

**计算机网络**

**课程实验报告**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 实验名称 | HTTP 代理服务器的设计与实现 | | | | | |
| 姓名 | 王丁子睿 | | 院系 | 计算机科学与技术 | | |
| 班级 | 1803104 | | 学号 | 1183710211 | | |
| 任课教师 | 李全龙 | | 指导教师 | 李全龙 | | |
| 实验地点 | 格物楼213 | | 实验时间 | 2020/10/31 | | |
| 实验课表现 | 出勤、表现得分(10) |  | 实验报告  得分(40) |  | 实验总分 |  |
| 操作结果得分(50) |  |
| 教师评语 | | | | | | |
|  | | | | | | |

****

|  |
| --- |
| 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。 |
| 实验内容： |
| (1)设计并实现一个基本HTTP代理服务器。要求在指定端口（例如8080）接收来自客户的HTTP请求并且根据其中的URL地址访问该地址所指向的HTTP服务器（原服务器），接收HTTP服务器的响应报文，并将响应报文转发给对应的客户进行浏览。  (2)设计并实现一个支持Cache功能的HTTP代理服务器。要求能缓存原服务器响应的对象，并能够通过修改请求报文（添加if-modified-since头行），向原服务器确认缓存对象是否是最新版本。（选作内容，加分项目，可以当堂完成或课下完成）  (3)扩展HTTP代理服务器，支持如下功能：（选作内容，加分项目，可以当堂完成或课下完成）  a)网站过滤：允许/不允许访问某些网站；  b)用户过滤：支持/不支持某些用户访问外部网站；  c)网站引导：将用户对某个网站的访问引导至一个模拟网站（钓鱼）。 |
| 实验过程： |
| 1. 浏览器使用代理，以edge为例，依次打开设置→系统→打开计算机的代理设置→启用代理服务器并设置相应的端口号。      1. 启动代理服务器      1. 使用浏览器浏览特定的网页，记录实验结果 |
| 实验结果： |
| 1. **基本代理功能：**   利用浏览器打开today.hit.edu.cn，可以正常显示，并打印相关信息     1. **缓存功能：**   在第一步执行完毕后，会发现缓存文件夹中新生成了缓存，重新打开today.hit.edu.cn，可以发现打印的信息中出现了304 not modified，表明代理服务器直接调用了缓存     1. **网站过滤：**   设置过滤网站www.whu.edu.cn，用浏览器打开该网页，会提示网站过滤信息     1. **网站重定向：**   设置重定向www.bit.edu.cn→www.hit.edu.cn，用浏览器打开前一网站，会发现实际跳转到的为后一网站     1. **用户过滤：**   设置过滤IP为本机IP127.0.0.1，重新访问today.hit.edu.cn，会发现无法访问 |
| 问题讨论： |
| 1. **Socket 编程的客户端和服务器端主要步骤**   客户端(client.py)：声明socket，将该socket与代理服务器连接，并向代理服务器发送信息，获取来自代理服务器的信息。（该代码主要用于测试，实际实验中该部分由浏览器完成）  服务器端(proxy.py)：创建服务器端socket，不断循环，每次新建一个进程来处理一条socket接收到的请求。若请求信息合法，则新建一个socket，向相应的网页发送请求，获取信息，转发给客户端，并关闭该socket。   1. **HTTP代理服务器的基本原理**   将客户端、代理服务器、目标服务器看作三个节点，按次序用双向边将三个节点相连。  每次客户端发送请求，会首先被代理服务器接收，接收后代理服务器会解析出客户端的IP和想要访问的地址，检查IP和地址是否合法，或是否需要处理。执行完上述操作后，代理服务器会继续向目标服务器发送处理后的请求。  此外，代理服务器可以存储一部分网址的信息，直接发送回客户端，从而省去向目标服务器发送请求的过程。   1. **HTTP代理服务器的流程图**      1. **实现 HTTP 代理服务器的关键技术及解决方案**    1. 套接字连接：声明一个socket实例，参数为socket.AF\_INET和socket.SOCK\_STREAM，从而能够支持IPv4和TCP。每次想要访问某个网站，先用send向目标服务器发送请求，每个接收的buffer再发送给客户端。    2. 服务器缓存：检查缓存是否存在和存发缓存较为简单，主要介绍如何检查缓存是否被修改。获取本地缓存的修改时间，构建请求信息If-Modified-Since，时间为格林威治时间，插入原请求信息的第二行，并向目标服务器发送。若返回的头部中的状态码为304，表示缓存未被修改，直接发送本地的缓存即可；若为200，表示缓存已被修改，则接下来执行的操作与未缓存一致；若为其他状态码，则说明网站不会处理该请求、请求格式有误等。 2. **HTTP 代理服务器实验验证过程以及实验结果**   （见实验流程、实验结果两部分）   1. **HTTP 代理服务器源代码**   （见随报告一同提交的代码） |
| 心得体会： |
| 该实验的主要精力集中在请求的解析和构建过程中，由于请求有较为严格的格式要求，所以在方便了解析的同时，也部分提高了重构请求的难度。若请求构建的格式有误，很容易出现向目标服务器发送请求后，无法获得返回信息的情况。 |