**实验 1：HTTP 代理服务器的设计与实现**

|  |  |
| --- | --- |
| **学号：** | 1130310128 |
| **姓名：** | 杨尚斌 |
| **专业：** | 计算机科学与技术 |
| **指导老师：** | Xxx |

1. **实验目的**

熟悉并掌握 Socket 网络编程的过程与技术；深入理解 HTTP 协议， 掌握 HTTP 代理服务器的基本工作原理；掌握 HTTP 代理服务器设计与 编程实现的基本技能。

1. **实验环境**

* 接入 Internet 的实验主机；
* Windows 10, Visual Studio 2013；
* 开发语言：C++。

1. **实验内容**
2. **实验总结**
   * + 1. **Socket 编程的客户端和服务器端主要步骤**

服务器端：

* + - * 1. 使用 socket 函数创建一个 socket 描述符，来唯一标识一个 socket。函数原型为： int socket(int domain, int type, int protocol)，函数返回一个
        2. 使用 bind 函数绑定 IP 地址，端口信息等。函数原型为：int bind(int sockfd, const struct sockaddr \* addr, socklen\_t addrlen)
        3. 使用 listen 函数进行监听创建的 socket。函数原型为 int listen(int sockfd, int backlog)
        4. 使用 accept 函数接收请求，此时 socket 连接也就建立好了。函数原型为 int accept(int sockfd, struct sockaddr \* addr, socklen\_t \* addrlen)
        5. 使用 read(), write() 等函数调用网络I/O 进行读写操作，来实现网络中不同进程之间的通信
        6. 使用 close 函数关闭网络连接，即关系相应的 socket 描述字。函数原型为 int close(int fd)，注：close 操作知识是相应的 socket 描述字的引用计数-1，只有当引用计数为0的时候，才会触发客户端想服务器发送终止连接请求

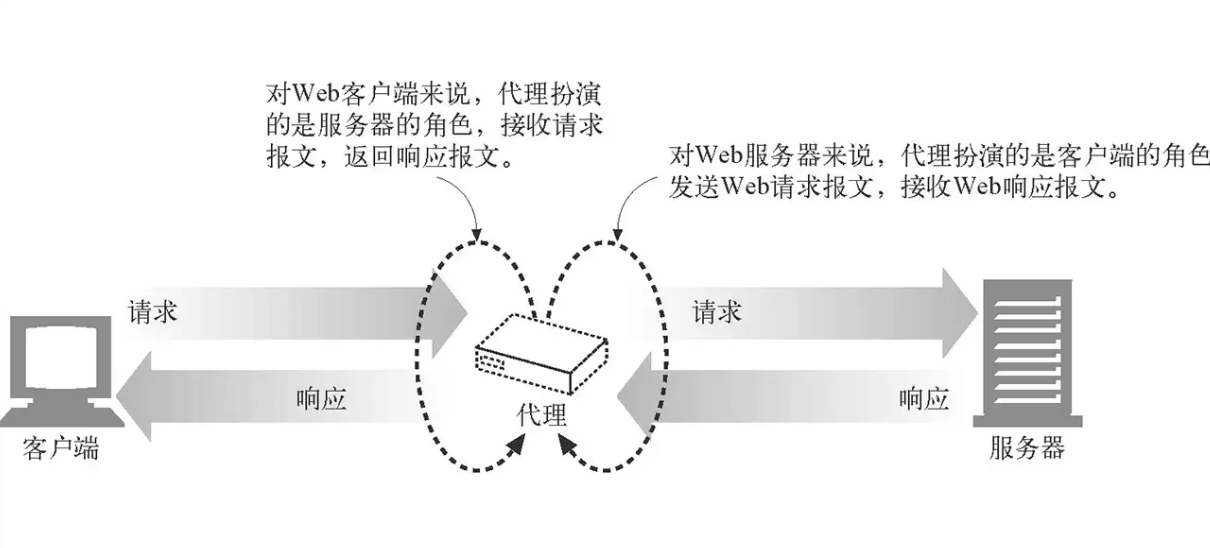
客户端：

1. 创建 socket
2. 绑定 IP 地址，端口信息
3. 设置要连接对方的 IP 地址和端口属性
4. 使用connect 函数连接服务器。 函数原型为int connect(int sockfd, const struct sockaddr \* addr, socklen\_t addrlen)
5. 使用 read(), write()等函数进行网络 I/O 的读写
6. 关闭网络连接
   * + 1. **HTTP 代理服务器的基本原理**

正常情况下去访问 HTTP 协议的网站，一般是浏览器向服务器发送一个请求，之后服务器会产生响应，从而客户端得到相应的数据。

如果我们在请求与响应的中间加入一层，即 浏览器向代理服务器发送请求，代理服务器去解析，然后向真正的服务器去发送请求，然后得到真正服务器所得到的响应，然后再发送给客户端。

也就是说，加了代理服务器之后，相对于客户端来说，代理服务器就是服务器；相对于真正的服务器来说，代理服务器就是客户端。代理服务器在整个过程中充当的是一个媒介的作用，用图来表示就是这样的：



* + - 1. **HTTP 代理服务器的程序流程图**
      2. **实现 HTTP 代理服务器的关键技术及解决方案**
      3. **HTTP 代理服务器实验验证过程以及实验结果**
      4. **HTTP 代理服务器源代码（带有详细注释）**