sweet—here’s a **one-click build script** for the **MSYS2 UCRT64** shell that compiles all four CGI apps and drops the needed DLLs into C:\xampp\cgi-bin\. It also (safely) backs up any libmysql.dll so your CGIs load MariaDB’s client.

Save as build\_ucrt64.sh and run it from the **MSYS2 UCRT64** shell.

**how to use**

1. open **MSYS2 UCRT64** shell
2. put this build\_ucrt64.sh next to your sources (auth.cpp, bid\_sell.cpp, listings.cpp, transactions.cpp).
3. run:

nginx

CopyEdit

bash build\_ucrt64.sh

#!/usr/bin/env bash

# Build all CGI binaries with UCRT64 GCC and copy required DLLs to XAMPP cgi-bin.

set -euo pipefail

# --- Config (override via env if needed) ---

XAMPP\_ROOT="${XAMPP\_ROOT:-/c/xampp}"

CGI\_DIR="${CGI\_DIR:-$XAMPP\_ROOT/cgi-bin}"

SRC\_DIR="${SRC\_DIR:-$PWD}" # folder containing \*.cpp sources

UCRT\_PREFIX="/ucrt64"

INCLUDE\_DIR="$UCRT\_PREFIX/include"

LIB\_DIR="$UCRT\_PREFIX/lib"

BIN\_DIR="$UCRT\_PREFIX/bin"

CXX="${CXX:-g++}"

CXXFLAGS="${CXXFLAGS:--O2 -std=c++17}"

INCS="-I$INCLUDE\_DIR"

LIBS="-L$LIB\_DIR -lmariadb -lws2\_32"

# List of programs to build (add db\_probe.cpp if you keep it around)

SOURCES=(

"auth.cpp"

"bid\_sell.cpp"

"listings.cpp"

"transactions.cpp"

)

# DLLs your CGIs typically need (copy if present)

DLLS=(

libmariadb.dll

libstdc++-6.dll

libgcc\_s\_seh-1.dll

libwinpthread-1.dll

# common extras that libmariadb might depend on; copied only if found:

libcurl-4.dll

libzstd.dll

zlib1.dll

libiconv-2.dll

libnghttp2-14.dll

libidn2-0.dll

libunistring-2.dll

libssh2-1.dll

libcrypto-3-x64.dll

libssl-3-x64.dll

)

echo "==> MSYSTEM: ${MSYSTEM:-unknown}"

if [[ "${MSYSTEM:-}" != "UCRT64" ]]; then

echo "!! This script is meant for the MSYS2 UCRT64 shell. You are in: ${MSYSTEM:-unknown}" >&2

echo " Open 'MSYS2 UCRT64' from Start Menu and run again." >&2

exit 1

fi

# Ensure directories exist

mkdir -p "$CGI\_DIR"

# Ensure required packages are installed

echo "==> Checking toolchain & client libs..."

if ! pacman -Qq mingw-w64-ucrt-x86\_64-gcc >/dev/null 2>&1; then

echo "==> Installing UCRT64 GCC..."

pacman -S --needed --noconfirm mingw-w64-ucrt-x86\_64-gcc

fi

if ! pacman -Qq mingw-w64-ucrt-x86\_64-libmariadbclient >/dev/null 2>&1; then

echo "==> Installing MariaDB client dev libs..."

pacman -S --needed --noconfirm mingw-w64-ucrt-x86\_64-libmariadbclient

fi

# Compile all programs

echo "==> Building CGI programs into $CGI\_DIR"

for src in "${SOURCES[@]}"; do

if [[ ! -f "$SRC\_DIR/$src" ]]; then

echo "!! Missing source: $SRC\_DIR/$src" >&2

exit 1

fi

out="$CGI\_DIR/${src%.cpp}.cgi"

echo " - $src -> $(cygpath -w "$out")"

"$CXX" $CXXFLAGS $INCS "$SRC\_DIR/$src" -o "$out" $LIBS

chmod +x "$out" || true

done

# Copy runtime DLLs next to the CGIs

echo "==> Copying runtime DLLs into $CGI\_DIR"

for dll in "${DLLS[@]}"; do

[[ -f "$BIN\_DIR/$dll" ]] || continue

cp -u "$BIN\_DIR/$dll" "$CGI\_DIR/" && echo " + $dll"

done

# If libmysql.dll is present, back it up so libmariadb.dll is used

if [[ -f "$CGI\_DIR/libmysql.dll" ]]; then

ts=$(date +%Y%m%d-%H%M%S)

mv -f "$CGI\_DIR/libmysql.dll" "$CGI\_DIR/libmysql.dll.bak.$ts"

echo "==> Backed up libmysql.dll -> libmysql.dll.bak.$ts"

fi

echo "==> Done!"

echo "CGI dir: $(cygpath -w "$CGI\_DIR")"

echo "Try: http://localhost/cgi-bin/auth.cgi"

(Override defaults if your XAMPP lives elsewhere:)

ini

CopyEdit

XAMPP\_ROOT=/d/xampp bash build\_ucrt64.sh

That’s it—this compiles, copies DLLs, and makes the CGIs runnable in C:\xampp\cgi-bin\.

Sources

Ask ChatGPT