

计算机网络实验报告

实验名称 DNS中继服务器的实现

班级 2014211304

学号 2014211237 2014210441

姓名 代亚暄 张晓宇

系统的功能设计

支持普通的DNS服务

- 该程序可以完全取代原DNS服务程序，支持使用正常的ping，ftp，IE等功能，名字解析工作正常
- 当域名在本地Cache表中存在时，使用本地cache表的处理结果
- 表中未检测到该域名，则向因特网DNS服务器发出查询，并将结果返给客户端
- 当外部请求超时时，返回“域名不存在”的消息

支持针对本地cache表中的域名的特殊处理

- 检索结果为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）
- 检索结果为普通IP地址，则向客户返回这个地址（服务器功能）

支持多个客户端同时发送请求

- 通过多线程及消息ID转换实现同时服务多个客户端的要求

支持指定cache文件路径

- 通过指定不同的 filename 实现加载不同的本地域名文件

支持双级Debug模式

- 通过设置不同的debug level，打印出不同程度的debug信息，方便调试

模块划分

模块划分如下：

- DNSRelay
 - 对DNS请求进行解析
 - Cache中未出现的进行中继转发
 - 对外部服务器的回复进行转发
- Cache

- CacheFile基于Btree的实现,
- 提供Cache的载入, 索引, 更新和清空等功能, 以便DNSrelay调用
- DNSrecs
 - 对CacheFile的增删改, 以及列表显示, 便于Debug

软件流程图

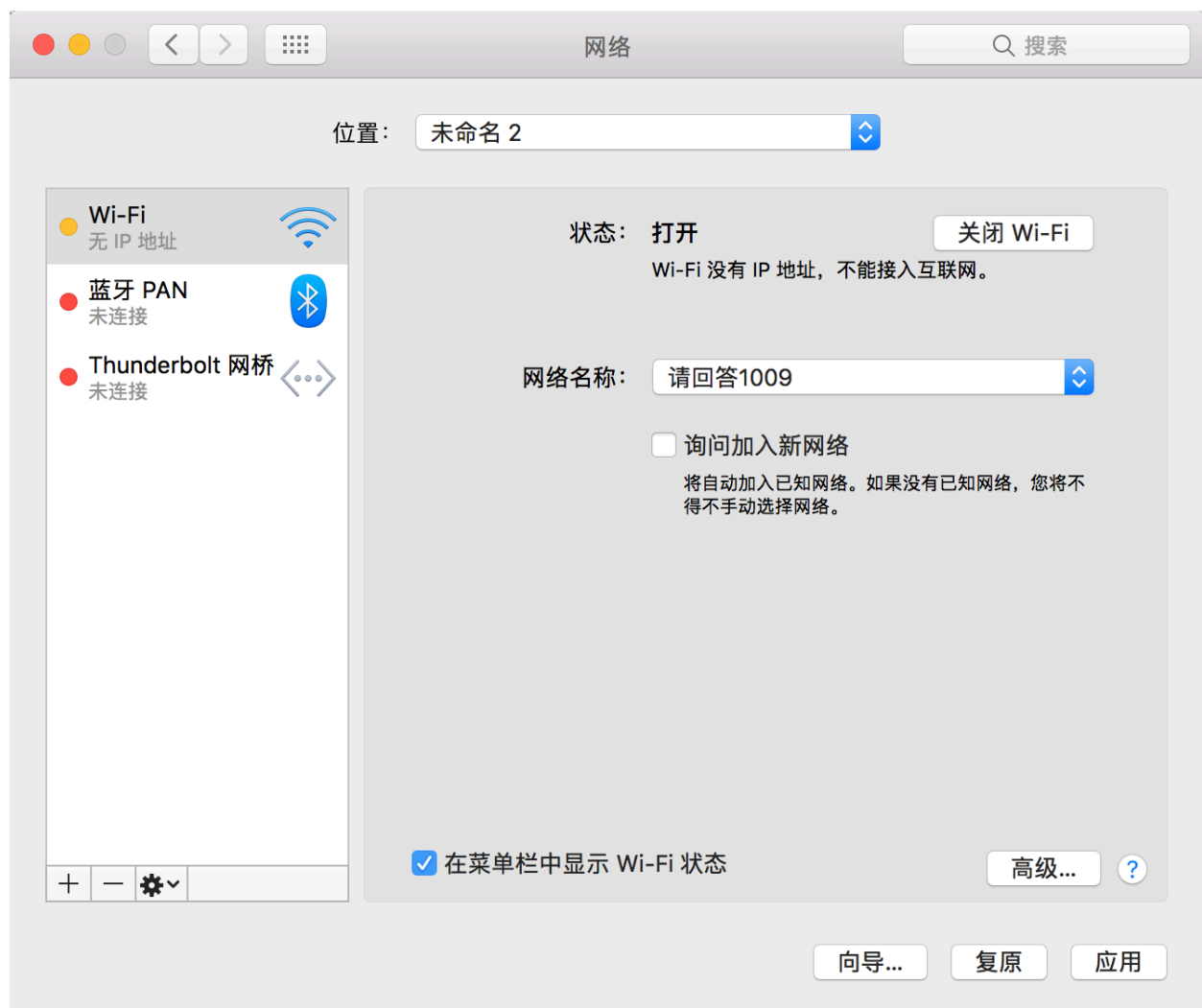
运行环境

操作系统: Mac OS

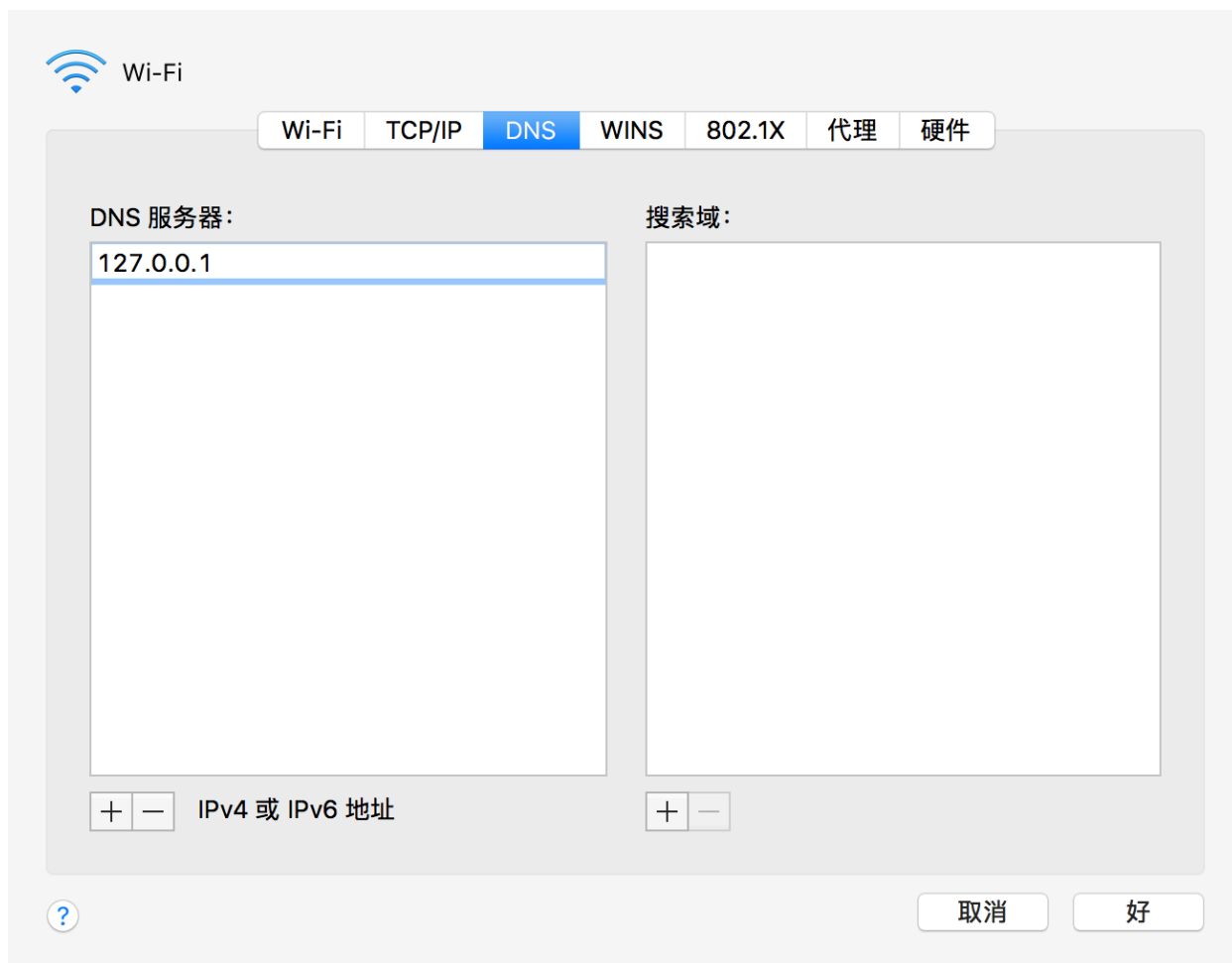
python 版本: python 2.7

使用步骤

1. 在系统偏好设置->网络中添加新的位置, 如 未命名2



2. 选择高级, 在DNS Tab下添加DNS服务器127.0.0.1



3. 修改位置为自定义2 的位置，然后就可以在53端口启动自己的dnsrelay.py了。注意需要在管理员模式下运行。
4. 运行命令为

```
sudo python dnsrelay.py -d d -f filename
```

说明：

d 对应debug level，可以为0，1，2，对应3个debug等级。

- 0 无调试信息输出
- 1 仅输出时间坐标，序号，客户端IP地址，查询的域名
- 2 输出cache查找结果，处理方式，是否超时，外部response详情等冗长的调试信息

filename为加载的cache文件路径。

测试用例及运行结果

测试Dnsrecs.py

```
Python Dnsrecs.py dnsrelay.txt all
```

```

zhangxiaoyudeMacBook-Pro:final zhangxiaoyu$ python dnsrecs.py dnsrelay.txt all
dnsrelay.txt
Loaded 906 records in.
Initial the new cache, capacity: 2000
Loaded DNSFile dnsrelay.txt
74.125.207.132 0-focus-opensocial.googleusercontent.com
74.125.207.132 0-open-opensocial.googleusercontent.com
byr.bt.pku 0.0.0.0
74.125.207.189 0.client-channel.google.com
74.125.207.189 0.docs.google.com
74.125.207.189 0.drive.google.com
0.0.0.0 008.cn
74.125.207.132 1-focus-opensocial.googleusercontent.com
74.125.207.132 1-open-opensocial.googleusercontent.com
74.125.207.132 1-ps.googleusercontent.com
74.125.207.132 1.bp.blogspot.com
74.125.207.189 1.client-channel.google.com
74.125.207.189 1.docs.google.com
74.125.207.189 1.drive.google.com
74.125.207.189 10.docs.google.com
74.125.207.189 10.drive.google.com
74.125.207.189 11.docs.google.com
74.125.207.189 11.drive.google.com
74.125.207.189 12.docs.google.com
74.125.207.189 12.drive.google.com
74.125.207.189 13.docs.google.com

```

```
Python dnsrecs.py dnsrelay.txt add tagineer.cn 0.0.0.0
```

```

zhangxiaoyudeMacBook-Pro:final zhangxiaoyu$ python dnsrecs.py dnsrelay.txt add t
agineer.cn 0.0.0.0
dnsrelay.txt
Loaded 906 records in.
Initial the new cache, capacity: 2000
Loaded DNSFile dnsrelay.txt
added tagineer.cn 0.0.0.0
saved modified cache file indnsrelay.txt

```

```
Python dnsrecs.py dnsrelay.txt del bupt.edu.cn
```

```

zhangxiaoyudeMacBook-Pro:final zhangxiaoyu$ python dnsrecs.py dnsrelay.txt del b
upt.edu.cn
dnsrelay.txt
Loaded 907 records in.
Initial the new cache, capacity: 2000
Loaded DNSFile dnsrelay.txt
delete bupt.edu.cn
saved modified cache file indnsrelay.txt

```

测试DNSrelay.py

打开dnsrelay.py

```
zhangxiaoyudeMacBook-Pro:final zhangxiaoyu$ sudo python dnsrelay.py
Password:
dnsrelay.txt
Loaded 907 records in.
Initial the new cache, capacity: 200
Listening
```

测试屏蔽功能

```
nslookup 008.cn
```

```
zhangxiaoyudeMacBook-Pro:~ zhangxiaoyu$ nslookup 008.cn
Server:          127.0.0.1
Address:         127.0.0.1#53

Non-authoritative answer:
*** Can't find 008.cn: No answer
```

测试cache中存在的结果

```
nslookup 0.drive.google.com
```

```
zhangxiaoyudeMacBook-Pro:~ zhangxiaoyu$ nslookup 0.drive.google.com
Server:          127.0.0.1
Address:         127.0.0.1#53

Non-authoritative answer:
Name:   0.drive.google.com
Address: 74.125.207.189
```

测试cache中不存在的结果

```
nslookup github.com
```

```
zhangxiaoyudeMacBook-Pro:~ zhangxiaoyu$ nslookup github.com
Server:          127.0.0.1
Address:         127.0.0.1#53

Non-authoritative answer:
Name:   github.com
Address: 192.30.255.112
```

测试不同的 debug level 的结果

默认情况下，debug level 为0，不输出任何中间结果

```
zhangxiaoyudeMacBook-Pro:final zhangxiaoyu$ sudo python dnsrelay.py
Password:
dnsrelay.txt
Loaded 907 records in.
Initial the new cache, capacity: 200
Listening
```

debug level 为1时，输出简单的中间结果

```
zhangxiaoyudeMacBook-Pro:final zhangxiaoyu$ sudo python dnsrelay.py -d 1
dnsrelay.txt
Loaded 907 records in.
Initial the new cache, capacity: 200
Listening
Time: 1496340671.964128, no = 31230, ip = ('127.0.0.1', 59740) , q = 'github.com.'
Time: 1496340674.683762, no = 34445, ip = ('127.0.0.1', 52523) , q = '0.drive.google.com.'
Time: 1496340678.275321, no = 59139, ip = ('127.0.0.1', 51757) , q = '008.cn.'
```

debug level 为2时，输出所有debug信息

```
zhangxiaoyudeMacBook-Pro:final zhangxiaoyu$ sudo python dnsrelay.py -d 2
dnsrelay.txt
Loaded 907 records in.
Initial the new cache, capacity: 200
Listening
Time: 1496340755.690283, no = 64692, ip = ('127.0.0.1', 59030) , q = '008.cn.'
Query qname='008.cn', eixt in cache
ip='0.0.0.0'
Time: 1496340760.113646, no = 31604, ip = ('127.0.0.1', 56752) , q = '0.drive.google.com.'
Query qname='0.drive.google.com', eixt in cache
ip='74.125.207.189'
Time: 1496340763.640364, no = 54243, ip = ('127.0.0.1', 49581) , q = 'github.com.'
Query qname='github.com', doesn't exit in cache,
rd=1,request from foreign server
foreign_res=''; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 56633\n;; flags: qr rd ra; QUERY:
1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 0\n;; QUESTION SECTION:\n;github.com.
IN      A\n;; ANSWER SECTION:\ngithub.com.          15      IN      A      192.30.255.113\ngit
hub.com.          15      IN      A      192.30.255.112'
```

调试中遇到并解决的问题

1. test1等不完整域名会在调用dnspython模块功能，通过nslookup进行查询时自动在解析后得到域名后加.workgroup后缀，导致无法在cache中按应有查询得到地址。对这种情况在cache.py中加入了判断和过滤并单独处理。
2. 绑定端口报错。使用sudo并将相关占用端口的进程关闭。

心得体会

1. 事先进行独立功能的模块划分；
2. 使用信息尽量丰富的中间输出，并结合python报错帮助定位和调试代码bug；
3. 善用文档和标准库帮助提高开发效率。

