中国科学技术大学计算机学院

《计算机网络实验报告》

实验题目：DNS Relay

学生姓名：郑振东

学生学号：PB18111703

完成日期：2020 年 12 月 25 号

计算机实验教学中心制

2019 年 09 月

【实验要求】

开发一个程序，从配置文件加载“域名-IP 地址”列表，并按如下方式处理 DNS 查询。

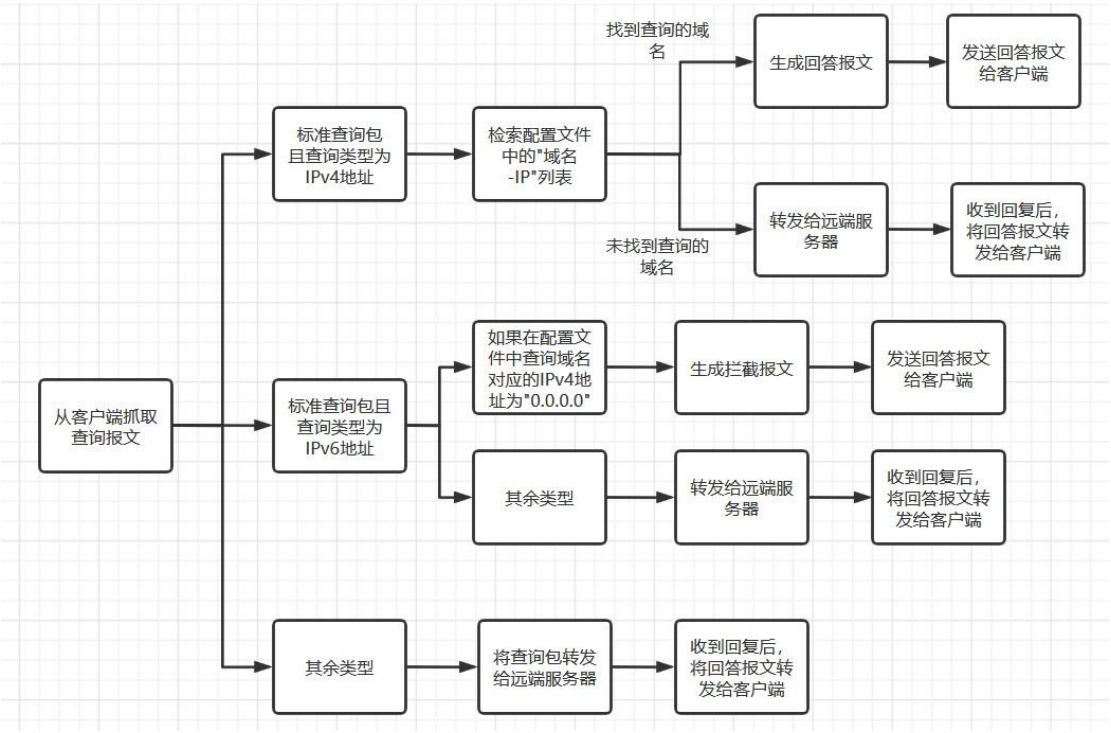
- 拦截：如果查询的名称在列表中，其关联的 IP 地址为“0.0.0.0”，则向客户端响应 0.0.0.0。
- 本地解析：如果查询到的名称在列表中，并且有关联的有意义的 IP 地址，则响应该 IP 地址。
- 中继：如果域名不在列表中，则在服务器和客户端之间中继查询和响应。

【实验环境】

1. Windows 操作系统;
2. VSCode;

【实验过程】

处理一个 DNS 数据包的大致流程



代码讲解

只有一个类：DNSRelay

在初始化阶段会做以下 4 件事：

- 装载配置文件
- 生成 UDP 套接字端口
- 生成线程池
- 生成线程锁（可去掉）

```
01 def __init__(self, file_name):
02         self.file_data = []
03         with open(file_name) as file_object:
04             for line in file_object:
05                 if line != '\n':
06                     self.file_data.append(tuple(line.rstrip().split()))
07         self.socketRecv = socket.socket(socket.AF_INET, socket.SOCK_DGRAM, 0)
08         self.socketRecv.bind(("localhost", 53))
09         self.pool = ThreadPoolExecutor(max_workers = 4)
10         self.lock = threading.RLock()
```

当挂起 DNS 服务器之后，服务器会不断从本机 53 端口抓取 DNS 数据包，然后放入到线程池中等待 solve 函数处理。

```
def run(self):
        while True:
            try:
                message, address = self.socketRecv.recvfrom(1024)
                self.pool.submit(self.solve, message, address)
            except ConnectionResetError:
                ...
```

处理一个 DNS 数据包时，solve 函数会首先从数据包的第 12 个字节后开始读出查询的域名，域名读取完毕后再截取出查询类型。

```
# 得到域名
        NAME = ''
        i = 12
        if message[i] != 0:
            while True:
                for j in range(1, message[i] + 1):
                    NAME = NAME + chr(message[i + j])
                i = message[i] + i + 1
                if message[i] == 0:
                    break
                NAME = NAME + '.'
        # 查询类型
        TYPE = message[i+1:i+3]
```

如果 DNS 数据包是标准查询包且查询类型是 IPv4 地址，就去配置文件中检索“域名-IP 地址”列表。

若能在配置文件中找到查询域名的 IP 地址，先调用 gen_response 函数生成回答报文，再向客户端发送回答报文。

若配置文件中找不到查询域名的 IP 地址，调用 forward 函数处理。

```
if message[2] >> 3 == 0 and TYPE == b'\x00\x01':
            print('------是标准查询包且查询类型为IPv4地址')
```

```
            print('------在配置文件中查找')
            for (ip, domain) in self.file_data:
                if domain == NAME:
                    print("------找到对应IP地址")
                    response = self.gen_response(message, ip)
                    print("------回答报文生成完毕")
                    self.socketRecv.sendto(response, address)
                    print('------发送回答报文给 ' + address[0])
                    end = time.time()
                    print("---------------------------------解析完毕  用时: %.03f 秒" %(end - start))
                    self.lock.release()
                    break
            else:
                print('------配置文件中未找到')
                self.forward(message, address, start)
```

如果 DNS 数据包是标准查询包且查询类型是 IPv6 地址，就去配置文件中检索“域名-IP 地址”列表。

若配置文件中查询域名对应的 IPv4 地址为“0.0.0.0”，将此查询报文拦截，先调用 gen_response 函数生成拦截报文，然后给客户端发送回答报文。

其余情况均调用 forward 函数处理。

```
elif message[2] >> 3 == 0 and TYPE == b'\x00\x1C':
      for (ip, domain) in self.file_data:
          if domain == NAME and ip == "0.0.0.0":
              print("------此查询IPv6报文的域名对应的IPv4地址在配置文件中为'0.0.0.0'")
              response = self.gen_response(message, ip)
              print("------回答报文生成完毕")
              self.socketRecv.sendto(response, address)
              print('------发送回答报文给 ' + address[0])
              end = time.time()
              print("---------------------------------解析完毕  用时: %.03f 秒" %(end - start))
              self.lock.release()
              break
```

```
13          else:
14              print('------其余查询类型')
15              self.forward(message, address, start)
```

剩余的 DNS 数据包均调用 forward 函数处理。

```
1 else:
2       print('------其余查询类型')
3       self.forward(message, address, start)
```

在 forward 函数中，首先生成一个 UDP 套接字端口，然后利用这个端口转发来自客户端的 DNS 数据包给远端 DNS 服务器（其 IPv4 地址为：222.5.5.5），收到来自远端 DNS 服务器的回答报文后，再将回答报文转发给客户端，最后将生成这个 UDP 套接字端口关闭。

```
01 def forward(self, message, address, start):
02          forward_socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM, 0)
03          forward_socket.settimeout(5)
04          try:
05              forward_socket.sendto(message, ('223.5.5.5', 53))
06              print('------查询报文转发给远端DNS服务器，其IP地址为：223.5.5.5')
07              response, response_addr = forward_socket.recvfrom(1024)
08              print('------得到本地DNS服务器的回复')
09              self.socketRecv.sendto(response, address)
10              print('------转发回答报文给 ' + address[0])
11              end = time.time()
12              print("--------------------------------解析完毕  用时：%.03f 秒
" %(end - start))
13              forward_socket.close()
14          except ConnectionResetError:
15              forward_socket.close()
16          except:
17              print("------TIME OUT")
18              print("--------------------------------解析完毕")
19              forward_socket.close()
20          self.lock.release()
```

gen_response 函数中，首先要做以下 11 件事情:

- 截取查询报文的会话标志 ID（2 字节）；
- 将 QR 置 1（响应报文标志）；
- 设置 Opcode 为‘-0b0000’（标准响应类型）；
- 置 AA 为 0，表示应答服务器不是该域名的权威解析服务器;
- 置 TC 为 0，表示报文未截断;
- 置 RD 为 1，表示期望进行递归查询;
- 置 RA 为 1，表示应答服务器支持递归查询;
- 置 Z 为 0;
- 置 AD 为 0，表示应答服务器未验证该查询相关的 DNSSEC 数字签名;
- 置 CD 为 0，表示服务器已经进行了相关 DNSSEC 数字签名的验证;
- 置 NSCOUNT 和 ARCOUNT 均为 b’\x00\x00’（权威区域和附加区域为空）;

如果查询报文中对应的 IPv4 地址为‘0.0.0.0’，表示该报文需要拦截。首先置应答码 Rcode 为‘-0b0011’，指出解析的域名不存在；然后置 QDCOUNT 为 b‘\x00\x01’，表示报文请求段中的问题记录数为 1，置 ANCOUNT 为 b‘\x00\x00’，表示报文回答段中的回答记录数为 0；最后将查询报文的问题区域附到回答报文的后面即可。

其余置应答码 Rcode 为‘-0b0000’，表示一切正常；然后置

QDCOUNT 为 b‘\x00\x01’，表示报文请求段中的问题记录数为 1，置 ANCOUNT 为 b‘\x00\x01’，表示报文回答段中的回答记录数为 1；接下来将查询报文的问题区域附到回答报文后面；最后在回答报文的问答区域中，先置域名指针置为 b‘\xC0\x0C’，然后置类型为 IPv4 查询，查询类为 IN，接下来将该记录的生存时间为 86400 秒，最后设置资源数据的长度为 4，在资源数据部分附上返回的 IPv4 地址。

```
01 def gen_response(self, message, ip):
02     response = message[:2]
03     if ip == "0.0.0.0":
04         print("------查询报文已拦截")
05         # QR = '1'
06         # Opcode = '0000'
07         # AA = '0'
08         # TC = '0'
09         # RD = '1'
10         # RA = '1'
11         # Z ='0'
12         # AD = '0'
13         # CD = '0'
14         # Rcode = '0011'
15         # QDCOUNT = b'\x00\x01'
16         # ANCOUNT = b'\x00\x00'
17         # NSCOUNT = b'\x00\x00'
18         # ARCOUNT = b'\x00\x00'
19         response += b'\x81\x83\x00\x01\x00\x00\x00\x00\x00\x00'
20         # 问题区域
21         response += message[12:]
22     else:
23         print("------为合法查询报文")
24         # QR = '1'
25         # Opcode = '0000'
26         # AA = '0'
27         # TC = '0'
28         # RD = '1'
29         # RA = '1'
30         # Z ='0'
```

```
31          # AD = '0'
32          # CD = '0'
33          # Rcode = '0000'
34          # QDCOUNT = b'\x00\x01'
35          # ANCOUNT = b'\x00\x01'
36          # NSCOUNT = b'\x00\x00'
37          # ARCOUNT = b'\x00\x00'
38          response += b'\x81\x80\x00\x01\x00\x01\x00\x00\x00\x00'
39          # 问题区域
40          response += message[12:]
41          # 指针，指向请求部分的域名
42          # 高两位识别指针，12为首部区域的长度
43          response += b'\xC0\x0C'
44          # 类型为IPv4地址查询，查询类为IN
45          response += b'\x00\x01\x00\x01'
46          # 生存时间：一天
47          response += b'\x00\x01\x51\x80'
48          # 资源数据长度
49          response += b'\x00\x04'
50          ip = ip.split('.')
51          for i in range(4):
52              response += int(ip[i]).to_bytes(1, 'big')
53      return response
```

实验结果：

nslookup

拦截

配置文件（截取）

```
127.0.0.1 www.test1.com
0.0.0.0 pic1.zhimg.com
0.0.0.0 pic2.zhimg.com
```

输出

```
C:\Users\zhengzhendong>nslookup
默认服务器:  localhost
Address:  127.0.0.1

> pic1.zhimg.com
服务器:  localhost
Address:  127.0.0.1

*** localhost 找不到 pic1.zhimg.com: Non-existent domain
>
```

本地服务器输出

```
从 127.0.0.1 抓取一个包
---------------------------------开始解析
域名为：pic1.zhimg.com
查询类型为：b'\x00\x01'
------是标准查询包且查询类型为IPv4地址
------在配置文件中查找
------找到对应IP地址
------查询报文已拦截
------回答报文生成完毕
------发送回答报文给 127.0.0.1
---------------------------------解析完毕  用时：0.003 秒
```

本地解析

配置文件（截取）

```
127.0.0.1 www.test1.com
0.0.0.0 pic1.zhimg.com
0.0.0.0 pic2.zhimg.com
```

输出

```
> www.test1.com
服务器:  localhost
Address:  127.0.0.1

非权威应答:
名称:    www.test1.com
Address:  127.0.0.1
```

本地服务器输出

```
从 127.0.0.1 抓取一个包
--------------------------------开始解析
域名为：www.test1.com
查询类型为：b'\x00\x01'
------是标准查询包且查询类型为IPv4地址
------在配置文件中查找
------找到对应IP地址
------为合法查询报文
------回答报文生成完毕
------发送回答报文给 127.0.0.1
---------------------------------解析完毕  用时：0.003 秒
```

中继

输出

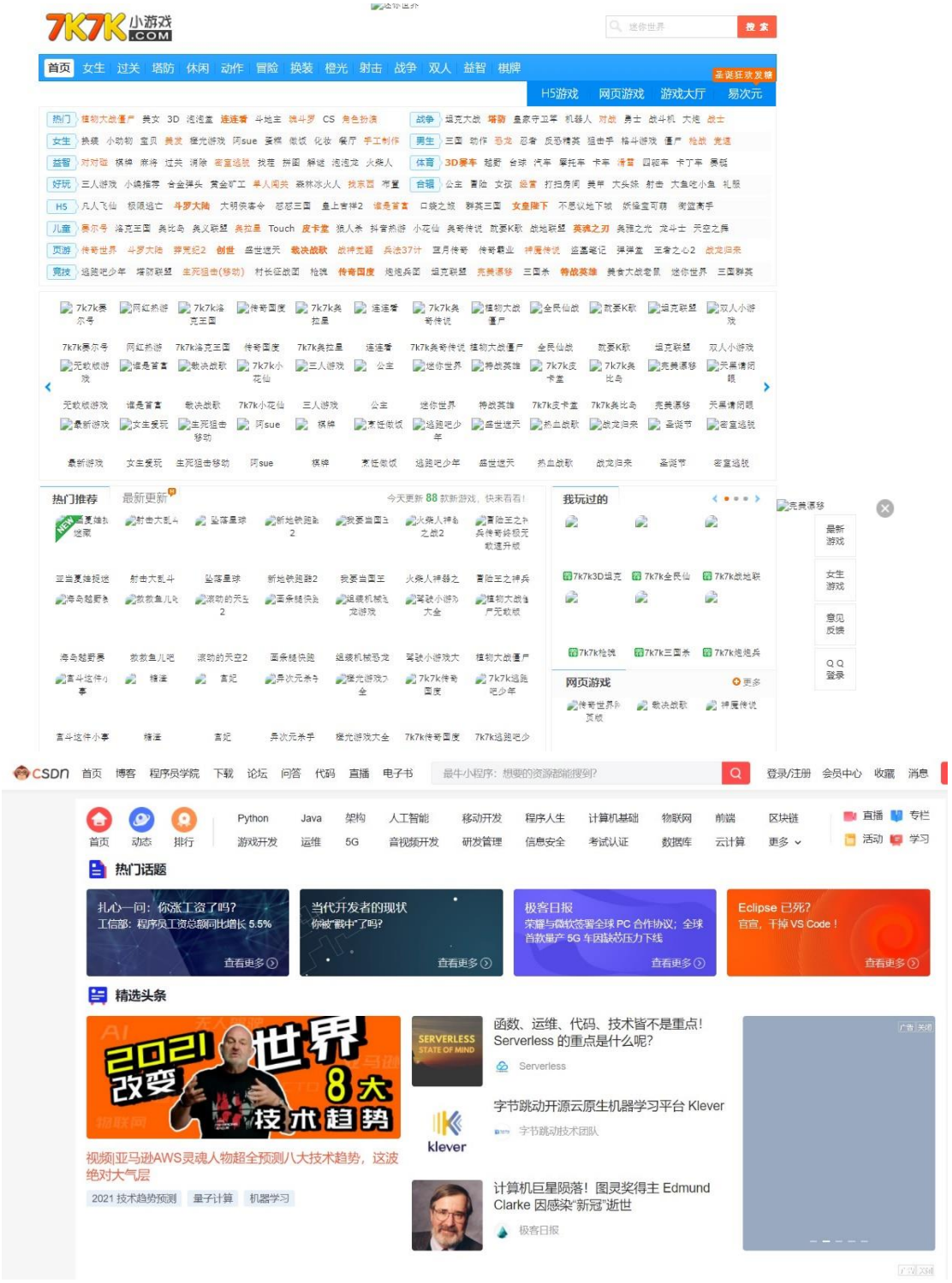
```
> www.taobao.com
服务器:  localhost
Address:  127.0.0.1

非权威应答:
名称:    www.taobao.com.danuoyi.tbcache.com
Addresses:  2409:8c20:a12:104:2::3f9
          2409:8c20:a12:104:2::3fa
          223.111.255.232
          223.111.255.233
Aliases:  www.taobao.com
```

本地服务器输出

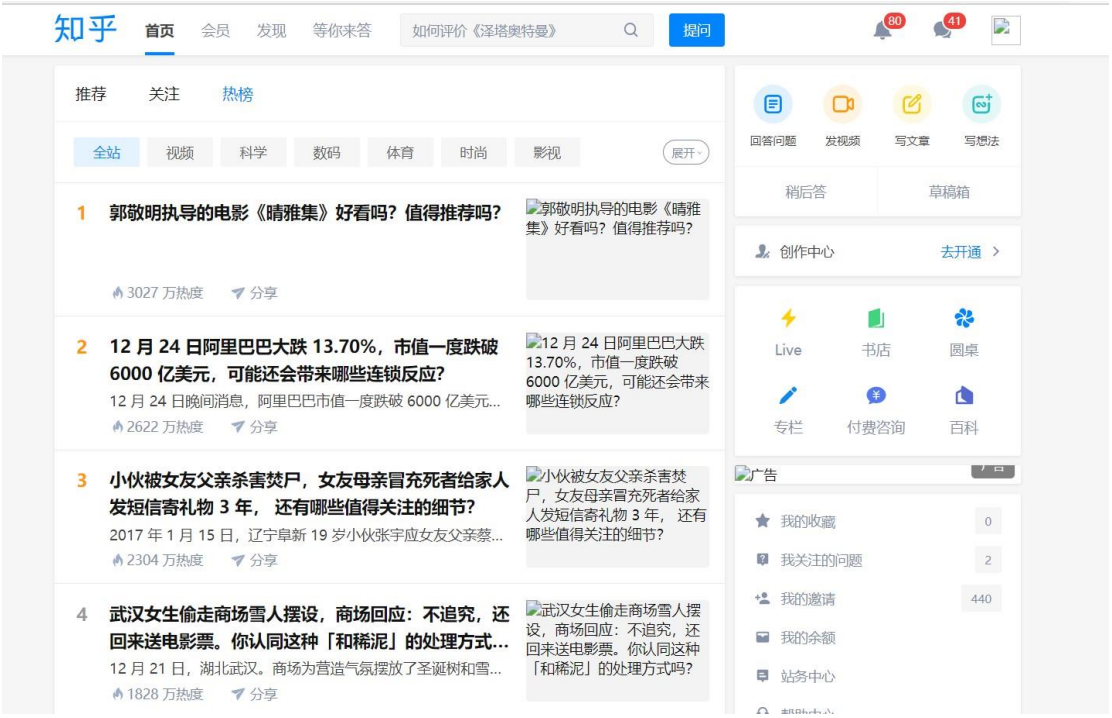
```
从 127.0.0.1 抓取一个包
--------------------------------开始解析
域名为：www.taobao.com
查询类型为：b'\x00\x01'
------是标准查询包且查询类型为IPv4地址
------在配置文件中查找
------配置文件中未找到
------查询报文转发给远端DNS服务器，其IP地址为：223.5.5.5
------得到本地DNS服务器的回复
------转发回答报文给 127.0.0.1
---------------------------------解析完毕  用时：0.073 秒
```

广告屏蔽


7K7K小游戏.COM
首页 女生 过关 塔防 休闲 动作 冒险 换装 橙光 射击 战争 双人 益智 棋牌
H5游戏 网页游戏 游戏大厅 易次元
热门推荐
最新更新
我玩过的
网页游戏
最新游戏
女生游戏
意见反馈
QQ登录

CSDN 首页 博客 程序员学院 下载 论坛 问答 代码 直播 电子书
登录/注册 会员中心 收藏 消息
首页 动态 排行
Python Java 架构 人工智能 移动开发 程序人生 计算机基础 物联网 前端 区块链
游戏开发 运维 5G 音视频开发 研发管理 信息安全 考试认证 数据库 云计算 更多
直播 专栏 活动 学习
热门话题
扎心一问：你涨工资了吗？
工信部：程序员工资总额同比增长 5.5%
当代开发者的现状
你被"戳中"了吗？
极客日报
荣耀与微软签署全球 PC 合作协议；全球首款量产 5G 车因缺芯压力下线
Eclipse 已死？
官宣，干掉 VS Code！
查看更多
精选头条
2021 改变世界 8大技术趋势
视频|亚马逊AWS灵魂人物超全预测八大技术趋势，这波绝对大气层
2021 技术趋势预测 量子计算 机器学习
函数、运维、代码、技术皆不是重点！Serverless 的重点是什么呢？
Serverless
字节跳动开源云原生机器学习平台 Klever
字节跳动技术团队
计算机巨星陨落！图灵奖得主 Edmund Clarke 因感染"新冠"逝世
极客日报


掘金
首页
沸点
小册
活动
探索掘金
写文章
登录
推荐
后端
前端
Android
iOS
人工智能
开发工具
代码人生
阅读
热门
最新
热榜
掘金酱 · 8天前
掘力计划创作者训练营第一期，开营了！
快来学习写作爆款文章！
杨村长 · 6小时前 · 前端
备战2021：vite工程化实践，建议收藏
269
47
设计稿智能生成代码 · 2天前 · 前端
阿里前端智能化方向负责人 2020年终技术回顾
6
1
程序员内点事 · 1小时前 · Java / 后端
千万不要给女朋友解释 什么是 “羊群效应”
8
7
隐冬 · 18小时前 · JavaScript
我用JS开发了一款桌面应用
23
18
掘金 - juejin.cn
一个帮助开发者成长的社区
+86
请输入手机号
验证码
获取验证码
立即登录
注册登录即表示
同意 用户协议 、 隐私政策
广告
下载掘金客户端
一个帮助开发者成长的社区


知乎
首页
会员
发现
等你来答
如何评价《泽塔奥特曼》
提问
80
41
推荐
关注
热榜
全站
视频
科学
数码
体育
时尚
影视
展开
1
郭敬明执导的电影《晴雅集》好看吗？值得推荐吗？
3027 万热度
分享
郭敬明执导的电影《晴雅集》好看吗？值得推荐吗？
2
12 月 24 日阿里巴巴大跌 13.70%，市值一度跌破 6000 亿美元，可能还会带来哪些连锁反应？
12 月 24 日晚间消息，阿里巴巴市值一度跌破 6000 亿美元...
2622 万热度
分享
12 月 24 日阿里巴巴大跌 13.70%，市值一度跌破 6000 亿美元，可能还会带来哪些连锁反应？
3
小伙被女友父亲杀害焚尸，女友母亲冒充死者给家人发短信寄礼物 3 年，还有哪些值得关注的细节？
2017 年 1 月 15 日，辽宁阜新 19 岁小伙张宇应女友父亲蔡...
2304 万热度
分享
小伙被女友父亲杀害焚尸，女友母亲冒充死者给家人发短信寄礼物 3 年，还有哪些值得关注的细节？
4
武汉女生偷走商场雪人摆设，商场回应：不追究，还回来送电影票。你认同这种「和稀泥」的处理方式...
12 月 21 日，湖北武汉。商场为营造气氛摆放了圣诞树和雪...
1828 万热度
分享
武汉女生偷走商场雪人摆设，商场回应：不追究，还回来送电影票。你认同这种「和稀泥」的处理方式吗？
回答问题
发视频
写文章
写想法
稍后答
草稿箱
创作中心
去开通
Live
书店
圆桌
专栏
付费咨询
百科
广告
我的收藏
0
我关注的问题
2
我的邀请
440
我的余额
站务中心