HW1 Report ---Local DNS server

Name: 韩铠赢 Student ID: 121090155

1. Introduction

I use OOP for this local DNS server, to change the flag, please change the value in the `def __init___()`. `self.flag = 0` is for iterative DNS query from the root server, `self.flag = 1` is for DNS query from pubilc DNS server. Since the Local DNS IP and port I used is the same as given in the tutorial, please use UNIX commmand `dig` to test the program, e.g. `dig @127.0.0.1 -p 1234 www.baidu.com`.

```
class dnserver:
    def __init__(self):
        self.ip = "127.0.0.1"
        self.port = 1234
        self.cache = {}
        self.flag = 0
        self.sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
        self.sock.bind((self.ip, self.port))
```

- **Notation** I choose to output the whole process to the server terminal, so you can see where the program stucks if it meets any mistakes, please try to test more times since the program may meet timeout or refused when doing iterative query. For public server, I set the default as `8.8.8.8` of the google server, so it may need to change to another one in the public server pool or use proxy to connect to global Internet. One more important thing is that, DNS cache poisoning (DNS污染) is unavoidable since DNS query via UDP is not encrypted, so ISPs may easily change the ressponse package which may lead to error IP address, resulting in 404 or 503 error when connect via IP using browsers.

2. Local DNS server

This is the methods I used to implement the local DNS server:

1. `__init__(self)` to initialize the settings of the local DNS server.
2. `listen_q(self)` using socket to listen to the client terminal, and determine whether using cache or doing iterative/public DNS query to response.
3. `update_cache(self, qname, response)` to update the cache with the requested domain and cooresponding response package.
4. `ask_publicdns(self, data, clientaddr)` transmit the client query to public DNS server and transmit the public server's response to client.
5. `extract_addRR(self, source, dest)` to add the CNAME RR to the final response to client when handling DNS query that need CNAME.
6. `handlecname(self, cname, data, clientaddr, pre_response)` start a new iterative request with the CNAME domain, which is similar to normal, only to handling CNAME cases.
7. `ask_dns(self, data, clientaddr)` query the client's requested domain name by name in reverse order doing NS query, e.g. `1. com. 2. google.com`, and start a A query when the

domain becomes `www.google.com` in case `dig @127.0.0.1 -p 1234 www.google.com`. If meets CNAME response in the final step, it will turn to `handlecname()` to deal with it.

8. `main()` to start the DNS server

Test result:

- Note that the server pass by is root ip + ip(if it responsed) + corresponding domain + ...

```
linux@linux-virtual-machine:~$ dig @127.0.0.1 -p 1234 www.baidu.com
;; Warning: query response not set

; <<>> DiG 9.18.12-0ubuntu0.22.04.3-Ubuntu <<>> @127.0.0.1 -p 1234 www.baidu.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 52856
;; flags: rd; QUERY: 1, ANSWER: 3, AUTHORITY: 5, ADDITIONAL: 9
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;www.baidu.com.                 IN      A

;; ANSWER SECTION:
www.a.shifen.com.       120     IN      A       157.148.69.80
www.a.shifen.com.       120     IN      A       157.148.69.74
www.baidu.com.          1200    IN      CNAME   www.a.shifen.com.

;; AUTHORITY SECTION:
a.shifen.com.           1200    IN      NS      ns4.a.shifen.com.
a.shifen.com.           1200    IN      NS      ns5.a.shifen.com.
a.shifen.com.           1200    IN      NS      ns1.a.shifen.com.
a.shifen.com.           1200    IN      NS      ns2.a.shifen.com.
a.shifen.com.           1200    IN      NS      ns3.a.shifen.com.

;; ADDITIONAL SECTION:
ns1.a.shifen.com.       300     IN      A       110.242.68.42
ns2.a.shifen.com.       600     IN      A       220.181.33.32
ns3.a.shifen.com.       300     IN      A       36.155.132.12
ns3.a.shifen.com.       300     IN      A       153.3.238.162
ns4.a.shifen.com.       300     IN      A       14.215.177.229
ns4.a.shifen.com.       300     IN      A       111.20.4.28
ns5.a.shifen.com.       600     IN      A       180.76.76.95
ns5.a.shifen.com.       600     IN      AAAA    240e:bf:b801:1006:0:ff:b04f:346b
ns5.a.shifen.com.       600     IN      AAAA    240e:940:603:a:0:ff:b08d:239d

;; Query time: 1243 msec
;; SERVER: 127.0.0.1#1234(127.0.0.1) (UDP)
;; WHEN: Sun Oct 15 19:19:37 HKT 2023
;; MSG SIZE  rcvd: 348

linux@linux-virtual-machine:~$ S
```

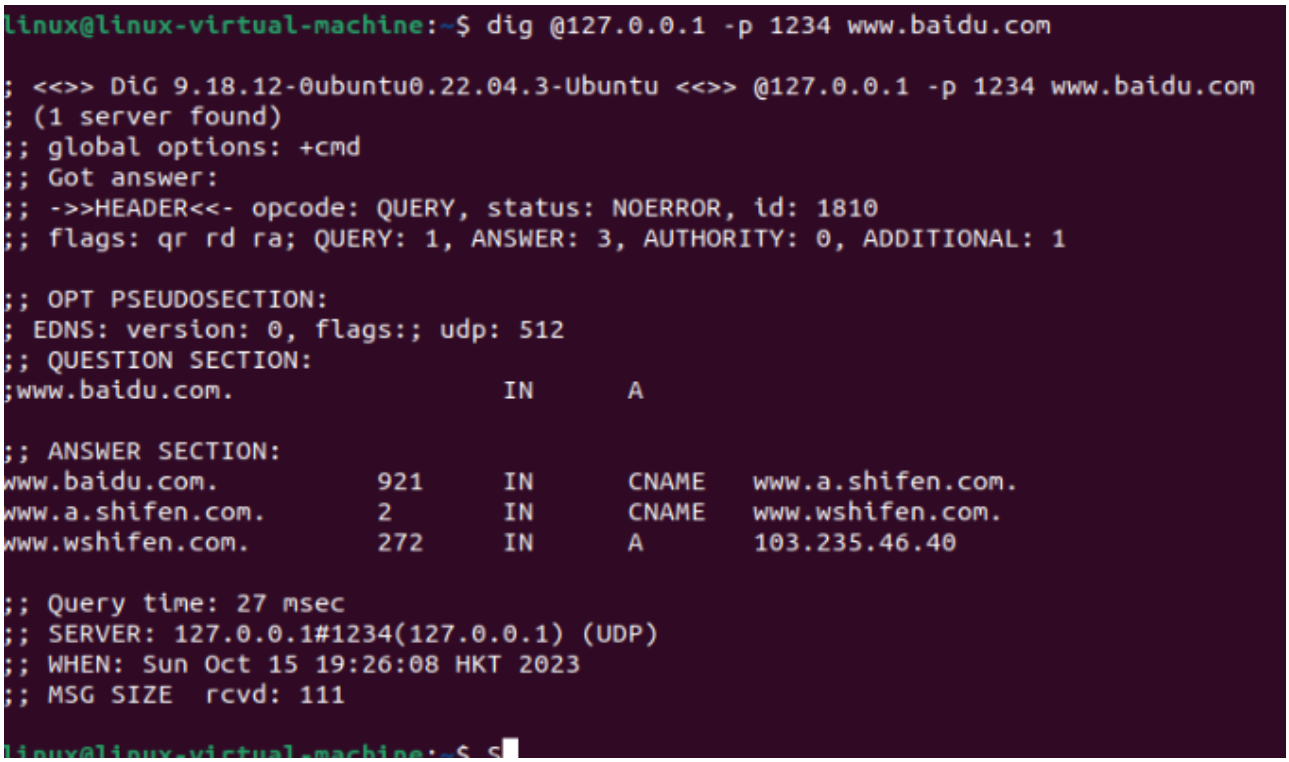
```
The servers pass-by during iterative query:
192.33.4.12
192.55.83.30
b.gtld-servers.net.
220.181.33.31
ns2.baidu.com.
```

```
The servers pass-by during iterative query for cname:
192.33.4.12
192.55.83.30
c.gtld-servers.net.
110.242.68.134
dns.baidu.com.
```

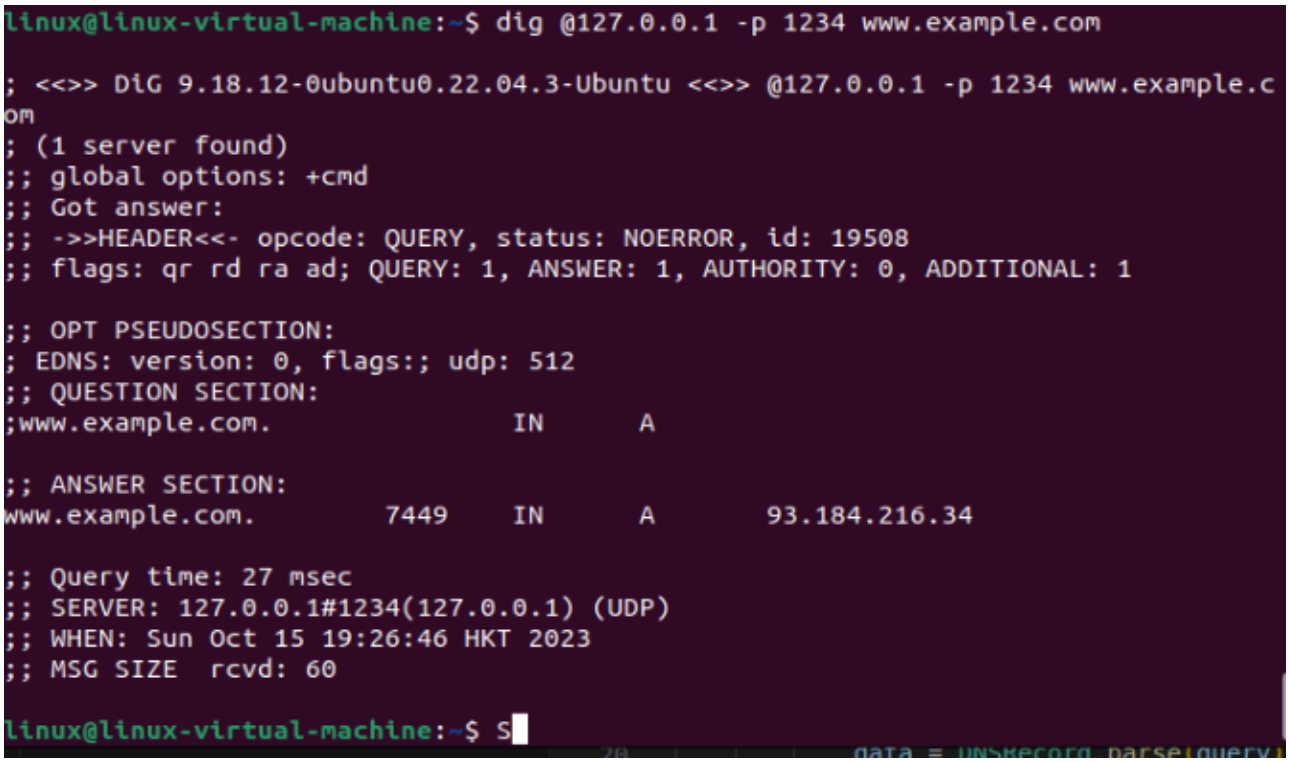
```
linux@linux-virtual-machine:~$ dig @127.0.0.1 -p 1234 www.example.com

; <<>> DiG 9.18.12-0ubuntu0.22.04.3-Ubuntu <<>> @127.0.0.1 -p 1234 www.example.c
om
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 27843
;; flags: qr aa rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        86400   IN      A       93.184.216.34

;; Query time: 623 msec
;; SERVER: 127.0.0.1#1234(127.0.0.1) (UDP)
;; WHEN: Sun Oct 15 19:23:48 HKT 2023
;; MSG SIZE  rcvd: 49

linux@linux-virtual-machine:~$
```

```
The servers pass-by during iterative query:
192.33.4.12
192.55.83.30
i.gtld-servers.net.
a.iana-servers.net.
```

```
^Clinux@linux-virtual-machine:~$ dig @127.0.0.1 -p 1234 www.google.com

; <<>> DiG 9.18.12-0ubuntu0.22.04.3-Ubuntu <<>> @127.0.0.1 -p 1234 www.google.co
m
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 17470
;; flags: qr aa rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;www.google.com.                        IN      A

;; ANSWER SECTION:
www.google.com.         300     IN      A       142.250.66.100

;; Query time: 639 msec
;; SERVER: 127.0.0.1#1234(127.0.0.1) (UDP)
;; WHEN: Sun Oct 15 19:25:17 HKT 2023
;; MSG SIZE  rcvd: 48

linux@linux-virtual-machine:~$
```



```
linux@linux-virtual-machine:~$ dig @127.0.0.1 -p 1234 www.baidu.com

; <<>> DiG 9.18.12-0ubuntu0.22.04.3-Ubuntu <<>> @127.0.0.1 -p 1234 www.baidu.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 1810
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;www.baidu.com.                 IN      A

;; ANSWER SECTION:
www.baidu.com.          921     IN      CNAME   www.a.shifen.com.
www.a.shifen.com.       2       IN      CNAME   www.wshifen.com.
www.wshifen.com.        272     IN      A       103.235.46.40

;; Query time: 27 msec
;; SERVER: 127.0.0.1#1234(127.0.0.1) (UDP)
;; WHEN: Sun Oct 15 19:26:08 HKT 2023
;; MSG SIZE  rcvd: 111

linux@linux-virtual-machine:~$ S
```



```
linux@linux-virtual-machine:~$ dig @127.0.0.1 -p 1234 www.example.com

; <<>> DiG 9.18.12-0ubuntu0.22.04.3-Ubuntu <<>> @127.0.0.1 -p 1234 www.example.c
om
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 19508
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        7449    IN      A       93.184.216.34

;; Query time: 27 msec
;; SERVER: 127.0.0.1#1234(127.0.0.1) (UDP)
;; WHEN: Sun Oct 15 19:26:46 HKT 2023
;; MSG SIZE  rcvd: 60

linux@linux-virtual-machine:~$ S
```