MySQL Encryption

Encryption Types in MySQL

1. **Data-at-Rest Encryption**:
   - Tablespace Encryption: Encrypts the entire tablespace, including the InnoDB tables.
   - Column-Level Encryption: Encrypts specific columns in a table.
2. **Data-in-Transit Encryption**:
   - SSL/TLS Encryption: Encrypts data transmitted between MySQL clients and servers using SSL/TLS.

==========

Data-at-Rest Encryption

> MyISAM does not support encryption natively. Tables will need to be converted to InnoDB before encryption is implemented.

1. Enable the Keyring Plugin

The keyring plugin is used to store and manage encryption keys securely within MySQL.

Install the Keyring Plugin

- If you are using MySQL 5.7 or later, the keyring_file plugin is included by default.
- Add the following lines to the MySQL configuration file

```
[mysqld]
early-plugin-load = keyring_file.so
keyring_file_data = /var/lib/mysql-keyring/keyring
```

Create the directory for the keyring file if it doesn't exist

```
sudo mkdir /var/lib/mysql-keyring
sudo chown mysql:mysql /var/lib/mysql-keyring
```

Restart the MySQL server to load the plugin

Verify the Keyring Plugin is Enabled:

```
SHOW PLUGINS;
```

2. Enable InnoDB Tablespace Encryption

Enable InnoDB File-Per-Table:

Ensure that `innodb_file_per_table` is enabled, which is the default setting in MySQL 5.6 and later.

```
[mysqld]
innodb_file_per_table = 1
```

Enable InnoDB Encryption:

```
[mysqld]
innodb_encrypt_tables = ON
innodb_encrypt_log = ON
innodb_encryption_threads = 4
```

Restart MySQL.

```
systemctl restart mysql
```

3. Encrypt Existing Tables

Encrypt a Specific Table

```
ALTER TABLE mytable ENCRYPTION='Y';
```

4. Verify Encryption

Check Encryption Status

You can verify if a table is encrypted by querying the `information_schema.tables` table:

```
SELECT table_schema, table_name, create_options
FROM information_schema.tables
WHERE create_options LIKE '%ENCRYPTION="Y"%';
```

Binary Log Encryption

You can replay unencrypted binary logs onto encrypted tables.

Enable Binary Log Encryption

Add the following configuration to your `my.cnf`

```
[mysqld]
binlog_encryption = ON
```

Verify Binary Log Encryption

```
SHOW VARIABLES LIKE 'binlog_encryption';
```

Replaying encrypted binary logs

Replaying encrypted binary logs involves ensuring that the encrypted logs are decrypted and applied correctly on the MySQL server.

Use mysqlbinlog to Read Encrypted Binary Logs

The `mysqlbinlog` utility will handle the decryption transparently if the keyring plugin is properly configured.

```
mysqlbinlog /path/to/binlog.000001 | mysql -u username -p
```

The process for replaying binary logs whether encrypted or not is largely the same, providing that the keyring plugin is enabled. See here for more info.

==========

Data-in-Transit Encryption

Data-in-transit encryption refers to the protection of data as it moves between systems, such as between a client and a server, or between servers. This type of encryption ensures that data remains confidential and integral during transmission, preventing unauthorized access and tampering.

Data-in-transit encryption typically uses Transport Layer Security (TLS) or its predecessor, Secure Sockets Layer (SSL), to secure communications.

Implementing Data-in-Transit Encryption in MySQL

1. Generate or Obtain Certificates and Keys:

Generate self-signed certificates using OpenSSL or obtain them from a trusted CA. Certificate needs to cover the MySQL server hostname.

Example command to generate a self-signed certificate using OpenSSL:

```
openssl req -newkey rsa:2048 -nodes -keyout server-key.pem -x509 -days 365 -out server-cert.pem
```

2. Configure MySQL Server

Edit the MySQL configuration file (`my.cnf` or `my.ini`) to include the paths to the certificates and keys.

```
[mysqld]
ssl-ca = /path/to/ca-cert.pem
ssl-cert = /path/to/server-cert.pem
ssl-key = /path/to/server-key.pem
```

> To force the use of SSL when connecting to a MySQL server, you can add the below to the my.cnf:
> `require_secure_transport = ON`

3. Restart MySQL Server

```
systemctl restart mysql
```

4. Verify SSL/TLS Configuration

Verify that SSL/TLS is enabled on the server.

```
SHOW VARIABLES LIKE '%ssl%';
```

==========

Revision #4
Created 2024-06-30 13:49:56 UTC by Daniel
Updated 2024-06-30 14:24:41 UTC by Daniel