

Business Case

Author: Benjamin Esenwein

(TINF20C, SWE I Praxisprojekt 2021/2022)

Project: WebSocket's in a LwIP HTTP Server

Customer: Rentschler & Holder
Rotebühlplatz 41
70178 Stuttgart

Supplier: Team 4: Laura Reeken, inf20051@lehre.dhbw-stuttgart.de
Benjamin Esenwein, inf20074@lehre.dhbw-stuttgart.de
Yannis Plaschko, inf20093@lehre.dhbw-stuttgart.de
Maximilian Meier, inf20084@lehre.dhbw-stuttgart.de
Lucas Kaczynski, inf20147@lehre.dhbw-stuttgart.de
Isabel Schwalm, inf20085@lehre.dhbw-stuttgart.de
Rotebühlplatz 41
70178 Stuttgart

Version	Date	Author	Comment
V0.1	09.04.2022	Benjamin Esenwein	created, filled document
V1.0	29.04.2022	Benjamin Esenwein	Added table of contents
V1.1	04.05.2022	Benjamin Esenwein	Added code highlighting

CONTENTS

1	Welcome to the lwIP Installationguide	3
2	Setting up the directory structure and the environment variable .	3
2.1	Create directory for the development environment	3
2.2	Download and unpack lwIP and contrip.....	3
2.3	Install WinPCap	4
2.4	Set the environment variable "PCAP_DIR"	4
3	Set up the development environment.....	5
3.1	Download Visual Studio Community	5
3.2	Open the Solution	5
3.3	Retargeting the project.....	5
3.4	Do not treat warnings as errors.....	6
3.5	Rebuild the solution.....	8
4	Configuration of the network settings for lwIP	8
4.1	Locate the relevant files	8
4.2	Adjust the relevant files	8

1 Welcome to the lwIP Installationguide

The following text describes the installation process of lwIP on Windows. It also describes basic errors that can occur during the installation and how to solve them.

The current version of lwIP no longer supports Windows. Instead, lwIP is supported on Linux. For lwIP under Linux there are also the most resources on the Internet that can help you with errors. Although we discuss the most common errors in this installation guide, there may still be errors that we cannot solve, or for which we do not provide help. Therefore, we advise only experienced users to use lwIP on Windows. If you still want to install lwIP on Windows, just follow the instructions specified here.

This guide uses a specific directory structure. It is recommended to use the same directory structure to better follow the instructions.

2 Setting up the directory structure and the environment variable

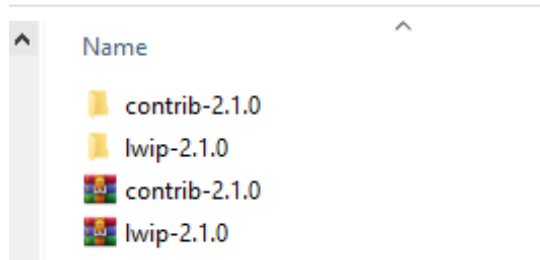
2.1 Create directory for the development environment

Create a folder for it under Windows. E.g. with the name lwIP-Project.

2.2 Download and unpack lwIP and contrib

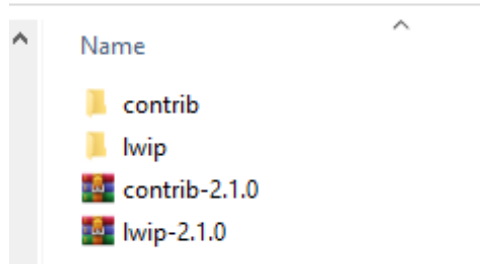
Download the lwip and contrib files (version 2.1.0 from September 26, 2018) from the release repository. Unzip the two files in the folder you created in step 2.1. Your folder should now look like this:

Dieser PC > Downloads > lwIP-Project >



Remove the version name from both the lwip and contrib folders. Your folder should now look like this:

Dieser PC > Downloads > lwIP-Project

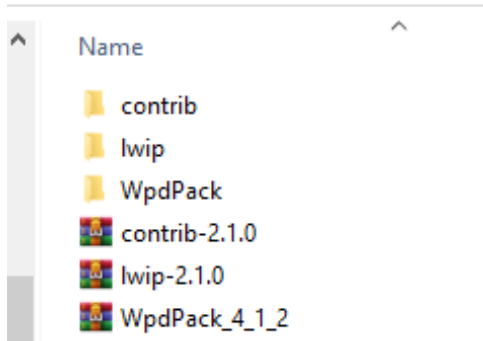


2.3 Install WinPCap

For lwIP to work under Windows, you need WinPcap (version 4.1.3). Download WinPcap for users. [WinPcap](#) There is an alternative solution called Npcap, but using Npcap requires additional configuration steps.

If you have installed the WinPcap version for users, download the WinPcap Developer's Pack (version 4.1.2). [WinPcap Developer's Pack](#) Unpack the file in a development directory. Your folder should look like this:

Dieser PC > Downloads > lwIP-Project >



2.4 Set the environment variable "PCAP_DIR"

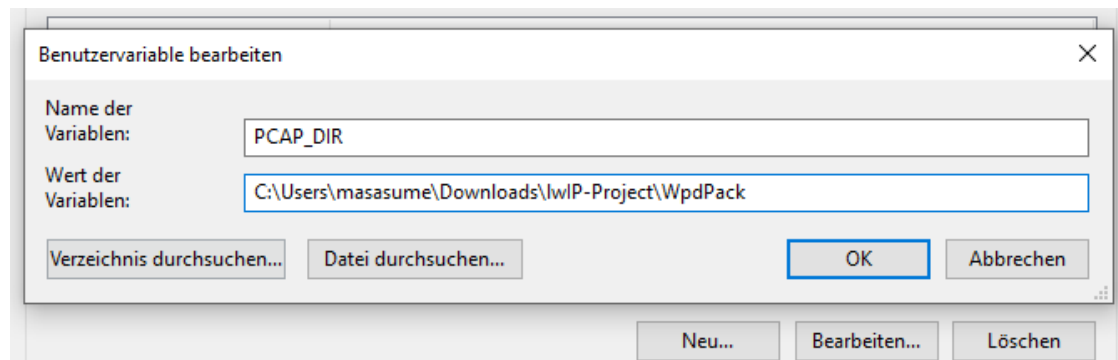
In order for lwIP to use WinPcap, you must define an environment variable that points to your development directory, or to the directory where you placed the WinPcap Developer's Pack.

Open the Windows search and enter Environment Variable. A window should open. Click in the window on the Environment Variables field.

In the window that opens, you can set environment variables for individual users or system-wide.

Create an environment variable for you as a user. To do this, press New. A window opens in which you can specify the system variable PCAP_DIR and the path to your development directory.

Give the environment variable the name PCAP_DIR and specify the path to your WpdPack folder.



3 Set up the development environment

3.1 Download Visual Studio Community

In this tutorial we will use Visual Studio Community Edition. We recommend using Visual Studio if you want to work with lwIP. [Visual Studio Community Edition](#) (2015 is officially supported by lwIP, but the 2022 version will also work).

When installing Visual Studio using the installer, there is no need to select "Workload". All required files are downloaded by opening the solutions in step 3.2.

3.2 Open the Solution

Open the Solution in the contrib/ports/win32/msvc/lwipTest.sln folder. The solution will then import the following projects into the development environment:

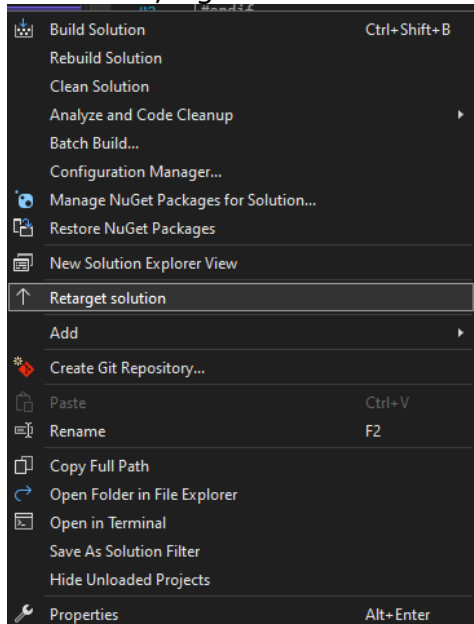
- lwIP
- lwIP pcapif
- lwIP_test
- makefsdata

Visual Studio should warn you that additional files need to be downloaded in order to work with the contrib Solution. Install the C/C++ runtime. After the installation is finished, the solution should be opened again.

3.3 Retargeting the project

In order for the opened project (lwipTest.sln) to compile successfully, the project must be retargeted.

To do this, right-click on the solution and select Retarget Projects.



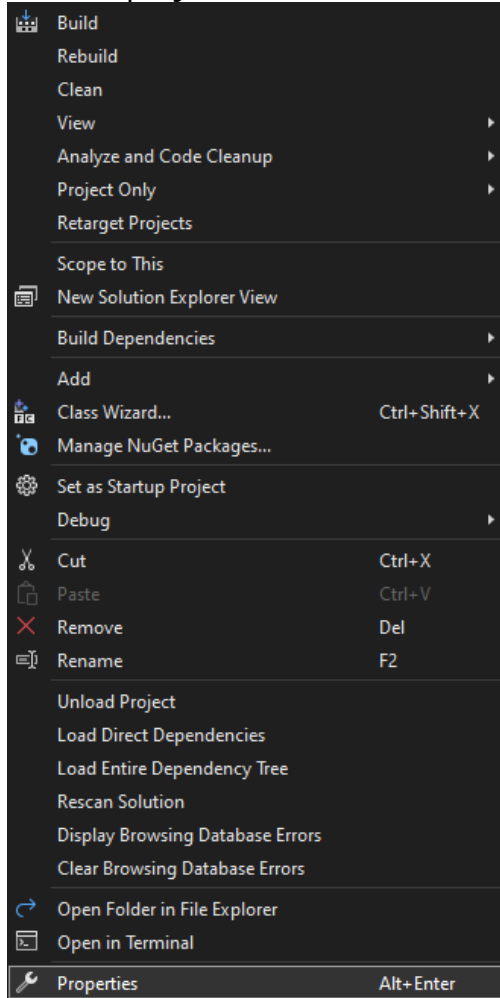
Then select the Windows SDK version 10.0. All projects in the solution should then be raised from SDK version 7 to SDK version 10.

3.4 Do not treat warnings as errors

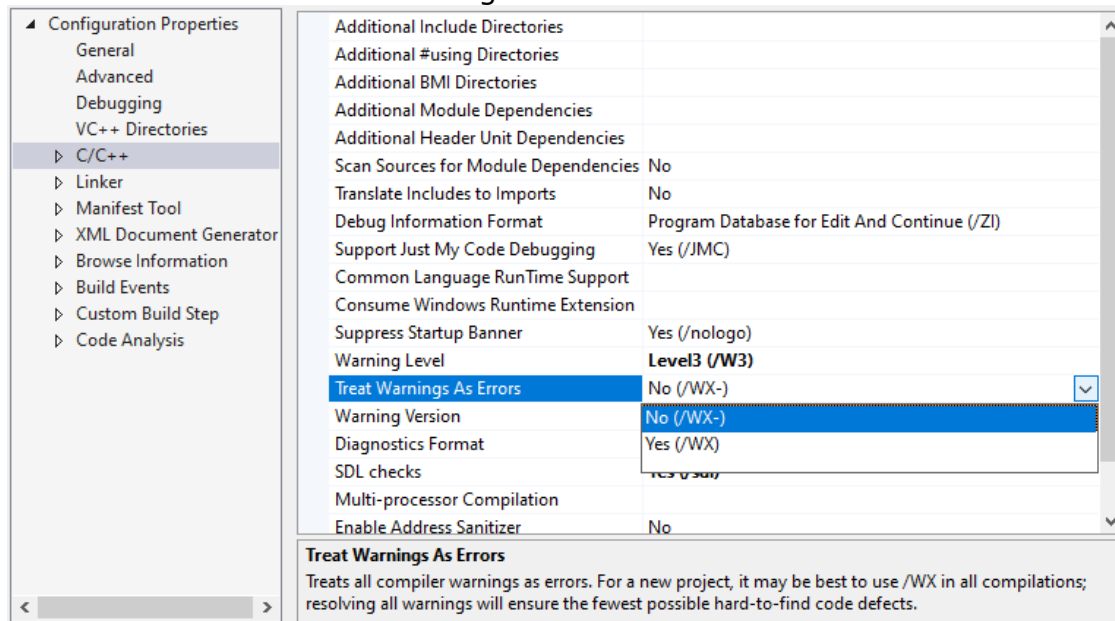
The project will compile successfully only if warnings from compiler are not treated as errors.

Repeat the steps below for all four projects in the solution.

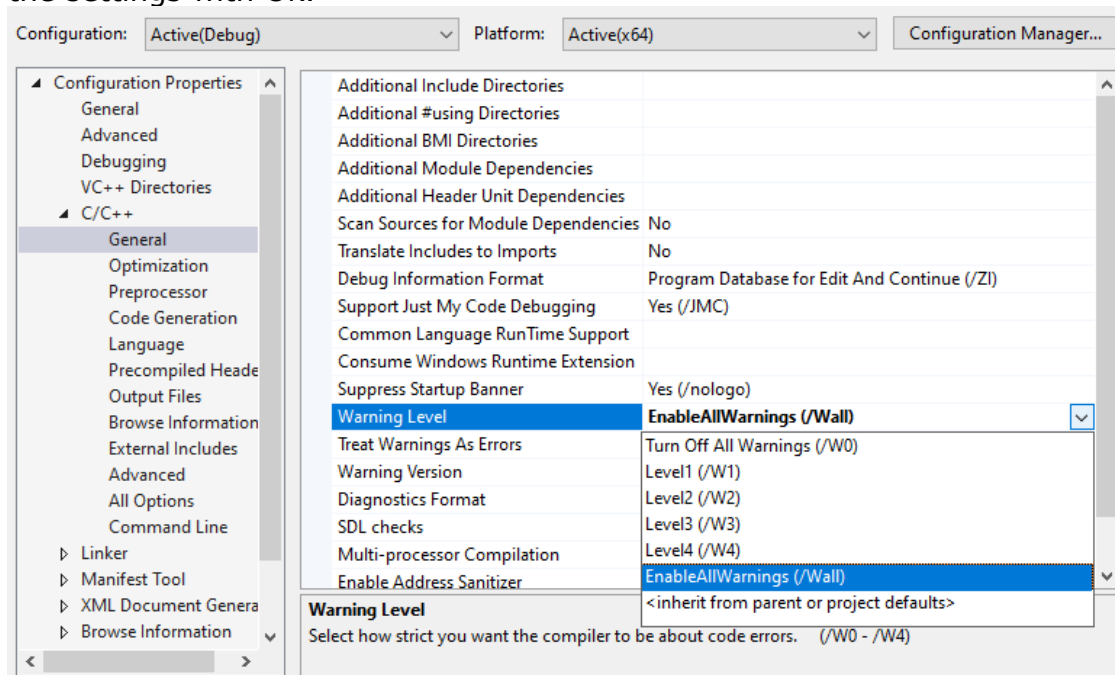
Go to a project with the mouse and right click. Then select Properties.



A window will open. In this window, navigate to the "C/C++" section in the left tree. Then select No for Treat Warnings As Errors.

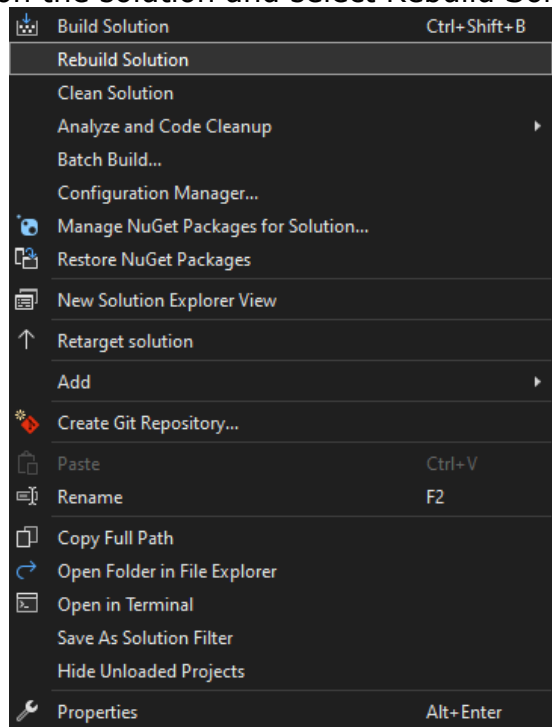


The entry Warning Level should be set to /Wall. This helps when debugging. Accept the settings with Ok.



3.5 Rebuild the solution

In order for your changes to take effect, you must rebuild the solution. Right-click on the solution and select Rebuild Solution.



If the Rebuild Solution fails, step 3.4 must be repeated for all projects of the Unittest Solution "lwIP_unittests". Open the solution in the folder /contrib/ports/win32/msvc/lwIP_unittests.sln and follow the process described in step 3.4.

4 Configuration of the network settings for lwIP

4.1 Locate the relevant files

For this step the files "lwipopts.h" and "lwipcfg.h" are needed. The latter is available as "lwipcfg.h.example" in the "lwip" project and should be duplicated and then renamed.

These files are in the folder "example_app".

It is recommended to connect the computer on which the lwIP project is to be developed via LAN. lwIP does not have WIFI drivers. However, these can be developed by the user.

4.2 Adjust the relevant files

In order to run and debug lwip under Windows, the network settings must be changed, because the network card cannot send checksums or sends incorrect checksums, which affect the use of lwip.

For this the following should be inserted in the file "lwipopts.h" under the line 244:


```
/* ----- Disable Checksum -----*/
```

```
#define CHECKSUM_CHECK_IP    0
```

```
#define CHECKSUM_CHECK_UDP    0
```

```
#define CHECKSUM_CHECK_TCP    0
```

```
#define CHECKSUM_CHECK_ICMP    0
```

Now only the IP configuration is missing. For this the file "lwipcfg.h" is needed. This can be found in the same directory as "lwipopts.h".

- The line #define PACKET_LIB_ADAPTER_NR contains the ID of the network adapter. This ID can be set later.
- Furthermore DHCP and AUTOIP can be activated. This way lwip will automatically request the necessary information regarding IP address, gateway and netmask from the developer computer's network.

```
#define USE_DHCP    1
```

```
#define USE_AUTOIP 1
```

- Alternatively, the options under b can also be deactivated, i.e. set to 0. Then IP address, gateway and netmask must be specified manually.
 - In this case, the command line (cmd) can be opened under Windows and the information can be retrieved with the command 'ipconfig /all'.

These can be entered in the file "lwipopts.h" at the appropriate place separated by commas. An unused address of the network should be used as IP address. For example "192.168.178.60".

Attention: "USE_DHCP" and "USE_AUTOIP" must be set to 0 in this case.

```
#define USE_DHCP    0
```

```
#define USE_AUTOIP  0
```

```
#define LWIP_PORT_INIT_IPADDR(addr)    IP4_ADDR((addr),  
192,168,178,60)
```

```
#define LWIP_PORT_INIT_GW(addr)        IP4_ADDR((addr),  
192,168,178,1)
```

```
#define LWIP_PORT_INIT_NETMASK(addr)    IP4_ADDR((addr),  
255,255,255,0)
```

Now you should be able to use lwIP under Windows.