

All About Encryption in MySQL



Jericho Rivera

Database Engineer

November 2, 2024



Hanoi, Vietnam 2024

PERCONA
UNIVERSITY

Open Source Databases Meetup

In partnership with



Who Am I

Jericho Rivera

Database Engineer

Percona Support

APAC



Coverage

- Architecture
- Keyring Components and Plugins
- Data at Rest Encryption
- Block Device Encryption
- Application-level Encryption
- Data-in-transit Encryption

Architecture

MySQL uses a two-tier architecture for data at-rest encryption with the following components

- Master encryption key – used to encrypt and decrypt tablespace keys
- Tablespace key for each tablespace – encrypts the tablespace pages and written in the tablespace header

<https://www.percona.com/blog/mysql-encryption-talking-about-keyrings/>

<https://www.percona.com/blog/mysql-encryption-master-key-encryption-in-innodb/>

MySQL Keyring

Keyring Components

- component_keyring_file
- component_keyring_encrypted_file*
- component_keyring_vault**
- component_keyring_oci*
- component_keyring_kmip**
- component_keyring_kms**

* MySQL Enterprise only feature

** Percona Server 8.4

Keyring Plugins

- keyring_file
- keyring_encrypted_file*
- keyring_vault
- keyring_okv*
- keyring_aws*
- keyring_hashicorp*
- keyring_oci

Keyring File plugin

```
mysql> select * from information_schema.plugins where plugin_name like 'keyring_file'\G
***** 1. row *****
      PLUGIN_NAME: keyring_file
     PLUGIN_VERSION: 1.0
    PLUGIN_STATUS: ACTIVE
    PLUGIN_TYPE: KEYRING
  PLUGIN_TYPE_VERSION: 1.1
    PLUGIN_LIBRARY: keyring_file.so
PLUGIN_LIBRARY_VERSION: 1.11
    PLUGIN_AUTHOR: Oracle Corporation
  PLUGIN_DESCRIPTION: store/fetch authentication data to/from a flat file
    PLUGIN_LICENSE: GPL
      LOAD_OPTION: ON
1 row in set (0.01 sec)

mysql> show global variables like 'keyring_file_data';
+-----+-----+
| Variable_name | Value                               |
+-----+-----+
| keyring_file_data | /var/lib/mysql-keyring/keyring |
+-----+-----+
1 row in set (0.01 sec)
```

```
early_plugin_load="keyring_file=keyring_file.so"
keyring-file-data=/var/lib/mysql-keyring/keyring
```

Keyring File component

```
[root@ps80 ~]# cat /sbin/mysqld.my
{
  "components": "file://component_keyring_file"
}
[root@ps80 ~]# cat /usr/lib64/mysql/plugin/component_keyring_file.cnf
{
  "path": "/var/lib/mysql-keyring/keyring_file", "read_only": false
}
[root@ps80 ~]# mysql -e "select * from performance_schema.keyring_component_status"
+-----+-----+
| STATUS_KEY          | STATUS_VALUE          |
+-----+-----+
| Component_name      | component_keyring_file |
| Author              | Oracle Corporation    |
| License             | GPL                   |
| Implementation_name | component_keyring_file |
| Version             | 1.0                   |
| Component_status    | Active                |
| Data_file           | /var/lib/mysql-keyring/keyring_file |
| Read_only           | No                    |
+-----+-----+
```

MySQL Data-at-Rest Encryption

Data-at-Rest encryption – the encryption or decryption of data as it is written into or read from disk.

Transparent data encryption – the use of encryption to protect data on the disk but does not affect application interactions with the database.

- Supports the Advanced Encryption Standard (AES) block-based encryption algorithm.
- Uses the Electronic Codebook (ECB) block encryption mode for tablespace key encryption.
- Cipher Block Chaining (CBC) block encryption mode for data encryption.

<https://dev.mysql.com/doc/refman/8.0/en/innodb-data-encryption.html#innodb-data-encryption-limitations>

MySQL Data-at-Rest Encryption

With INNODB transparent data encryption, we can choose which files will be encrypted:

- **File-per-Table tablespace**
 - CREATE/ALTER TABLE ... ENCRYPTION='Y';
- **General tablespace** – Worklog #9286
 - CREATE/ALTER TABLESPACE ... ENCRYPTION='Y';
- **mysql System tablespace** – Worklog #12063
 - ALTER TABLESPACE mysql ENCRYPTION='Y';
- **innodb_parallel_dblwr_encrypt** in 5.7 and <8.0.23 – Worklog #13775
- **innodb_redo_log_encrypt** – Worklog #9290
- **innodb_undo_log_encrypt** – Worklog #9289
- **Temporary File** – *innodb_temp_tablespace_encrypt (encrypt-tmp-files)*

Default encryption setting for schemas and general tablespaces:

- default_table_encryption (8.0.16)

MySQL Data-at-Rest Encryption

Binary log and Relay log file encryption

- **encrypt_binlog** = 1 | 0 (PS 5.7.20-19 to PS 8.0.14)
- **binlog_encryption** = ON | OFF (PS 8.0.15-5)
- **binlog_rotate_encryption_master_key_at_startup**
- <https://dev.mysql.com/worklog/task/?id=10957>
- the binary log master key is stored in the keyring
- the file password is stored in the log's file header
- the file password is encrypted using AES-CBC

MySQL Data-at-Rest Encryption

Audit Log Filter Encryption support in Percona Server v8.0

Requires OpenSSL v3.x

To enable this feature, the audit log filter needs to be installed:

```
# mysql < /usr/share/percona-server/audit_log_filter_linux_install.sql
```

- `audit_log_filter_encryption = NONE | AES`
- `audit_log_encryption_password_set()`
- `audit_log_encryption_password_get()`

To decrypt the encrypted audit log file:

```
openssl enc -d -aes-256-cbc -pass pass:password
```

```
-iter iterations -md sha256
```

```
-in audit.timestamp.log.pwd_id.enc
```

```
-out audit.timestamp.log
```

Block Device Encryption

- Linux block device
 - dm-crypt – a Linux kernel-level encryption tool
 - Linux Unified Key Setup (LUKS) – a generic key store
- Windows block device – Bitlocker encryption
- Cloud platforms' disk encryption
 - Amazon Web Services
 - EBS encryption
 - RDS Storage Encrypted database
 - Google Cloud Platform
 - default encryption at rest
 - Azure Cloud
 - Azure disk storage server-side encryption
 - Encryption at host
 - Azure disk encryption
 - Confidential disk encryption

Application-level Encryption

- Percona Server for MySQL 8.0/8.4

- component_encryption_udf

```
mysql> INSTALL COMPONENT 'file://component_encryption_udf';
```

```
mysql> select create_digest('md5','Hello!');
```

```
mysql> select create_asymmetric_priv_key('RSA', 3072);
```

```
mysql> SELECT AES_ENCRYPT('mytext','mykeystring', '', 'hkdf', 'salt', 'info');
```

- MySQL Enterprise Encryption

- <https://dev.mysql.com/doc/refman/8.0/en/mysql-enterprise-encryption.html>

Data-in-transit Encryption

- Added support to TLSv1.3 since v8.0.16
- Removed support to TLSv1.1 since v8.0.28
- Support for Weak ciphers in v8.4+
 - Conforms to proper TLS version (TLS v1.2 or TLSv1.3, as appropriate)
 - Provides perfect forward secrecy
 - Uses SHA2 in cipher, certificate, or both
 - Uses AES in GCM or any other AEAD algorithms or modes
- Added `--ssl-session-data[-continue-on-failed-reuse]` in v8.0.29
- Added `--tls-ciphersuites` in v8.0.16



Thank you!

<https://www.linkedin.com/in/riveraja/>

In partnership with

