开发过程文档

作为一个 HTTP 代理服务器，得出软件的雏形是较为容易的，但在软件的测试使用过程中遇到了很多问题，发现并解决这些问题才占据了这个软件开发的大部分时间，以下列举了我在开发过程中遇到的各种问题以及解决方案。

1. **难以找到测试所用网页**

   程序最开始只是想要处理 HTTP 功能，但是在当下，使用了 HTTPS 的网站才是占据了主流，很难找到可以测试程序功能的网页，所幸最后发现教务处网站还是使用的 HTTP 协议，可以选取教务处网站作为测试对象。

2. **对缓存文件名的处理**

   在测试时发现缓存文件无法写入，经调试发现最初在用的将请求地址作为文件名的方法导致了文件名中有着操作系统所不允许的'/'和'?'，于是用字符串替换处理成'_'后问题得以解决。

3. **文件所在目录混乱**

   最初就仅仅是把缓存文件放在程序所在文件夹内，但在调试时发现既不美观，也使得清除缓存这一行为变得不便。所以导入了 os 库，在程序所在目录建立了 cache 文件夹，将缓存放入文件夹内，分网站保存缓存文件，便于查看以及删除。

4. **网页访问速度过慢**

   在打开网页时发现尽管可以打开，但访问速度缓慢的让人难以忍受，一个 js 脚本或者样式表都要请求 5 秒之久，尽管已经添加了对于多线程的处理依旧无济于事。最终经过调试发现时间主要耗费在阻塞性的 socket.recv()方法上。于是我查询了文档发现可以使用通过超时来控制的 socket，在一段时间没有接受或发送消息后会断开连接，这样就可以避免循环调用 recv()方法时被阻塞了。

5. **无法处理 HTTPS 连接**

   在当下，使用了 HTTPS 的网站占据了主流，但时最开始设计的 HTTP 代理服务器仅能够处理 HTTP 的 GET 方法，在经过代理的情况下不能够访问任何使用了 HTTPS 的网页。这样的一个代理服务器在现在是可以说是完全不可用的，所以我想着采取一些方法来支持这一方面的功能。

   通过抓包以及查找网络资料发现，客户端往往会采用 HTTP 隧道的方法通过 HTTP 代理服务器来访问 TLS 网站，最常见形式是通过 HTTP CONNECT 方法。而抓取到的 HTTP CONNECT 包也证明了这一点。在这种机制下，客户端要求 HTTP 代理服务器将 TCP 连接转发到所需的目的地。代理服务器只要与远程服务器建立连接后，将发送到代理服务器的所有数据都将原样转发到即可。

   在加入了对 HTTP CONNECT 的支持之后，通过此代理处理对 HTTPS 的访问请求，此时不进行缓存。

6. **无法处理上传**

   在测试时，我测试了在一些网站上进行上传，大多无法上传成功。在抓包之后发现上传会用到 HTTP POST，而这尚未在程序中得到支持。而这也会极大的影响到正常使用，于是我在程序中添加了对于 HTTP POST 的支持。由于网站对于不同的 POST 请求会有不同的响应，且 POST 请求内容并未放在头部，此时代理服务器不进行缓存。