MySQL Security: Best Practices

Manizha Davis
Oracle Enterprise Architect
Manizha.Davis@oracle.com
Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions.

The development, release, and timing of any features or functionality described for Oracle’s products remains at the sole discretion of Oracle.
89% of Organizations Experienced Data Breaches, According to New Ponemon Report

Source: *Sixth Annual Benchmark Study on Privacy & Security of Healthcare Data*, conducted by Ponemon Institute

66% of the largest businesses in the UK have suffered a cyberattack or data breach within the past twelve months

Source: *UK government’s Cyber Security Breaches Survey 2016*

25% experience a repeated breach at least one a month

Source: *UK government’s Cyber Security Breaches Survey 2016*
Goals are simple

• Protect Data
  – From those who shouldn’t see it
  – Don’t lose the data

• Maintain Data Availability
As a DBA / Developer / ... we need to

- Ensure only users who should get in, can get in
- Limit what users and user applications can do
- Limit from where users and applications can access data
- Watching What is happening, and when it happened
- Make sure we can back things up (don’t lose data)
- Make sure we keep history of what happened
- Minimize attack surface
Database Vulnerabilities

- Poor Configurations
  - Set controls and change default setting
- Over Privileged Accounts
  - Privilege Policies
- Weak Access Control
  - Dedicated Administrative Accounts
- Weak Authentication
  - Strong Password Enforcement
- Weak Auditing
  - Compliance & Audit Policies
- Lack of Encryption
  - Data, Backup, & Network Encryption
- Proper Credential & Key Management
  - Use mysql_config_editor, Key Vaults
- Unsecured Backups
  - Encrypted Backups
- No Monitoring
  - Security Monitoring, Users, Objects
- Poorly Coded Applications
  - Database Firewall
Attack Vectors and Targets for Databases

Cloud/OS/DB Admin  Test/Dev  Hackers  AppUser  HelpDesk

Risk Multipliers
High Availability
Database Consolidation
Legacy Applications
Outsourcing
Cloud

Applications  DB  OS  Backup  Network
Database Malicious Actions

• Information Disclosure: Obtain credit card and other personal information
  – Defense: Encryption – Data and Network, Tighter Access Controls

• Denial of Service: Run resource intensive queries
  – Defense: Resource Usage Limits – Set various limits – Max Connections, Sessions, Timeouts, ...

• Elevation of Privilege: Retrieve and use administrator credentials
  – Defense: Stronger authentication, Access Controls, Auditing

• Spoofing: Retrieve and use other credentials
  – Defense: Stronger account and password policies

• Tampering: Change data in the database, Delete transaction records
  • Defense: Tighter Access Controls, Auditing, Monitoring, Backups
Use most limited Grants

• For users and privileges be specific
• It's more work – but far better security
• Don’t go crazy with wildcards esp for the priv level
• Avoid GRANT priv_type ON *.*
• Use options like - SSL on GRANTS, MAX_<op>_PER_HOUR, if data is esp. sensitive.
• http://dev.mysql.com/doc/refman/5.6/en/grant.html
MySQL Password Policies

• Accounts without Passwords
  – Assign passwords to all accounts to prevent unauthorized use

• Password Validation Plugin
  – Enforce Strong Passwords

• Password Expiration/Rotation
  – Require users to reset their password

• Account lockout (in v. 5.7)

• Password Retry Rules (in v. 5.7.16+)
Forced Password Expiration

• As in admin if you need to force a user to reset their password, but not set one for them (or set a temp password)

• And Enterprise you may want to force this (and verify via Auditing)


• Note: end user will need to use a client application that supports the password reset.
MySQL Security Overview

- MySQL
- Linux / LDAP
- Windows AD
- Custom

- SSL/TLS
- Public/Private Key
- Transparent Encryption
- Key Management

- Privilege Management Administration
- Database & Objects
- Proxy Users

- Login and Query Activities
- Block Threats
- Auditing
- Regulatory Compliance
- Firewall & Auditing

Security
Regulatory Compliance

• Regulations
  – PCI – DSS: Payment Card Data
  – HIPAA: Privacy of Health Data
  – Sarbanes Oxley, GLBA, The USA Patriot Act:
    Financial Data, NPI "personally identifiable financial information"
  – FERPA – Student Data
  – EU General Data Protection Directive: Protection of Personal Data (GDPR)
  – Data Protection Act (UK): Protection of Personal Data

• Requirements
  – Continuous Monitoring (Users, Schema, Backups, etc)
  – Data Protection (Encryption, Privilege Management, etc.)
  – Data Retention (Backups, User Activity, etc.)
  – Data Auditing (User activity, etc.)
MySQL Authorization

- Administrative Privileges
- Database Privileges
- Session Limits and Object Privileges
- Fine grained controls over user privileges
MySQL Authentication

• Built in Authentication
  – user table stores users and encrypted passwords

• External Authentication with MySQL Enterprise Authentication
  – Microsoft Active Directory
  – Linux PAMs (Pluggable Authentication Modules)
    • Support LDAP and more

• X.509
  – Server authenticates client via certificates

• MySQL Native, SHA 256 Password plugin
  – Native uses SHA1 or plugin with SHA-256 hashing and per user salting for user account passwords.
MySQL Enterprise Audit

• Out-of-the-box logging of connections, logins, and query
• Simple to fine grained policies for filtering, and log rotation
• Dynamically enabled, disabled: no server restart
• XML-based audit stream
  – Send data to a remote server / audit data vault
    • Oracle Audit Vault
    • Splunk, etc.

Adds regulatory compliance to MySQL applications (HIPAA, Sarbanes-Oxley, PCI, etc.)
Logging For Audit

• Proper logging is always a requirement for security.
  – FIPS, HIPAA, PCI-DSS, SOX, DISA STIG, …

• MySQL built-in logging infrastructure:
  – general log, error log, NDB logs.

• MySQL Audit plugin:
  – Granularity made for auditing.
  – Can be modified live.
  – Contains additional details.
  – Compatible with Oracle Audit Vault
MySQL Enterprise Audit - Work Flow

1. DBA Enables Audit Plugin
   - Defines Filters and Options
   - Who, What, Where, When, How

2. User Connects from a Host
   - Authenticates
   - Runs Queries
   - Alters Tables, etc.

3. DBA Reviews Local Audit Events
   - MySQL Workbench EE
   - Or other XML file viewer

4. IT Sec Archives to Audit Vault
   - Globally Assesses Audit Trail
MySQL Enterprise Encryption

• MySQL encryption functions
  – Symmetric encryption AES256 (All Editions)
  – Public-key / asymmetric cryptography – RSA

• Key management functions
  – Generate public and private keys
  – Key exchange methods: DH

• Sign and verify data functions
  – Cryptographic hashing for digital signing, verification, & validation – RSA, DSA
MySQL Enterprise Encryption
Encryption/Decryption within MySQL

Sensitive Data

Encryption
Public Key

Encrypted Data

Decryption
Private Key

Sensitive Data

Private / Public Key Pairs
- Generate using MySQL Enterprise Encryption Functions
- Use externally generated (e.g. OpenSSL)
MySQL Enterprise Transparent Data Encryption

• Improves Security
  – Added Layer– enforces access controls
  – Simple to use and manage

• Meets Security and Regulatory Requirements
  – Fit for cases where encryption is required
    • Healthcare, FiServ, Government, etc.

• Secures and Manages Keys
  – Supports Standard KMIP 1.2 protocols
  – Supports Oracle Key Vault and other Key Stores
Database **Firewall**

- **SQL Injection Attacks**
  - #1 Web Application Vulnerability
  - 77% of Web Sites had vulnerabilities

- **MySQL Enterprise Firewall**
  - Monitor database statements in real-time
  - Automatic White List “rules” generation for any application
  - Block SQL Injection Attacks
  - Intrusion Detection System
MySQL Enterprise Firewall

• Block SQL Injection Attacks
  – Allow: SQL Statements that match Whitelist
  – Block: SQL statements that are not on Whitelist

• Intrusion Detection System
  – Detect: SQL statements that are not on Whitelist
    • SQL Statements execute and alert administrators

```sql
- Select * from employee where id=22
- Select * from employee where id=22 or 1=1
```

- Allow
- Block

Applications → White List → Rule → Detect & Alert Intrusion Detection → MySQL
MySQL Enterprise Monitor

- Enforce MySQL Security Best Practices
- Monitoring & Alerting
- Configuration Management
- Centralized User Management

<table>
<thead>
<tr>
<th>Security Item</th>
<th>Info</th>
<th>Coverage</th>
<th>Schedule</th>
<th>Event Handling</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Account Has An Overly Broad Host Specifier</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>5m</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- Account Has Global Privileges</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>5m</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- Account Has Old Insecure Password Hash</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>6h</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- Account Has Strong MySQL Privileges</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>5m</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- Account Requires Unavailable Authentication Plugins</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>6h</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>- Insecure Password Authentication Option Is Enabled</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>6h</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- Insecure Password Generation Option Is Enabled</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>6h</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- LOCAL Option Of LOAD DATA Statement Is Enabled</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>5m</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- Non-Authorized User Has DB, Table, Or Index Privileges On All Databases</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>5m</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- Non-Authorized User Has GRANT Privileges On All Databases</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>5m</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- Non-Authorized User Has Server Admin Privileges</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>5m</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- Policy-Based Password Validation Does Not Perform Dictionary Checks</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>5m</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- Policy-Based Password Validation Is Weak</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>5m</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- Policy-Based Password Validation Not Enabled</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>5m</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- Privilege Attentions Detected: Privileges Granted</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>5m</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- Privilege Attentions Detected: Privileges Revoked</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>5m</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- Privilege Attentions Have Been Detected</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>5m</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- Root Account Can Login Remotely</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>5m</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- Root Account Without Password</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>5m</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- SHA-256 Password Authentication Not Enabled</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>5m</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>- Server Contains Default &quot;test&quot; Database</td>
<td>✔️</td>
<td>100% (103/103)</td>
<td>5m</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

Problem Description
When users create weak passwords (e.g., "password" or "short") it compromises the security of the server, making it easier for unauthorized people to guess the password and gain access to the server. Starting with MySQL Server 5.6, MySQL offers the "validate_password" plugin that can be used to test passwords and improve security. With this plugin you can implement and enforce a policy for password strength (e.g., passwords must be at least 8 characters long, have both lowercase and uppercase letters, and contain at least one special non-alphanumeric character).

Links and Further Reading
- MySQL Manual: The Password Validation Plugin
- MySQL Manual: Keeping Passwords Secure
- Blog: New 5.6 password verification plugin (and impacts to PASS/WORD()) function
- Blog: Implementing a password policy in MySQL

Expression
%{status} &%{validate_password政策} &%THRESHOLD
Oracle Enterprise Manager for MySQL

- Availability monitoring
- Performance monitoring
- Configuration monitoring
- All available metrics collected
  - Allowing for custom threshold based incident reports
- MySQL auto-detection
MySQL Enterprise Backup

• Online Backup for InnoDB (scriptable interface)
• Full, Incremental, Partial Backups (with compression)
• Strong Encryption (AES 256)
• Point in Time, Full, Partial Recovery options
• Metadata on status, progress, history
• Scales – High Performance/Unlimited Database Size
• Windows, Linux, Unix
• Certified with Oracle Secure Backup, NetBackup, Tivoli, others
Index

• MySQL Enterprise Security
• MySQL Enterprise Authentication
• MySQL Enterprise Firewall
• MySQL Enterprise Transparent Data Encryption
• MySQL Enterprise Audit
Thank You