**深 圳 大 学 实 验 报 告**

**课程名称：­ 计算机网络（Computer Networks）**

**实验名称： Application Layer Assignment**

**学院： 电子与信息工程学院**

**专业： 电子信息工程**

**指导教师： 毕宿志**

**报告人：何雨璇 班级：电信6班 学号： 2022280445**

**实验时间： 2023 年 11 月 1 日**

**实验报告提交时间： 2023 年 11 月 4 日**

**教务部制**

|  |
| --- |
| 1. **Purpose of experiment** 2. 了解Web服务器的基本原理和运作方式，利用python来接收和处理HTTP请求，并向客户端发送HTTP响应 3. 了解ping程序的基本概念和功能解DNS的基本原理和功能，利用python实现客户端向服务器发送ping消息、服务器返回pong消息以及计算往返时间（RTT） 4. 了解FTP协议的基本原理和实现，使用Python和套接字编程来实现文件的上传和下载功能，并处理相关的文件传输逻辑 5. **Experimental principle** 6. Web：我们建立了一个简单的Web服务器，它会根据HTTP协议接收客户端发送的HTTP请求，处理请求并返回HTTP响应。在Web服务中，通过Socket编程，服务会监听指定端口，接收客户端连接，并解析请求中的请求偷请求行、请求头和请求体等信息。服务器在根据请求中的路径，读取相应文件内容，并封装成符合HTTP协议的响应消息返回给客户端。 7. UDP-based ping ：程序使用UDP协议实现基于ping-pong消息的简单网络延迟测量。客户端向服务器发送ping消息，服务器接收到ping消息后立即返回pong消息。客户端记录发送ping消息的时间戳，并在接收到pong消息时计算往返时间（RTT）。这里我利用随机数刻意的产生了一些错误，但是由于属于内网传输它们很大一部分都是正确传输的。 8. TCP-based FTP ：程序使用TCP协议实现文件的上传和下载。客户端和服务器之间通过TCP Socket建立连接，并使用TCP的可靠数据传输机制进行文件传输。FTP客户端发送文件请求消息给服务器，服务器根据请求消息进行文件的读取或写入操作，并将文件数据封装成TCP数据包发送给客户端。客户端接收到数据包后，将文件数据写入本地文件。 9. **Content** 10. Web : 11. 新建两个文件，http\_server.py文件与index.html文件； 12. 简单编辑index； 13. 在http\_server文件中，编辑两个函数，一个用于接收http请求一个用于运行后端服务并使其在控制台输出文字检错； 14. 运行http\_server，在浏览器中访问网址http://localhost:8080/index.html，查看结果； 15. 在浏览器中运行网址http://localhost:8080/none.html，查看结果。 16. UDP-based ping ： 17. 新建两个文件，UDP\_pinger\_client.py文件与 UDP\_pinger\_server.py文件； 18. 编辑UDP\_pinger\_server.py文件作为服务器基于UDP接收请求并响应请求； 19. 编辑 UDP\_pinger\_client.py文件作为客户端基于UDP发送请求，记录并输出等待的时间，其中利用随机数模拟一下出错，注意文件中的端口要与服务器的一致； 20. 运行UDP\_pinger\_server.py，接着运行UDP\_pinger\_client.py，查看结果。 21. TCP-based FTP ： 22. 新建两个文件FTP\_client.py 和 FTP\_server.py，并新建一个文件夹files，其中包含index.html，这里的index直接使用了Web中的index； 23. 编辑FTP\_server.py作为服务器基于TCP接收请求，查询是否存在文件并做出响应； 24. 编辑FTP\_client.py作为客户端基于TCP发送请求，并请求输入需要请求的文件名； 25. 运行FTP\_server.py，再运行FTP\_client.py，在控制台输入index.html，查看结果 26. 关闭刚刚运行的文件，重新再按顺序运行两个文件，在控制输入none.html，查看结果。 |
| 1. **Conclusion and discussion** 2. Web :   conclusion:  访问网址http://localhost:8080/index.html后，结果显示为编写后的index.html  访问网址http://localhost:8080/none.html后，界面显示“404File Not Found”  discussion:  在8080这个端口由于我只编写了index.html这个文件所以，客户端无法获取到none.html这个文件   1. UDP-based ping :   conclusion:  运行UDP\_pinger\_server.py，接着运行UDP\_pinger\_client.py后，  我们可以看到结果：（由于太多只展示一半）  Response : Pong Ping 1 1699111901.119885 (Error)  RTT: 0.004360 seconds  Response : Pong Ping 2 1699111901.124386  RTT: 0.000420 seconds  Response : Pong Ping 3 1699111901.124817  RTT: 0.000606 seconds  Response : Pong Ping 4 1699111901.125453  RTT: 0.000250 seconds  Response : Pong Ping 5 1699111901.125731 (Error)  RTT: 0.000518 seconds  discussion：  在这个实验中我输出了每个response对应的 Pong Ping和它们发送的时间，并且输出了每个Ping Pong的RTT，可以看到这里的Error并不是RTT大于1造成的，是因为我折翼了一个随机数使得某些请求即使获得及时回应也会显示Error。   1. TCP-based FTP :   conclusion：  运行FTP\_server.py，再运行FTP\_client.py，在控制台输入index.html后，我们会在UDP\_pinger\_client 同一文件夹下发现index.html文件  关闭刚刚运行的文件，重新再按顺序运行两个文件，在控制台输入none.html后，控制台显示“Error: File 'none.html' not found”  discussion:  只有在输入正确的文件名是才能获取正确的文件，并且就我设置的这个端口而言，files这个文件夹中不存在的文件客户端是无法获取的 |
| 指导教师批阅意见：  成绩评定：  指导教师签字：  年 月 日  备注： |

注：1、报告内的项目或内容设置，可根据实际情况加以调整和补充。

2、教师批改学生实验报告时间应在学生提交实验报告时间后10日内。