

# Hook 和长连接技术的授课软件设计方案

金子富, 王海珍, 蒋宁馨, 朱 瑞, 张桂香

(齐齐哈尔大学 计算机与控制工程学院, 黑龙江 齐齐哈尔 161006)

**摘要:** 针对网络带宽不足给网络授课带来的压力, 文章研究了在 Windows 系统下基于 C++ 语言, 利用 HOOK 技术与长连接技术研究网络授课软件的设计方案, 通过实现键盘和鼠标的动作监控、指令传输等操作, 不仅可以实现 PC 机与 PC 机之间实时传输的功能, 而且节省了带宽资源, 提高了数据传输效率。

**关键词:** HOOK 技术; 长连接技术; 网络授课

**中图分类号:** TP368; G642

**文献标识码:** A

**文章编号:** 1674-1064 (2022) 01-192-03

**DOI:** 10.12310/j.issn.1674-1064.2022.01.065

随着网络技术的发展, 越来越多的企业针对各样的技能培训建立了网络授课模式<sup>[1-2]</sup>, 多以视频和直播为主, 模式单一。Hook 技术是 Windows 操作系统中必须具有的重要组成部分, 在程序响应和系统设计等各个领域都有着广泛的应用<sup>[3]</sup>。

文章基于 Windows 操作系统的 Hook 和长连接技术设计网络授课软件, Hook 技术实现键盘、鼠标操作指令的传输, 可以节省带宽资源, 减少因地理位置等原因导致的网络效果差的情况。长连接技术主要用于维护服务器和客户端之间长一段时间的连接<sup>[4-5]</sup>, 客户端定时向服务器和客户端发送有关维持连接的信息, 通过服务端的检测机制, 检测该连接是否还是处于可用状态, 并由客户端与服务端通过长连接互相发送数据。其可以提高整个软件的及时性、便捷性, 使课堂互动变得更加轻松, 有助于推动网络授课模式的发展。

## 1 软件功能分析

笔者经过查阅、分析相关资料, 确定网络授课软件的主要功能如下:

**桌面共享功能。**该功能使学生都可以看到老师讲解的内容, 在上课的时候, 学生能够看到老师的板书或者课件, 并听老师的讲解。

**高清语音功能。**由于网络授课的对象、地点都不固定, 可能在家里, 也可能在宾馆或酒店, 所以使用的网络也不固定, 或者网通, 或者电信, 如果网络的线路质量差, 授课就卡。因此, 需要融合电话拨入功能作为网络备份。

**发送文件功能。**网络授课过程中, 需要模拟现场发试卷或者资料的效果, 具备远程发送文件功能。

**远程互动功能。**现场上课时, 老师可以提问学生, 学生可以走到黑板前答题。网络授课时, 可以远程控制老师当前讲的课件或者白板, 实现远程点名提问或答题。

**对学生发言权限的控制。**正常网络授课时, 禁止学生发言, 以防学生端杂音传入。需要学生发言时, 允许学生举手, 经老师审核决定是否发言。

## 2 设计方案

Hook 技术能够监控键盘、鼠标等外设设备, 长连接技术能够实现网络实时通信。即 Hook 事件监视对授课端主机上的直播软件中的 PPT 进行鼠标和键盘的操作记录, 在授课端主机对 PPT 操作的指令事件会被封装传输, 传输到听课端主机再拆包并重新还原, 以传输指令。

### 2.1 Hook 技术及应用

#### 2.1.1 Hook 技术的特点

Hook 技术可以拦截操作系统实际运行中所有存在的单条线程信息, 使用户在实际运行中主动地完成单条线程信息的操作和应用。

例如, Local Hook 拦截系统中的某一特定线程, 可以大幅提高整个系统的工作运行质量和系统的运行效果, 对于系统的影响很小。Remove Hook 可以拦截整个操作系统中正常运行的其余线程, 这种拦截的方法既可以拦截操作系统中某一或多个区域的总体线程, 也可以拦截整个操作系统的线程, 因

基金项目: 2020 年省级大学生创新创业训练计划资助项目: 基于 Hook 技术和长连接技术的授课软件 (202010232055); 黑龙江省高等教育教学改革研究项目: “新工科”背景下基于区块链技术的混合教学模式研究与实践 (SJGY20200770); 黑龙江省省属高等学校基本科研业务费科研项目: 基于传感器阵列与回归分析的智能鞋垫系统研究与应用 (135209245)

作者简介: 金子富 (1999—), 男, 黑龙江绥化人, 本科在读, 研究方向: 软件设计与开发。

通讯作者: 王海珍 (1976—), 女, 山东临沂人, 硕士, 副教授, 研究方向: 嵌入式技术、密码分析与设计、网络安全。

为需要同时拦截多个总体线程，对系统的影响也会变大，会降低系统运行的稳定性。

### 2.1.2 Hook的安装和释放

调用函数HHOOK SetWindowsHookEx ( int idHook, HOOKPROC lpfn, HINSTANCE hMod, DWORD dwThreadId ) 安装Hook，如果函数返回一个句柄则安装成功，返回NULL则安装失败。得到了一个关于hook句柄的动态回归函数值。idhook表示Hook处理的消息类型，用lpfn指向Hook的函数地址，最后两个参数则共同定义了Hook是作用于全局还是局部。当dwThreadId为空时，Hook作用于全局；hMod保存程序实例句柄，hMod为空时，Hook作用局部。此时，dwThreadId可以由GetCurrentThreadId()填充进程ID。

需要特别注意的是，hook存在两种使用方式，一是仅关注自己系统中所在的一些进程事件，二是作为一种远程hook直接观测其他进程中的一些具体线程事件，或者直接从网络中捕获到的系统中所有进程事件发生的情况，这种方式会花费大量时间处理数据。Hook使用完后，可以使用UnHookWindowsHookEx ( HHOOK hkh ) 进行Hook的卸载，参数hkh为SetWindowsHookEx返回的句柄。

### 2.1.3 Hook技术的应用

笔者以键盘、鼠标的监控为例说明Hook技术的应用。首先，操作系统将hook安装到前台键盘上，并将热键安装到前台软件上，将热键被按下的事件设置成一个宏用以监听，这样在操作系统正常运行时，系统就可以直接自动地完成被检测到键盘的工作。在检测到热键被按下的事件时，即使是前台软件能正常运行，系统也能以正确的处理方法进行工作。

对于鼠标而言，安装hook可以实时分析大量数据，而且可以分析鼠标当前位置和按下键的事件，并协助描绘鼠标运动状态和轨迹。当鼠标未被按下且静止时，系统可以应用Hook中的相关功能函数进行鼠标光标的位置定位，并对窗口中的相关因素进行直观的文字描述。

## 2.2 长连接及其应用

### 2.2.1 长连接的特点

TCP连接的建立需要三次握手，如果每个操作都是先建立连接再进行，那么处理速度会大大降低。笔者由此提出了长连接技术，一个操作完成后连接不断开，再次操作时就不用建立连接了。长连接多用于操作频繁、点对点的通信，但长连接对于服务端来说会耗费一定的资源，所以连接数不能太多。

### 2.2.2 长连接的应用

笔者采用C++语言，基于MFC应用框架<sup>[6]</sup>，应用Windows平台上的Socket网络编程接口，进行网络实时通信程序设计。即采用CSocket类设计，该类派生于MFC库的CAsync Socket类，对Win Sock API进行了继承并且有了进一步抽象封装，便于开发者快速构建一个网络通信框架而不需要深入的底层研究。

CSocket支持两种通信模式：SOCK\_STREAM（流模式）和SOCK\_DGRAM（报文模式），流模式是基于TCP协议，进行一个有连接的数据传输，而报文模式则是基于UDP协议进行无连接的数据传输。流模式更加安全可靠，本方案采用流模式，而且使用长连接。这样，客户端和服务端之间用于传输HTTP数据的TCP连接不会关闭，客户端再次访问这个服务器时，会继续使用这一条已经建立的连接，从而提高数据传输的实时性。

## 3 程序设计与实现

### 3.1 程序设计流程

本方案通过MFC框架实现的C/S架构分为服务端和客户端，客户端线程和服务端线程实现的主要流程如图1所示。服务端线程主要是通过Hook技术获取客户端手动触发的事件，如单击、双击鼠标的左键、右键，或者键盘按下的哪一个按键。

客户端线程获取服务端线程的操作，并结合相应的Windows底层实现远程控制，客户端线程和服务端线程采用基于长连接的TCP协议实现的多用户连接访问，服务端线程需要指定端口，客户端线程需要进行指定端口的访问并登录，才能对其进行操作，同时将提交的信息返回给服务端线程。

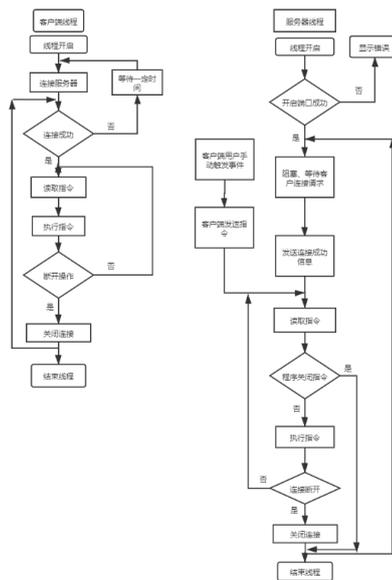


图1 程序设计流程图

### 3.2 关键技术实现

#### 3.2.1 服务器推技术实现

该技术使服务器主动与浏览器交流信息，即实现了教师端向学生端发起的通信，获得每一个学生端的信息，如IP地址、当前浏览页面、sessionID等信息。文章利用这些信息在服务器端创建一个伪HTTP请求，使服务器作出响应，更新该客户端的信息，从而实现服务器推技术。

### 3.2.2 服务器端长连接实现

```
void Ser_Lis_Port(unsigned short port,
std::function<bool(SOCKET stu_clientSocket)>listenCallback) {
//初始化
WORD sockVersion = MAKEWORD(2,2);
WSADATA wsiData;
if(WSAStartup(sockVersion,&wsiData)!=0 {
std::cout<< "WSAStartup error!" << std::endl;
return;
}
//创建套接字
SOCKET tea_serverSocket = socket(AF_INET,SOCK_
STREAM, IPPROTO_TCP);
if(tea_serverSocket == INVALID_SOCKET) {
std::cout<< "create socket error!" << std::endl;
return;
}
//绑定IP地址和端口
Socketaddr_in serin;
sin.sin_family = AF_INET;
sin.sin_port = htons(port);
sin.sin_addr.S_un.S_addr = INADDR_ANY;
if (bind (tea_serverSocket, (LPSOCKADDR) &serin,
sizeof(serin)) == SOCKET_ERROR) {
std::cout<< "bind error!" << std::endl;
return;
}
//监听端口
If (listen(tea_serverSocket,5) == SOCKET_ERROR) {
std::cout<< "listen error!" << std::endl;
return;
}
//等待学生端连接
SOCKET stu_socketClient;
Sockaddr_in remoteAddr;
int remoteAddrLen = sizeof (remoteAddr);
while (1) {
stu_clientSocket = accept(tea_serverSocket, (SOCKADDR *)
&remoteAddr, &remoteAddrLen);
std::thread clientConnectThread = std:::(listenCallback, stu_
clientSocket);
clientConnectThread.detach();
}
}
}
```

### 3.3 主要界面

教师作为服务器端，学生作为客户端，教师端运行界

面如图2所示，学生端运行界面如图3所示。点击得到屏幕按钮，如图4所示，可以看到教师授课的共享桌面。



图2 服务器端界面

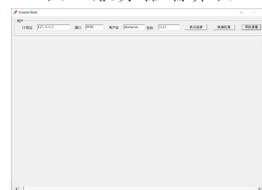


图3 控制端界面



图4 控制效果图

## 4 结语

文章基于Windows操作系统的Hook技术和长连接技术，提出了网络授课软件的设计方案，即基于长连接技术设计客户端、服务器端程序。服务端通过Hook技术获取客户端手动触发外设的事件，并接收相应的控制指令，进行相关控制。长连接技术的长连接可以减少TCP建立、关闭的请求操作，提高数据传输效率，但随着时间推移，存在达到服务器允许的上限问题。因此，本方案需要考虑哪些客户端需要和服务器建立长连接，使其更符合网络授课的需求。

## 参考文献

- [1] 穆阳,都松阳.基于网络技术的教育培训模式的探究[J].教育考试,2017(6):77-78.
- [2] 孙立擎,杜晓艳,张乘.“互联网+”模式下对企业培训发展的思考[J].教育现代化(电子版),2017(4):226.
- [3] 温卫.Hook技术及其在软件研发中的应用探讨[J].电脑编程技巧与维护,2019(6):52-54.
- [4] 尹慧平,姚全珠,何瓦特.基于TCP长连接的移动设备分屏交互技术研究[J].云南民族大学学报(自然科学版),2016,25(2):174-178.
- [5] 严腾飞.基于长连接的数据处理实现方法、系统及装置[P].中国:CN103634323B,2019.
- [6] 周亚文.以MFC为框架实现C/S通信的Socket编程[J].数码世界,2018(8):28-29.