**6.3.1 Configuring MySQL to Use Encrypted Connections**

Several configuration parameters are available to indicate whether to use encrypted connections, and to specify the appropriate certificate and key files. This section provides general guidance about configuring the server and clients for encrypted connections:

* [Server-Side Startup Configuration for Encrypted Connections](https://dev.mysql.com/doc/refman/5.7/en/using-encrypted-connections.html#using-encrypted-connections-server-side-startup-configuration)
* [Client-Side Configuration for Encrypted Connections](https://dev.mysql.com/doc/refman/5.7/en/using-encrypted-connections.html#using-encrypted-connections-client-side-configuration)
* [Configuring Encrypted Connections as Mandatory](https://dev.mysql.com/doc/refman/5.7/en/using-encrypted-connections.html#mandatory-encrypted-connections)

Encrypted connections also can be used in these contexts:

* Between master and slave replication servers. See [Section 16.3.8, “Setting Up Replication to Use Encrypted Connections”](https://dev.mysql.com/doc/refman/5.7/en/replication-solutions-encrypted-connections.html).
* Among Group Replication servers. See [Section 17.5.2, “Group Replication Secure Socket Layer (SSL) Support”](https://dev.mysql.com/doc/refman/5.7/en/group-replication-secure-socket-layer-support-ssl.html).
* By client programs that are based on the MySQL C API. See [Section 27.7.14, “C API Support for Encrypted Connections”](https://dev.mysql.com/doc/refman/5.7/en/c-api-encrypted-connections.html).

Instructions for creating any required certificate and key files are available in [Section 6.3.3, “Creating SSL and RSA Certificates and Keys”](https://dev.mysql.com/doc/refman/5.7/en/creating-ssl-rsa-files.html).

**Server-Side Startup Configuration for Encrypted Connections**

On the server side, the [--ssl](https://dev.mysql.com/doc/refman/5.7/en/server-options.html#option_mysqld_ssl) option specifies that the server permits but does not require encrypted connections. This option is enabled by default, so it need not be specified explicitly.

To require that clients connect using encrypted connections, enable the [require\_secure\_transport](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_require_secure_transport) system variable. See [Configuring Encrypted Connections as Mandatory](https://dev.mysql.com/doc/refman/5.7/en/using-encrypted-connections.html#mandatory-encrypted-connections).

These system variables on the server side specify the certificate and key files the server uses when permitting clients to establish encrypted connections:

* [ssl\_ca](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_ssl_ca): The path name of the Certificate Authority (CA) certificate file. ([ssl\_capath](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_ssl_capath) is similar but specifies the path name of a directory of CA certificate files.)
* [ssl\_cert](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_ssl_cert): The path name of the server public key certificate file. This certificate can be sent to the client and authenticated against the CA certificate that it has.
* [ssl\_key](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_ssl_key): The path name of the server private key file.

For example, to enable the server for encrypted connections, start it with these lines in the my.cnf file, changing the file names as necessary:

[mysqld]

ssl\_ca=ca.pem

ssl\_cert=server-cert.pem

ssl\_key=server-key.pem

To specify in addition that clients are required to use encrypted connections, enable the [require\_secure\_transport](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_require_secure_transport) system variable:

[mysqld]

ssl\_ca=ca.pem

ssl\_cert=server-cert.pem

ssl\_key=server-key.pem

require\_secure\_transport=ON

Each certificate and key system variable names a file in PEM format. Should you need to create the required certificate and key files, see [Section 6.3.3, “Creating SSL and RSA Certificates and Keys”](https://dev.mysql.com/doc/refman/5.7/en/creating-ssl-rsa-files.html). MySQL servers compiled using OpenSSL can generate missing certificate and key files automatically at startup. See [Section 6.3.3.1, “Creating SSL and RSA Certificates and Keys using MySQL”](https://dev.mysql.com/doc/refman/5.7/en/creating-ssl-rsa-files-using-mysql.html). Alternatively, if you have a MySQL source distribution, you can test your setup using the demonstration certificate and key files in its mysql-test/std\_data directory.

The server performs certificate and key file autodiscovery. If no explicit encrypted-connection options are given other than [--ssl](https://dev.mysql.com/doc/refman/5.7/en/server-options.html#option_mysqld_ssl) (possibly along with [ssl\_cipher](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_ssl_cipher)) to configure encrypted connections, the server attempts to enable encrypted-connection support automatically at startup:

* If the server discovers valid certificate and key files named ca.pem, server-cert.pem, and server-key.pem in the data directory, it enables support for encrypted connections by clients. (The files need not have been generated automatically; what matters is that they have those names and are valid.)
* If the server does not find valid certificate and key files in the data directory, it continues executing but without support for encrypted connections.

If the server automatically enables encrypted connection support, it writes a note to the error log. If the server discovers that the CA certificate is self-signed, it writes a warning to the error log. (The certificate is self-signed if created automatically by the server or manually using [**mysql\_ssl\_rsa\_setup**](https://dev.mysql.com/doc/refman/5.7/en/mysql-ssl-rsa-setup.html).)

MySQL also provides these system variables for server-side encrypted-connection control:

* [ssl\_cipher](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_ssl_cipher): The list of permissible ciphers for connection encryption.
* [ssl\_crl](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_ssl_crl): The path name of the file containing certificate revocation lists. ([ssl\_crlpath](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_ssl_crlpath) is similar but specifies the path name of a directory of certificate revocation-list files.)
* [tls\_version](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_tls_version): Which encryption protocols the server permits for encrypted connections; see [Section 6.3.2, “Encrypted Connection TLS Protocols and Ciphers”](https://dev.mysql.com/doc/refman/5.7/en/encrypted-connection-protocols-ciphers.html). For example, you can set [tls\_version](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_tls_version) to prevent clients from using less-secure protocols.

**Client-Side Configuration for Encrypted Connections**

For a complete list of client options related to establishment of encrypted connections, see [Command Options for Encrypted Connections](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#encrypted-connection-options).

By default, MySQL client programs attempt to establish an encrypted connection if the server supports encrypted connections, with further control available through the [--ssl-mode](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-mode) option:

* In the absence of an [--ssl-mode](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-mode) option, clients attempt to connect using encryption, falling back to an unencrypted connection if an encrypted connection cannot be established. This is also the behavior with an explicit [--ssl-mode=PREFFERED](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-mode) option.
* With [--ssl-mode=REQUIRED](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-mode), clients require an encrypted connection and fail if one cannot be established.
* With [--ssl-mode=DISABLED](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-mode), clients use an unencrypted connection.
* With [--ssl-mode=VERIFY\_CA](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-mode) or [--ssl-mode=VERIFY\_IDENTITY](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-mode), clients require an encrypted connection, and also perform verification against the server CA certificate and (with VERIFY\_IDENTITY) against the server host name in its certificate.

Attempts to establish an unencrypted connection fail if the [require\_secure\_transport](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_require_secure_transport) system variable is enabled on the server side to cause the server to require encrypted connections. See [Configuring Encrypted Connections as Mandatory](https://dev.mysql.com/doc/refman/5.7/en/using-encrypted-connections.html#mandatory-encrypted-connections).

The following options on the client side identify the certificate and key files clients use when establishing encrypted connections to the server. They are similar to the [ssl\_ca](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_ssl_ca), [ssl\_cert](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_ssl_cert), and [ssl\_key](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_ssl_key) system variables used on the server side, but [--ssl-cert](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-cert) and [--ssl-key](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-key) identify the client public and private key:

* [--ssl-ca](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-ca): The path name of the Certificate Authority (CA) certificate file. This option, if used, must specify the same certificate used by the server. (--ssl-capath is similar but specifies the path name of a directory of CA certificate files.)
* [--ssl-cert](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-cert): The path name of the client public key certificate file.
* [--ssl-key](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-key): The path name of the client private key file.

For additional security relative to that provided by the default encryption, clients can supply a CA certificate matching the one used by the server and enable host name identity verification. In this way, the server and client place their trust in the same CA certificate and the client verifies that the host to which it connected is the one intended:

* To specify the CA certificate, use [--ssl-ca](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-ca) (or [--ssl-capath](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-capath)), and specify [--ssl-mode=VERIFY\_CA](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-mode).
* To enable host name identity verification as well, use [--ssl-mode=VERIFY\_IDENTITY](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-mode) rather than [--ssl-mode=VERIFY\_CA](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-mode).

Note

Host name identity verification with VERIFY\_IDENTITY does not work with self-signed certificates that are created automatically by the server or manually using [**mysql\_ssl\_rsa\_setup**](https://dev.mysql.com/doc/refman/5.7/en/mysql-ssl-rsa-setup.html) (see [Section 6.3.3.1, “Creating SSL and RSA Certificates and Keys using MySQL”](https://dev.mysql.com/doc/refman/5.7/en/creating-ssl-rsa-files-using-mysql.html)). Such self-signed certificates do not contain the server name as the Common Name value.

Host name identity verification also does not work with certificates that specify the Common Name using wildcards because that name is compared verbatim to the server name.

MySQL also provides these options for client-side SSL control:

* [--ssl-cipher](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-cipher): The list of permissible ciphers for connection encryption.
* [--ssl-crl](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-crl): The path name of the file containing certificate revocation lists. (--ssl-crlpath is similar but specifies the path name of a directory of certificate revocation-list files.)
* [--tls-version](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_tls-version): The permitted encryption protocols; see [Section 6.3.2, “Encrypted Connection TLS Protocols and Ciphers”](https://dev.mysql.com/doc/refman/5.7/en/encrypted-connection-protocols-ciphers.html).

Depending on the encryption requirements of the MySQL account used by a client, the client may be required to specify certain options to connect using encryption to the MySQL server.

Suppose that you want to connect using an account that has no special encryption requirements or that was created using a [CREATE USER](https://dev.mysql.com/doc/refman/5.7/en/create-user.html) statement that included the REQUIRE SSL clause. Assuming that the server supports encrypted connections, a client can connect using encryption with no [--ssl-mode](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-mode) option or with an explicit [--ssl-mode=PREFFERED](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-mode) option:

mysql

Or:

mysql --ssl-mode=PREFERRED

For an account created with a REQUIRE SSL clause, the connection attempt fails if an encrypted connection cannot be established. For an account with no special encryption requirements, the attempt falls back to an unencrypted connection if an encrypted connection cannot be established. To prevent fallback and fail if an encrypted connection cannot be obtained, connect like this:

mysql --ssl-mode=REQUIRED

If the account has more stringent security requirements, other options must be specified to establish an encrypted connection:

* For accounts created with a REQUIRE X509 clause, clients must specify at least [--ssl-cert](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-cert) and [--ssl-key](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-key). In addition, [--ssl-ca](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-ca) (or [--ssl-capath](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-capath)) is recommended so that the public certificate provided by the server can be verified. For example (enter the command on a single line):
* mysql --ssl-ca=ca.pem
* --ssl-cert=client-cert.pem

--ssl-key=client-key.pem

* For accounts created with a REQUIRE ISSUER or REQUIRE SUBJECT clause, the encryption requirements are the same as for REQUIRE X509, but the certificate must match the issue or subject, respectively, specified in the account definition.

For additional information about the REQUIRE clause, see [Section 13.7.1.2, “CREATE USER Statement”](https://dev.mysql.com/doc/refman/5.7/en/create-user.html).

To prevent use of encryption and override other --ssl-*xxx* options, invoke the client program with [--ssl-mode=DISABLED](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-mode):

mysql --ssl-mode=DISABLED

To determine whether the current connection with the server uses encryption, check the session value of the [Ssl\_cipher](https://dev.mysql.com/doc/refman/5.7/en/server-status-variables.html#statvar_Ssl_cipher) status variable. If the value is empty, the connection is not encrypted. Otherwise, the connection is encrypted and the value indicates the encryption cipher. For example:

mysql> SHOW SESSION STATUS LIKE 'Ssl\_cipher';

+---------------+---------------------------+

| Variable\_name | Value |

+---------------+---------------------------+

| Ssl\_cipher | DHE-RSA-AES128-GCM-SHA256 |

+---------------+---------------------------+

For the [**mysql**](https://dev.mysql.com/doc/refman/5.7/en/mysql.html) client, an alternative is to use the STATUS or \s command and check the SSL line:

mysql> \s

...

SSL: Not in use

...

Or:

mysql> \s

...

SSL: Cipher in use is DHE-RSA-AES128-GCM-SHA256

...

**Configuring Encrypted Connections as Mandatory**

For some MySQL deployments it may be not only desirable but mandatory to use encrypted connections (for example, to satisfy regulatory requirements). This section discusses configuration settings that enable you to do this. These levels of control are available:

* You can configure the server to require that clients connect using encrypted connections.
* You can invoke individual client programs to require an encrypted connection, even if the server permits but does not require encryption.
* You can configure individual MySQL accounts to be usable only over encrypted connections.

To require that clients connect using encrypted connections, enable the [require\_secure\_transport](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_require_secure_transport) system variable. For example, put these lines in the server my.cnf file:

[mysqld]

require\_secure\_transport=ON

With [require\_secure\_transport](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_require_secure_transport) enabled, client connections to the server are required to use some form of secure transport, and the server permits only TCP/IP connections that use SSL, or connections that use a socket file (on Unix) or shared memory (on Windows). The server rejects nonsecure connection attempts, which fail with an [ER\_SECURE\_TRANSPORT\_REQUIRED](https://dev.mysql.com/doc/refman/5.7/en/server-error-reference.html#error_er_secure_transport_required) error.

To invoke a client program such that it requires an encrypted connection whether or not the server requires encryption, use an [--ssl-mode](https://dev.mysql.com/doc/refman/5.7/en/connection-options.html#option_general_ssl-mode) option value of REQUIRED, VERIFY\_CA, or VERIFY\_IDENTITY. For example:

mysql --ssl-mode=REQUIRED

mysqldump --ssl-mode=VERIFY\_CA

mysqladmin --ssl-mode=VERIFY\_IDENTITY

To configure a MySQL account to be usable only over encrypted connections, include a REQUIRE clause in the [CREATE USER](https://dev.mysql.com/doc/refman/5.7/en/create-user.html) statement that creates the account, specifying in that clause the encryption characteristics you require. For example, to require an encrypted connection and the use of a valid X.509 certificate, use REQUIRE X509:

CREATE USER 'jeffrey'@'localhost' REQUIRE X509;

For additional information about the REQUIRE clause, see [Section 13.7.1.2, “CREATE USER Statement”](https://dev.mysql.com/doc/refman/5.7/en/create-user.html).

To modify existing accounts that have no encryption requirements, use the [ALTER USER](https://dev.mysql.com/doc/refman/5.7/en/alter-user.html) statement.