MySQL Database Security

Michael Messina
Senior Managing Consultant, Rolta-AdvizeX
mmessina@advizex.com / mike.messina@rolta.com

Introduction

- Michael Messina
- Senior Managing Consultant Rolta AdvizeX,
- Working with Oracle Approximately 25 years, MySQL about 12 years.
- Background includes Performance Tuning, High Availability and Disaster Recovery
- MySQL Cloud Service 2018 Implementation Specialist
- Oracle Database OCP
- Oracle RAC Certified Expert
- Oracle Exadata Implementation Specialist
- Oracle ACE
- MMESSINA@ADVIZEX.COM / MIKE.MESSINA@ROLTA.COM
- www.advizex.com

ROLTA ADVIZEX

Agenda

- MySQL History
- Why MySQL
- Security
- Data Security
- MySQL Database Hardening Best Practices
- User Accounts/Roles
- User Locking
- Password Management
- Password Validation
- Password Rotation
- Connection Encryption
- Connection-Control Plugin

ROLTA ADVIZEX

Agenda

- Transparent Data Encryption (TDE)
- Auditing
- MySQL Database Firewall
- Data Masking
- Security Management / Monitoring
- Questions/Discussion

ROLTA ADVIZEX

MySQL History

- MySQL was created by a Swedish company, MySQL AB
- First release on 05/23/1995, making a 30 year History for MySQL
- Version 3.20 released January 1997
- Windows Version of MySQL Release January 1998
- Version 3.21 production release in 1998
- Version 4.0 Beta release August 2002
- Version 4.0 Production release March 2003
  - Unions
- Version 5.0 Beta release March 2005
- Version 5.0 Production release October 2005
  - cursors, stored procedures, triggers, views, XA transactions

MySQL History

- Sun Acquires MySQL AB in 2008
- Version 5.1 Production Release November 2008
  - event scheduler, partitioning, row-based replication, server log to tables
- Oracle Acquires Sun Microsystems January 2010
- MySQL Server 5.5 December 2010
  - Unicode character sets utf16, utf32, and utf8mb4
  - New Partitioning options
- Version 5.6 February 2013
  - full-text search for innodb
  - NoSQL memcache
- Version 5.7 October 2015
  - Group Replication

ROLTA ADVIZEX

Why MySQL

- Long History of a Solid Scalable Database (Proven over 30 year)
- Great Proven Performance
- Worlds Most Popular Open Source Database
- Open Source and run Community Editions for Free
- Large Database Company Oracle more then 20,000 developers
- Enterprise Options for Support by Oracle at very low cost
- Rich tool set to manage and monitor available
- Enterprise Version has Database Firewall a security option from Oracle that only the Oracle Database can complete with no other open source database has this.
- Large Support Community

ROLTA ADVIZEX

Why MySQL

- Run in Cloud / Virtualized / Dedicated Hosts
  - Cloud Infrastructure as a Service – IAAS
  - Cloud Database as a Service – DAAS
  - Oracle Cloud is much like IAAS and Enterprise Edition
- Full and Incremental Database Backup Capability
  - Enterprise Edition MySQL Enterprise Backup
  - Community Edition 3rd party Backup tools
- Point in Time Recovery
- Highly Scalable
  - Many scalability options
- Can Be Made Highly Available
  - Many Options to Provide Redundancy for availability based on needs

ROLTA ADVIZEX

Security

- Average cost of a data breach is $7.35 million,
- $225 per stolen record
- Damage to victims (our customers and clients), brand (our business reputation), and business (revenue hit)
- HIPAA
- GDPR
- SOX
- The List goes On

ROLTA ADVIZEX

Security

- Insider threat often more overlooked then outside threat so need to ensure focus on security includes the more likely insider threat to data breach. (ie. Separation of Duties)
- Physical Security typically has had more focus then data security though that has been changing.
- Data Security is getting more focus as more breaches are highlighted in Media as they happen.
- Regulatory Compliance must be adhered to for Health Care and Financial information some critical business data is left unprotected leaving organization exposed to that data being stolen by competitors and organization may not even realize it.

ROLTA ADVIZEX

Protect Our Organizations Data

4 Pillars of Data Protection

- Assessment
  - Investigate your Risks, know you data and where it is
  - Discover what needs protecting so you can take action
  - Oracle Database Security Assessment Tools can help with this
- Goverance
  - Policy and procedures
- Training
  - Make sure your staff have the knowledge and training to protect data
- Response
  - Data Breaches
  - Loss of data
  - Take proper action
  - I say this means be proactive, monitor and consistently assess

Source: https://securitythinkingcap.com/four-pillars-of-data-protection-for-the-modern-enterprise/

ROLTA ADVIZEX

Data Security

- Data becoming the most valuable assess for an organization
  - Focus on protecting data becoming important
    - Regulatory Compliance
    - Data Exposure in the News
    - Lawsuits
    - Company/Organization Reputation
  - Forbes - Data: Your Most Ignored And Valuable Asset
    - https://www.forbes.com/sites/forbesagencycouncil/2018/02/12/data-your-most-ignored-and-valuable-asset/#694f3b70715b
- Protect the Data
  - Prevent Unauthorized Access to data
  - Detect Invalid Access to data
  - Continual Security Improvements for Data Protection
  - Continual Security Improvements for Data Access Detection
  - Compliance Reporting
  - E-Discovery

ROLTA ADVIZEX

MySQL Database Hardening Best Practices

- Strong Passwords
- Remove all Anonymous users from database
  - drop user “”@”localhost”;
- Least Privilege
  - Only grant Required permissions eliminate any permission not absolutely required
  - Utilized Roles to standardize permissions based on requirements
- Encrypt All Client to Database Network Traffic
- Encrypt Database at Rest (Transparent Data Encryption)
- Monitor User Access and Data Access and Log it and review for abnormal patterns and/or policy violations
- Limit Access to Database Servers to essential personnel only and limit file system privileges for all database related files.
- Maintain server and database administrative control otherwise highly privileged database admin users can access your data!

ROLTA ADVIZEX

User Accounts/Roles

- User Accounts
  - Designed to make sure only authorized users get in.
  - Designed to make sure Authorized Users only see what they are allowed to see to control access to database objects/data
- Enterprise Authentication
  - Integrate with Central Authentication Service
    - Active Directory
    - Linux PAM / Centrify
      - authentication_pam.so
      - Externally Identified accounts
    - LDAP Native Authentication
      - authentication_ldap_sasl.so
      - authentication_ldap_sasl_server_host=127.0.0.1
      - authentication_ldap_sasl_bind_base_dn="dc=example,dc=com"
      - plugin-load-add=authentication_ldap_simple.so
      - authentication_ldap_simple_server_host=127.0.0.1
      - authentication_ldap_simple_bind_base_dn="dc=example,dc=com"

ROLTA ADVIZEX

User Accounts/Roles

- Roles
  - Implements roles to group privileges
  - Grant roles to accounts to standardize privileges
  - Control privileges to many uses through the privileges of roles
  - Allows modification of privileges for groups of users with a single change to a role
  - Allows groups of users to have standard/common privileges based on job role or data access level requirements.
  - Allows security to standardize privileges and quickly validate access as well as access outside standard.

ROLTA ADVIZEX

User Locking

- Lock Account
  - ALTER USER ‘user_name'@'localhost' ACCOUNT LOCK;
- UnLock Account
  - ALTER USER ‘user_name'@'localhost' ACCOUNT UNLOCK;
- Example when account is locked:
  - ERROR 3118 (HY000): Access denied for user 'user_name'@'localhost'. Account is locked.
  - This is not the same as password Expired:
    - ERROR 1820 (HY000): You must reset your password using ALTER USER statement before executing this statement.

Password Management

- Password Expiration Policy
  - Password expiration, to require passwords to be changed periodically.
  - Manual: ALTER USER username@'localhost' PASSWORD EXPIRE ;
  - Auto: MySQL parameter default_password_lifetime
  - ERROR 1820 (HY000): You must reset your password using ALTER USER statement before executing this statement.
- Password reuse restrictions, to prevent old passwords from being chosen again.
  - Password Validation
- Password verification, to require that password changes also specify the current password to be replaced.
  - Password Validation
- Password strength assessment, to require strong passwords.
  - Password Validation

ROLTA ADVIZEX

Password Validation

- Plugin < MySQL 8 Configuration
  - default_password_lifetime=90
  - disconnect_on_expired_password=ON
  - validate_password_check_user_name=ON
  - validate_password_dictionary_file
  - validate_password_length=8
  - validate_password_mixed_case_count=1
  - validate_password_number_count=1
  - validate_password_policy=MEDIUM
  - validate_password_special_char_count=1

Password Validation

- Component MySQL 8 and above Configuration
  - default_password_lifetime=90
  - disconnect_on_expired_password=ON
  - password_history=3
  - password_require_current=OFF
  - password_reuse_interval=365
  - validate_password.check_user_name=ON
  - validate_password.dictionary_file
  - validate_password.length=8
  - validate_password.mixed_case_count=1
  - validate_password.number_count=1
  - validate_password.policy=MEDIUM
  - validate_password.special_char_count=1

Password Validation (Examples)

- To prohibit reusing any of the last 6 passwords or passwords newer than 365 days
  - password_history=6
  - password_reuse_interval=365
- To Ensure that current password is required to change password
  - password_require_current=ON
- Ensure Password must have 8 characters, 2 special characters, at least 1 upper and 1 lowercase character and at least 1 number
  - validate_password.length=8
  - validate_password.mixed_case_count=1
  - validate_password.number_count=1
  - validate_password.special_char_count=2
- Make Users Change their Password every 90 days
  - default_password_lifetime=90

ROLTA ADVIZEX

Password Rotation

- **Dual Password Support**
- As of MySQL 8.0.14, a user account is permitted to have dual passwords
- primary and secondary passwords.
- Makes it possible to seamlessly perform credential changes rotate in new passwords for accounts
- Example:
  - A system has a large number of MySQL servers with replication.
  - Multiple applications connect to different MySQL servers.
  - Periodic credential changes must be made to the account or accounts used by the applications to connect to mysql servers.
  - Using dual passwords existing application servers passwords will continue to work with a password change until those application servers can be updated.

Connection Encryption

- Encrypt the traffic between the database and clients accessing the data
- Uses keys from database sever
- OpenSSL
- Server Side Configuration
  - [mysqld] section of database parameter file (my.cnf)
  - ssl_ca = /opt/mysql/data/data/ca.pem
  - ssl_cert = /opt/mysql/data/data/client-cert.pem
  - ssl_key = /opt/mysql/data/data/client-key.pem
- Force Connections to utilize encryption
  - require_secure_transport=ON
  - Default is OFF

ROLTA ADVIZEX

Connection Encryption

- Client Side Configuration
  - --ssl-mode=REQUIRED
    - Default is preferred, by default will attempt a secure connection
    - Require will enforce that connection be secure
  - --ssl-mode=PREFERRED
    - Will attempt and Prefer a secure connection, but not require it
    - This is the default mode
  - --ssl-mode=VERIFY_CA
    - Requires secure connection and verifies identity of server hostname
  - --ssl-mode=VERIFY_IDENTITY
    - Does not work with self signed certificates
    - Self Signed Certificates are created by the mysql server or mysql_ssl_rsa_setup
  - Parameters for Keys
    - ssl_ca = /opt/mysql/data/data/ca.pem
    - ssl_cert = /opt/mysql/data/data/client-cert.pem
    - ssl_key = /opt/mysql/data/data/client-key.pem

Connection-Control Plugin

- Focus on Denial of Service Attacks
- Controls and denies connections when connection has failed past threshold.
- Install
  - INSTALL PLUGIN CONNECTION_CONTROL SONAME 'connection_control.so';
  - INSTALL PLUGIN CONNECTION_CONTROL_FAILED_LOGIN_ATTEMPTS SONAME 'connection_control.so';
- Configure
  - connection_control_failed_connections_threshold=3
    - After 3 failed attempts disallow connection for specified delay
    - For each 3 attempts after the connection delay escalates to max connection delay
  - connection_control_max_connection_delay=2147483647
  - connection_control_min_connection_delay=1000

ROLTA ADVIZEX

Transparent Data Encryption (TDE)

- Encryption at Rest with AES algorithm
  - Database/Table Level
  - Database Backups
  - Protection is transparent to applications accessing the database
  - High Performance – Little to No Noticeable Performance Impact
  - Two-tier encryption
    - Master key
    - Tablespace keys (table)
    - Keys can be managed in Oracle Key Vault, Thales Vormetric, AWS KMS, others
    - Encryption Key can also be file based on the local file system with database
  - Compliance HIPAA – PCI DSS
  - No Downtime Required to Implement

ROLTA ADVIZEX

Transparent Data Encryption

- Local File System TDE Key
  - Not intended for a regulatory compliance solution
  - Better then nothing when key ring standard service does not exist
  - keyring_file.so
  - keyring_file_data – Directory for key file
- Example Local File System
  - early_plugin_load=keyring_file.so
  - keyring_file_data=/opt/mysql/tde/keyring
- KMIP-compatible product as a back end for keyring storage
  - Oracle Key Vault, Gemalto KeySecure, Thales Vormetric
  - keyring_okv.so (AES Supported Type)
  - keyring_okv_conf_dir - directory location with required files
    - okvclient.ora - details of the KMIP back end
    - ssl - directory contains the certificate and key files required to establish a secure connection with the KMIP back end: CA.pem, cert.pem, and key.pem

ROLTA ADVIZEX

Transparent Data Encryption

- Example KMIP
  - early-plugin-load=keyring_okv.so
  - keyring_okv_conf_dir=/opt/mysql/tde
- Communicates with the Amazon Web Services Key Management Service
  - keyring_aws (AES Type)
  - keyring_aws_conf_file
  - must obtain a secret access key that provides credentials for communicating with AWS KMS and write it to a configuration file
- Example AWS KMS
  - early-plugin-load=keyring_aws.so keyring_aws_cmk_id='arn:aws:kms:us-west-2:111122223333:key/abcd1234-ef56-ab12-cd34-ef56abcd1234’
  - keyring_aws_conf_file=/usr/local/mysql/mysql-keyring/keyring_aws_conf
  - keyring_aws_data_file=/usr/local/mysql/mysql-keyring/keyring_aws_data

Transparent Data Encryption - Backup

- MySQL Enterprise Backup
- File Based TDE for Database
  - Must Utilize –encrypt-password option with password for keyring encrypted file
  - Does not decrypt files encrypted tablespace files are copied into backup
  - When the database uses encrypted InnoDB tables, MySQL Enterprise Backup always stores the master key for encryption in an encrypted file inside the backup, irrespective of the kind of keyring plugin the server uses.
-

Transparent Data Encryption - Backup

- Keyring plugin TDE other than file for Database
  - accesses the keyring to obtain the master key and uses it to decrypt the encrypted tablespace keys
  - master key is then put into a keyring data file, named keyring_kef and saved in the meta folder in the backup
  - The date files are encrypted with the user password supplied with the option --encrypt-password
- Example:
  - ${MYSQL_BASE}/meb/bin/mysqlbackup --user=${USR} --password=${PWD} --encrypt-password=${TDEPWD} --compress –backup-dir=$BACKUP_DIR/$TODAY

ROLTA ADVIZEX

Auditing

- Track the Changes to the database
- Track Database Data Access
- E-Discovery for Regulatory Compliance
  - GDPR, PCI DSS, HIPAA, HITECH, SOX, etc.
- Bring Audit Trail into Qradar, Oracle Audit Vault, etc.
- Detect and Report on any possible unauthorized access to data
- Written to database file system in specified format
- Format options
  - OLD - Old XML Format Legacy
  - NEW - New XML Format (default)
  - JSON

ROLTA ADVIZEX

Auditing

- Configuration
  - audit_log=FORCE_PLUS_PERMANENT
  - audit_log_file=/opt/mysql/audit/audit_log.log
  - audit_log_format=NEW
  - audit_log_policy=ALL
  - audit_log_strategy=ASYNCHRONOUS
  - audit_log_statement_policy=ALL
  - audit_log_buffer_size=1048576
  - audit_log_rotate_on_size=5242880
  - audit_log_connection_policy=ALL
- Additional Configuration Options
  - audit_log_compression
    - Gzip the audit log files after switching to new audit log
  - audit_log_encryption
    - Encrypt the audit log with AES-256-CBC cipher encryption

ROLTA ADVIZEX

Auditing (Example)


```
<AUDIT_RECORD>
 <TIMESTAMP>2019-01-11T23:58:44 UTC</TIMESTAMP>
 <RECORD_ID>46386613_2018-11-16T01:31:29</RECORD_ID>
 <NAME>Query</NAME>
 <CONNECTION_ID>1843438</CONNECTION_ID>
 <STATUS>0</STATUS>
 <STATUS_CODE>0</STATUS_CODE>
 <USER>mem_admin[mem_admin] @ servername.domain.net [xx.xx.xxx.xx]</USER>
 <OS_LOGIN/>
 <HOST>servername.domain.net</HOST>
 <IP>xx.xx.xxx.xx</IP>
 <COMMAND_CLASS>show_variables</COMMAND_CLASS>
 <SQLTEXT>SHOW VARIABLES LIKE 'log_bin'</SQLTEXT>
</AUDIT_RECORD>
```


ROLTA ADVIZEX

MySQL Database Firewall

- Provides protection for the database without any changes to application
- Modes
  - Allow All SQL on “Whitelist”
  - Block Block SQL not on “Whitelist”
  - Detect Record SQL and notify of SQL not on “Whitelist”
- Realtime Threat Protection
  - Can Block SQL not on the “Whitelist”
    - Block SQL Injection Attacks
  - Can Record and SQL Not on the “Whitelist”
    - Know SQL that is being executed that was not a known SQL
    - Know when SQL is being Executed that does not fall in policy
- Database Intrusion Detection
  - Get Notified when SQL not on Whitelist is being submitted for execution
- Automatically learn and build “Whitelist” of SQL
- Tracks Allowed and Block SQL for Analysis and Reporting

ROLTA ADVIZEX

MySQL Database Firewall

- Install MySQL Database Firewall Plugin
  - Linux
    - linux_install_firewall.sql
  - Windows
    - win_install_firewall.sql
  - Example:
    - mysql -u root -p mysql < linux_install_firewall.sql
- Turn on MySQL Database Firewall
  - my.cnf parameter -> mysql_firewall_mode=ON
  - my.cnf parameter -> mysql_firewall_trace=ON
  - Can be set dynamically SET GLOBAL mysql_firewall_mode=ON
- Generate WhiteList by setting recording
  - CALL mysql.sp_set_firewall_mode('fwuser@localhost', 'RECORDING’);
- After you have a good white list goto protecting
  - CALL mysql.sp_set_firewall_mode('fwuser@localhost', 'PROTECTING');

ROLTA ADVIZEX

MySQL Database Firewall

- Message Back to Client that Attempts to Execute un-approved statement
  - ERROR 1045 (28000): Statement was blocked by Firewall
- To set detection mode where where firewall will record un-approved statements but no block them
  - CALL mysql.sp_set_firewall_mode('fwuser@localhost', 'DETECTING');
- Important Status Variables (Show global status like 'Firewall%')
  - Firewall_access_denied
    - Total statements rejected by MySQL Database Firewall.
  - Firewall_access_granted
    - Total statements accepted by MySQL Database Firewall.
  - Firewall_access_suspicious
  - Total statements recorded by MySQL Enterprise Firewall as suspicious for users who are in DETECTING mode. (ie. SQL Not on Whitelist)

ROLTA ADVIZEX

Data Masking

- Plugin data_masking
  - INSTALL PLUGIN data_masking SONAME 'data_masking.so';
  - Then execute create function statements for installation.
- Mask and De-Identify Data
- Great for Moving Production Data to Non-Production Environments
- Data Privacy Mandates
  - GDPR, PCI DSS, HIPAA, HITECH, SOX, etc.
- Transform
  - Remove any characteristics of the data the identify it
  - Example transform SSN to XXX-XXX-3334 where it leaves the last 4 digits of SSN but transforms the rest thus masking it.
- Generate
  - Generate random data for email addresses, names, credit card numbers, etc.
  - Use characteristics or dictionary for generation of data

ROLTA ADVIZEX

Data Masking

- mask_inner()
  - Argument 1: String to Mask
  - Argument 2: Position to start Masking at
  - Argument 3: How Many characters to not mask at end of string
  - Argument 4: Character to Mask With default is X
  - Example:

```
SELECT mask_inner('This is a string', 1, 5);
+---------------------------------------+
| mask_inner('This is a string', 1, 5) |
+---------------------------------------+
| TXXXXXXXXXXtring                      |
+---------------------------------------+
```

ROLTA ADVIZEX

Data Masking

- mask_outer()
  - Argument 1: String to Mask
  - Argument 2: Position to start Masking from the End of String
  - Argument 3: How Many characters to not mask at beginning of string
  - Argument 4: Character to Mask With default is X
  - Example:

```
SELECT mask_outer('This is a string', 1, 5);
+---------------------------------------+
| mask_outer('This is a string', 1, 5) |
+---------------------------------------+
| Xhis is a sXXXXX                      |
+---------------------------------------+
```

ROLTA ADVIZEX

Data Masking

- mask_pan
  - Mask all but last 4 digits of a number
  - Argument 1: Number to be masked
- mask_pan_relaxed
  - Does not mask first 6 digits
  - Argument 1: Number for masking
- mask_ssn
  - Mask Social Security Number
  - Format XXX-XXX-3334
  - Argument 1: Socal Security Number to Mask

Data Masking

- gen_range()
  - Generate a randome number within range provided
  - Argument 1: First number in Range
  - Argument 2: Second Number in Range
- gen_rnd_email()
  - Generate a random email address
- gen_ran_pan()
  - Generates a random payment card number
- gen_ran_ssn()
  - Generates a Random Social Security Number
- gen_ran_us_phone()
  - Generates a random U.S Phone Number

ROLTA ADVIZEX

Security Management / Monitoring

- Security is about diligence, discipline and standardization
  - Not Sexy New program development
  - Behind the wall protecting
- Assessment
  - Know where you data is
  - Know the tools and capabilities you have to protect it
  - Constantly Monitor Data Access Activities and alert to abnormalities
  - Utilize Automation to your advantage here
- Governance
  - Adhere to standards
  - Have policies and procedures that focus on protecting the data
  - Communicate policies and procedures to everyone and ensure everyone adheres to them

ROLTA ADVIZEX

Security Management / Monitoring

- Training
  - Ensure you stay up to date on latest tools and capabilities that exist to protect your data
  - Ensure you continue to train an update staff on latest threats
  - Ensure you train your staff on the latest tools and capabilities to protect the data
- Response
  - The best laid plans and even the most secure environments may be come a victim to exposure be ready
  - Ensure you have the data you need to see what was exposed
  - Ensure you have the tools and capabilities to close the gap
  - Ensure you are diligent in you monitoring and assessment to catch any exposures quickly so response is fast and gaps closure swift to limit damage.

ROLTA ADVIZEX

Take Action

- We all Must do our part to secure our data for the benefit of our organization, but also our customers who are counting on us to protect their information. We have a responsibility.

Credits

- MySQL Documentation, Oracle Corporation

ROLTA ADVIZEX

Thank You!

Any Further Questions?

ROLTA ADVIZEX