 and higher inherently obscure feedback of authentication information.

Allocation of Resources (NIST SA-2)

1. The CIO is responsible for budgeting and capital planning for information technology, which will include allocating proper resources for information security.
2. The information security officer will prepare request for new services or products based on risk management decisions and forward them to the CIO for consideration and review.
3. The information security officer shall determine information security requirements for the information system or information system service in mission/business process planning.

Denial of Service Protection (NIST SC-5)

1. Network edge security devices (Unified Threat Management (UTM) or next generation firewalls (NGFW)) shall be employed to protect information system components on internal organizational networks from being directly affected by denial of service attacks.
2. Database shall be employed on a server cluster such that denial of service attacks are mitigated.
3. Appropriate network, access control lists, IPS controls, and proactive monitoring shall be implemented to decrease risk of denial of service attacks (internal and external) on critical systems.

Malicious Code Protection (NIST SI-3)

1. ClamAV shall be deployed on database server (next section) to detect and eradicate malicious code.
2. ClamAV shall be updated whenever new releases are available.
3. ClamAV shall be configured to:
    a. Perform daily scans of the server and real-time scans of files from external sources as the files are downloaded, opened, or executed
    b. Block and quarantine malicious code and sends alert to administrator in response to malicious code detection
    c. Addresses the receipt of false positives during malicious code detection and eradication and the resulting potential impact on the availability of the information system.

ClamAV Installation and Configuration (NIST SI-3)

Implement the following steps on a terminal console for the desired database server:

1. sudo apt-get update

2. sudo apt-get -y install clamav-daemon

3. sudo systemctl enable clamav-daemon

4. sudo systemctl start clamav-daemon

5. $ tail /var/log/clamav/clamav.log

   Response:
   ```
   Sun Jun 28 19:08:32 2020 -> Portable Executable support enabled.
   Sun Jun 28 19:08:32 2020 -> ELF support enabled.
   Sun Jun 28 19:08:32 2020 -> Mail files support enabled.
   Sun Jun 28 19:08:32 2020 -> OLE2 support enabled.
   Sun Jun 28 19:08:32 2020 -> PDF support enabled.
   Sun Jun 28 19:08:32 2020 -> SWF support enabled.
   Sun Jun 28 19:08:32 2020 -> HTML support enabled.
   Sun Jun 28 19:08:32 2020 -> XMLDOCS support enabled.
   Sun Jun 28 19:08:32 2020 -> HWP3 support enabled.
   Sun Jun 28 19:08:32 2020 -> Self checking every 3600 seconds.
   ```

6. Run a scan:
   ```
   sudo clamdscan –fdpass
   ```

   Response:
   ```
   /home/testuser: OK
   ----------- SCAN SUMMARY -----------
   Infected files: 0
   Time: 0.015 sec (0 m 0 s)
   ```

7. Test by downloading a virus file:
   ```
   wget www.eicar.org/download/eicar.com
   ```

8. Rescan:
   ```
   sudo clamdscan –fdpass
   ```

   Response:
   ```
   /home/ubuntu/eicar.com: Win.Test.EICAR_HDB-1 FOUND
   ----------- SCAN SUMMARY -----------
   Infected files: 1
   Time: 0.005 sec (0 m 0 s)
   ```

9. sudo mkdir /root/quarantine

10. Exclude certain directories from being scanned (all one line):
    ```
    sudo printf "ExcludePath ^/proc\nExcludePath ^/sys\nExcludePath
    ^/run\nExcludePath ^/dev\nExcludePath ^/snap\nExcludePath
    ^/var/lib/lxcfs/cgroup\nExcludePath ^/root/quarantine\n" | sudo
    tee -a /etc/clamav/clamd.conf
    ```

11. Restart clamav-daemon:
    ```
    sudo systemctl restart clamav-daemon
    ```

12. sudo su –

13. Execute clamdscan every night at 1:00 am (this is one continuous line):
    ```
    echo "0 1 * * * root /usr/bin/clamdscan --fdpass --
    log=/var/log/clamav/clamdscan.log --move=/root/quarantine /" |
    tee /etc/cron.d/clamdscan
    ```

14. Exit (from sudo su -)

15. Test full system scan (with quarantine file movement):
    ```
    sudo /usr/bin/clamdscan --fdpass --
    log=/var/log/clamav/clamdscan.log --move=/root/quarantine /
    ```

    Results:
    ```
    ---------------------------------------
    /snap: Excluded
    /run: Excluded
    /dev: Excluded
    /proc: Excluded
    /sys: Excluded
    /var/lib/lxcfs/cgroup: Excluded
    WARNING: /var/lib/lxd/unix.socket: Not supported file type
    /home/ubuntu/eicar.com: Win.Test.EICAR_HDB-1 FOUND
    /home/ubuntu/eicar.com: moved to ‘/root/quarantine/eicar.com’
    /root/quarantine: Excluded
    ----------- SCAN SUMMARY -----------
    Infected files: 1
    Total errors: 1
    Time: 235.497 sec (3 m 55 s)
    ```
16. Begin to enable real-time scanning:

```
sudo printf "OnAccessIncludePath /home\nOnAccessIncludePath
/var/www\nOnAccessExcludeUname clamav\nOnAccessExcludeRootUID
true" | sudo tee -a /etc/clamav/clamd.conf
```

17. Create a systemd file for clamonacc:

```
[Unit]
Description=ClamAV On Access Scanner
Requires=clamav-daemon.service
After=clamav-daemon.service syslog.target network.target

[Service]
Type=simple
User=root
ExecStartPre=/bin/bash -c "while [ ! -S /var/run/clamav/clamd.ctl
]; do sleep 1; done"
ExecStart=/usr/bin/clamonacc -F --config-
file=/etc/clamav/clamd.conf --log=/var/log/clamav/clamonacc.log -
-move=/root/quarantine

[Install]
WantedBy=multi-user.target
```

18. Enable and start the clamonacc daemon:

```
sudo systemctl enable clamonacc
sudo systemctl start clamonacc
```

19. Download a virus file to see if it's automatically detected and quarantined:

```
wget www.eicar.org/download/eicar.com
```

20. See if the file was moved:

```
ls -l
```

    Result:

```
Total: 0
```

21. Check the quarantine folder:

```
sudo ls /root/quarantine
```

    Result:

```
eicar.com
```

22. Check the logs:

```
sudo tail /var/log/clamav/clamonacc.log
```

    Result:

```
-------------------------------------
ClamInotif: watching '/home' (and all sub-directories)
ClamInotif: watching '/var/www' (and all sub-directories)
/home/testuser/eicar.com: Win.Test.EICAR_HDB-1 FOUND
/home/testuser/eicar.com: moved to '/root/quarantine/eicar.com'
```

MySQL Account Management Control Policy (NIST AC-2)

1. Create Roles and Create Users .

2. Assigned Roles to the Users.

3. The account will be automatically locked for a period of 1 day when the maximum number of unsuccessful logon attempts is exceeded.

MySQL Account Management Configuration (NIST AC-2)

1. Creating Users and roles :

```
CREATE ROLE manager;
CREATE ROLE readonly;
CREATE USER 'ge'@'localhost' IDENTIFIED BY 'MypPassword$123';
CREATE USER 'el'@'localhost' IDENTIFIED BY 'MyPassword$321';
```

Expected response for all the above:

```
Query OK, 0 rows affected (0.02 sec)
```

2. Granting roles access to database:

```
GRANT SELECT ON GroupProj.* to manager;
GRANT SHOW DATABASES on *.* to manager;
GRANT SELECT, INSERT, UPDATE, DELETE on GroupProj.* to manager;

GRANT SELECT ON GroupProj.* to readonly;
```

Expected response for all the above:

```
Query OK, 0 rows affected (0.02 sec)
```

3. Assigning the roles to the users:

```
GRANT manager TO 'el'@'localhost';
GRANT readonly TO 'ge'@'localhost';
```

Expected response for all the above:

```
Query OK, 0 rows affected (0.02 sec)
```

MySQL Separation of Duties Control Policy (NIST AC-5)

1. Have separate duties for each user .

2. Those users shouldn't be able to access other user stuff .

MySQL Separation of Duties Configuration (NIST AC-5)

Already have users and roles created from last nist-2 controls which are

1. USERS and Their Roles:

```
GRANT SELECT ON GroupProj.* to manager;
GRANT SHOW DATABASES on *.* to manager;
GRANT SELECT, INSERT, UPDATE, DELETE on GroupProj.* to manager;

GRANT SELECT ON GroupProj.* to readonly;
```

Expected response for all the above:

```
Query OK, 0 rows affected (0.02 sec)
```

2. Testing controls for manager:

```
Open sql terminal
Mysql –u el –p;
SET ROLE manager;
SELECT CURRENT_ROLE;
ADD TABLE TEST;
DROP TABLE TEST:
```

Expected response for all the above:

```
Query OK, 0 rows affected (0.02 sec)
```

3. Testing controls for manager:

```
Open sql terminal
Mysql –u el –p;
SET ROLE readonly;
SELECT CURRENT_ROLE;
ADD TABLE TEST;
```

Expected response for all the above:

```
Query OK, 0 rows affected (0.02 sec)
```

Got an error while adding the table TEST

ERROR 1044(42000): Acess denied for user ‘ge’@’localhost’
CREATE TABLE TEST denied in database ‘GroupProj’

MySQL System Use Control Policy (NIST AC-8)

1. Create or have some sort of Notification that will show the Usage

MySQL System Use Configuration (NIST AC-8)

1. In work Bench Go to Scripting
2. Go to scripting shell
3. Notification
4. Find notification of GroupProj

MySQL Role-Based Security Training Control Policy (NIST AT-3)

1. We don’t have any role based security training for our database.

2. In order to get that done the user should go through the policies made by the organization for their specific role.

3. The user should view all the training material.

4. The user should follow all the tools provided to make sure that the security is taken care of well.

MySQL Content of Audit Records Policy (NIST AU-3)

1. MySQL Audit records will be used to show the audits .

2. It will show the audits available under content of audit records.

3. Coordinates the security audit function with other organizational entities requiring audit-related information to enhance mutual support and to help guide the selection of auditable events.

MySQL Content of Audit Records Configure (NIST AU-3)

4. Copy audit_log.so to /usr/lib/mysql/plugin

5. In a connected and logged-in mysql prompt, run the following:
   ```
   install plugin audit_log soname 'audit_log.so';
   ```

   Expected response:
   ```
   Query OK, 0 rows affected (0.03 sec)
   ```

6. In a connected and logged-in mysql prompt, run the following:
   ```
   Show plugins;
   ```

   Expected response (near the bottom):
   ```
   | audit_log | ACTIVE | AUDIT | audit_log.so | GPL |
   46 rows in set (0.00 sec)
   ```

7. Add the below line to /etc/mysql/mysql.conf.d/mysql.cnf and save the file:
   ```
   audit_log_file = /var/log/mysql/audit.log
   ```

8. Restart MySQL:
   ```
   sudo systemctl restart mysql
   ```

9. In a connected and logged-in mysql prompt, run the following:
   ```
   mysql> show global variables like 'audit%';
   ```

   Expected result (or similar):
   ```
   +--------------------------+-------------------------+
   | Variable_name            | Value                   |
   +--------------------------+-------------------------+
   | audit_log_buffer_size    | 1048576                 |
   | audit_log_file           | /var/log/mysql/audit.log |
   | audit_log_flush          | OFF                     |
   | audit_log_format         | OLD                     |
   | audit_log_handler        | FILE                    |
   | audit_log_policy         | ALL                     |
   | audit_log_rotate_on_size | 0                       |
   | audit_log_rotations      | 0                       |
   | audit_log_strategy       | ASYNCHRONOUS            |
   ```

```
| audit_log_syslog_facility | LOG_USER                 |
| audit_log_syslog_ident    | percona-audit            |
| audit_log_syslog_priority | LOG_INFO                 |
+---------------------------+--------------------------+
12 rows in set (0.00 sec)
```

MySQL Time Stamp Control Policy (NIST AU-8)

1. Have some sort of time stamp on the data

2. We didn't had any time stamp on our tables so created one with time stamp.

MySQL Time Stamp Configuration (NIST AU-8)

1. Creating Table groupmemtitle tied to Groupmem so technically created two tables :

CREATE TABLE groupmemtitle(id INT PRIMARY KEY AUTO_INCREMENT, title varchar (300), content TEXT, publishdate DATETIME, expiredate DATETIME, updated TIMESTAMP default NOW() ON UPDATE NOW(), groupmem_id INT, FOREIGN KEY (groupmem_id) REFERENCES groupmem (id) ON DELETE NO ACTION) ENGINE = INNODB;

Expected response for all the above:

```
Query OK, 0 rows affected (0.02 sec)
```

2. Added data:

```
INSERT INTO groupmemtitle (title, groupmem_id) Values ('ODD',1);
INSERT INTO groupmemtitle (title, groupmem_id) Values ('PL',2);
INSERT INTO groupmemtitle (title, groupmem_id) Values
('PA/SR',4);
INSERT INTO groupmemtitle (title, groupmem_id) Values
('PA/SD',3);
INSERT INTO groupmemtitle (title, groupmem_id) Values
('PA/SC',5);
```

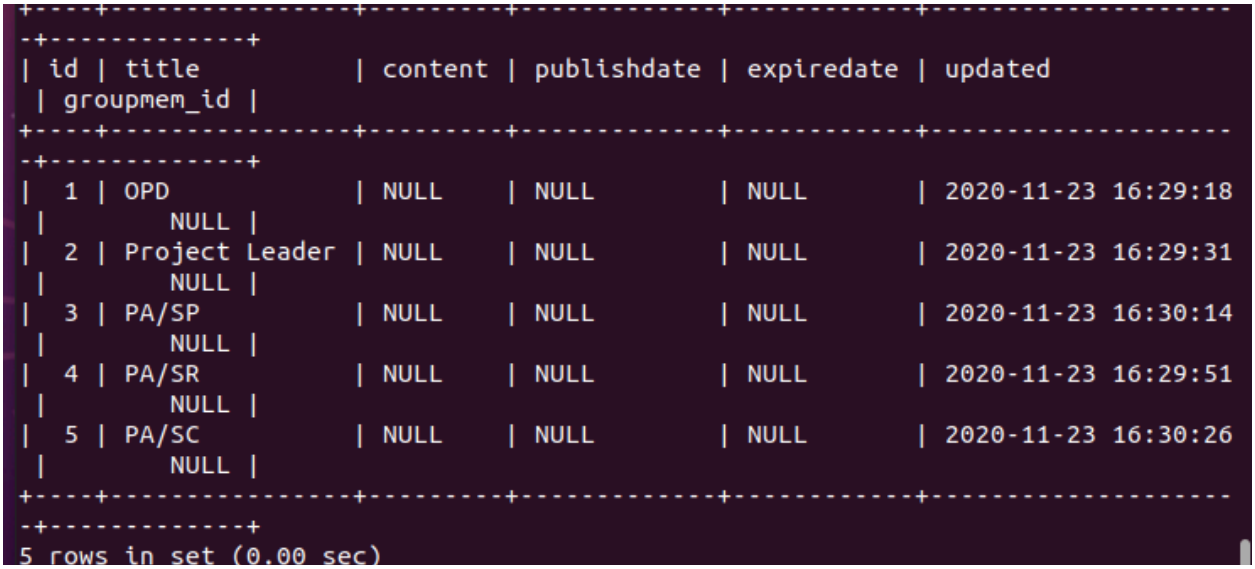
Expected response for all the above:

```
Query OK, 0 rows affected (0.02 sec)
```

3. Assigning the roles to the users:

```
SELECT * FROM groupmemtitle;
```

```
Shows the updated table
```



```
+----+----------------+---------+-------------+------------+---------------------
-+-------------+
| id | title          | content | publishdate | expiredate | updated
 | groupmem_id |
+----+----------------+---------+-------------+------------+---------------------
-+-------------+
|  1 | OPD            | NULL    | NULL        | NULL       | 2020-11-23 16:29:18
 |        NULL |
|  2 | Project Leader | NULL    | NULL        | NULL       | 2020-11-23 16:29:31
 |        NULL |
|  3 | PA/SP          | NULL    | NULL        | NULL       | 2020-11-23 16:30:14
 |        NULL |
|  4 | PA/SR          | NULL    | NULL        | NULL       | 2020-11-23 16:29:51
 |        NULL |
|  5 | PA/SC          | NULL    | NULL        | NULL       | 2020-11-23 16:30:26
 |        NULL |
+----+----------------+---------+-------------+------------+---------------------
-+-------------+
5 rows in set (0.00 sec)
```

MySQL Configuration Control Policy (NIST AT-3)

1. We don’t have any configuration control policy or change in this database.

2. If needed currently I am running this on a hard drive but can be moved to a secure measure like windows workbench or something else.

MySQL Fire Protection Control Policy (NIST PE-13)

1. Facilities need to maintain their server rooms, data centers, and mainframe computer rooms safe from fire.
2. Facilities will determine the best method to prevent Fire from damaging any of the equipment.

MySQL Fire Protection Configuration (NIST PE-13)

1. We don’t have any Fire Protection configuration.
2. If we needed to had one we will make sure that our area is equipped with best fire alarms, smoke detector and they are connected directly to 911 system.

Access Enforcement (NIST AC-3)

1. Information Systems should enforce approved authorizations for logical access to informations and system resources
2. Access enforcement can also be employed at the application level

MySQL Access Enforcement (NIST AC-3)

3. Login to mysql as root user which has full access `/usr/local/mysql/bin/mysql -u root -p`
4. Connect to database `use GroupProj;`
   Expected output: > Database changed
5. Attempt logical access to database `select City from Austria; delete from Austria where City = "Wels";`
   Expected output: > Query OK, 1 row affected (0.01 sec)

Permitted Actions Without Identification or Authentication (NIST AC-14)

1. Organization must specify what type of user actions can be performed on the information system without identification or authentication
2. Access is restricted for any user without authentication or identification on our system

MySQL Permitted Actions Without Identification or Authentication (NIST AC-14)

3. Open sql terminal
4. Create a user with not authentication `create user 'noAuth';`
   Expected output > Query OK, 0 rows affected (0.10 sec)
5. Try accessing the database `show databases;`
   Expected output >
```
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.03 sec)
```
6. No access to the database ‘GroupProj’

Remote Access (NIST AC-17)

1. Organization established and documents user restrictions, connection requirements and implementation guidance for each type of remote access it allows
2. Organization authorize remote access prior to allowing connections

MySQL Remote Access (NIST AC-17)

3. Attempt remote access mysql -u root -h <host> -P 3306 -D GroupPorj -p

Publicly Accessible Content (AC-22)

1. Organization designated who can post information on publicly accessible information systems
2. Organization ensures that public information does not contain non public information
3. Organization reviews content before posting

MySQL Publicly Accessible Content (AC-22)

4. Create database from public information found at https://www.start.umd.edu/data-tools/global-terrorism-database-gtd
5. Connect to database as root user or any authorized user /usr/local/mysql/bin/mysql -u root -p
   use GroupProj;
   Expected output: > Database changed

6. Access public data select * from Brazil limit 5;
   Expected output: > 5 rows in set (0.01 sec)

Incident Handling (NIST IR-4)

1. Organization implements incident handling capabilities for security incidents
2. Organization uses lessons learned from ongoing incident handling to implement better incident response procedures, training and testing
3. Incident response capability is dependent on the capabilities of the information system
4. Incident related information can be obtained from a variety of sources including audit monitoring, network monitoring and administration reports

Incident Handling with MySQL Workbench (NIST IR-4)

5. Check some sources for incident handling by clicking “Server” on the toolbar
6. Click “Server Status” to see some incidents happening on the server. ex-CPU load, traffic, buffer usage
7. Click “Server logs” in order to see the log of incidents on server

System Development Life Cycle (NIST SA-3)

1. Organization manages the system using its own defined development life cycle that incorporated information security considerations
2. Organization defines and documents information security roles and responsibilities
3. Our system implements 4 roles in our development life cycle: database_administrator, end_user, client, testing
4. Each have their own responsibilities which we specified with the privileges granted: grant select, insert, update, delete / grant select, insert, update / grant select / grant select, update for database_administrator, end_user, client and testing respectively

Boundary Protection (NIST SC-7)

1. Information system monitors and controls communication at the external boundaries of the system
2. Information system implements subnetworks for publicly accessible components separated from internal networks
3. Information system only connects to external networks through managed interfaces consisting of boundary protection devices
4. Managed interfaces include gateways, routers, firewalls, guards or encrypted tunnels

Boundary Protection with MySQL Workbench (NIST SC-7)

5. On the toolbar click ‘Server’ then ‘Option files’ then “security” to view the boundary security on the server

Plan of Action and Milestones (NIST CA-5)

1. Organization develops a plan of action and milestones that they want to accomplish for remedial actions to correct weaknesses or deficiencies noted during the assessment of security controls.

2. With their current plan of actions and milestones, it should be constantly updated with findings from the assessments.

Baseline Configuration (NIST CM-2)

1. Organizations should create documents regarding the baseline configurations for each of its devices.

2. Software applications needed for the organizations should be downloaded onto the new devices and or pre-existing devices for the workstation.

3. Network Configurations should also be configured and set up on each device for the workstation

Baseline Configuration for MySQL Database (NIST CM-2)

1. Database/schema should be set up using one device utilizing MySQL Workbench

2. After creating database and importing all data, push to GitHub for other workstations to utilize “git pull” to receive the database

3. All workstations should pull the database onto the computer

4. Desktop or laptop device needs to Import data to MySQL Workbench and verify that it is working

Information System Backup (NIST CP-9)

1. Information Systems shall back up databases and any data within the database, but there is not documentation in need of backup. Any documentation should be stored in the same place.

2. Keep all of the backups of information in a safe place

3. With most updated changes to database, create a backup of database in MySQL Workbench

4. Conduct a backup by Clicking “Server” and then “Data Export”

5. Export all information needed to a selected dump folder, export the data, and then store in a secure storage location along with any existing documents.

Information System Recovery and Reconstitution (NIST CP-10)

1. With information systems crashing or losing any data/documentation, all data can be recovered from a created backup

2. Create a schema with title similar to data that was backed up

3. Click “Server” and then direct to “Data Import” option in drop down menu

4. Import data from selected folder by clicking the ellipses and locating the data from the most recent backup in storage location.

5. Select all data needed to be imported, and Click “Import Data”

6. Validate that all data has been imported and is usable by refreshing the Schema tab

Media Access (NIST MP-2)

1. For organizations, data any important data should be stored with flash drives, external hard drives, compact disks, etc., which is the digital method of storing data and important records

2. Non-digital method includes storing all important files, documentation, and records, in a storage area within the building

3. Access should be restricted so the development team only is able to access the data

Media Access with MySQL (NIST MP-2)

1. With the data present within' MySQL Workbench, click "Server" and then click "Data Export" from the drop-down menu.

2. Export the selected data into a dump folder in the designated file directory

3. Insert External Hard Drive or Flash Drive to device and transfer data over into external storage device.

4. Restrict access by placing a password lock on storage device; therefore, development team should only have access to data

Visitor Access Records (NIST PE-8)

1. Organizations should document all visitors that are accessing the building by recording entrance and departure times, date of access, identification, and purpose of visits.

2. Documentation can be digital or non-digital and stored for safe keeping

Visitor Access Records with MySQL Workbench (NIST PE-8)

1. Users accessing the database should record in a shared document of date access, identification, time accessed/time completed, and purpose for accessing database along with any changes created.

Water Damage Protection (NIST PE-15)

1. Facilities need to maintain a safe area for the server rooms, data centers, and mainframe computer rooms.

2. Facilities should file a request in regards to any water valves that are present within the server rooms, data centers, or mainframe computer rooms.

3. Facilities will determine the best method to prevent water valves from damage any of the equipment.

Use of External Information System Control Policy (NIST AC-20)

1. External information system should be able to access the organization's information system with authorization, and also able to process, store, or transmit organization-controlled information
2. The organization should restrict the use of organization-controlled portable storage devices by authorized individuals on external information system
3. The organization should prohibit the use of non-organizationally owned systems/components/devices

Security Training Record Control Policy (NIST AT-4)

1. The organization should document and monitor the system security training activities including basic security awareness training and specific security training.
2. The organization should retain the training records for a period based on the security policy.

User-Installed Software Control Policy (NIST CM-11)

1. The organization should establish governing the installation of software by users.
2. The organization needs to build alert system to enforce software installation policies.
3. The organization should monitor policy compliance regularly.

Incident Monitoring (NIST IR-5)

1. The organization tracks and documents information system security incidents

Incident Monitoring with MySQL Workbench (NIST IR-5)

1. Check some sources for incident monitoring by clicking “Server” on the toolbar
2. Click “Server logs” to see the error log files of the incidents on server
3. Click “Performance reports” in order to see incident related performance
4. Click “Users and Privileges” to see user information and privileges, in order to obtain incidents related info and user/administrator reports, also can monitor the physical access

Rules of Behavior (NIST PL-4)

1. The organization needs to establish rules that describe the responsibilities and expected behaviors with regard to information and information system usage for individuals who require access to the information system
2. The organization needs to document the agreements signed by such individuals
3. The organization should review and update the rules of behavior regularly
4. Individuals who have signed the agreements need to re-sign a copy if there is any update

Incident Monitoring with MySQL Workbench (NIST PL-4)

1. Check some sources for monitor individual users' behavior by clicking "Server" on the toolbar
2. Click "Performance reports" in order to see "User Resource Use"
3. Click "User Resource Use" to see "User Behavior Statistic Data"

Access Agreement (NIST PS-6)

1. The organization develops and documents access agreements for organizational information systems
2. The organization needs to review and update the access agreements regularly
3. The organization needs to ensure the user sign appropriate access agreements prior to being granted access
4. The organization needs to ensure the user re-sign the documents when the access agreements have been updated.

Access Agreement in MySQL Workbench (NIST PS-6)

1. Create database in MySQL Workbench from public information found at https://www.start.umd.edu/data-tools/global-terrorism-database-gtd
2. Open terminal and connect to database as root user or any user have the privilege to create roles.
   ~$ mysql -u root -p;.
   Then enter the password for root user.
   Expected output: able to connect to the database.
3. Use GroupProj database
   mysql> use GroupProj;
   Expected output: database changed
4. Create user role
   mysql> create role end_user;
   Expected output: role created
5. Create user 'GlobalProj_user' at localhost with password
   mysql> create user 'GlobalProj_user'@'localhost' identified by 'pA$sw0rD';
   Expected output: user created
6. After signing the access agreements, users will be granted access
7. Grant access and privileges for user role
   mysql> grant end_user to 'GlobalProj_user'@'localhost';
   mysql> grant select on GlobalProj.* to end_user;
   Expected output: role user gets the access and privileges

Third Party Personnel Security (NIST PS-7)

1. The organization establish personnel security requirements including security roles and responsibilities for third party providers
2. The organization requires third-party providers to comply with personnel security policies and procedures established by the organization
3. The organization should document personnel security requirements
4. The organization requires third-party providers to notify any personnel transfers or terminations of third-party personnel who have information system privileges.

Third Party Personnel Security in MySQL Workbench (NIST PS-7)

1. Create database in MySQL Workbench from public information found at https://www.start.umd.edu/data-tools/global-terrorism-database-gtd
2. Open terminal and connect to database as root user or any user have the privilege to create roles.
   ~$ mysql -u root -p;.
   Then enter the password for root user.
   Expected output: able to connect to the database.
3. Use GroupProj database
   mysql> use GroupProj;
   Expected output: database changed
4. Create client role
   mysql> create role client;
   Expected output: role created
5. Create user 'GlobalProj_client' at localhost with password
   mysql> create user 'GlobalProj_client'@'localhost' identified by 'pA$sw0rD';
   Expected output: user created
6. Grant access and privileges for client role
   mysql> grant client to 'GlobalProj_client'@'localhost';
   mysql> grant select on GlobalProj.* to client;
   Expected output: role client gets the access and privileges
7. After going through the personnel security policy and procedures and signing the agreements, third-party providers can get the access as client and should be able to use role client to connect to the database