绕过 cdn 探测真实 ip 方法大全

目录：

渗透测试过程中，需要寻找真实 IP 的情况，无非就是目标使用了 cdn 或者有云防护。

CDN

> CDN 即内容分发网络，主要解决因传输距离和不同运营商节点造成的网络速度性能低下的问题。
> 说的简单点，就是一组在不同运营商之间的对接点上的高速缓存服务器，把用户经常访问的静态数据资源直接缓存到节点服务器上，当用户再次请求时，会直接分发到离用户近的节点服务器上响应给用户，当用户有实际数据交互时才会从远程 Web 服务器上响应，这样可以大大提高网站的响应速度及用户体验。

CDN 的优势

> - 提高用户访问速率，优化用户使用体验。
> - 隐藏真实服务器的 IP
> - 提供 WAF 功能，目前很多 CDN 也提供了 WAF 的功能，我们的访问请求会先经过 CDN 节点的过滤，该过滤可对 SQL 注入、XSS、Webshell 上传、命令注入、恶意扫描等攻击行为进行有效检测和拦截。CDN 节点将认为无害的数据提交给真实的主机服务器。

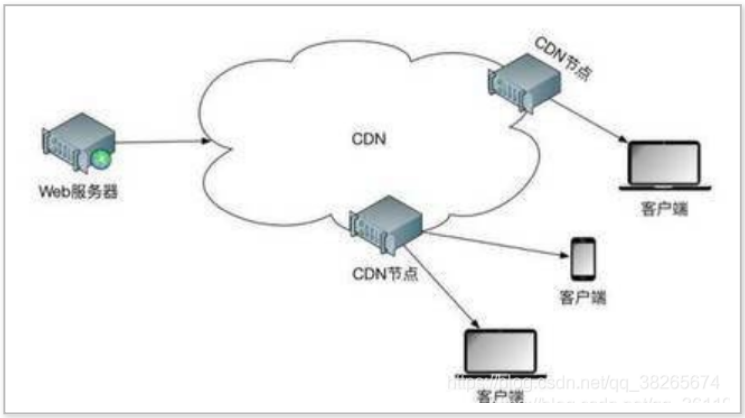
几种访问方式的不同

> - 传统访问：用户访问域名—> 解析服务器 IP—> 访问目标主机
> - 普通 CDN：用户访问域名—>CDN 节点—> 真实服务器 IP—> 访问目标主机
> - 带 WAF 的 CDN：用户访问域名—>CDN 节点（云 WAF）—> 真实服务器 IP—> 访问目标主机

CDN 的配置

1. 将域名的 NS 记录指向 CDN 厂商提供的 DNS 服务器。

2. 给域名设置一个 CNAME 记录，将它指向 CDN 厂商提供的另一个域名。



> 所以在渗透测试中，为了要知道网站服务器的真实 IP，我们必须绕过 CDN 查找出网站的真实 ip 地址。

判断是否使用 cdn

多地 ping

> 不同地区的服务器 -> 访问 ->ip，假如使用了 cdn->ip 会众多。
> 假如使用了双线 -> ip 一般只有几个。
> 这是区分 cdn 跟多线服务器的很好的方法。
> 不同的地方去 Ping 服务器，如果 IP 不一样，则目标网站肯定使用了 CDN。
> 在线 ping:
> http://ping.chinaz.com/
> http://ce.cloud.360.cn/
> http://www.webkaka.com/ping.aspx
> https://asm.ca.com/en/ping.php

nslookup 进行检测

>

使用 nslookup 进行检测，原理同上，如果返回域名解析对应多个 IP 地址多半是使用了 CDN。有 CDN 的示例：

> www.163.com
> 服务器: public1.114dns.com
> Address: 114.114.114.114
>
> 非权威应答:
> 名称: 163.xdwscache.ourglb0.com
> Addresses: 58.223.164.86
>
> 125.75.32.252
> Aliases: www.163.com
>
> www.163.com.lxdns.com

无 CDN 的示例：

> xiaix.me
> 服务器: public1.114dns.com
> Address: 114.114.114.114
> 非权威应答:
> 名称: xiaix.me
> Address: 192.3.168.172

CDN 的绕过思路

网上有很多绕过 CDN 的思路，但是存在很多问题，以下是收集并总结的思路。
在站长的角度，不可能每个站都会用上 CDN。
站在 DNS 服务商的角度，历史解析记录可能不受 CDN 服务商控制。
站在 CDN 服务商的角度，提供 CDN 服务的区域有限制，CDN 流量有限制。

绕过 cdn 探测真实 ip 方法大全

1、多地 ping

参考如上的

> 1. 使用在线网站 ping 目标，如果得出不同的 ip，我们可以判断不同地区是否使用 cdn 来得出真实 ip。
> 2. 国内的 CDN 一般只对国内的用户访问加速，
> 所以使用国外在线代理网站：
> https://asm.ca.com/en/ping.php

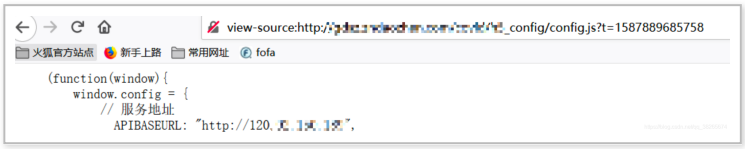
2、泄露文件

> 这个需要用工具扫或者爬，但是找到的成功率不是很高。
> - 服务器日志文件
> - 探针文件
> 扫描目标 web 目录 获取 phpinfo 探针类文件 ，基本上都存有服务器真实 ip 信息泄露。
> 在 phpinfo 里真实 IP 对应项为：
> 
> | | |
> |---|---|
> | SERVER_NAME | 127.0.0.1 |
> | SERVER_ADDR | 127.0.0.1 |

3、信息收集

> 首先我们可以进行正常的信息收集过程，尽可能抓取各种 IP 地址（通过 host、nslookup、whois、地址段等），然后检查哪些服务器启用了 Web 服务（通过 netcat、nmap、masscan 等）。
> 一旦获取到 Web 服务器 IP 地址，下一步就是检查目标域名是否以虚拟主机方式托管在某个平台上。
> 如果不采用这种方式，我们就可以看到默认的服务器页面或者默认配置的站点页面，否则我们就找到了切入点。
> - 服务器信息

- 错误信息
- js 文件信息



>- 旁站

- 子域名或者父域名
  子域名法：由于成本问题，可能某些厂商并不会将所有的子域名都部署 CDN，所以如果我们能尽量的搜集子域名，或许可以找到一些没有部署 CDN 的子域名，拿到某些服务器的真实 ip/ 段
  工具收集
    - wydomain:
      https://github.com/ring04h/wydomain
    - subDomainsBrute:
      https://github.com/lijiejie/
    - Sublist3r:
      https://github.com/aboul3la/Sublist3r
    - layer 子域名挖掘机

  在线收集
    - https://dnsdb.io/zh-cn/
    - https://phpinfo.me/bing.php
    - http://www.webscan.cc

- 通过搜索引擎查找公网上的相同站点（开发环境，备份站点等）
  参考： 子域名探测方法大全

- TXT 记录

- 搜索引擎
  常见的有钟馗之眼，shodan，fofa 搜索。
  以 fofa 为例，只需输入: title:“网站的 title 关键字”或者 body:“网站的 body 特征”就可以找出 fofa 收录的有这些关键字的 ip 域名，很多时候能获取网站的真实 ip。

例如：

考虑到站长建站不可能用一个域名。

假如是做非法产业，黑色产业，一般都需要购买一定的量的域名，域名被拦截的时候，方便指向，继续安全访问。

方法是 whois–> 联系信息 –> 社工 –> 反查域名或子级域名

4、漏洞利用

漏洞利用，比如 SSRF、XXE、XSS、文件上传等漏洞，或者我们找到的其他突破口，注入包含我们自己服务器地址的 payload，然后在服务器上检查对应的日志。

5、SSL 证书

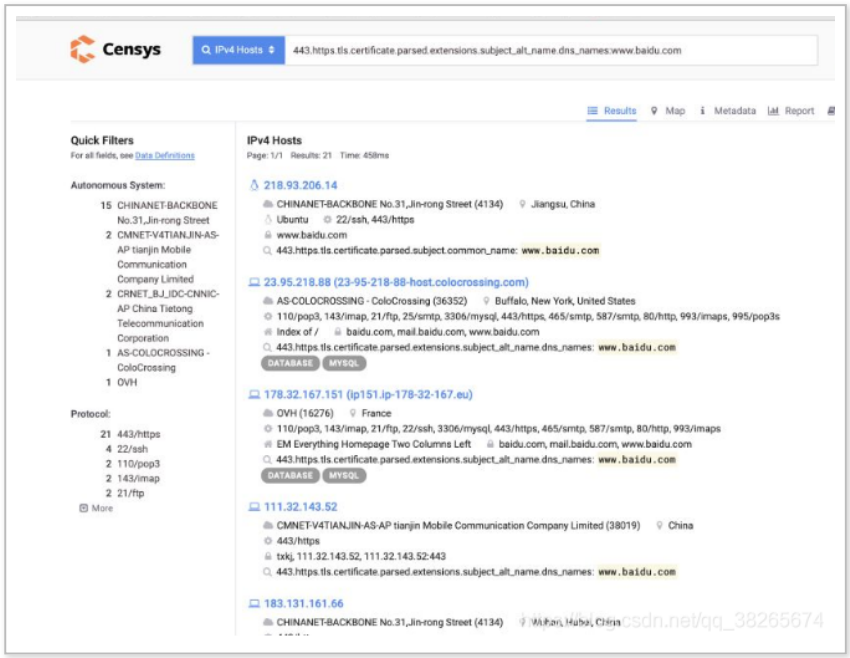
假如在 www.wangsu.com 上托管了一个服务，原始服务器 IP 是 136.23.63.44，而 wangsu 会为你提供 DDoS 保护，Web 应用程序防火墙等服务，以保护你的服务免受攻击。为此，你的 Web 服务器就必须支持 SSL 并具有证书。

Censys 工具就能实现对整个互联网的扫描，Censys 是一款用以搜索联网设备信息的新型搜索引擎，能够扫描整个互联网，Censys 会将互联网所有的 ip 进行扫面和连接，以及证书探测。若目标站点有 https 证书，并且默认虚拟主机配了 https 证书，我们就可以找所有目标站点是该 https 证书的站点。

https://censys.io/ipv4

例如：

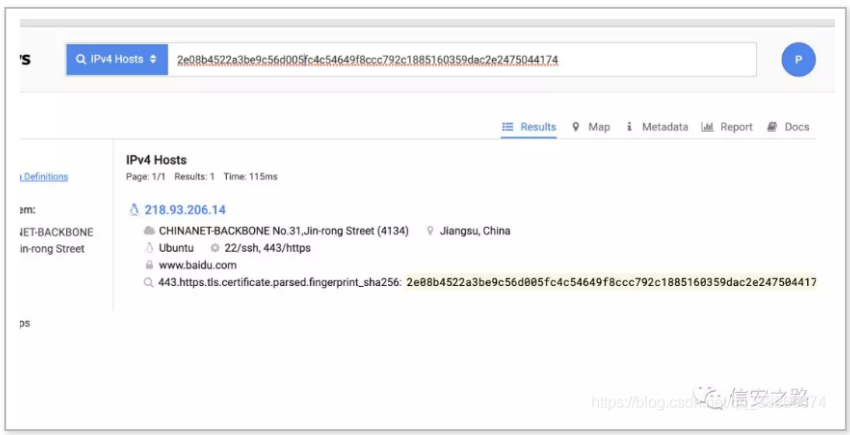
443.https.tls.certificate.parsed.extensions.subject_alt_name.dns_names:www.baidu.com



还有一种方式，就是搜集 SSL 证书 Hash，然后遍历 ip 去查询证书 hash，如果匹配到相同的，证明这个 ip 就是那个域名同根证书的服务器真实 ip

简单来说，就是遍历 0.0.0.0/0:443，通过 ip 连接 https 时，会显示证书

当然，也可以用 censys 等引擎



再放一个搜集证书的网站:

https://crt.sh

一个小脚本，可以快速搜集证书

```
# -*- coding: utf-8 -*-
# @Time    : 2019-10-08 22:51
# @Author  : Patrilic
# @FileName: SSL_subdomain.py
# @Software: PyCharm
```

```
import requests
import re

TIME_OUT = 60
def get_SSL(domain):

    domains = []
    url = 'https://crt.sh/?q=%25.{}'.format(domain)
    response = requests.get(url,timeout=TIME_OUT)
    # print(response.text)
    ssl = re.findall("<TD>(.*?).{}</TD>".format(domain),respo
nse.text)
    for i in ssl:
        i += '.' + domain
        domains.append(i)
    print(domains)

if __name__ == '__main__':
    get_SSL("baidu.com")
```

6、DNS 解析

> 一般网站从部署开始到使用 cdn 都有一个过程，周期如果较长的话 则可以通过这类 历史解析记录查询等方式获取源站 ip，查看 IP 与域名绑定的历史记录，可能会存在使用 CDN 前的记录。

> 找国外的比较偏僻的 DNS 解析服务器进行 DNS 查询，因为大部分 CDN 提供商只针对国内市场，而对国外市场几乎是不做 CDN，所以有很大的几率会直接解析到真实 IP 。

> 全世界 DNS 地址：
> http://www.ab173.com/dns/dns_world.php
> https://dnsdumpster.com/
> https://dnshistory.org/
> http://whoisrequest.com/history/
> https://completedns.com/dns-history/
> http://dnstrails.com/
> https://who.is/domain-history/
> http://research.domaintools.com/research/hosting-history/
> http://site.ip138.com/
> http://viewdns.info/iphistory/
> https://dnsdb.io/zh-cn/

https://www.virustotal.com/
https://x.threatbook.cn/
http://viewdns.info/
http://www.17ce.com/
http://toolbar.netcraft.com/site_report?url=
https://tools.ipip.net/cdn.php

利用 SecurityTrails 平台，攻击者就可以精准的找到真实原始 IP。
他们只需在搜索字段中输入网站域名，然后按 Enter 键即可，这时“历史数据”就可以在左侧的菜单中找到。
https://securitytrails.com/

7、被动获取

被动获取就是让目标服务器主动链接我们的服务器，获取来源 IP。

很多站点都有发送邮件的功能，如 Rss 邮件订阅、找回密码、邮箱注册等。而且一般的邮件系统很多都是在内部，没有经过 CDN 的解析。可以通过邮件源码寻找服务器的真实 IP。

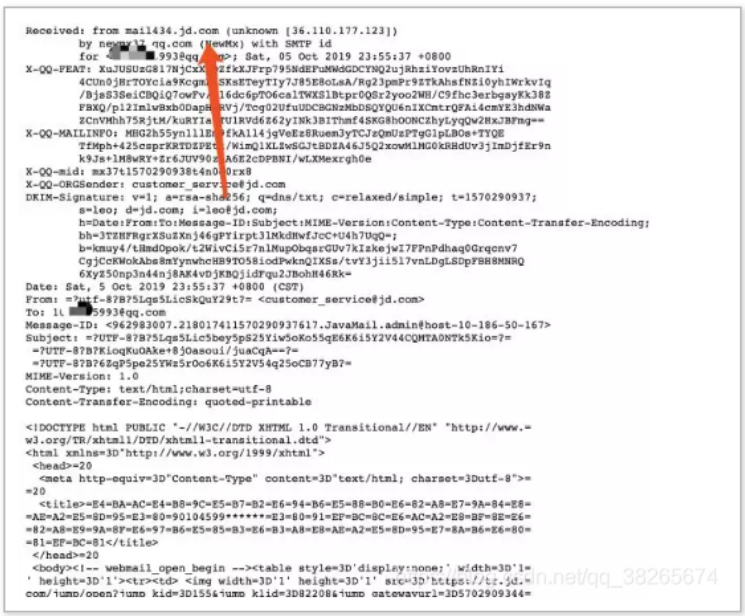
- SSRF
  SSRF 漏洞，服务器主动向外发起连接，泄露真实 IP 地址
  如 DZ SSRF 漏洞 exp 为：
  http:// 域名 / forum.php?mod=ajax&action=downremoteimg&message=[img=1,1]http://13.250.114.92:3319/aq9w.jpg[/img]

- MX 记录
  基本原理：
  就是想办法让目标 Web 服务器向我们自己的服务器 / Collaborator 发起请求。
  1、如果目标系统有发件功能，通常在注册用户 / 找回密码等地方，
  2、下一步就是提取目标邮件中的头部信息，比如我

们可以订阅目标服务，创建账户，使用“忘记密码”功能，或者订购某些 产品…… 总之，我们要想办法让目标给我们发送一封邮件（这种场景下我们可以使

用 Burp Collaborator）。

3、收到邮件后，我们可以查看源代码，特别是其中的邮件头，记录下其中的所有 IP 地址，包括子域名，这些信息很可能与托管服务有关。然后，我们可以尝试通过这些地址访问目标。



- RSS
  待补充

8、流量攻击

发包机可以一下子发送很大的流量。
这个方法是很笨，但是在特定的目标下渗透，建议采用。
cdn 除了能隐藏 ip，可能还考虑到分配流量，
不设防的 cdn 量大就会挂，高防 cdn 要大流量访问。
经受不住大流量冲击的时候可能会显示真实 ip。
站长 –> 业务不正常 –>cdn 不使用 –> 更换服务器。

9、全网扫描

- https://github.com/3xp10it/xcdn

- https://github.com/boy-hack/w8fuckcdn

10、长期关注

在长期渗透的时候，每天访问目标站。可能有新的发现与惊喜。

11、对比 banner

获取目标站点的 banner，在全网搜索引擎搜索，也可以使用 AQUATONE，在 Shodan 上搜索相同指纹站点。
可以通过互联网络信息中心的 IP 数据，筛选目标地区 IP，遍历 Web 服务的 banner 用来对比 CDN 站的 banner，可以确定源 IP。
欧洲：
http://ftp.ripe.net/pub/stats/ripencc/delegated-ripencc-latest
北美：
https://ftp.arin.net/pub/stats/arin/delegated-arin-extended-latest
亚洲：
ftp://ftp.apnic.net/public/apnic/stats/apnic/delegated-apnic-latest
非洲：
ftp://ftp.afrinic.net/pub/stats/afrinic/delegated-afrinic-latest
拉美：
ftp://ftp.lacnic.net/pub/stats/lacnic/delegated-lacnic-extended-latest
获取 CN 的 IP
http://www.ipdeny.com/ipblocks/data/countries/cn.zone ，
例如：
找到目标服务器 IP 段后，可以直接进行暴力匹配 ，使用 zmap、masscan 扫描 HTTP banner，然后匹配到目标域名的相同 banner

root@kali:~# zmap -p 80 -w bbs.txt -o 80.txt

设置 http-req，将其设置为如下的值：

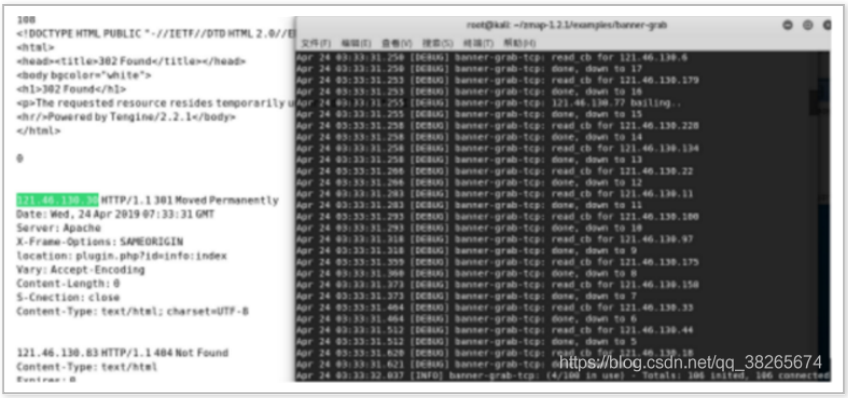
```
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:66.0) Gec
ko/20100101 Firefox/66.0
Host: %s
```

使用 zmap 的 banner-grab 对扫描出来 80 端口开放的主机进行 banner 抓取。

root@kali:~/zmap-1.2.1/examples/banner-grab# cat /root/bbs.txt |./banner-grab-tcp -p 80 -c 100 -d http-req -f ascii > http-banners.out

根据网站返回包特征，进行特征过滤

location: plugin.php?id=info:index



1、ZMap 号称是最快的互联网扫描工具，能够在 45 分钟扫遍全网。

https://github.com/zmap/zmap

https://github.com/zmap/zmap
2、Masscan 号称是最快的互联网端口扫描器，最快可以在六分钟内扫遍互联网。
https://github.com/robertdavidgraham/masscan

12、利用老域名

在换新域名时，常常将 CDN 部署到新的域名上，而老域名由于没过期，可能未使用 CDN，然后就可以直接获取服务器真实 ip。
例如 patrilic.top > patrilic.com
域名更新时，可能老域名同时解析到真实服务器，但是没有部署 CDN
这个可以通过搜集域名备案的邮箱去反查，可能会有意外收获

13、favicon_hash 匹配

利用 shodan 的 http.favicon.hash 语法，来匹配 icon 的 hash 值, 直接推:
https://github.com/Ridter/get_ip_by_ico/blob/master/get_ip_by_ico.py

14、CloudFlare Bypass

对 CloudFlare 客户网站进行真实 IP 查询
（因为很多网站都使用 CloudFlare 提供的 CDN 服务）

- 通过在线网站 CloudFlareWatch:
  http://www.crimeflare.us/cfs.html#box
- 一个绕过 CloudFlare 的简单 Ruby 脚本，可以发现您的真实 IP 地址:
  https://github.com/HatBashBR/HatCloud
- CloudFail 是一种战术侦察工具，旨在收集有关受 Cloudflare 保护的目标的足够信息，以期发现服务器的位置:
  https://github.com/m0rtem/CloudFail

- CloudFlair 是一种工具，用于查找受公开公开的 CloudFlare 保护的网站的原始服务器，并且不会限制对 CloudFlare IP 范围的网络访问:
  https://github.com/christophetd/CloudFlair
- 通过滥用 DNS 历史记录绕过防火墙:
  https://github.com/vincentcox/bypass-firewalls-by-DNS-history
- 免费版的 cf，我们可以通过 DDOS 来消耗对方的流量，只需要把流量打光，就会回滚到原始 ip
  还有利用 cloudflare 的改 host 返回示例:
  https://blog.detectify.com/2019/07/31/bypassing-cloudflare-waf-with-the-origin-server-ip-address/
  里面给了详细的介绍，我们可以通过 HOST 来判断是否是真实 ip, 具体看文章即可

15、配置不当导致绕过

在配置 CDN 的时候，需要指定域名、端口等信息，有时候小小的配置细节就容易导致 CDN 防护被绕过。

- 案例 1：为了方便用户访问，我们常常将 www.test.com 和 test.com 解析到同一个站点，而 CDN 只配置了 www.test.com，通过访问 test.com，就可以绕过 CDN 了。
- 案例 2：站点同时支持 http 和 https 访问，CDN 只配置 https 协议，那么这时访问 http 就可以轻易绕过。

16、APP

如果网站有 APP，使用 Fiddler 或 Burp Suite 抓取 APP 请求，从中找到真实 IP。

17、F5 LTM 解码法

当服务器使用 F5 LTM 做负载均衡时，通过对 set-cookie 关键字的解码真实 ip 也可被获取，例如：Set-Cookie: BIGipServerpool_8.29_8030=487098378.24095.0000，

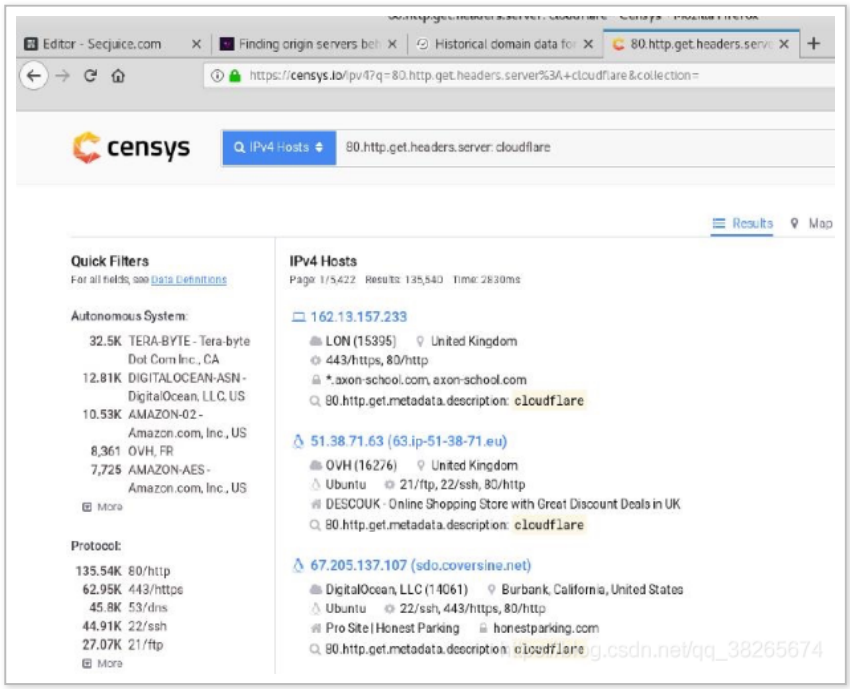
先把第一小节的十进制数即 487098378 取出来，然后将其转为十六进制数 1d08880a，接着从后至前，以此取四位数出来，也就是 0a.88.08.1d，最后依次把他们转为十进制数 10.136.8.29，也就是最后的真实 ip。

18、社工 CDN 平台

19、利用 HTTP 标头寻找真实原始 IP

例如，Censys 上用于匹配服务器标头的搜索参数是 80.http.get.headers.server :，查找由 CloudFlare 提供服务的网站的参数如下：

80.http.get.headers.server:cloudflare



20、利用网站返回的内容寻找真实原始 IP

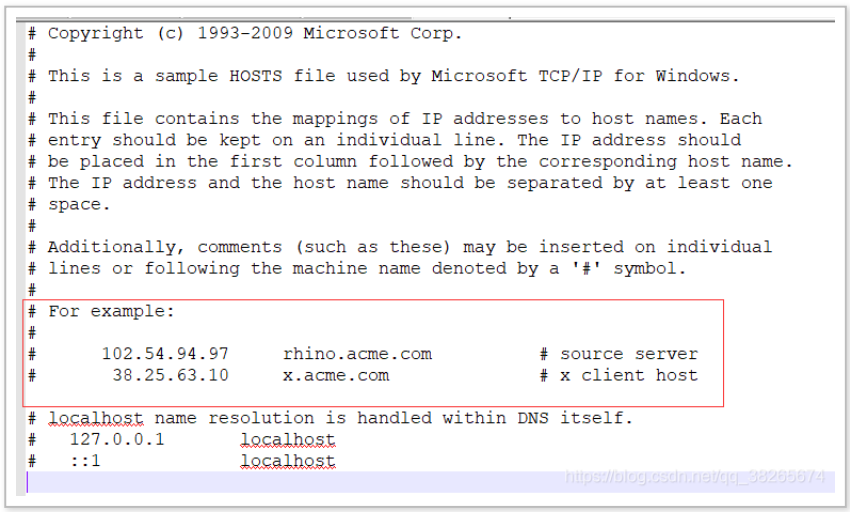
如果原始服务器 IP 也返回了网站的内容，那么可以在网上搜索大量的相关数据。

绕过 cdn 的流程

找到真实 ip 之后，与 hosts 文件绑定，实现能够直接用 ip 或域名访问目标真实服务器。

hosts 文件

电脑在进行 DNS 请求以前，系统会先检查自己的 Hosts 文件中，是否有这个地址映射关系，如果有则调用这个 IP 地址映射，如果没有再向已知的 DNS 服务器提出域名解析。
所以 Hosts 文件是用来提高解析效率的。
也可以理解为，Hosts 的请求级别比 DNS 高，当 Hosts 文件里面有对应的 IP 时，它会直接访问那个 IP，而不通过 DNS。
hosts 文件路径：
C:\Windows\System32\drivers\etc\hosts
使用方法如下：



参考链接：

1. 绕过 CDN 查找网站真实 ip
2. 绕过 CDN 寻找真实 IP 地址的各种姿势
3. 绕过 CDN 寻找真实 IP 的 8 种方法
4. https://www.cnblogs.com/qiudabai/p/9763739.html
   此文章由网上的文章总结而来，有一些文章没有链接记录望谅解。