# 《计算机网络》课程设计实验报告

姓名：

学号：

班级：班

**系统的功能设计**

基本要求

“DNS中继服务器”的实现

设计一个DNS服务器程序，读入“IP地址-域名”对照表，当客户端查询域名对应的IP地址时，用域名检索该对照表，有三种可能检索结果：

检索结果：ip地址0.0.0.0，则向客户端返回“域名不存在”的报错消息（不良网站拦截功能）

检索结果：普通IP地址，则向客户端返回该地址（服务器功能）

表中未检到该域名，则向因特网DNS服务器发出查询，并将结果返给客户端（中继功能）

考虑多个计算机上的客户端会同时查询，需要进行消息ID的转换

**实验环境**

Windows 7

C++（编译器Dev-C++ 4.9.9.2）

**模块划分**

主函数int main(int argc, char\*\* argv)

运行套接字服务及管理DNS

函数int GetTable(char \*tablePath);

读取dnsrelay.txt信息，获取域名解析表

函数void GetUrl(char \*recvbuf, int recvnum);

从DNS Request中解析QName字段，获取域名

函数int IsFind(char\* url, int num);

判断是否在表中找到DNS请求中的域名

函数unsigned short RegisterNewID (unsigned short oID, SOCKADDR\_IN temp);

为向外部名字服务器发出DNS Request 做准备，划分新的ID

函数void DisplayInfo(unsigned short newID, int find);

展示调试信息（时间坐标，ID，服务功能：域名及IP）

函数void introduce(void);

版本介绍

**软件流程图**

1.输入dnsrelay [-d | -dd] [dns-server-ipaddr] [filename]指令执行程序

2.获取域名解析表dnsrelay.txt

3.初始化

4.创建本地DNS和外部DNS套接字

5.接收来自ping,ftp,IE等的DNS Request（使用nslookup 指令测试及应用浏览器）

在域名解析表中查找，并区分使用的功能（中继，服务器，屏蔽）

5.1中继：

将DNS Request报文更改Header中的ID，转发给外部名字服务器，接收DNS Response，归还原有ID，并响应DNS请求。

增加了功能：解析DNS Response，分析Resource Record部分，找到Type=A, Length=4的响应，并取出RDATA，即其IPv4。

5.2服务器

构造DNS Response报文（原有DNS Request修改AnCount=1+RR部分），响应DNS请求。

5.3屏蔽

构造DNS Response报文，AnCount=0,RR部分返还0.0.0.0，响应DNS请求。

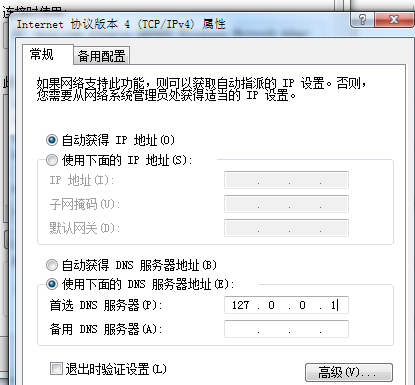
6.循环以维持DNS中继服务器服务

**测试用例以及运行结果**

使用ipconfig/all,记下当前DNS服务器



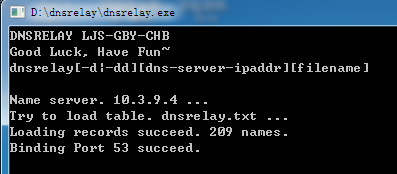
将DNS设置为127.0.0.1(本地主机)



运行dnsrelay程序（在程序中把外部DNS服务器设为前面记下的10.3.9.5）

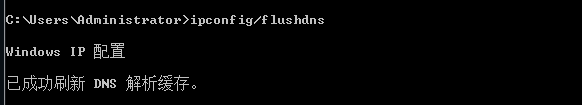
#define DEF\_DNS\_ADDRESS "10.3.9.5" //外部DNS服务器地址

#define LOCAL\_ADDRESS "127.0.0.1" //本地DNS服务器地址



正常使用ping，ftp，IE等，名字解析工作正常

清空缓存



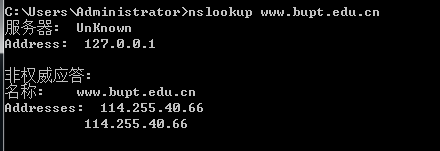
中继功能

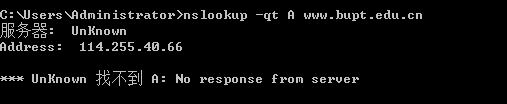




服务器功能

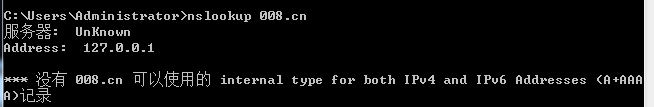






屏蔽功能



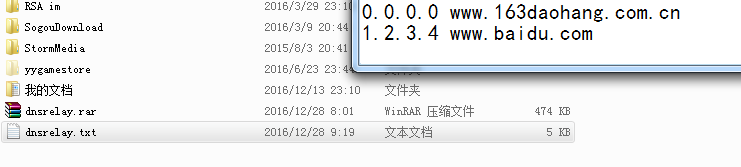


设置了环境变量之后，可执行的dnsrelay指令





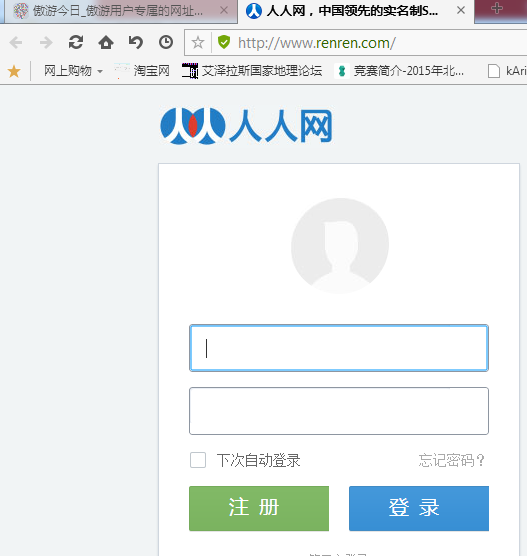
为验证其使用了不同目录下的域名解析表（txt），将D:\dnsrelay.txt中添加假想项1.2.3.4 www.baidu.com





证明使用了不同的域名解析表。

实际应用，在使用dns中继器的情况下可以打开网页。



**调试中遇到并解决的问题**

1. 在配置编译环境时的问题

我们用到的编译器Dev-C++ 4.9.9.2是比较老的版本，在链接库文件时出现了一系列问题使得链接不成功。经过尝试，我们创建了工程，并将libws2\_32.a加入了工程连接器。

同时参考了课件中的建议：

为使用winsock函数库，vc编程增加下面语句：

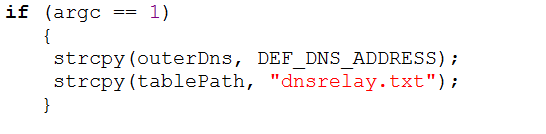
#pragma comment(lib,"Ws2\_32.lib")

也可以不加此语句，但链接时须增加wsock32.lib库

1. 在Internet协议属性配置时，关闭了TCP/IPv6以避免影响
2. 反复修改了代码，以优化结构，删除了大量实用度不高的变量，以及考虑了对函数的权限限制。
3. 在实现dnsrelay [-d | -dd] [dns-server-ipaddr] [filename]命令的过程中发现的问题

4.1配置相关联的环境变量

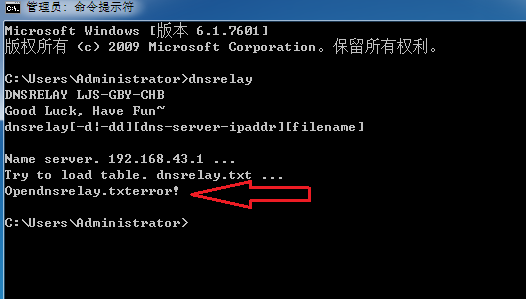
4.2使用缺省的[filename]时，需要将路径引导至文件所在的目录，或将代码中设置的默认文件地址写做绝对地址。



操作样例：

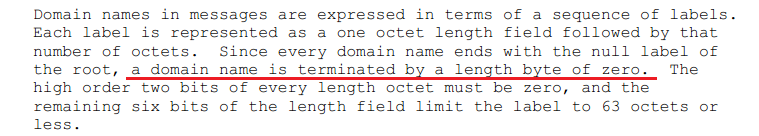


出错的情况：



1. 在解析DNS Response部分时发现的问题

5.1在最初思路中，我们想先过滤掉Header及Question部分，由于Question部中QName的长度不确定，我们采取了在QName中搜索直至查到“00”以确定其尾部。

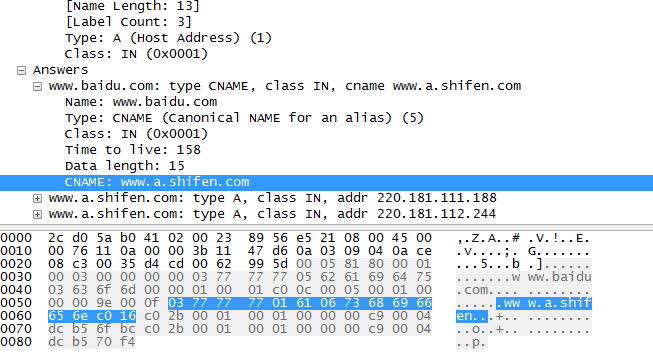


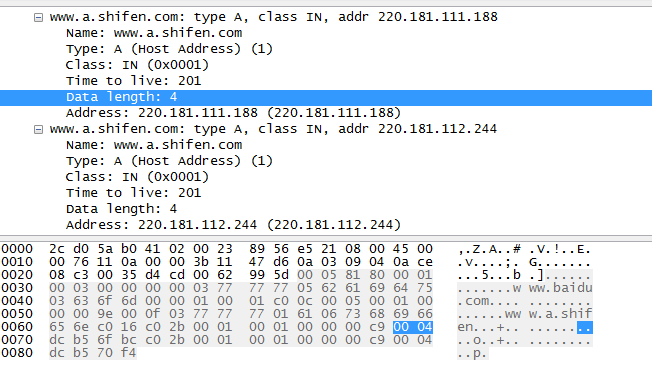
（RFC-1035 3.1 [Page 10]）

后经思考，发现可以直接使用接收DNS Request时套接字函数的返回值iSend = sendto()来表示Header及Question的长度。使得代码极大的化简。

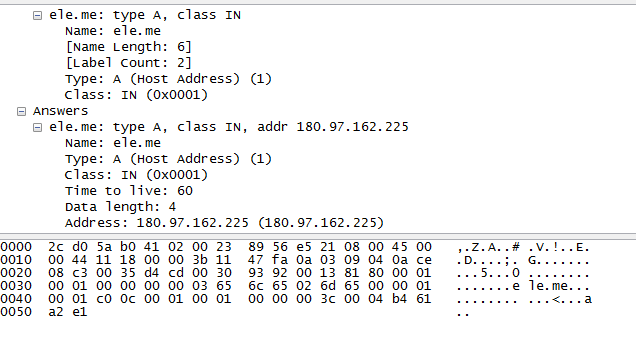
5.2在探索Answer部分时候，最初我们使用的www.baidu.com来观察其DNS Response，但是发现其返回了一些引用到其镜像服务器（如www.a.shifen.com）的内容，使得我们对数据报的处理无从下手，后尝试发现域名ele.me的返回DNS Response报文及其工整，方便观察。

www.baidu.com：

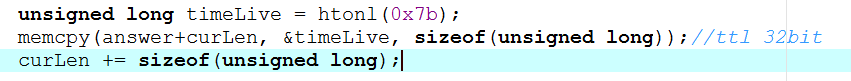




ele.me：

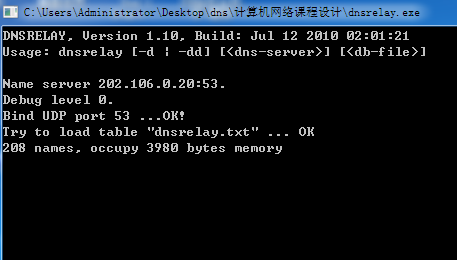


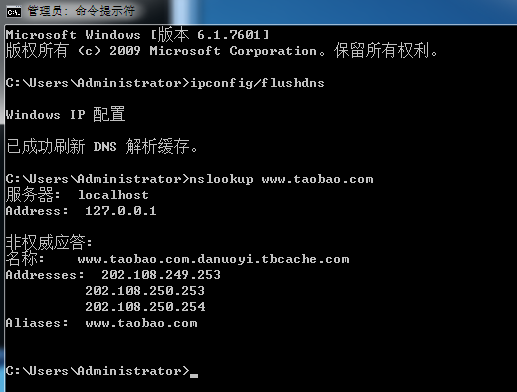
6. 构造DNS Response的RR部分时，TTL的类型长度占32bit,需要使用unsigned long类型，及对应的函数htonl()。

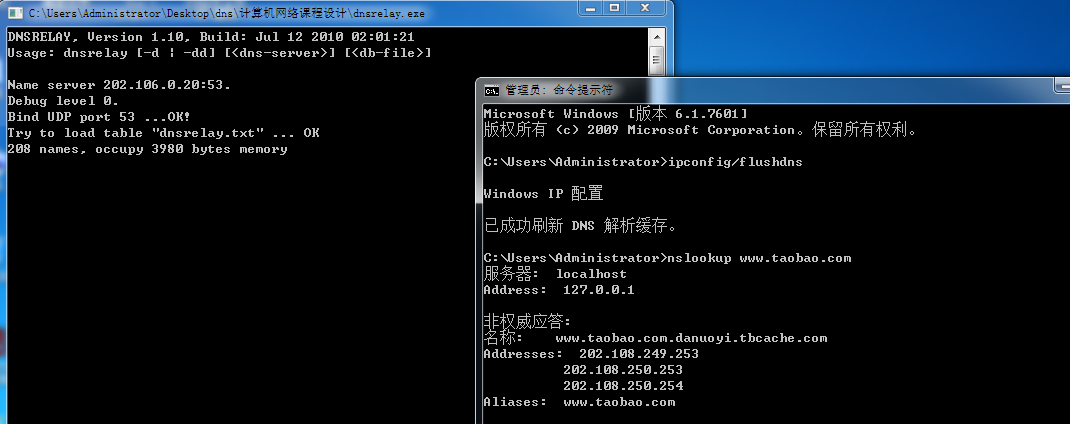


1. 针对nslookup指令，我们查询了网上的一些信息。但是仍然留有一些疑问。

运行示范代码：



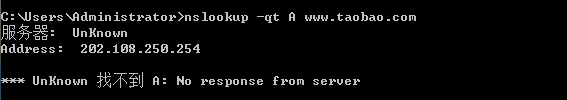




以老师提供的示范代码运行为例：

执行nslookup www.taobao.com关于为何会返回两个相同的IP地址(202.108.249.253)，我的考虑解释是指令发出了两个DNS请求分别对应IPv4及IPv6。

对nslookup 指令加以描述：nslookup –qt A www.taobao.com



返回了1个IP地址。

**课程设计工作总结**

这次实验是我们大四学年非常重要的一次实验，所以我们组成员对其极为重视，并且希望能够靠努力获得回报，我们从学期中期便开始筹划，并研究方法。在谈论中我们逐步认知了DNS Request 和 DNS Response报文的结构。在处理字节的网络顺序和主机顺序的问题中学习了htons(), htonl(),ntohs(), ntohl()函数的用法。细致讨论并修改代码耗费了我们连续数天时间。我们希望做出的程序能够满足老师给出的要求，并且一直尝试为其添加新功能，（如作为中继服务器时解析外部名字服务器返还的DNS Response里的IP）在研究过程中我们使用了WireShark来直观地观察数据报，来为程序的修改提供应用上的支持。此外还阅读了老师给的所有文档，以尽量丰富理论基础。希望我们做的成果能够得到认可。

**源代码及编译配置文件附于文件夹**