

### The Ultimate Web Hosting Package.

From **\$3.95**/mo

[True Hosting Pricing](#)  
Advertised = Renewal

[Sign Up Now](#)

**Lifetime Hosting Option**  
Available for a limited time.

**Free Migration Available**  
We can help you move your email and websites as desired.

- ✓ Host Unlimited Domains
- ✓ FREE SSL Included
- ✓ Unlimited [HTTPS Hosting](#)
- ✓ Unlimited Web Space
- ✓ Unlimited Visitors
- ✓ [WordPress Hosting](#)
- ✓ [LiveCode Hosting](#)
- ✓ [MariaDB Hosting](#)
- ✓ [LiteSpeed Hosting](#)
- ✓ 280+ Plugin Web Apps
- ✓ Instant Activation
- 🔗 [Full List of Features](#)

🏠 / Tutorials & Guides

## Get a LiveCode Desktop or Mobile App to Communicate with a MariaDB/MySQL Database

Your LiveCode desktop or mobile app may need to store data on, or retrieve data from, a MariaDB/MySQL database located on a remote server, such as on your HostM hosting account.

For security reasons, we recommend creating a **HTTPS-based API layer** for this purpose, rather than having your LiveCode desktop or mobile app connect remotely to MariaDB/MySQL via its direct port 3306.

### 100% FREE LiveCode-MariaDB/MySQL HTTPS-Based API Solution/Demo

We've created a basic 100%-LiveCode API solution with full **UTF-8 support** that you can use as a starting point for your projects.

#### System Requirements

On the server side, you'll need a hosting setup running a minimum of **LiveCode Server 6 (8.1.0 or later recommended)**, access to either **MariaDB** or **MySQL**, and an **SSL-secured** domain. HostM offers all of these features, but you may use any hosting setup that meets these requirements.

Remember to have an **SSL certificate** installed on your domain to ensure that all communication is encrypted. If you have a hosting account with HostM, contact us via the Client Lounge to request for the installation of a **FREE SSL certificate** from Let's Encrypt on your hosted domain.

On the client side, you may use a minimum of **LiveCode 6.7.9 through 6.7.11**, or, for built-in Unicode support, **LiveCode 7.1.2 and up (8.1.0 or later recommended)**.

#### Features

The following features are currently available:

- HTTPS POST for secure transmissions
- Built-in authentication via API key
- Integrity verification via SHA-1 checksums
- Full UTF-8 support for multilingual support
- Manage your data via native LiveCode arrays
- Auto-conversion to industry-standard JSON for transmission

The API solution comprises:

- **.lc** server-side script
- **.livecode** server-side sample stack
- **.livecode** server-side library stacks
- **.sql** sample MariaDB/MySQL database
- **.livecode** client-side sample stack
- **.livecode** client-side library stacks

#### Free Download

The latest version of the HTTPS-Based LiveCode-MariaDB/MySQL API solution, which we'll update from time to time, can be downloaded below:

[Download](#) LiveCode-MariaDB/MySQL HTTPS API Solution 1.1.4

#### Instructions

The download consists of two folders: the **client-side** folder contains stacks you run from your desktop, while the **server-side** folder contains files that you need to set up on the server.

##### Setting Up the Test MariaDB/MySQL Database

1. Sign in to your cPanel and click MySQL Databases.
  2. Create a MySQL database named **example\_apitest** (replace "example" with your actual cPanel username).
  3. Create a MySQL user named **example\_apitest** (replace "example" with your actual cPanel username). Make a note of the password, as you'll need it later.
  4. On the same page, under "Add User To Database", assign the newly-created user to the newly-created database, with all privileges checked.
  5. Click the phpMyAdmin icon in your cPanel.
  6. In the left-column, click the **+** sign next to your cPanel username, then click the **name** of your newly-created database.
  7. Click the SQL tab at the top.
  8. In the **server-side** folder from the download, you'll find a file named **users.sql**.
  9. Open that file in a plain text editor, then copy and paste its content into the large field in phpMyAdmin.
  10. Click the Go button. You should see the message "MySQL returned an empty result set (i.e. zero rows)", and there should be no errors. Your demo MariaDB/MySQL database is now ready!
- To further verify that you've added the demo table correctly, click the Structure tab at the top and you should see the **users** table listed. Click Browse and you'll see a default sample entry listed, with test@example.com as its email address.

##### Configuring Your LiveCode Stacks

1. In the **server-side** folder from the download, you'll find a stack named **api-demo-server-mariadb.livecode**.
2. Open that stack in LiveCode, then edit the stack script and modify the database name, user and password to the ones you've just created in cPanel.
3. You'll notice that there's also an API key specified within the stack script. Change it to any strong random string you wish, then copy it, as you'll need to paste it into another stack shortly.
4. In the **client-side** folder from the download, you'll find a stack named **api-demo-client-mariadb.livecode**.
5. Open that stack in LiveCode, then edit the stack script and paste in the API key appropriately.
6. You'll notice that there's also a HTTPS URL specified within the stack script. Change the domain **example.com** to your actual domain name.
7. Edit the scripts of each of the three buttons and change the value of tProfile to match the one you're using in the stack script.
8. Save both stacks, then exit LiveCode.

##### Uploading the Server-Side Stacks

1. Sign in to your hosting account via FTP and create a folder in its root (**outside** the **public\_html** folder) and name it **stacks**.
2. Using your FTP app, upload the stacks ending with **.livecode** from the **server-side** folder into the newly-created **stacks** folder on the server.

##### Uploading the Server-Side Script

1. In the **server-side** folder from the download, you'll find a file named **api-demo-script-mariadb.lc**.
2. Open that file in a plain text editor, then change the path (if necessary) to the exact location of the stack named **api-demo-server-mariadb.livecode** on the server.
3. Using your FTP app, upload the **api-demo-script-mariadb.lc** script into the main web folder for your domain name.

If you're on a cPanel-based account, you would typically upload the file into the folder **public\_html/example.com**, where **example.com** is your actual domain name.

##### Testing Your Demo Setup

1. In the **client-side** folder from the download, you'll find a stack named **api-demo-client-mariadb.livecode**.
2. Open that stack in LiveCode.
3. Without modifying any of the fields, click the **Create User** button.

If all goes well, you should see the message: **There is already a user in the database with the email: [test@example.com]**.

This is because the demo database does indeed already contain a user with that email.

4. Click the **Retrieve User** button.

In the **Output** field, you should see the details of the record retrieved from the MariaDB/MySQL database on the server.

5. Experiment with entering various bits of data, clicking the buttons, and checking out their scripts (as well as the stack script) to understand how the client app works.
6. Also check out the stack script of the server-side stack **api-demo-server-mariadb.livecode** to see how you can customize it to perform tasks as an intermediary between your LiveCode client app and your MariaDB/MySQL database.

##### Building a Test Standalone from the Demo

1. In the **client-side** folder from the download, you'll find a stack named **api-demo-client-mariadb.livecode**.
2. Open that stack in LiveCode.
3. Select **File > Standalone Application Settings**.
4. If using **LiveCode 6, 7, or 8.0.x (but not 8.1 and up)**:
  - If building for **Mac, Windows, and/or Linux**, click **General** in the top icon bar, then enable the **Internet** and **SSL & Encryption** script libraries.
  - If building for **iOS**, click **iOS** in the top icon bar, then enable the **Encryption** external.
  - If building for **Android**, click **Android** in the top icon bar, then enable the **SSL & Encryption** external.
5. If using **LiveCode 6 or 7**:
  - Click **Copy Files** in the top icon bar, then click **Add File...** and select the accompanying stacks **libHttpsApiClient.livecode** and **libJson.livecode**.
6. If using **LiveCode 8.0.x (but not 8.1 and up)**:
  - Click **Copy Files** in the top icon bar, and ensure that **com.livecode.library.json** is selected. Also click **Add File...** and select the accompanying stack **libHttpsApiClient.livecode**.
7. If using **LiveCode 8.1 and up**:
  - Click **General** in the top icon bar, then enable the option **Select inclusions for the standalone application**.
  - Click **Copy Files** in the top icon bar, then click **Add File...** and select the accompanying stack **libHttpsApiClient.livecode**.
  - Click **Inclusions** in the top icon bar, then enable the following:
    - JSON Library
    - Internet
    - SSL & Encryption
    - Answer Dialog
    - Ask Dialog
    - tsNet (if available)
  - If building for **Android**, click **Android** in the top icon bar, then under **Application Permissions**, enable **Internet**.
8. Close the **Standalone Application Settings** window.
9. Save your stack.
10. Select **File > Save as Standalone Application** to build your standalone application(s).

#### Troubleshooting Checklist

If you're running into issues connecting to the API layer, have a look at the following:

##### Connection or timeout errors when connecting from within LiveCode as well as from standalones

- Ensure that you have a valid SSL certificate installed on the domain from which you're running the API layer. If you're hosting with HostM, contact us via the Client Lounge to request for a free SSL certificate to be installed on your domain.

##### Connects fine from within LiveCode, but not from standalones

- Review the **Building a Test Standalone from the Demo** section above to ensure that your standalone settings are correct.

#### Create Your Own LiveCode HTTPS-Based API Layer

Use ideas and knowledge gained from the demo stacks to create your very own HTTPS-based API layer to allow your LiveCode desktop and mobile apps to communicate with MariaDB/MySQL databases. Feel free to extract and implement any code snippets from the demo as needed for your projects.

#### What About LiveCode HTML5 Apps?


This HTTPS-based API solution can be used in conjunction with LiveCode HTML5 apps, just as with desktop and mobile apps. Nevertheless, most LiveCode-based web projects don't actually require or involve LiveCode HTML5 at all.


For more information about the differences between LiveCode Server and LiveCode HTML5, check out our article [LiveCode Server vs LiveCode HTML5 — Which Should I Use for My Web Project?](#)


#### Feedback & Support

Should you have any questions or suggestions, or require further assistance, simply open a Client Care ticket via the secure **Client Lounge** and we will be happy to assist. Even if you're not a hosting client, feel free to get in touch via our [visitor contact form](#).

Love our guides? **HostM** offers professional and helpful **LiveCode Hosting** services with [unlimited features](#) and renewal rates that [actually match](#) our advertised rates.

  
100% Mac, Windows and Linux friendly!





[Click here to get started.](#)  
It's fast, easy and secure.

[Sign Up Now](#)