

# Lab5 Web Proxy Server

- 学号:1813075
- 姓名:刘茵

## Lab5 Web Proxy Server

1. 运行过程baidu
2. 运行过程linux
3. 源代码

- 采用测试链接:

[www.baidu.com](http://www.baidu.com)

[www.linux.com](http://www.linux.com)

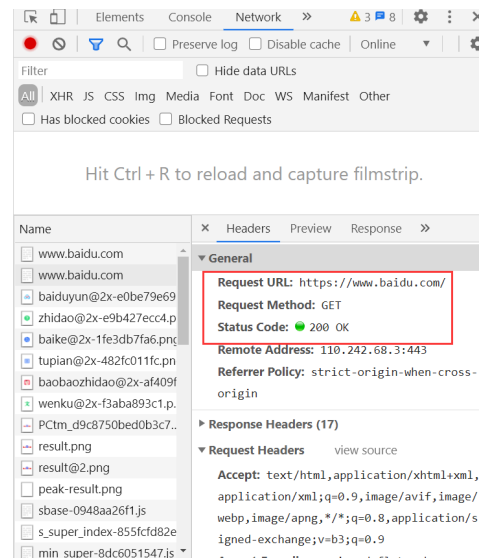
## 1. 运行过程baidu

1. 在本地运行代理服务器程序

```
D:\pythonProject\untitled>ProxyServer.py localhost
Ready to serve...
```

2. 输入 <http://localhost:8990/www.baidu.com> 开始测试

视频 贴吧 学术 更多



(网页最终显示结果)

- 运行成功，第一遍搜索结果显示：

```
D:\pythonProject\untitled>ProxyServer.py localhost
Ready to serve...
Received a connection from: ('127.0.0.1', 65465)
从客户端传来的信息 GET /www.baidu.com HTTP/1.1
Host: localhost:8999
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Accept-Encoding: gzip, deflate, br
Accept-Language: zh-CN,zh;q=0.9
```

◦

```
/www.baidu.com
filename: www.baidu.com
filetouse /www.baidu.com

开始检查代理服务器中是否存在文件：
文件不在代理服务器当中，开始向远端服务器请求网页
HOST NAME baidu.com
buffer: b'HTTP/1.1 200 OK\r\nDate: Wed, 23 Dec 2020 14:05:54 GMT\r\nServer: Apache\r\nLast-Modified: Tue, 12 Jan 2010 13:48
ETag: "51-47cf7e6ee8400"\r\nAccept-Ranges: bytes\r\nContent-Length: 81\r\nCache-Control: max-age=86400\r\nExpires: Thu, 24
:05:54 GMT\r\nConnection: Close\r\nContent-Type: text/html\r\n\r\n<html>\n<meta http-equiv="refresh" content="0;url=http://
om/">\n</html>\n'
```

代理服务器获得客户端的GET请求

代理服务器中不存在查询的html文件，向远端请求网页并缓存到本地。

远端服务器会响应代理服务器的请求，本地会生成一个缓存文件，存于代理服务器同一目录下，且网页会显示链接内容。

- 第二次循环显示结果：

```
Ready to serve...
Received a connection from: ('127.0.0.1', 65466)
从客户端传来的信息 GET /www.baidu.com HTTP/1.1
Host: localhost:8999
Connection: keep-alive
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/sig
=b3;q=0.9
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Accept-Encoding: gzip, deflate, br
Accept-Language: zh-CN,zh;q=0.9
```

```
/www.baidu.com
filename: www.baidu.com
filetouse /www.baidu.com
```

开始检查代理服务器中是否存在文件：  
文件存在在代理服务器中

代理服务器获得客户端的GET请求

文件存在于代理服务器

且相对于第一次搜索来说，网页加载速度较原来更快。

- favicon.ico一直不在本地

```
Cache-Control: no-cache
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Accept: image/avif,image/webp,image/apng,image/*,*/*;q=0.8
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: no-cors
Sec-Fetch-Dest: image
Referer: http://localhost:8999/www.baidu.com
Accept-Encoding: gzip, deflate, br
Accept-Language: zh-CN,zh;q=0.9

/favicon.ico
filename: favicon.ico
filetouse /favicon.ico

开始检查代理服务器中是否存在文件：
文件不在代理服务器当中，开始向远端服务器请求网页
HOST NAME favicon.ico
Illegal request
Ready to serve...
Received a connection from: ('127.0.0.1', 65483)
```

## 2. 运行过程linux

```

D:\pythonProject\untitled>ProxyServer.py ""
Ready to serve...
Received a connection from: ('127.0.0.1', 57992)
从客户端传来的信息 GET /www.linux.com HTTP/1.1
Host: localhost:8999
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/we
;v=b3;q=0.9
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Accept-Encoding: gzip, deflate, br
Accept-Language: zh-CN,zh;q=0.9

/www.linux.com
filename: www.linux.com
filetouse /www.linux.com

开始检查代理服务器中是否存在文件:
文件不在代理服务器当中, 开始向远端服务器请求网页
HOST NAME: linux.com
buffer: b'HTTP/1.1 503 Max restarts limit reached\r\nRetry-After: 0\r\nContent-Length: 0\r\nDate: Sat, 2
onnection: close\r\nServer: Pantheon\r\nX-Served-By: cache-hkg17922-HKG\r\nX-Cache: MISS\r\nX-Cache-Hits:
963387,VS0,VE2\r\nAge: 0\r\nAccept-Ranges: bytes\r\nVia: 1.1 varnish\r\n\r\n'
/www.linux.com
filename: www.linux.com
filetouse /www.linux.com

开始检查代理服务器中是否存在文件:
文件存在在代理服务器中

```

```
HTTP/1.1 503 Max restarts limit reached
Retry-After: 0
Content-Length: 0
Date: Sat, 26 Dec 2020 08:39:55 GMT
Connection: close
Server: Pantheon
X-Served-By: cache-hkg17922-HKG
X-Cache: MISS
X-Cache-Hits: 0
X-Timer: S1608971996.963387,VS0,VE2
Age: 0
Accept-Ranges: bytes
Via: 1.1 varnish
```

### 3. 源代码

注：因为相应的远端html链接和python版本的不适配修改了部分原始代码，并增加相应的输出以便输出更易读。

```
from
socket
import *
import
sys

if
len(sys.
argv) <=
1:
```

```
print('
Usage :
"python
ProxySer
ver.py
server_i
p"\n[ser
ver_ip :
It is
the IP
Address
Of Proxy
Server')
```

```
sys.exi
t(2)
```

```
# Create
a server
socket,
bind it
to a
port and
start
listenin
g
tcpSerSo
ck =
socket(A
F_INET,
SOCK_STR
EAM)
# Fill
in
start.
SERVER_I
P =
sys.argv
[1]
PORT_NUM
= 8990
tcpSerSo
ck.bind(
(SERVER_
IP,
PORT_NUM
))
tcpSerSo
ck.liste
n(1)
# Fill
in end.
while 1:
```

```
#
Strat
receivin
g data
from the
client

print('
Ready to
serve...'
')

tcpClis
ock,
addr =
tcpSerSo
ck.accep
t()

print('
Received
a
connecti
on
from:',
addr)

message
=
tcpCliSo
ck.recv(
1024).de
code()

print('
从客户端
传来的信
息',
message)

print(m
essage)

#
Extract
the
filename
from the
given
message

print(m
essage.s
plit()
[1])
```

```
filename = message.split()[1].partition("/")[2]

print('filename:', filename)

fileExist = "false"

filetouse = "/" + filename

print('filetouse', filetouse)

print('\n')
try:

    # Check whether the file exist in the cache

    print('开始检查代理服务器中是否存在文件: ')

    f = open(filetouse[1:], "r")

    outputdata = f.read()

    fileExist = "true"
```

```
print('
文件存在
在代理服
务器中')

#
ProxySer
ver
finds a
cache
hit and
generate
s a
response
message

tcpcli
sock.send
("HTTP/1
.1 200
OK\r\n".
encode()
)

tcpcli
sock.send
("Conten
t-
Type:tex
t/html\r
\n".enco
de())

# Fill
in
start.

for i
in
range(0,
len(outp
utdata))
:

tcpcli
sock.send
(outputd
ata[i].e
ncode())

# Fill
in end.
```

```
        print('
Read
from
cache')
        #
        Error
        handling
        for file
        not
        found in
        cache

    except
    IOError:

        print('
cache no
set')

        if
        fileExis
        t ==
        "false":

            print('
文件不在
代理服务
器当中，
开始向远
端服务器
请求网
页')

            #
            Create a
            socket
            on the
            proxyser
            ver

            c =
            socket(AF_INET,
            SOCK_STR
            EAM)

            hostn
            =
            filename
            .replace
            ("www.",
            "",1)

            print(h
            ostn)
```



```
try:

    #
    Connect
    to the
    socket
    to port
    80

    #
    Fill in
    start.

    c.conne
    ct((host
    n, 80))

    #
    Fill in
    end.

    #
    Create a
    temporar
    y file
    on this
    socket
    and ask
    port 80
    for the
    file
    requeste
    d by the
    client

    fileobj
    =
    c.makefi
    le('rwb'
    , 0)

    fileobj
    .write((
    "GET " +
    "http://"
    " +
    filename
    + "
    HTTP/1.0
    \n\n").e
    ncode())
```

```
#  
Read the  
response  
into  
buffer
```

```
#  
Fill in  
start.
```

```
buf =  
fileobj.  
read()
```

```
print("  
buffer:"  
, buf)
```

```
#  
Fill in  
end.
```

```
#  
Create a  
new file  
in the  
cache  
for the  
requeste  
d file.
```

```
#  
Also  
send the  
response  
in the  
buffer  
to  
client  
socket  
and the  
correspo  
nding  
file in  
the  
cache
```

```
tmpFile  
=  
open("./  
" +  
filename  
, "wb")
```

```
        #
        Fill in
        start.

        tmpFile
        .write(b
        uf)

        tmpFile
        .close()

        #
        Fill in
        end.

    except:

        print("
        Illegal
        request"
        )

        else:

            #
            HTTP
            response
            message
            for file
            not
            found

            #
            Fill in
            start.

            tcpClis
            ock.send
            ('HTTP/1
            .1 404
            Not
            Found\r\
            n'.encod
            e())
```

```
tcpclis  
ock.send  
( 'Conten  
t-type:  
text/htm  
l;  
charset=  
UTF-  
8\r\n\r\n'  
n'.encod  
e()
```

```
tcpclis  
ock.send  
( '404  
Not  
Found\r\n'  
n'.encod  
e()
```

```
#  
Fill in  
end.
```

```
tcpclis  
ock.clos  
e()
```