

【C++】开源：websocketpp安装与使用

原创

Frank学习路上



已于 2023-07-19 22:02:17 修改



1523



收藏

3

版权

分类专栏：

c++开源项目学习

文章标签：

websocket

c++

网络协议



c++开源项目学习 专栏收录该内容

6 订阅

44 篇文章

订阅专栏

😄★,°:☆(▽ ▽)/\$.:°★😄

这篇文章主要介绍websocketpp的安装与使用。

学其所用，用其所学。——梁启超

欢迎来到我的博客，一起学习，共同进步。

喜欢的朋友可以关注一下，下次更新不迷路👉

文章目录

:smirk:1. websocket介绍

:blush:2. websocketpp安装

:satisfied:3. websocketpp使用

😄1. websocket 介绍

WebSocket 是一种在单个 TCP 连接上进行全双工通信的协议，它使得客户端和服务端之间的数据交流变得更加实时、高效。相比传统的 HTTP 请求-响应机制，WebSocket 直接建立连接，并通过数据帧（Data Frame）来交换消息，从而避免了每次通信都要建立、断开连接的开销。

WebSocket 协议 最早由 HTML5 规范提出，它可以用于 Web 应用程序、移动应用程序等不同类型的场景中。WebSocket 除了支持文本格式的消息外，还可以使用二进制数据格式发送数据，这使得 WebSocket 可以处理包括音频、视频等复杂类型的数据。

WebSocket 的优点包括：

实时性：WebSocket 支持实时通信，使得客户端与服务端可以快速响应对方的请求。高效性：相比 HTTP 协议，WebSocket

不需要反复 连接 、断开连接，因此减少了网络通信的延迟时间和资源消耗。跨平台：WebSocket 协议可以跨越多种浏览器和操作系统，从而为开发者和用户提供了更广泛的选择。

总的来说，WebSocket 提供了一种快速、实时、高效的通信方式，使得 Web 应用程序和移动应用程序等可以更加灵活地处理复杂的数据交互需求。

😊 2. websocketpp安装

以ubuntu18.04为例：

websocketpp库依赖boost_system，因此首先安装boost库：

```
1 | # apt安装
2 | sudo apt-get install libboost-dev
```

安装websocketpp库（这里用0.8.2版本）：

github地址：<https://github.com/zaphoyd/websocketpp>

```
1 | # 编译安装
2 | cd websocketpp
3 | mkdir build && cd build
4 | cmake ..
5 | make
6 | sudo make install
```

安装完成：

```
-- Installing: /usr/local/include//websocketpp/server.hpp
-- Installing: /usr/local/include//websocketpp/connection.hpp
-- Installing: /usr/local/include//websocketpp/endpoint.hpp
-- Installing: /usr/local/include//websocketpp/common
-- Installing: /usr/local/include//websocketpp/common/network.hpp
-- Installing: /usr/local/include//websocketpp/common/type_traits.hpp
-- Installing: /usr/local/include//websocketpp/common/cpp11.hpp
-- Installing: /usr/local/include//websocketpp/common/system_error.hpp
-- Installing: /usr/local/include//websocketpp/common/platforms.hpp
-- Installing: /usr/local/include//websocketpp/common/functional.hpp
-- Installing: /usr/local/include//websocketpp/common/chrono.hpp
-- Installing: /usr/local/include//websocketpp/common/asio.hpp
-- Installing: /usr/local/include//websocketpp/common/connection_hdl.hpp
-- Installing: /usr/local/include//websocketpp/common/time.hpp
-- Installing: /usr/local/include//websocketpp/common/random.hpp
-- Installing: /usr/local/include//websocketpp/common/md5.hpp
-- Installing: /usr/local/include//websocketpp/common/stdint.hpp
-- Installing: /usr/local/include//websocketpp/common/thread.hpp
-- Installing: /usr/local/include//websocketpp/common/memory.hpp
-- Installing: /usr/local/include//websocketpp/common/regex.hpp
-- Installing: /usr/local/include//websocketpp/common/asio_ssl.hpp
-- Installing: /usr/local/include//websocketpp/uri.hpp
-- Installing: /usr/local/include//websocketpp/connection_base.hpp
```

😓 3. websocketpp使用

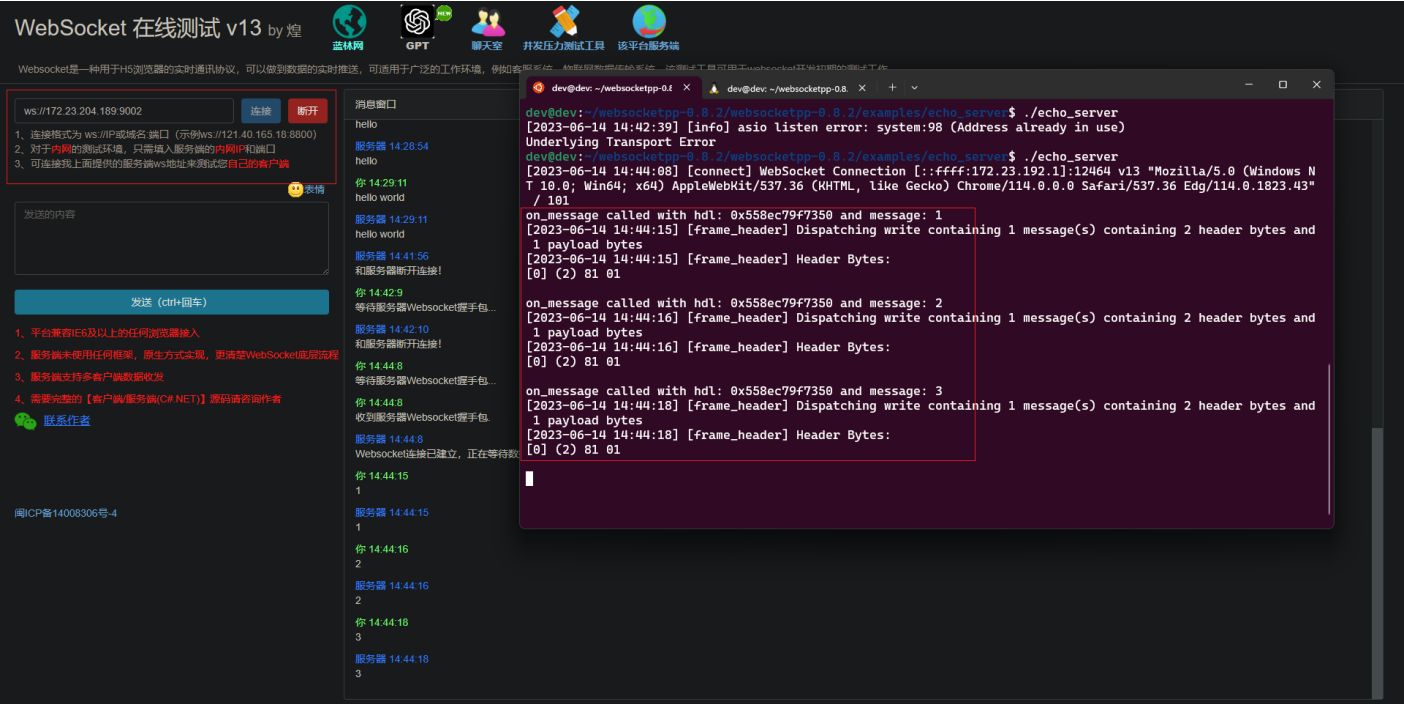
通信例程测试：

```
1 # 服务端
2 cd websocketpp/examples/echo_server
3 g++ echo_server.cpp -o echo_server -lboost_system -lpthread
4 ./echo_server
5 # 客户端
6 cd websocketpp/examples/echo_client
7 g++ echo_client.cpp -o echo_client -lboost_system -lpthread
8 ./echo_client
9 # 默认通信在本地的9002端口
```

在线websocket收发测试：

测试地址：<http://www.websocket-test.com/>

测试如下：



端口重用问题可以看：<http://t.csdn.cn/Z7AyU>（一般关掉服务后直接重启会报错，等一会就可以）

根本解决就是在listen函数前加入这条，这样在Linux端重启程序后就不会报错了：

```
1 server.set_reuse_addr(true); // 加入端口复用
2 server.listen(websocketpp::lib::asio::ip::tcp::v4(), uPort);
```

参考：

```
1 https://zhuanlan.zhihu.com/p/59925926
2
```



以上。