How to Configure SSL in MySQL Server

In this article, we can have a look important aspects of configuring and managing SSL in **MySQL** Server. These would include the default configuration, disabling SSL, and enabling and enforcing SSL on a MySQL server.

Our observations are based on the community version of MySQL 5.7.25 on Ubuntu 16.04 Version.

```
mysql> status
--------------
mysql  Ver 14.14 Distrib 5.7.25, for Linux (x86_64) using  EditLine wrapper

Connection id:          3
Current database:
Current user:           root@localhost
SSL:                    Not in use
Current pager:          stdout
Using outfile:          ''
Using delimiter:        ;
Server version:         5.7.25-0ubuntu0.16.04.2-log (Ubuntu)
Protocol version:       10
Connection:             Localhost via UNIX socket
Server characterset:    latin1
Db     characterset:    latin1
Client characterset:    utf8
Conn.  characterset:    utf8
UNIX socket:            /var/run/mysqld/mysqld.sock
Uptime:                 34 sec

Threads: 1  Questions: 5  Slow queries: 0  Opens: 110  Flush tables: 1  Open
tab
les: 26  Queries per second avg: 0.147
--------------


mysql> show variables like 'have_ssl';
+---------------+----------+
| Variable_name | Value    |
+---------------+----------+
| have_ssl      | DISABLED |
+---------------+----------+
1 row in set (0.01 sec)

mysql>

root@ubuntu6-30pm:~# tail -f /var/log/mysql/error.log
2019-03-31T05:31:03.396503Z 0 [Note] InnoDB: Buffer pool(s) load completed at
190331 11:01:03
2019-03-31T05:31:03.406918Z 0 [Warning] Failed to set up SSL because of the
following SSL library error: SSL context is not usable without certificate
and private key
```

```
2019-03-31T05:31:03.406961Z 0 [Note] Server hostname (bind-address):
'127.0.0.1'; port: 3310
2019-03-31T05:31:03.406978Z 0 [Note]   - '127.0.0.1' resolves to '127.0.0.1';
2019-03-31T05:31:03.407024Z 0 [Note] Server socket created on IP:
'127.0.0.1'.
2019-03-31T05:31:04.232381Z 0 [Note] Failed to start slave threads for
channel ''
2019-03-31T05:31:08.213244Z 0 [Note] Event Scheduler: Loaded 0 events
2019-03-31T05:31:08.213660Z 0 [Note] /usr/sbin/mysqld: ready for connections.
Version: '5.7.25-0ubuntu0.16.04.2-log'  socket: '/var/run/mysqld/mysqld.sock'
port: 3310  (Ubuntu)
```

```
mysql> SHOW VARIABLES LIKE '%ssl%';
+---------------+----------+
| Variable_name | Value    |
+---------------+----------+
| have_openssl  | DISABLED |
| have_ssl      | DISABLED |
| ssl_ca        |          |
| ssl_capath    |          |
| ssl_cert      |          |
| ssl_cipher    |          |
| ssl_crl       |          |
| ssl_crlpath   |          |
| ssl_key       |          |
+---------------+----------+
9 rows in set (0.01 sec)
```

Generating SSL/TLS Certificates and Keys

To enable SSL connections to MySQL, we first need to generate the appropriate certificate and key files. A utility called mysql_ssl_rsa_setup is provided with MySQL 5.7 and above to simplify this process.

Ubuntu 16.04 has a compatible version of MySQL, so we can use this command to generate the necessary files.

The files will be created in MySQL's data directory, located at /var/lib/mysql. We need the MySQL process to be able to read the generated files, so we will pass mysql as the user that should own the generated files:

```
root@ubuntu6-30pm:~# mysql_ssl_rsa_setup --uid=mysql
Generating a 2048 bit RSA private key
.....................................................................
.....................................................................
.....................................+++
.....................................+++
writing new private key to 'ca-key.pem'
-----
Generating a 2048 bit RSA private key
.....................................+++
```

```
.....................................................................
.........+++
writing new private key to 'server-key.pem'
-----
Generating a 2048 bit RSA private key
.....................................................................+
++
.....................................................+++
writing new private key to 'client-key.pem'
```

```
root@ubuntu6-30pm:~# find /var/lib/mysql -name '*.pem' -ls
  1190130      4 -rw-------   1 mysql    mysql    1679 Mar 31 11:25
/var/lib/mysql/client-key.pem
  1189131      4 -rw-------   1 mysql    mysql    1679 Mar 31 11:25
/var/lib/mysql/server-key.pem
  1183407      4 -rw-------   1 mysql    mysql    1679 Mar 31 11:25
/var/lib/mysql/ca-key.pem
  1187171      4 -rw-r--r--   1 mysql    mysql    1107 Mar 31 11:25
/var/lib/mysql/ca.pem
  1190734      4 -rw-r--r--   1 mysql    mysql    1107 Mar 31 11:25
/var/lib/mysql/client-cert.pem
  1183231      4 -rw-r--r--   1 mysql    mysql     451 Mar 31 11:25
/var/lib/mysql/public_key.pem
  1190081      4 -rw-r--r--   1 mysql    mysql    1107 Mar 31 11:25
/var/lib/mysql/server-cert.pem
  1183043      4 -rw-------   1 mysql    mysql    1679 Mar 31 11:25
var/lib/mysql/private_key.pem
root@ubuntu6-30pm:~#
```

The last column shows the generated files. The central columns that show "mysql" indicate that the generated files have the correct user and group ownership.

These files are the key and certificate pairs for the certificate authority (starting with "ca"), the MySQL server process (starting with "server"), and for MySQL clients (starting with "client"). Additionally, the private_key.pem and public_key.pem files are used by MySQL to securely transfer password when not using SSL.

Enable SSL Connections on the MySQL Server

MySQL versions will look for the appropriate certificate files within the MySQL data directory when the server starts. Because of this, we don't actually need to modify the MySQL configuration to enable SSL.

We can just restart the MySQL Server :-

```
root@ubuntu6-30pm:~# service mysql restart
```

```
mysql> show variables like 'have_ssl';
+---------------+-------+
| Variable_name | Value |
+---------------+-------+
| have_ssl      | YES   |
+---------------+-------+
1 row in set (0.01 sec)

mysql> SHOW VARIABLES LIKE '%ssl%';
+---------------+-----------------+
| Variable_name | Value           |
+---------------+-----------------+
| have_openssl  | YES             |
| have_ssl      | YES             |
| ssl_ca        | ca.pem          |
| ssl_capath    |                 |
| ssl_cert      | server-cert.pem |
| ssl_cipher    |                 |
| ssl_crl       |                 |
| ssl_crlpath   |                 |
| ssl_key       | server-key.pem  |
+---------------+-----------------+
9 rows in set (0.00 sec)



root@ubuntu6-30pm:/var/log/mysql# tail error.log
2019-03-31T06:05:00.041731Z 0 [Note] Found ca.pem, server-cert.pem and
server-key.pem in data directory. Trying to enable SSL support using them.
2019-03-31T06:05:00.042463Z 0 [Warning] CA certificate ca.pem is self signed.
2019-03-31T06:05:00.045564Z 0 [Note] Server hostname (bind-address):
'127.0.0.1'; port: 3310
2019-03-31T06:05:00.045618Z 0 [Note]   - '127.0.0.1' resolves to '127.0.0.1';


mysql> SHOW SESSION STATUS LIKE '%Ssl_version%';
+---------------+-------+
| Variable_name | Value |
+---------------+-------+
| Ssl_version   |       |
+---------------+-------+
1 row in set (0.01 sec)

mysql> CREATE USER 'ssluser'@'localhost' IDENTIFIED BY 'pwd123';
Query OK, 0 rows affected (0.00 sec)

mysql> GRANT USAGE ON *.* TO 'ssluser'@'localhost' REQUIRE ssl;
Query OK, 0 rows affected, 1 warning (0.01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.01 sec)
```

Old versions we have to connect with --ssl

```
root@ubuntu6-30pm:/etc/mysql/mysql.conf.d# mysql -u ssluser -p --ssl
WARNING: --ssl is deprecated and will be removed in a future version. Use --
ssl-mode instead.
```

```
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 5.7.25-0ubuntu0.16.04.2-log (Ubuntu)

Copyright (c) 2000, 2019, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.

You are enforcing ssl conection via unix socket. Please consider
switching ssl off as it does not make connection via unix socket
any more secure.
mysql> SHOW SESSION STATUS LIKE '%Ssl_version%';
+---------------+---------+
| Variable_name | Value   |
+---------------+---------+
| Ssl_version   | TLSv1.1 |
+---------------+---------+
1 row in set (0.00 sec)


root@ubuntu6-30pm# mysql -u ssluser -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 5.7.25-0ubuntu0.16.04.2-log (Ubuntu)

Copyright (c) 2000, 2019, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.

You are enforcing ssl conection via unix socket. Please consider
switching ssl off as it does not make connection via unix socket
any more secure.
mysql> SHOW SESSION STATUS LIKE '%Ssl_version%';
+---------------+---------+
| Variable_name | Value   |
+---------------+---------+
| Ssl_version   | TLSv1.1 |
+---------------+---------+
1 row in set (0.01 sec)

Connected with root user, you can verify below as
mysql -u root -ppassword

mysql> status
--------------
```

```
mysql  Ver 14.14 Distrib 5.7.25, for Linux (x86_64) using  EditLine wrapper

Connection id:          10
Current database:
Current user:           root@localhost
SSL:                    Cipher in use is DHE-RSA-AES256-SHA
Current pager:          stdout
Using outfile:          ''
Using delimiter:        ;
Server version:         5.7.25-0ubuntu0.16.04.2-log (Ubuntu)
Protocol version:       10
Connection:             Localhost via UNIX socket
Server characterset:    latin1
Db     characterset:    latin1
Client characterset:    utf8
Conn.  characterset:    utf8
UNIX socket:            /var/run/mysqld/mysqld.sock
Uptime:                 8 min 46 sec

Threads: 2  Questions: 21  Slow queries: 0  Opens: 121  Flush tables: 1  Open
ta
bles: 37  Queries per second avg: 0.039
```

Prepared by Sarrath Chandh
www.dbacentre.com
dbacentre.7@gmail.com