

1. 功能设计

实验的内容如下

设计一个DNS服务器程序，读入“域名-IP地址”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，要求实现三种检索结果：

- 不良网站（IP地址为0.0.0.0）拦截功能
检索结果IP地址为0.0.0.0，则向客户端返回“域名不存在”的报错消息
- 服务器功能 检索结果为普通IP地址，则向客户返回这个地址
- 中继功能：对照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

同时需要设计几个调试信息的等级

- 调试信息级别1
1.输出查询的域名及其对应的IP地址 2.使用默认名字服务器 3.使用默认配置文件
- 调试信息级别2
1.输出时间坐标，序号，客户端IP地址，查询的域名 2.使用指定的名字服务器 3.使用指定的配置文件
- 调试信息级别3
1.输出数据报的内容、接收和发送数据包的地址以及端口、查询的IP地址及其域名 2.使用指定的名字服务器 3.使用默认配置文件

功能的设计使用文档给出的即可。

2. 模块划分

其核心在于服务器的功能和DNS的解析。

为了支持多个客户端的功能，使用的服务器为 `socketserver.ThreadingUDPServer`

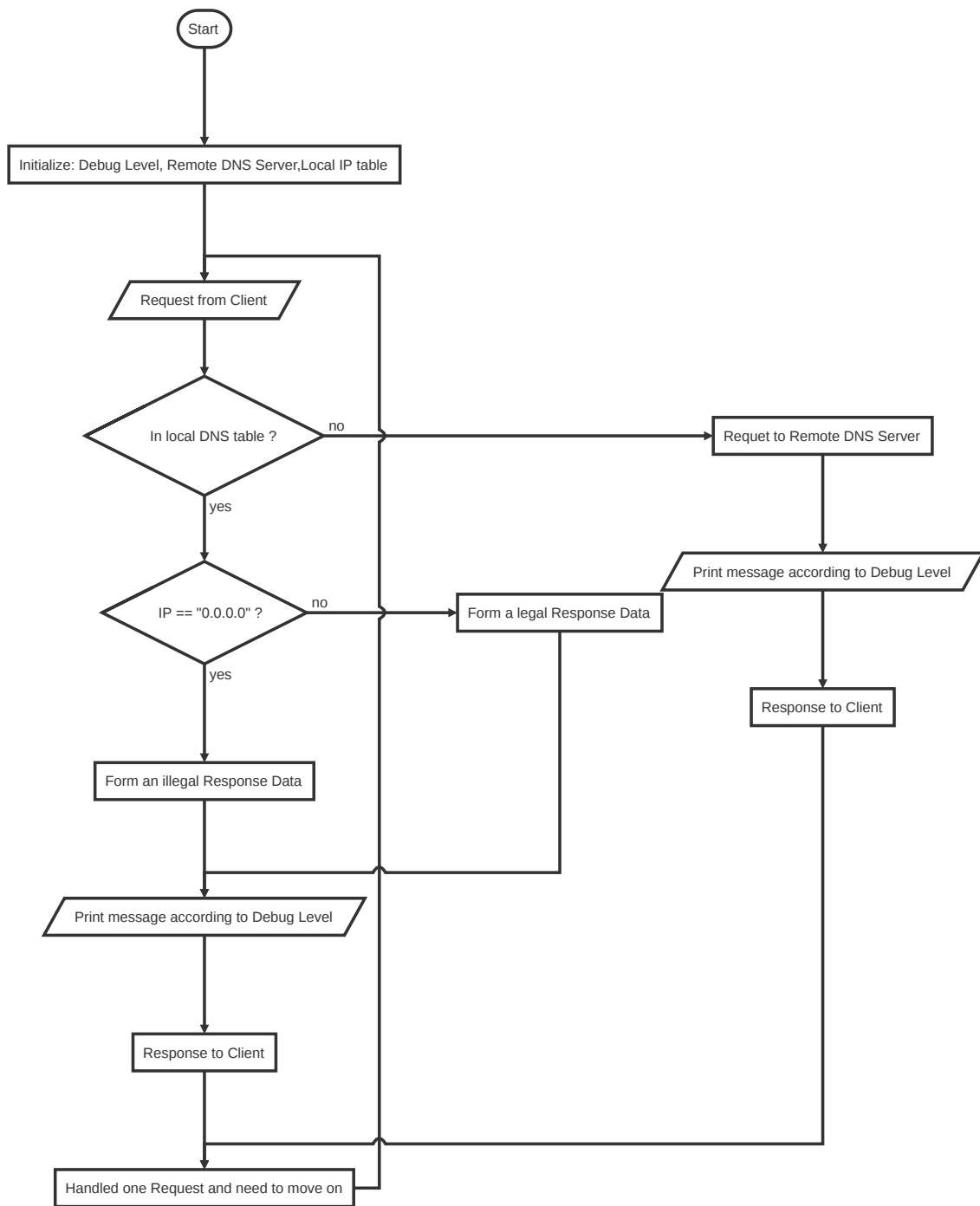
为了便于解析DNS数据，使用的库为 `dnslib`

充分利用库的情况下，模块的划分为：

- 根据参数判断调试级别。并且设定：
 - 外界的DNS服务器IP地址
 - 内部的IP文件
- DNS请求处理

需要判断是否在内部IP文件中找到，如果找到，就构建DNS回答，如果找不到就向外部的DNS服务器请求，然后再回答。

3. 系统流程图



4. 程序运行说明

使用原本在 `/etc/resolv.conf` 文件内的外部DNS服务器作为默认的，速度会比较快。

5. 测试用例及运行结果

5.1. 测试环境

为了模拟真实的环境，使用两台电脑。（以下实验在一台电脑上仍然可以顺利测试）

- Server
 - Arch Linux
 - Python 3.7.0
 - 需要使用的包均可以通过 `sudo pip3 install xxx` 来安装
- Client
 - Deepin 15.8 unstable

5.2. 基本使用

5.2.1 准备文件

- 在server机上
首先需要准备的ip地址文件(包括 `dnsrelay.txt` , `dns-table.txt`)格式如下

```
1 1.2.3.4 abc.com
2 0.0.0.0 wrong.com
3 185.199.109.153 drdh.com
```

5.2.2. 获取IP并修改

- 在server机上
`ip link` 或者 `ip addr show` 查看本机的ip地址 `114.214.176.57` 每次不一样
编辑 `/etc/resolv.conf` 使DNS服务器为上面的地址
编辑源码的ip字段, 即最前面的 `localIP="xxx.xxx.xxx.xxx"` 部分
- 在client机上
使用 `create_ap` 创建热点 `sudo create_ap wlp9s0 enp0s20f0u3 Arch archwifi` 在另一个电脑上连接,
(这一条不是必须的)
然后修改DNS服务器(`/etc/resolv.conf`)为上面的ip地址, 就可以正常查询了

5.3. 测试结果

调试级别1

命令为

```
1 sudo python3 DNS_Server.py 1
```

```
→ src git:(master) x sudo python3 DNS_Server.py 1
Debug level: 1
DNS Server: 202.106.0.20
IP table: ./dnsrelay.txt
The DNS server is running at %s ... 222.195.84.207
```

测试多线程如下：

开两个终端。依次输入如下的命令

```
→ temp nslookup wrong.com

→ ~ nslookup test0
Server:      222.195.84.207
Address:     222.195.84.207#53

Name:   test0
Address: 61.135.169.121
Name:   test0
Address: 61.135.169.121
```

可以发现第一个还没有输出成功，第二个就已经完成。

调试级别2

```
1 | sudo python3 DNS_Server.py 2 --addr 202.38.64.56 --ipfile dns-table.txt
```

依次查询如下

```
→ ~ nslookup abc.com
Server:      222.195.84.207
Address:     222.195.84.207#53

Name:   abc.com
Address: 1.2.3.4
Name:   abc.com
Address: 1.2.3.4

→ ~ nslookup wrong.com
Server:      222.195.84.207
Address:     222.195.84.207#53

wrong.com    text = "illegal website or website doesn't exist"
wrong.com    text = "illegal website or website doesn't exist"

→ ~ nslookup drdh.com
Server:      222.195.84.207
Address:     222.195.84.207#53

Name:   drdh.com
Address: 185.199.109.153
Name:   drdh.com
Address: 185.199.109.153

→ ~ nslookup baidu.com
Server:      222.195.84.207
Address:     222.195.84.207#53

Non-authoritative answer:
Name:   baidu.com
Address: 123.125.115.110
Name:   baidu.com
Address: 220.181.57.216
```

可以注意到每个分别代表了自定义的域名与ip, 定义为0.0.0.0的地址的返回错误信息, 转发到外部DNS Server的内容。

在server的输出如下

```

→ src git:(master) x sudo python3 DNS_Server.py 2 --addr 202.38.64.56 --ipfile dns-table.txt
[sudo] password for drdh:
Debug level: 2
DNS Server: 202.38.64.56
IP table: dns-table.txt
The DNS server is running at %s ... 222.195.84.207
Sun Dec 16 17:16:48 2018 2 ('114.214.178.57', 56179) app.jianguoyun.com.
Sun Dec 16 17:16:48 2018 2 ('114.214.178.57', 56179) app.jianguoyun.com.
Sun Dec 16 17:17:36 2018 3 ('114.214.178.57', 38370) drdh.com.
Sun Dec 16 17:17:36 2018 4 ('114.214.178.57', 42030) drdh.com.
Sun Dec 16 17:17:37 2018 5 ('114.214.178.57', 47000) app.jianguoyun.com.
Sun Dec 16 17:17:37 2018 6 ('114.214.178.57', 38200) app.jianguoyun.com.
Sun Dec 16 17:17:42 2018 8 ('114.214.178.57', 46318) wrong.com.
Sun Dec 16 17:17:42 2018 9 ('114.214.178.57', 37823) wrong.com.
Sun Dec 16 17:17:45 2018 10 ('222.195.84.207', 39343) clientservices.googleapis.com.
Sun Dec 16 17:17:45 2018 11 ('222.195.84.207', 34487) clientservices.googleapis.com.
Sun Dec 16 17:17:48 2018 12 ('114.214.178.57', 39538) abc.com.
Sun Dec 16 17:17:57 2018 14 ('222.195.84.207', 59763) www.archlinux.org.
Sun Dec 16 17:17:57 2018 14 ('222.195.84.207', 59763) www.archlinux.org.
Sun Dec 16 17:18:10 2018 15 ('114.214.178.57', 59938) abc.com.
Sun Dec 16 17:18:10 2018 16 ('114.214.178.57', 57625) abc.com.
Sun Dec 16 17:18:14 2018 17 ('114.214.178.57', 33501) wrong.com.
Sun Dec 16 17:18:14 2018 18 ('114.214.178.57', 53912) wrong.com.
Sun Dec 16 17:18:17 2018 19 ('114.214.178.57', 53900) drdh.com.
Sun Dec 16 17:18:17 2018 20 ('114.214.178.57', 42244) drdh.com.
Sun Dec 16 17:18:20 2018 21 ('114.214.178.57', 47564) baidu.com.
Sun Dec 16 17:18:20 2018 22 ('114.214.178.57', 48792) baidu.com.

```

注意到各类调试信息的正常输出。

测试打开浏览器如下

使用浏览器打开百度界面

```

→ src git:(master) x sudo python3 DNS_Server.py 2 --addr 202.38.64.56 --ipfile dns-table.txt
[sudo] password for drdh:
Debug level: 2
DNS Server: 202.38.64.56
IP table: dns-table.txt
The DNS server is running at %s ... 222.195.84.207
Sun Dec 16 17:46:03 2018 1 ('114.214.178.57', 56213) www.baidu.com.

```

界面可以正常打开。

调试级别3

命令如下

```
1 | sudo python3 DNS_Server.py 3 --addr 202.38.64.56
```

这一条主要看信息的输出。

测试命令如下

```
→ ~ nslookup test0
Server:      222.195.84.207
Address:     222.195.84.207#53

Name:   test0
Address: 61.135.169.121
Name:   test0
Address: 61.135.169.121

→ ~ nslookup baidu.com
Server:      222.195.84.207
Address:     222.195.84.207#53

Non-authoritative answer:
Name:   baidu.com
Address: 220.181.57.216
Name:   baidu.com
Address: 123.125.115.110
```

分别对应的输出信息如下

```
src git:(master) x sudo python3 DNS_Server.py 3 --addr 202.38.64.56
Debug level: 3
DNS Server: 202.38.64.56
IP table: ./dnsrelay.txt
The DNS server is running at %s ... 222.195.84.207
, 34487) clientservices.googleapis.com.
, 39538) abc.com.
, 59763) www.archlinux.org.
, 59763) www.archlinux.org.
, 59938) abc.com.
, 57625) abc.com.
, 33581) abc.com.
*****Request Data*****
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 20820
;; flags: rd; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;test0. baidu.com. IN A
*****Client Address*****
('114.214.178.57', 38778)
*****Search Name*****
test0.
*****Search IP*****
61.135.169.121

*****Request Data*****
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 581
;; flags: rd; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;test0. IN AAAA
*****Client Address*****
('114.214.178.57', 48557)
*****Search Name*****
test0.
*****Search IP*****
61.135.169.121
```

- 调试信息级别 2
 1. 输出时间坐标, 序号, 客户端 IP 地址, 查询
 2. 使用指定的名字服务器
 3. 使用指定的配置文件
- 调试信息级别 3
 1. 输出数据报的内容、接收和发送数据包的地
 2. 使用指定的名字服务器
 3. 使用默认配置文件

实验安排

实验环境

操作系统 Windows , Ubuntu
编程语言不限

实验报告

1. 功能设计
2. 模块划分
3. 系统流程图
4. 程序运行说明

*****Request Data*****
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 27691
;; flags: rd; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;baidu.com.
*****Client Address*****
('114.214.178.57', 60712)
*****Search Name*****
baidu.com.
*****Search IP*****
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 27691
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;baidu.com.
;; ANSWER SECTION:
baidu.com. 462 IN A 220.181.57.216
baidu.com. 462 IN A 123.125.115.110

实验安排

*****Request Data*****
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8331
;; flags: rd; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;baidu.com.
*****Client Address*****
('114.214.178.57', 59195)
*****Search Name*****
baidu.com.
*****Search IP*****
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8331
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0
;; QUESTION SECTION:
;baidu.com.
;; AUTHORITY SECTION:
baidu.com. 6939 IN SOA dns.baidu.com. sa.baidu.com. 2012139487 300 300 25920 00 7200

实验环境

*****Request Data*****
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8331
;; flags: rd; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;baidu.com.
*****Client Address*****
('114.214.178.57', 59195)
*****Search Name*****
baidu.com.
*****Search IP*****
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8331
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0
;; QUESTION SECTION:
;baidu.com.
;; AUTHORITY SECTION:
baidu.com. 6939 IN SOA dns.baidu.com. sa.baidu.com. 2012139487 300 300 25920 00 7200

提交内容

*****Request Data*****
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8331
;; flags: rd; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;baidu.com.
*****Client Address*****
('114.214.178.57', 59195)
*****Search Name*****
baidu.com.
*****Search IP*****
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8331
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0
;; QUESTION SECTION:
;baidu.com.
;; AUTHORITY SECTION:
baidu.com. 6939 IN SOA dns.baidu.com. sa.baidu.com. 2012139487 300 300 25920 00 7200

实验报告

*****Request Data*****
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8331
;; flags: rd; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;baidu.com.
*****Client Address*****
('114.214.178.57', 59195)
*****Search Name*****
baidu.com.
*****Search IP*****
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8331
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0
;; QUESTION SECTION:
;baidu.com.
;; AUTHORITY SECTION:
baidu.com. 6939 IN SOA dns.baidu.com. sa.baidu.com. 2012139487 300 300 25920 00 7200

程序运行说明

*****Request Data*****
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8331
;; flags: rd; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;baidu.com.
*****Client Address*****
('114.214.178.57', 59195)
*****Search Name*****
baidu.com.
*****Search IP*****
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8331
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0
;; QUESTION SECTION:
;baidu.com.
;; AUTHORITY SECTION:
baidu.com. 6939 IN SOA dns.baidu.com. sa.baidu.com. 2012139487 300 300 25920 00 7200

遇到的问题及心得体会

*****Request Data*****
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8331
;; flags: rd; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;baidu.com.
*****Client Address*****
('114.214.178.57', 59195)
*****Search Name*****
baidu.com.
*****Search IP*****
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8331
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0
;; QUESTION SECTION:
;baidu.com.
;; AUTHORITY SECTION:
baidu.com. 6939 IN SOA dns.baidu.com. sa.baidu.com. 2012139487 300 300 25920 00 7200

源代码、可执行文件、域名、IP 对照表

实验安排

实验环境

提交内容

实验报告

程序运行说明

遇到的问题及心得体会

源代码、可执行文件、域名、IP 对照表

6. 遇到的问题及心得体会

遇到的主要问题是需要了解DNS包的各个字段的含义，并依次来构建Response和Request。

心得体会是这实验挺有意思的。

Reference

[DNS 介绍与粗实现](#); [相关代码](#)

[DNS 使用dnslib案例](#); [相关代码](#)

[dnslib 介绍](#)