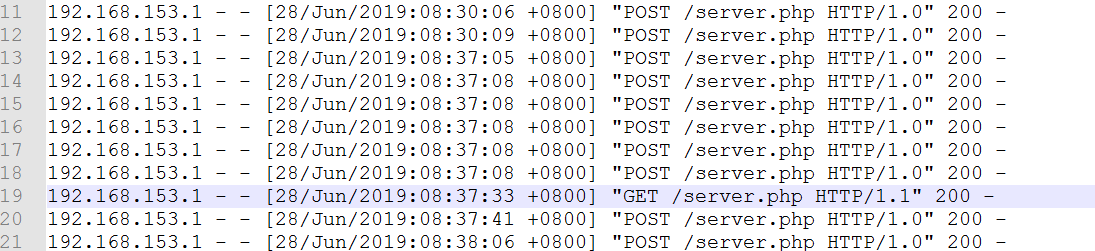
在应急响应中经常会从日志中看到一些常见的攻击行为，在渗透测试中也会利用一些攻击工具，为了能从日志中获取更多有价值的信息，来及时确定攻击来源，日志记录格式通常为，访问的ip地址，时间，访问路径，服务器响应状态，返回数据大小。

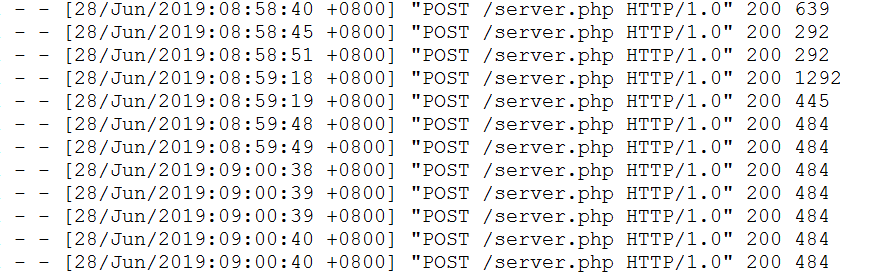
一、简单列举下常见的攻击日志

1、菜刀连接记录

利用菜刀连接一句话后门，在web日志中只能看到对菜刀马的post连接请求，但是具体看不到利用菜刀马做了哪些操作。



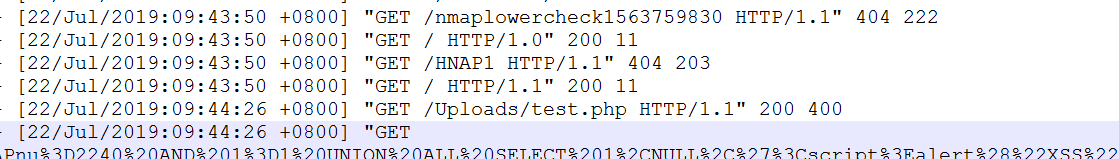
从日志中只能看到连接具体通过菜刀马的发送字节数，只能说明攻击者肯定是通过连接菜刀马读取或是新建、下载了相关文件，可从字节数的大小粗略判断读取文件的大小，但是貌似没啥用也。



虽然常见的安全设备已经对菜刀流量进行了拦截，可通过流量中转的方式尝试绕过安全设备的拦截。

2、端口扫描、sql注入攻击

端口扫描是常见的攻击方式，通过测试，当利用nmap进行端口扫描时，由于nmap常见使用方式为采用tcp、udp、ping扫描方式，所以access log中不记录相关日志，在日志只有一条get请求

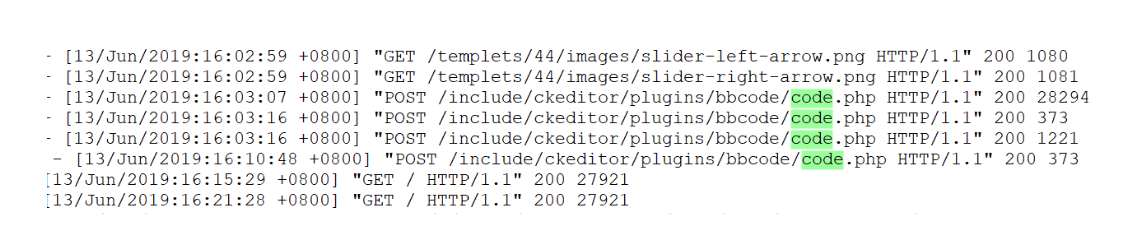


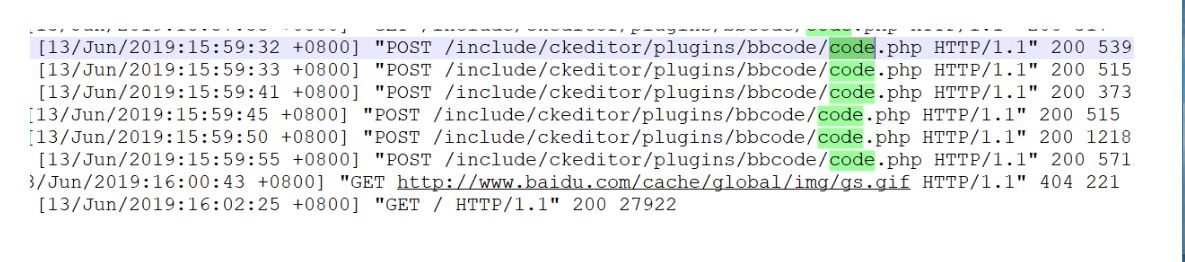
sql注入攻击流量，向指定参数提交sql查询



3、shell链接记录

服务器被拿下或进行了内容的篡改，多数是从一个shell的写入开始。shell通常的写入方式有以下几种，通过文件上传页面、常见中间件、cms、编辑器、框架命令执行漏洞、sql注入、phpmyadmin写入等方式。虽然从日志中我们可能无法看到具体写入的shell内容，当然get方式写入的除外，但是可以看到攻击者有没有对后门进行访问。

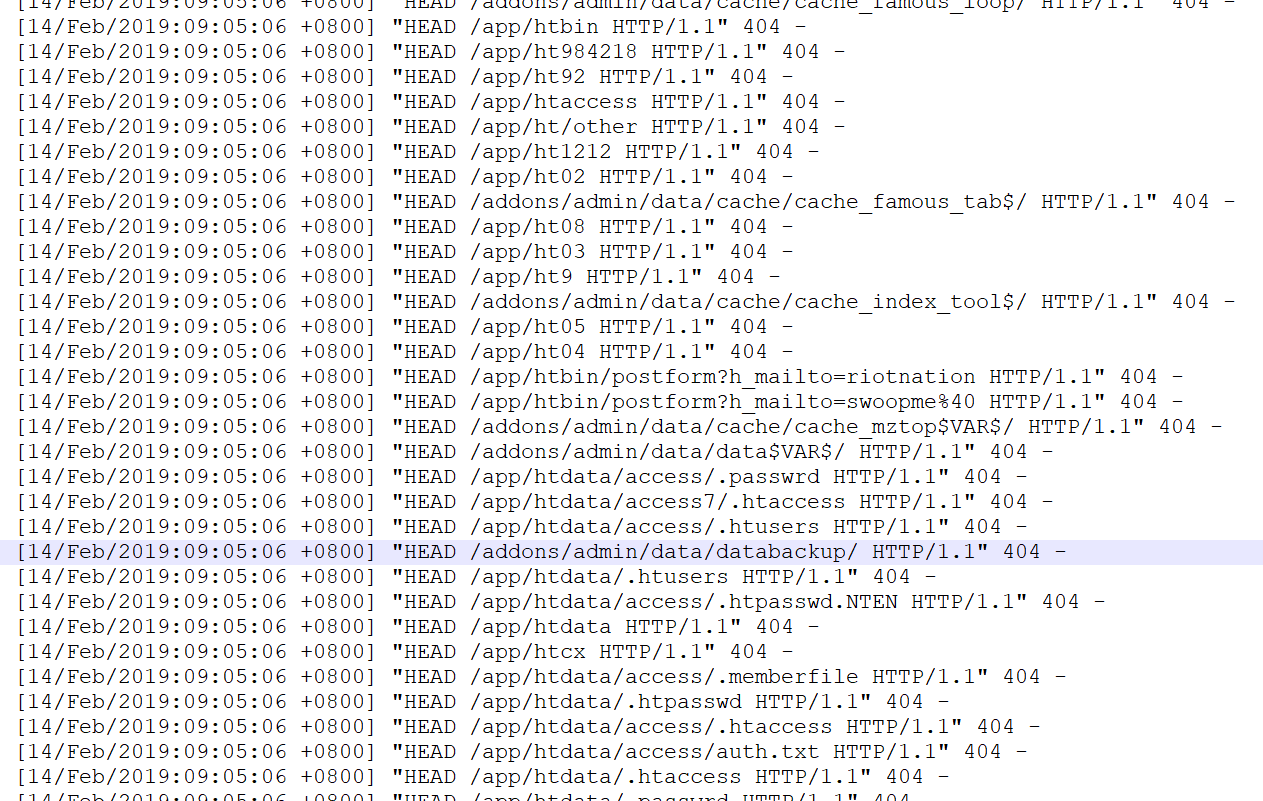




通过查看后门名称在日志中寻找攻击者是如何将后门传上去。

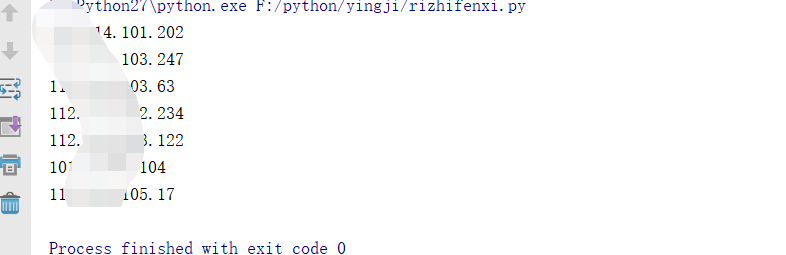
4、目录枚举

目录枚举是攻击的常见方式，通过目录枚举可以获取大量敏感信息或是未授权访问的文件。目录枚举通常是以GET或HEAD方式对url进行路径遍历，目录遍历多数是跑字典的方式，所以由于目标不存在目录而出现404的报错。这个时候写个小脚本收集下别人的枚举的目录，丰富下自己的字典其实也还不错。

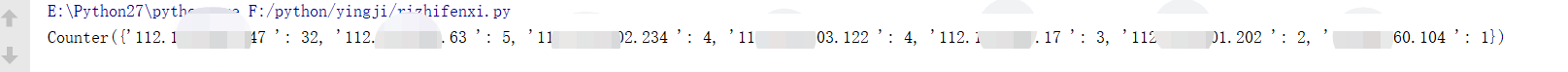


二、日志分析

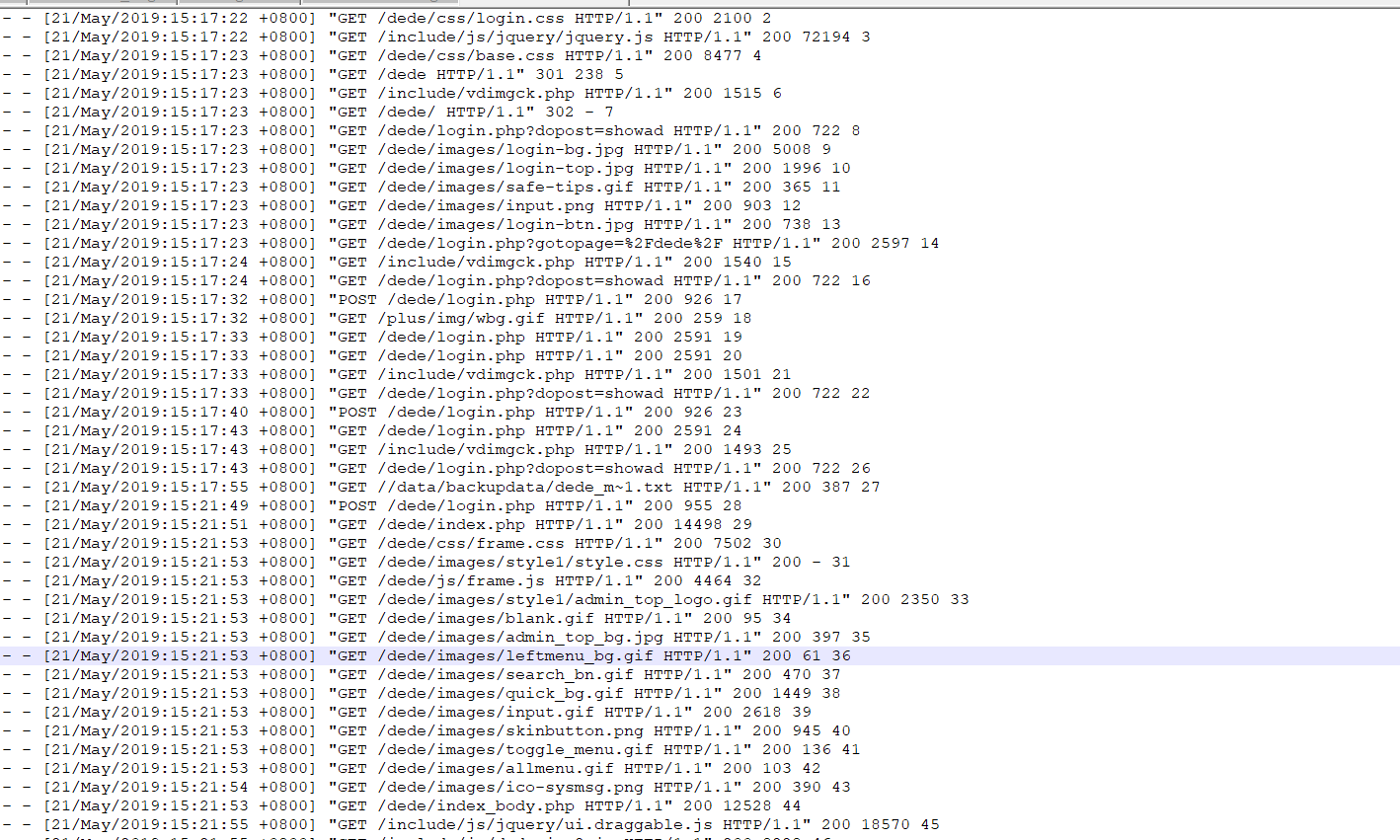
日志往往是很庞大的，并且从中记录的一些正常请求和非正常请求，所以可以尝试采用如下方式进行选择性日志分析查看。可通过扫描网站目录，确定webshell名称，在根据webshell名称从日志中找到访问过的ip，然后在过滤ip地址，看看该ip对系统做了哪些操作。假如现在已经通过查杀发现5678.php为异常后门文件，可通过python定位下攻击ip地址



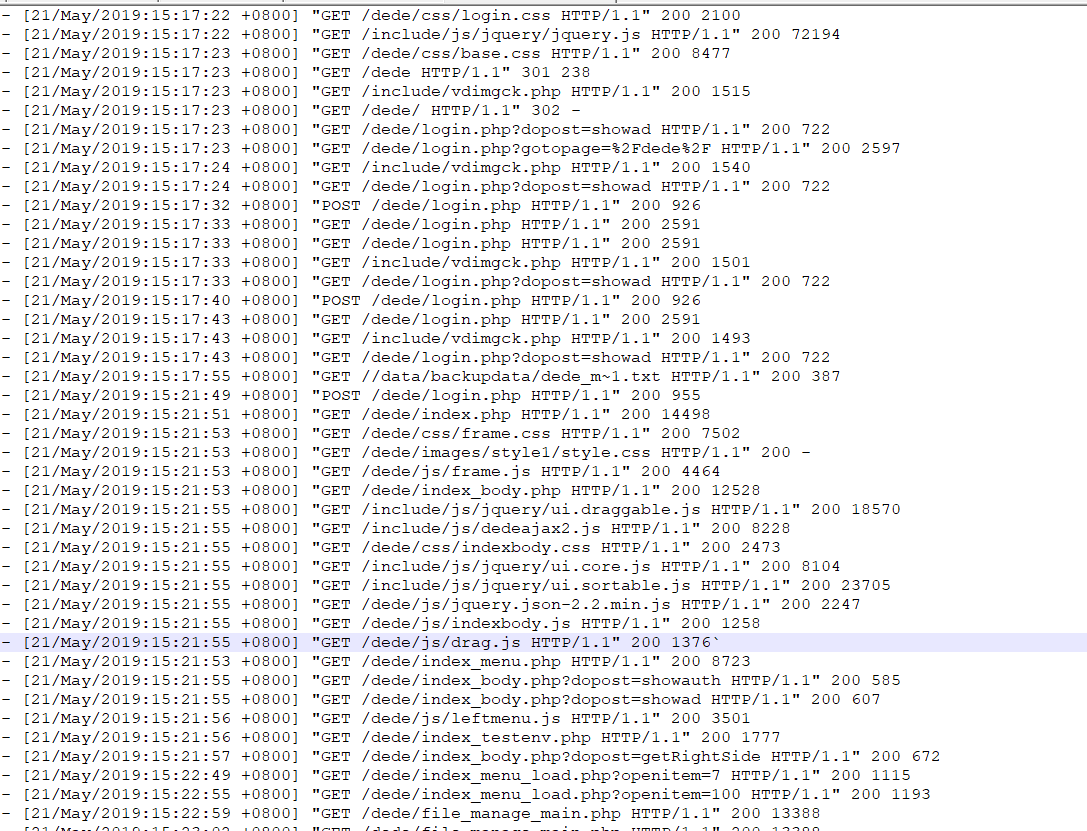
还可统计ip地址的访问频率，定位到访问次数比较多的ip地址，如下



在根据ip地址，定位出该ip地址都做了哪些操作，这样初步筛选下来日志的访问记录就已经减少了很多 了



虽然是减少了很多，但是如果人工去看，还是很费劲，通过观察请求还是可以继续精简的，比如过滤到get请求中的一些图片的链接。



这样下来在进行日志查看，就相对比较容易找到攻击源了，但是很多情况下日志数量要不这多的多，可以根据分析情况自行筛选，当然也可以采用自动化分析工具，很多牛人也写过日志分析脚本，不管用何种方式，找到适合的最好。可参考如下牛人写的脚本。

<https://www.freebuf.com/sectool/110644.html>

<https://www.freebuf.com/sectool/126698.html>

<https://www.freebuf.com/sectool/8982.html>

<https://www.freebuf.com/articles/web/96675.html>