Cloud SQL (https://cloud.google.com/sql/)

Documentation (https://cloud.google.com/sql/docs/)

MySQL (https://cloud.google.com/sql/docs/mysql/) Guides

Connecting to Cloud SQL from external applications

MySQL | PostgreSQL (https://cloud.google.com/sql/docs/postgres/connect-external-app) | SQL Server

This page describes how to establish a connection to Cloud SQL from an application running outside of Google Cloud.

For information about the various options for connecting to Cloud SQL, see Connection options for external applications (https://cloud.google.com/sql/docs/mysql/external-connection-methods). For information about configuring public IP, see Configuring public IP (https://cloud.google.com/sql/docs/mysql/configure-ip).

> **Note:** If you are connecting from within Google Cloud, use the instructions for your service:
>
> - Connecting from App Engine (https://cloud.google.com/sql/docs/mysql/connect-app-engine)
>
> - Connecting from Compute Engine (https://cloud.google.com/sql/docs/compute-engine-access)
>
> - Connecting from Google Kubernetes Engine (https://cloud.google.com/sql/docs/mysql/connect-container-engine)

Database connections consume resources on the server and the connecting application. Always use good connection management practices to minimize your application's footprint and reduce the likelihood of exceeding Cloud SQL connection limits (https://cloud.google.com/sql/docs/mysql/quotas#fixed-limits). For more information, see Managing database connections (https://cloud.google.com/sql/docs/mysql/manage-connections).

Before you start

Granting access to an application does not automatically enable a database user account to connect to the instance. Before you can connect to an instance, you must have a database user

account you can connect with. For new instances, this means you must have configured the
default user account. Learn more (https://cloud.google.com/sql/docs/mysql/create-manage-users).

You can connect to a Cloud SQL instance using the following methods:

- By using the proxy (#proxy) (Second Generation instances only)

- By configuring access for one or more public IP addresses (#appaccessIP)

- By using the JDBC Socket Factory (#java) (for the Java programming language, Second
  Generation instances only)

- By using the Cloud SQL Proxy library (#go) (for the Go programming language, Second
  Generation instances only)

Connecting from an external application using the proxy

If you are setting up the Cloud SQL Proxy for a local test environment (not for production), you
can use the Proxy Quickstart (https://cloud.google.com/sql/docs/mysql/quickstart-proxy-test) instead
of these instructions.

If you are using the Java or Go programming languages, you have some alternatives to using
the Cloud SQL Proxy. Learn more (#languages).

Note: The Cloud SQL Proxy is available only for Second Generation instances.

1. Enable the API

Enable the Cloud SQL Admin API.

ENABLE THE API (HTTPS://CONSOLE.CLOUD.GOOGLE.COM/FLOWS/ENABLEAPI?APIID=SQLADMIN&REDIR

2. Install the proxy client on your local machine

LINUX 64-BIT	LINUX 32-BIT	MORE ▾

1. Download the proxy:

```
wget https://dl.google.com/cloudsql/cloud_sql_proxy.linux.amd64 -O cloud_sql_
```

2. Make the proxy executable:

```
chmod +x cloud_sql_proxy
```

If your operating system isn't included here, you can also compile the proxy from source
 (http://github.com/GoogleCloudPlatform/cloudsql-proxy).

3. Determine how you will authenticate the proxy

Learn more about proxy authentication options
 (https://cloud.google.com/sql/docs/mysql/sql-proxy#authentication-options).

4. If required by your authentication method, create a service account

> **Note:** To create a service account with the required permissions, you must have
> `resourcemanager.projects.setIamPolicy` permission. This permission is included in the Project Owner,
> Project IAM Admin, and Organization Administrator roles.
> You must also have enabled the Cloud SQL Admin API.

1. Go to the **Service accounts** page of the Google Cloud Console.

   GO TO THE SERVICE ACCOUNTS PAGE (HTTPS://CONSOLE.CLOUD.GOOGLE.COM/IAM-ADMIN/SERV

2. Select the project that contains your Cloud SQL instance.

3. Click **Create service account**.

4. In the **Create service account** dialog, provide a descriptive name for the service account.

5. For **Role**, select one of the following roles:

   - **Cloud SQL > Cloud SQL Client**

   - **Cloud SQL > Cloud SQL Editor**

   - **Cloud SQL > Cloud SQL Admin**

   Alternatively, you can use the primitive Editor role by selecting **Project > Editor**, but the
   Editor role includes permissions across Google Cloud.

If you do not see these roles, your Google Cloud user might not have the
`resourcemanager.projects.setIamPolicy` permission. You can check your permissions
by going to the IAM page (https://console.cloud.google.com/iam-admin) in the Google Cloud
Console and searching for your user id.

6. Change the **Service account ID** to a unique, easily recognizable value.

7. Click **Furnish a new private key** and confirm that the key type is `JSON`.

8. Click **Create**.

   The private key file is downloaded to your machine. You can move it to another location.
   Keep the key file secure.

5. Determine how you will specify your instances for the proxy

Learn more about proxy instance specification options
 (https://cloud.google.com/sql/docs/mysql/sql-proxy#instances-options).

6. Start the proxy

The options you pass to the proxy depend on the authentication and instance specification
options you chose previously.

Depending on your language and environment, you can start the proxy using either TCP sockets
or Unix sockets.

TCP SOCKETS   UNIX SOCKETS

1. Copy your instance connection name from the **Instance details** page.

   For example: `myproject:us-central1:myinstance`.

2. If you are using a service account to authenticate the proxy, note the location on your client machine
   of the private key file that was created when you created the service account.

3. Start the proxy.
   Some possible proxy invocation strings:

   - Using Cloud SDK authentication:

   ```
   ./cloud_sql_proxy -instances=<INSTANCE_CONNECTION_NAME>=tcp:3306
   ```

   The specified port must not already be in use, for example, by a local database server.

- Using a service account and explicit instance specification (recommended for production environments):

```
./cloud_sql_proxy -instances=<INSTANCE_CONNECTION_NAME>=tcp:3306 \
                  -credential_file=<PATH_TO_KEY_FILE> &
```
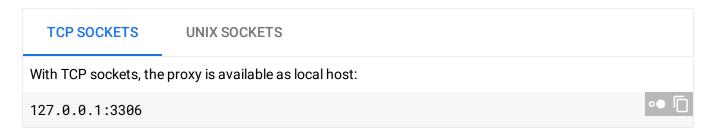
For more information about proxy options, see Options for authenticating the proxy (https://cloud.google.com/sql/docs/mysql/sql-proxy#authentication-options) and Options for specifying instances (https://cloud.google.com/sql/docs/mysql/sql-proxy#instances-options).

7. Update your application to connect to Cloud SQL using the proxy

The exact code statement required to use the proxy to connect to your Cloud SQL instance depends on the language and framework you are using.
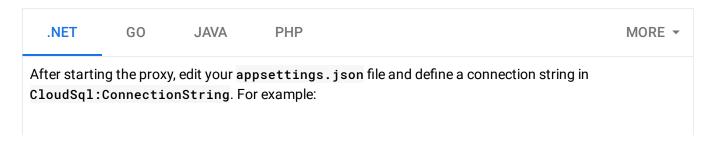
You connect to the proxy the same way you would to a TCP or Unix socket (depending on how you invoked the proxy):

TCP SOCKETS	UNIX SOCKETS

With TCP sockets, the proxy is available as local host:

```
127.0.0.1:3306
```

Need help? For help troubleshooting the proxy, see Troubleshooting Cloud SQL Proxy connections (https://cloud.google.com/sql/docs/mysql/sql-proxy#troubleshooting). Or, see our Cloud SQL Support page (https://cloud.google.com/sql/docs/mysql/support).

Language-specific information and examples

You can connect to the proxy from any language that enables you to connect to a Unix or TCP socket. Below are a few sample proxy invocation and connection statements to help you understand how they work together in your application.

.NET	GO	JAVA	PHP	MORE ▾

After starting the proxy, edit your `appsettings.json` file and define a connection string in `CloudSql:ConnectionString`. For example:

```
{
  "CloudSQL" : {
    ConnectionString": "Host=127.0.0.1;Uid=DATABASE_USER;Pwd=PASSWORD;Database=DAT
  }
}
```

Then, in your `Startup.cs` file, create a database connection:

```
var connectionString = new MySqlConnectionStringBuilder(
    Configuration["CloudSql:ConnectionString"])
{
    // Connecting to a local proxy that does not support ssl.
    SslMode = MySqlSslMode.None,
};
DbConnection connection =
    new MySqlConnection(connectionString.ConnectionString);
```

Configuring access for public IP connections

You can grant any application access to a Cloud SQL instance by underline authorizing the public IP addresses (https://cloud.google.com/sql/docs/mysql/configure-ip#add) that the application uses to connect.

You can not specify a private network (https://en.wikipedia.org/wiki/Private_network) (for example, 10.x.x.x) as an authorized network.

Public IP addresses for Second Generation instances:

- **IPv6:** Second Generation instances do not support IPv6.

- **IPv4:** Second Generation instances have a static IPv4 address automatically assigned. There is a small charge for the IP address any time your instance is off (deactivated).

Public IP addresses for First Generation instances:

Note: Second Generation is replacing First Generation; support for First Generation instances ends January 30, 2020. To upgrade a First Generation instance to Second Generation, see Upgrading a First Generation Instance to Second Generation (https://cloud.google.com/sql/docs/mysql/upgrade-2nd-gen).

For First Generation instances, Cloud SQL supports public IP connections over both IPv4 and IPv6 addresses. You can connect using either protocol, or both. However, you cannot mix the IP address versions in a single connection.

- **IPv6:** Each First Generation instance has a static IPv6 address automatically assigned to it; you do not need to assign an IPv6 address to your instance.

- **IPv4:** If you are connecting to your First Generation instance over IPv4, you must update your instance to have a public IPv4 address assigned to it. This IP address is static. There is a small charge for the IPv4 address any time your instance is off (stopped).

To configure access over IP connections:

CONSOLE   GCLOUD   CURL

---

1. Determine the IP address of your application. <u>Learn more.</u> (#findIP)

2. Authorize your application's IP address to connect to your instance.
   For more information, see <u>Adding an authorized address</u>
   (https://cloud.google.com/sql/docs/mysql/configure-ip#add).

3. If you are connecting to a First Generation instance over IPv4, assign an IPv4 address to the instance if you haven't done so already:

   a. Go to the Cloud SQL Instances page in the Google Cloud Console.

   <u>GO TO THE CLOUD SQL INSTANCES PAGE</u> (HTTPS://CONSOLE.CLOUD.GOOGLE.COM/SQL/IN

   b. Click the instance name to open its **Overview** page.

   c. Select the **IP address** tab.

   d. Click **Request IPv4 address**.

   Note that you are charged for an IPv4 address when the instance is off. For more information, see the <u>pricing</u> (https://cloud.google.com/sql/pricing#instance-ip) page.

4. You can find the IP address assigned to your instance in its **Instance details** page. This is the value you need for your application's connection string.

Need help? For help troubleshooting the proxy, see <u>Troubleshooting Cloud SQL Proxy connections</u> (https://cloud.google.com/sql/docs/mysql/sql-proxy#troubleshooting). Or, see our <u>Cloud SQL Support page</u> (https://cloud.google.com/sql/docs/mysql/support).

Configuring access for applications with dynamically assigned IP addresses

Some applications need to connect to your Cloud SQL instance using a dynamically assigned, or ephemeral, IP address. This is the case for Platform as a Service (Paas) applications, among others.

The best solution for these applications is to use a Cloud SQL Second Generation instance, and connect by using the Cloud SQL Proxy (https://cloud.google.com/sql/docs/mysql/sql-proxy). This solution provides the best access control for your instance.

If you cannot use a Second Generation instance, then you can either install your own proxy, or open up your firewall and apply SSL. But neither of these methods provide the security and control of the connectivity solutions provided by Second Generation instances.

Testing your connection

You can use the mysql client to test your ability to connect from your local environment. For more information, see Connecting the mysql client using IP addresses (https://cloud.google.com/sql/docs/mysql/connect-admin-ip) and Connecting the mysql client using the Cloud SQL Proxy (https://cloud.google.com/sql/docs/mysql/connect-admin-proxy).

Need help? For help troubleshooting the proxy, see Troubleshooting Cloud SQL Proxy connections (https://cloud.google.com/sql/docs/mysql/sql-proxy#troubleshooting). Or, see our Cloud SQL Support page (https://cloud.google.com/sql/docs/mysql/support).

Determining the IP address for your application

To determine the IP address of a computer running your application so you can authorize access to your Cloud SQL instance from that address, use one of the following options:

- If the computer is not behind a proxy, log in to the computer and use this link (http://ipv4.whatismyv6.com/) to determine its IP address.

- If the computer is behind a proxy, log in to the computer and use a tool or service like whatismyipaddress.com (https://www.whatismyip.com/proxy-check/) to determine its true IP address.

Note: If the computer running your application is assigned an IP address that can change over time (it is dynamically assigned), see Configuring access for applications with dynamically assigned IP addresses

(#dynamicIP).

What's next

- Learn about <u>managing database connections in your application</u> (https://cloud.google.com/sql/docs/mysql/manage-connections).

- Learn about <u>options for connecting external applications</u> (https://cloud.google.com/sql/docs/mysql/external-connection-methods).

- Learn about <u>connecting using a mysql client</u> (https://cloud.google.com/sql/docs/mysql/connect-admin-ip).

- Learn about <u>configuring IP connectivity</u> (https://cloud.google.com/sql/docs/mysql/configure-ip) .

- Learn about <u>connecting with other MySQL tools</u> (https://cloud.google.com/sql/docs/mysql/admin-tools).

- Learn about <u>MySQL connectors</u> (http://dev.mysql.com/downloads/connector/).

- Learn about <u>the proxy</u> (https://cloud.google.com/sql/docs/mysql/sql-proxy).

- Learn about <u>options for support</u> (https://cloud.google.com/sql/docs/mysql/support).