

DNS

# 白名单之殇：Specter僵尸网络滥用ClouDNS服务，github.com无辜躺枪



Hui Wang, Alex.Turing, litao3rd, YANG XU

Nov 18, 2021 • 11 min read

## 摘要

威胁情报的应用，始终存在着“漏报”和“误报”的平衡，为了减少可能的误报带来的业务影响，你的威胁情报白名单中是否静静的躺着 www.apple.com、www.qq.com、www.alipay.com 这样的流行互联网业务域名呢？你的机器学习检测模型，依照历史流量，是否会自动对 .qq.com、.alipay.com 这样的流量行为增加白权重呢？

但安全是对抗，白帽子想“判黑”，黑客想“洗白”。我们看到的白，不一定是真的白，可能只是黑客想让我们以为的白。

我们BotnetMon最近的跟踪发现，Specter僵尸网络家族的样本，会使用 `api.github.com` 这种域名作为CC域名来通信，通过“可定制化”的DNS服务，将“白域名”引导到黑IP上来实现自己恶意指令通信。

这种“过白”手法，在流量检测的场景下，让传统的IOC情报失陷判断与拦截方法失效，因为CC是绝对的白域名。从监测到阻拦，都对我们安全防守方提出了新挑战。

## 缘起

早在去年9月份，我们首次披露了 [Specter僵尸网络\(2020年9月\)](#)。该僵尸网络是一个针对 Linux 平台的远程控制木马（RAT），配置灵活，高度模块化/插件化，由 Dropper, Loader, Plugin三大部分组成，主要功能由Loader&Plugin决定，自从我们披露后一直活跃至今。

今年9月份，我们BotnetMon的CC自动抽取系统提醒我们，Specter的样本有更新，且自动提取的CC是 `api.github.com:80`。

众所周知github.com是全球公开的开源代码托管网站，之前很多恶意软件都有对github.com的使用，不过只是将其当成下载站，通过github.com来承载自己的恶意程序所需的代码/中间件，通过http下载使用。

但是直接请求到 `api.github.com:80` 为何能完成CC通信呢？是github被黑了，还是Specter僵尸网络的运营者出错了，亦或是我们自己CC自动抽取的bug？我们带着好奇开始了调查。

我们以样本(md5:2aec3fo6abd677f5f129ddb55d2cde67)为切入点，经过分析，发现Specter的这次更新主要集中在C2配置文件的结构上，在之前的版本中可以通过"SpectCF"字串来定位配置文件，而新版本中则剔除了此标志。此样本的C2配置文件解密后如下所示，可以看出绿色部分正是上文所说的

C2: `api.github.com:80`。

00000000:	44 FD 0D 9F 80 CF 49 23 9F D5 D9 6D AF 59 FE D9	D? ?€I#??m?Y??
00000010:	5C B4 CB DD 38 26 40 6A 8D E6 AB 66 50 D1 B2 92	\???8&@j???fP???
00000020:	70 61 75 65 36 7A 63 33 00 01 00 00 00 0E 00 00	pau6zc3 □ □
00000030:	00 61 70 69 2E 67 69 74 68 75 62 2E 63 6F 6D 02	api.github.com
00000040:	00 00 00 38 30 9E E9 9F 55 1E 01 3B 6C E1 B7 B6	80???U□,1???
00000050:	D9 A9 B4 CE B9 00 00 00 00 08 07 00 00 01 00 00	????? □□ □
00000060:	00 01 00 00 00 01 00 00 00 01 00 00 00 01 00 00	□ □ □ □
00000070:	00 01 00 00 00 01 00 00 00 01 00 00 00 0A 00 14	□ □ □ □
00000080:	00 0A 00 14 00	□

C2,PORT

DNS IP

而红色部分的数据则是这次更新多出来的结构，它又是什么呢？以小端格式解析后，发现它们是如下4个IP地址，隶属于DNS Hosting服务商[ClouDNS](#)。

85.159.233.158  
108.59.1.30  
217.182.183.225  
185.206.180.169

Specter新样本是通过以下代码片段对C2进行dns请求，它的逻辑为构造dns请求数据包，然后发往上文所述的DNS IP，最终得到真正的C2地址。

```

v18 = compose_dns_packet((const char *)((v10[7] << 24) | (v10[6] << 16) | (v10[5] << 8) | v10[4]), &v12, &v13, 1u);
if ( v18 )
{
    v15 = random_proc(0, 5);
    for ( i = 0; i <= 4; ++i )
    {
        v3 = v15 + i;
        v14 = v3 % 5;
        v4 = v3 % 5 + 2;
        if ( (v10[4 * v4 + 3] << 24) | (v10[4 * v4 + 2] << 16) | (v10[4 * v4 + 1] << 8) | v10[4 * v4] )
        {
            v17 = dnsquery(
                (int)v18,
                &v12,
                (v10[4 * (v14 + 2) + 3] << 24) | (v10[4 * (v14 + 2) + 2] << 16) | (v10[4 * (v14 + 2) + 1] << 8) | v10[4 * (v14 + 2)],
                0x35u,
                5,
                3u);
        }
    }
}

```

读者可以自行使用下面的dig命令，通过对比它们的输出结果，可以很清晰的看出这其中的区别。

```

dig api.github.com @8.8.8.8

dig api.github.com @85.159.233.158

```

至此迷雾褪去，Specter所使用的CC `api.github.com` 实际为在DNS Hosting服务商ClouDNS注册的ZONE `github.com` 下面的子域名，其中 `api` 为其子域名。这种域名的解析必须使用 ClouDNS 提供的解析服务器才能获取到特定的解析结果。

github没有被黑，Specter僵尸网络的运营者也没有出错，我们的CC自动提取也没有bug，而api.github.com这个白域名也确确实实的成为了CC域名。这种使用看起来正常白域名的行为的迷惑性非常强，很容易欺骗恶意软件分析人员，对基于黑白名单规则判定的安全工具也是一种巨大的挑战。

## ClouDNS可随意注册DNS Zone

鉴于上述利用过程，ClouDNS是核心的一个环节，我们对其进行了详细的探查。

[ClouDNS](#)是位于欧洲的全球托管DNS服务提供商，提供包括GeoDNS，Anycast DNS和DNS DDoS防护等服务。

ClouDNS允许随意注册DNS ZONE并添加子域名解析。我们注册了一个名为 `nsa.gov` 的DNS Zone，添加了一个子级域名 `test` 并解析到 `16.16.16.16`，ClouDNS给我们分配了4个Name Server用来解析这个域名，如下图所示：

创建成功后，便可使用平台分配的Name Server来解析我们创建的域名：

理论上，我们可以在ClouDNS上注册任何没有被注册或者没有被ClouDNS限制注册的Zone。前文提到的Specter C2 [api.github.com](https://api.github.com) 正是以这种方式产生出来的的域名。

不仅如此，ClouDNS在对“是否已经注册”的判定上也有“谜之逻辑”。前面说到github.com 在ClouDNS上已经被Specter团伙注册使用，但是当我们尝试重新注册github.com 这个Zone的时候，竟然也能成功，只是使用了和Specter团伙不同的一批NS。如图：

也既，ClouDNS支持将同一个zone根据不同的NS Server做不同的绑定。我们可以将任意一个白域名注册在不同的NS server上，然后为它绑定不同的IP。

以这种方式创建出来的域名和现有DNS体系完全混淆，如同平行宇宙一般，有着同样的名字的人但是过着截然不同的生活，同样的域名在不同的DNS体系下有着截然不同的功用。如果两个世界一直平行也无妨，但是黑暗世界的“福尔摩斯”来到人间作案，他的相貌可能会让他畅通无阻。目前我们已经看到的Specter利用这种方式创建出来的域名还有 [www.ibm.com](https://www.ibm.com)，正常世界的www.ibm[.]com 仍然是一个白域名，而平行世界的它正在被恶意团伙利用。

其实不只是ClouDNS，很多DNS托管服务商，在对托管域名的校验上，都存在类似的“漏洞”，这是另外一个宏大的话题，在此暂不展开。

## 对ClouDNS随意注册ZONE的探测

基于我们的Passive DNS数据，我们挑选了全网流行的TOP 1M二级域名进行探测。想探知有多少现有DNS系统里的域名的SLD有多少被在ClouDNS注册为新的Zone以及有多少可能被恶意注册。

探测结果显示其中大约有 1263 个二级域名在ClouDNS支持解析。考虑到ClouDNS本身也是合法的正规业务，只是业务存在被滥用的可能性，我们对支持解析的二级域名进行了过滤，过滤了明确非恶意注册的二级域后发现有超过 300 个二级域是恶意注册。部分在ClouDNS被恶意注册SLD如下：

safe.com  
consalud.cl  
godaddysites.com  
shopee.com  
jsdelivr.net  
afraid.org  
rumahweb.com  
mydomain.com  
crypto.com  
eq.edu.au  
adnxs.com  
webcindario.com  
web.com  
lamborghini.com  
manager-magazin.de  
toto.com  
migalhas.com.br  
googleadservices.com  
example.com  
dlink.com  
whitehouse.gov  
domain.com  
googlesyndication.com  
fb.com  
payeer.com  
ya.ru  
mql5.com  
aaa.com  
hola.com  
wukong.com  
mihanblog.com  
wpengine.com  
jumia.ma  
protonmail.com  
tasnimnews.com  
nintendo.com  
tabnak.ir  
lichess.org  
digitalocean.com  
asriran.com  
amazon.com.br  
akamaized.net  
yjc.ir  
office.net  
4399.com  
opera.com  
wp.com  
ytimg.com  
avast.com  
cloudflare.com  
playstation.com  
hespress.com  
leagueoflegends.com

wixsite.com  
skype.com  
googlevideo.com  
wp.pl  
wix.com  
samsung.com  
doubleclick.net  
weebly.com  
udemy.com  
speedtest.net  
godaddy.com  
zoom.us  
espn.com  
spotify.com  
amazonaws.com  
adobe.com  
wordpress.com  
apple.com  
msn.com  
github.com  
office.com  
alipay.com  
netflix.com  
360.cn  
amazon.com  
qq.com

另外我们还挑选了全网流行的TOP 1M的FQDN进行探测，探测结果显示有超过 300 个FQDN可以在ClouDNS产生非正常解析，过滤掉可能是正常业务和噪音数据后发现，至少有 192 个FQDN是恶意注册。

## 总结

我们大网探测结果显示，当前在我们的视野范围内还没有太多的恶意样本使用这种方式实现攻击。但这对我们来说是一个重要的预警，特殊的业务逻辑，导致表面正常网络行为的掩盖下，极有可能正在进行着恶意行为。这种方式并不需要任何额外的成本，就可以轻松实现 DNS 欺骗，进而实现对恶意行为的掩藏。

从阻拦的角度来说，ClouDNS的操作，已经让一个域名无法“自证清白”，常规的基于 IOC 匹配的安全检测行为完全失效。我们不可能将 github.com 加入到 IOC 列表中作为威胁情报输出到产品端。基于我们的搜索结果显示，以类似方式提供 DNS 托管的服务提供商还有很多，我们后续将新写一篇度量此类DNS的文章。同时考虑到这些服务提供商还提供着正常的业务，我们也无法将这些服务提供商的服务地址作为威胁情报输出到产品端。

当然，各安全业务方如果确定ClouDNS对自己生产系统没有影响，可以直接阻拦掉ClouDNS的所有NS IP的流量。仔细考虑ClouDNS如此粗暴的实现逻辑，这种阻拦方式可能也没有听起来那么粗暴，我们认为企业应该在自己的防护边界把这种未经许可的DNS访问都阻拦掉。

从域名解析商的角度，在用户添加ZONE的时候，应该采取必要的验证手段确保添加者是域名的合法持有人，我们可以看到好的提供商，比如国内的提供商[DNSPOD](#)需要验证通过后才支持解析。

从检测的角度来说，除了从样本角度出发的发现跟踪，在流量侧，需要有两个基础能力的支撑：

- 识别并监测非标准DNS server的能力
- 配合历史白数据，做异常DNS流量判定的能力

近期我们发布了[360 DTA](#)产品，该产品从 DNS 流量入手，结合我们运营多年的 PDNS 数据、botnet 数据，以及多维度数据关联分析等技术可以轻松检测到这种可能是恶意的 DNS 欺骗行为并给予正确的告警，帮助客户及时定位未知威胁、高级威胁，降低攻击影响面，提升情报生产及安全运营能力。

## IOC

### Sample MD5

```
0ffa01708fd0c67c78e9055b8839d24d  
162c245378b2e21bdab6ef35dfaad6b1  
2aec3f06abd677f5f129ddb55d2cde67
```

## CC

```
45.141.70.5  
www.ibm.com @pns101.cloudns.net  
api.github.com @ns103.cloudns.net
```



Start the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS



Name



Share

Best Newest Oldest

Be the first to comment.

Subscribe

Privacy

Do Not Sell My Data

— 360 Netlab Blog - Network Security Research Lab at 360 —

## DNS



新威胁：使用DNS Tunnel技术的Linux后门B1txor20正在通过Log4j漏洞传播

The Pitfall of Threat Intelligence Whitelisting: Specter Botnet is 'taking over' Top Legit DNS Domains By Using ClouDNS Service

七年一剑，360 DNS威胁分析平台

DNS

**The Pitfall of Threat Intelligence Whitelisting: Specter Botnet is 'taking over' Top Legit DNS Domains By Using ClouDNS Service**

Import 2022-11-30 11:16

**Malware uses namesilo Parking pages and Google's custom pages to spread**

Abstract Recently, we found a suspicious GoELFsample, which is a downloader mainly to spread mining malwares.

The interesting part is that we noticed it using namesilo's Parking page and Google's user-defined page to spread the sample and configuration. Apparently this is yet another attempt to hide

[See all 5 posts →](#)

Abstract In order to reduce the possible impact of false positives, it is pretty common practice for security industry to whitelist the top Alexa domains such as [www.google.com](http://www.google.com), [www.apple.com](http://www.apple.com), [www.qq.com](http://www.qq.com), [www.alipay.com](http://www.alipay.com). And we have seen various machine learning detection models that bypass



Nov 12, 2021 3 min read

Nov 18, 2021 6 min read

