记一次寻常的评估任务

原创六号刃部 酒仙桥六号部队
2020-08-31原文

这是 **酒仙桥六号部队** 的第 **71** 篇文章。

全文共计2754个字，预计阅读时长9分钟。

背景

一年一度的xx行动，各大行业或企业都要接受xx的检验。在正式检验前，都会做一些演习来对自家的系统摸摸底，查漏补缺。本次评估任务就是在这么一个情况下展开。

在某个风和日丽的下午接到上级交代的任务后，波澜不惊的我掀开本儿，点开目标列表，就开始了。又是一个寻常的历程。

前期信息收集

通过前期的踩点，对目标有了一个初步的判断。其本身的安全建设不是那么跟的上，但还是有一些防护，猜测应该是临时堆砌起来的一些设备，这种情况下安全设备本身的配置大概率会存在一些问题。

正常的信息收集走一波，目录、端口、子域名、web指纹、C段，这里有一点挺有意思的是在做信息收集的过程中发现客户给的其中一个站上了CDN，但是直接ping了一下就直接找到真实IP了，这是一

个典型的配置错误，因此我们很容易就拿到的真实IP，对后续C段的探测节省了时间。

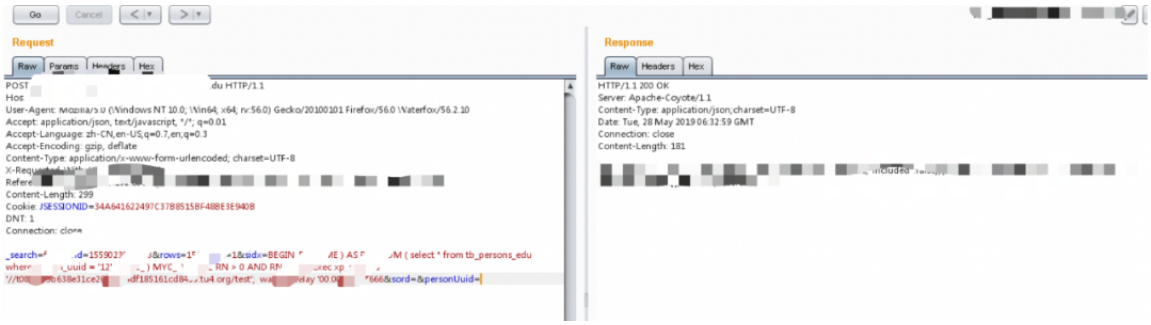
最后对给的目标站点，收集后的信息进行整理成列表，方便查漏补缺。梳理了下，值得突破的有几个后台，以及几个可能存在注入的站点。

通过分析，准备先拿其中一个站点的后台进行突破。

sql注入的滑铁卢

对前面信息收集到的部分后台，先拿没有验证码的后台入手，通过手动测试判断发现存在一个xx的测试账号，手动猜解密码无果，随即用上收集的字典，爆破了一波，运气不错，拿到密码登录后台。

登录系统后点点点的过程中，并没有找到可以直接拿权限的点，随之对其他漏洞进行测试，在一处进行查询的地方，看看查询返回的结果，首先得试试是否存在SQL注入，对查询处进行抓包测试，结合dnslog发现存在注入。



获取dnslog记录：

在确认存在SQL注入，且通过语句可以执行命令，准备进一步利用的时候，突然发现这个站点直接被关掉了，应该是爆破及进行SQL注入测试的时候动静太大，被安全设备检测到了，有人进行了处置，有点难顶。

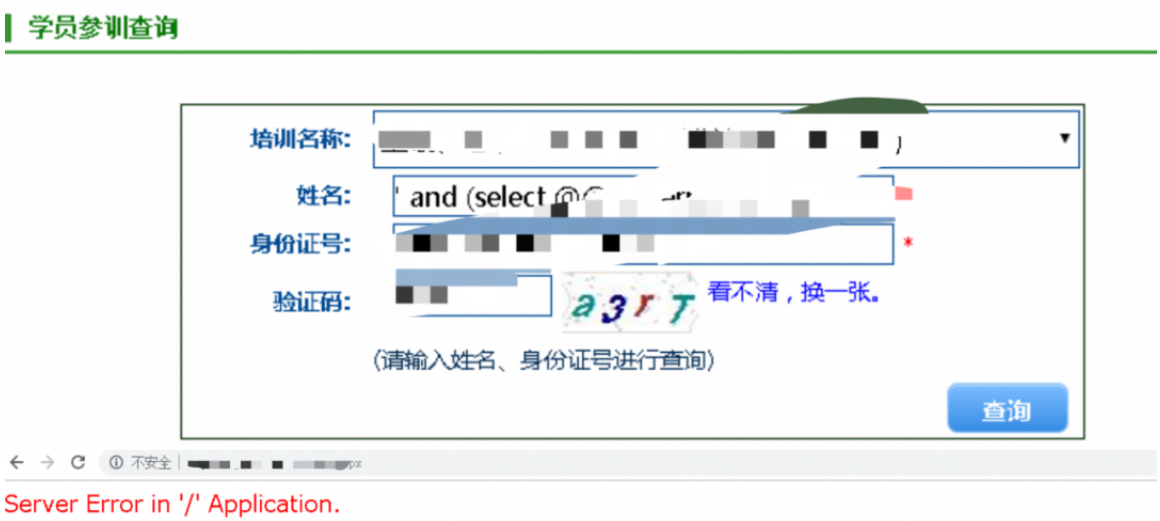
稳住，不慌，这条路走不通的时候，就走其他的路，只要把所有的路都走完，就可以让别人无路可走。

SQL注入到getshell

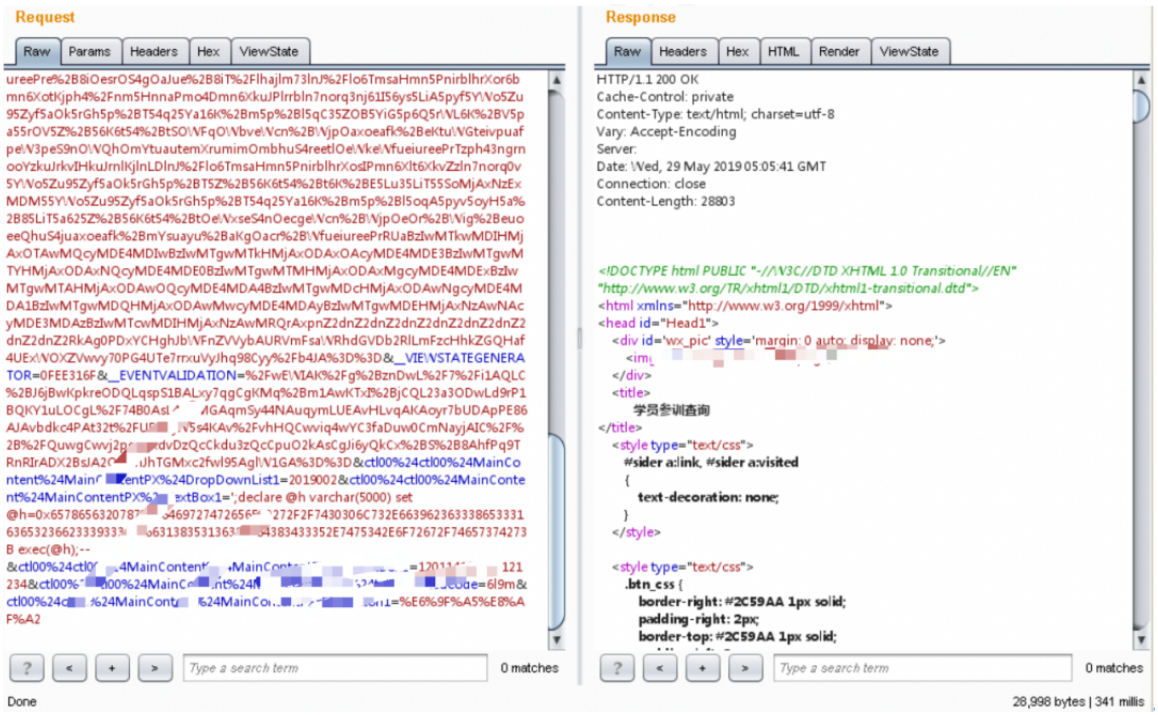
如上所说，发现SQL注入的站点被关停了，虽然有点难顶，但也不算太意外。

目标业务广，范围大，且收集到的可以作为突破点站也有那么几个，因此影响不那么大。

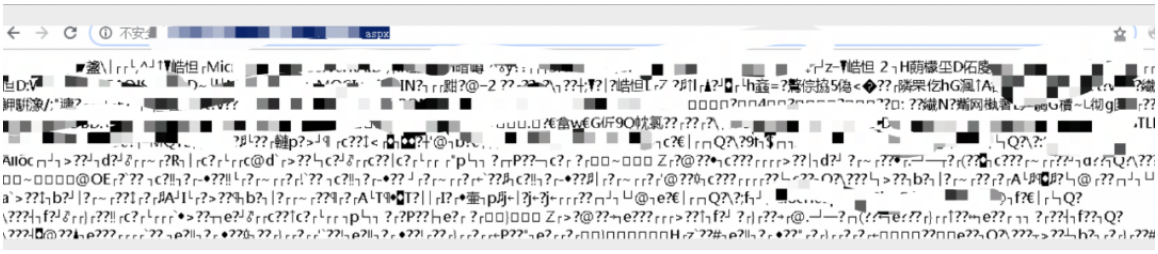
现在转过头来把精力放到另一个看起来比较老旧的站点，点点点的过程中，在前台存在一个培训查询的地方，老方法，测试一下SQL注入，发现姓名处存在SQL注入。



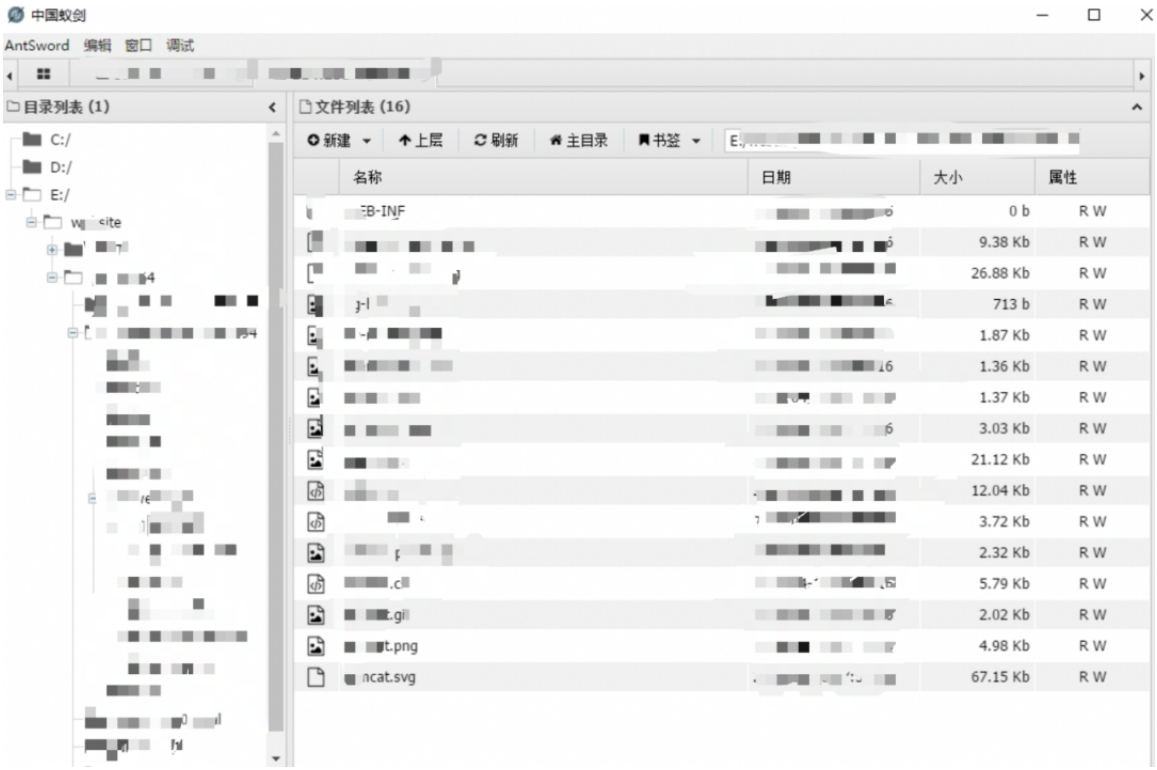
测试的过程中发现这里用的是mssql数据库，一个堆叠注入，手动打开xp_cmdshell。



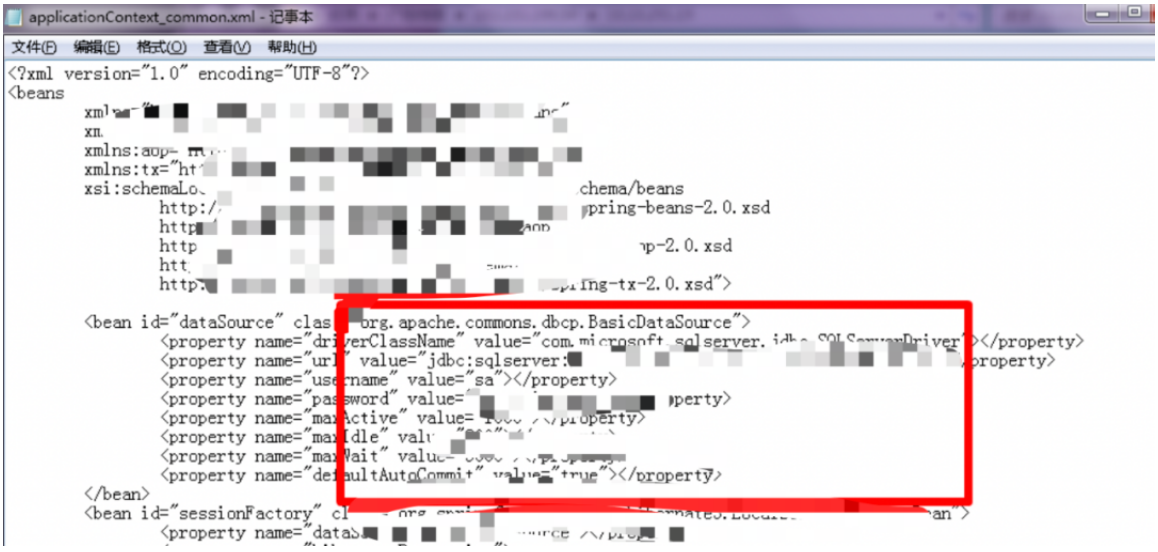
通过xp_cmshell执行系统命令来写入一句话木马后。



拿到shell后，使用蚁剑连接，至此拥有了web权限，随之对当前机器的进行信息收集，web.config，数据库密码及连接记录，当前开放的端口跟连接记录……



在applicationContext_common.xml找到mssql数据库sa用户的密码。



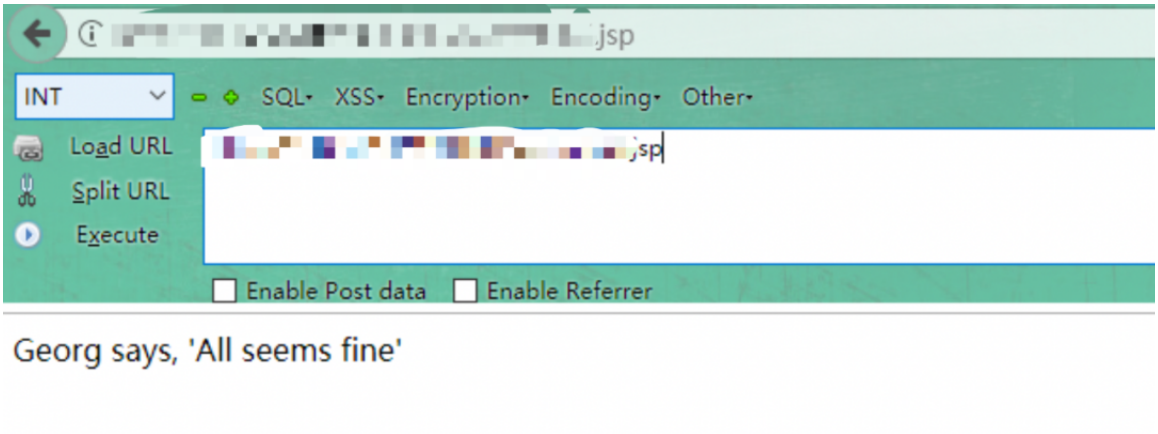
得知目前是system权限，这里直接上传mimikatz抓取密码（补图）。

通过收集整理，获取到administrator密码，mssql数据库的密码，当前机器有两张网卡，一张连入互联网，一张是通内网的，且通过netstat -ano收集到三个内网网段。

那么接下来的就是搭建代理，进入内网，进行愉快的内网游荡了。

内网代理reGeorg

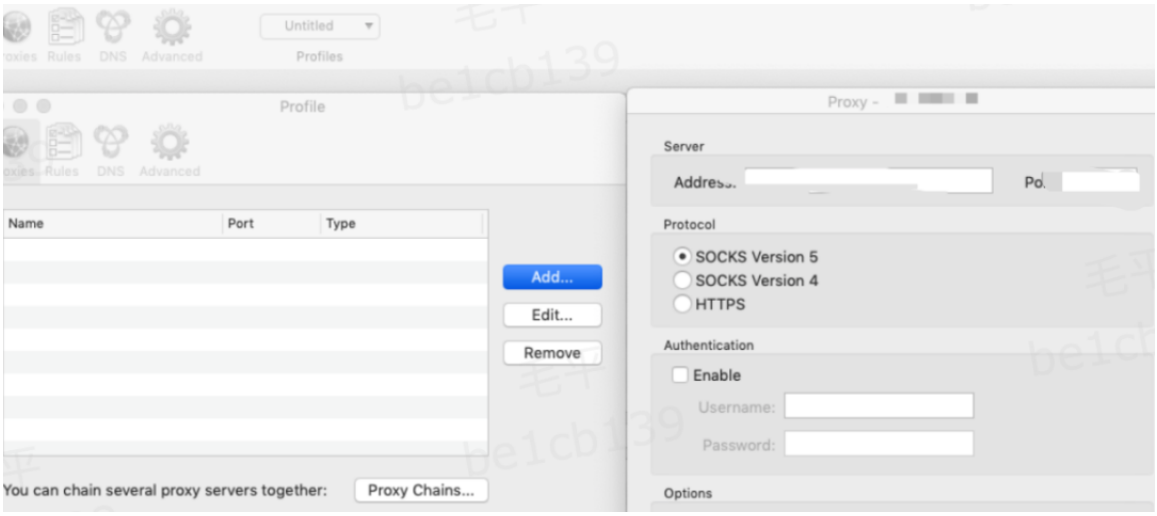
通过之前拿到的shell，当时直接使用reGeorg+Proxifier的方式进行内网的穿透，直接把tunnel.jsp扔到目录中去，之后访问地址加 xxx.tunnel.jsp ，显示“Georg says, ’All seems fine’”，表示脚本运行正常。



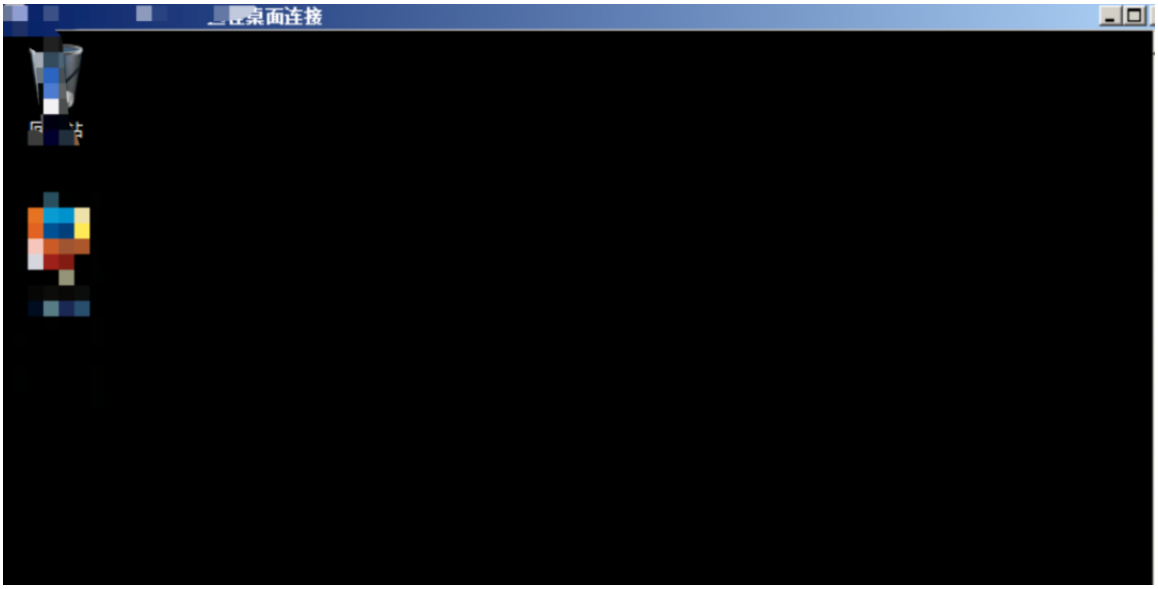
在服务器上执行如下命令：

```
reGeorgSocksProxy.py -p 9999 -u http://pentest.com/tunnel.jsp
```

接着我们配置Proxifier，运行Proxifier之后设置代理，在一切正常的情况下，我们即可访问内网资源。



使用之前通过mimikatz读取到的密码登录服务器。



这里由于已经拿到了服务器的权限，且通过代理成功的登陆了机器。渗透的本质是信息收集，继续对当前机器进行信息收集，翻一切有用的信息，txt 文本，各种配置文件……

这里使用BrowserGhost对当前浏览器存在的密码进行了读取，获取到部分内网服务的登陆权限。

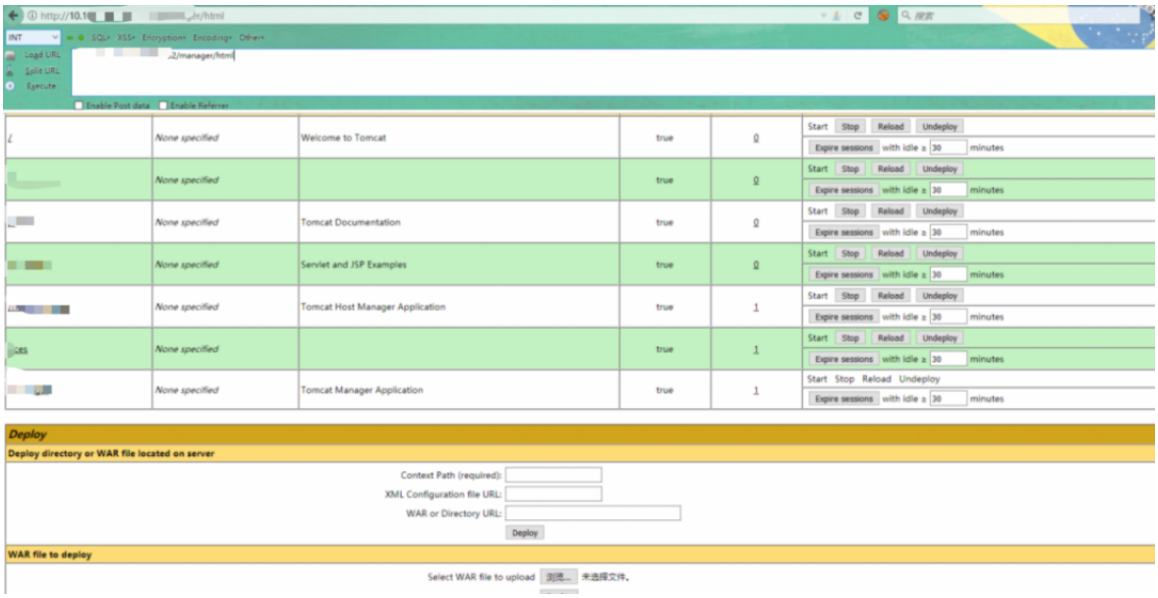
tomcat弱口令

有了据点后，接下来我对已知网段开放服务的情况进行了探测，进了内网后不要瞎扫，流量一大，安全设备一告警，很容易被管理员发现。

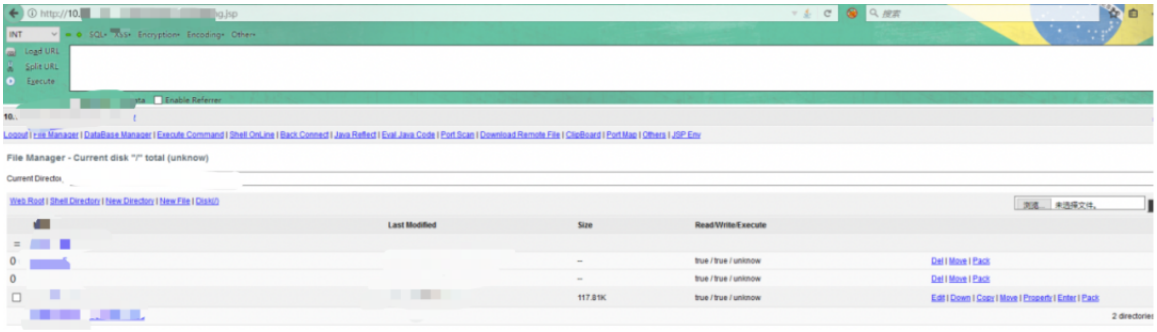
比较好的方式是想探测什么服务，就单单针对这个服务进行探测，线程调低一点，这样只要没有触发态势感知之类的阀值，我们就可以一直愉快的玩耍下去。

这里我使用F-NAScan.py单单针对Web服务做了探测，对其探测结果整理后，存在一部分tomcat的服务，且tomcat一般情况下是一个比较好的突破点。

接下来，这里对tomcat服务的弱口令进行了探测，一开始并不顺利，尝试了收集到的两个站点均以失败告终，在对第三个tomcat服务弱口令进行探测的时候，幸运降临，存在默认口令。使用xx登陆：



到这里，熟悉的页面，熟悉的操作，直接上传war包部署webshell。



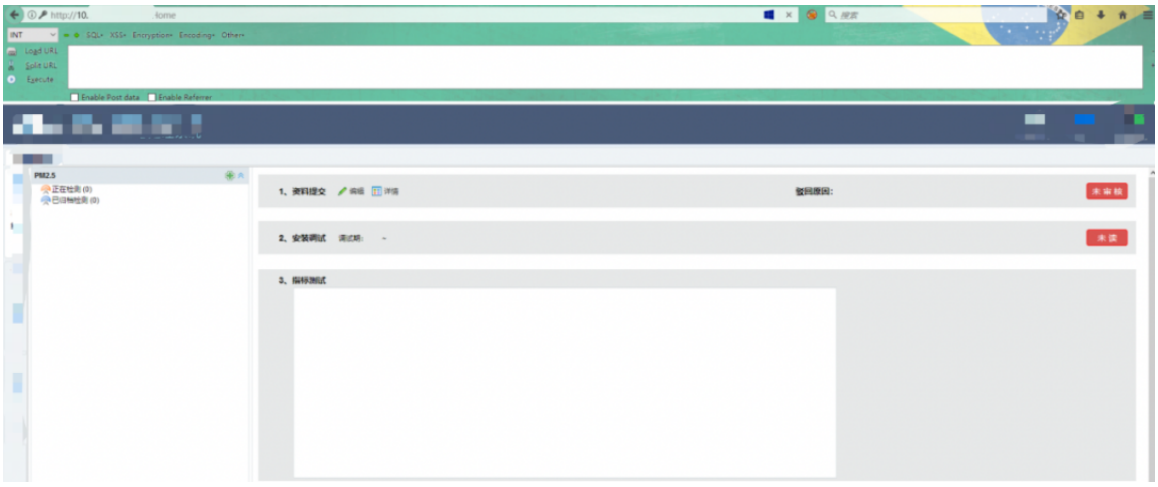
浏览器密码到getshell

在除了tomcat服务外，还获取到了其他的web服务，经过手动判断，发现存在管理员账户，手动猜解无果。随之用上burpsuite

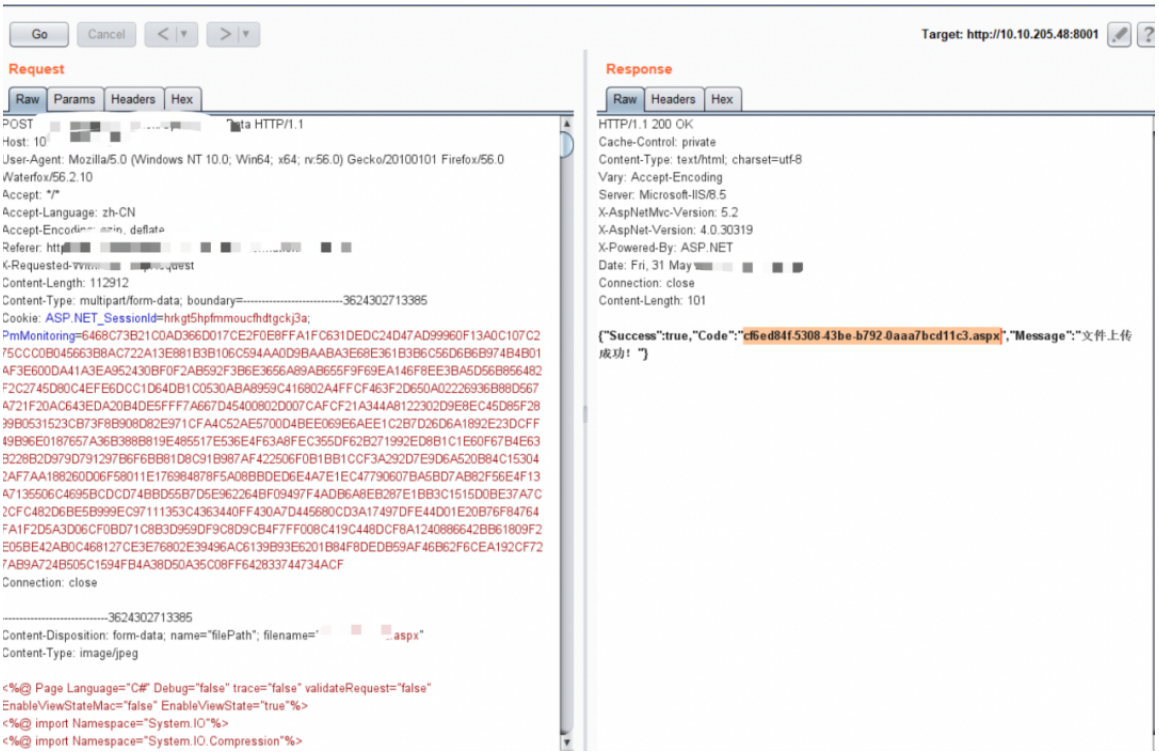
对密码进行爆破。当跑完一个字典没有结果的时候，我预感到不妙。再次试了我的备用字典，无果。

这时我正打算先放在一边看看其他有什么突破点的时候。突然想起我在做代理的那台机器上通过xx读取到的浏览器部分密码。死马当活马医，万一他们有些站点管理员是相同的人，那么密码也有可能是相通的。

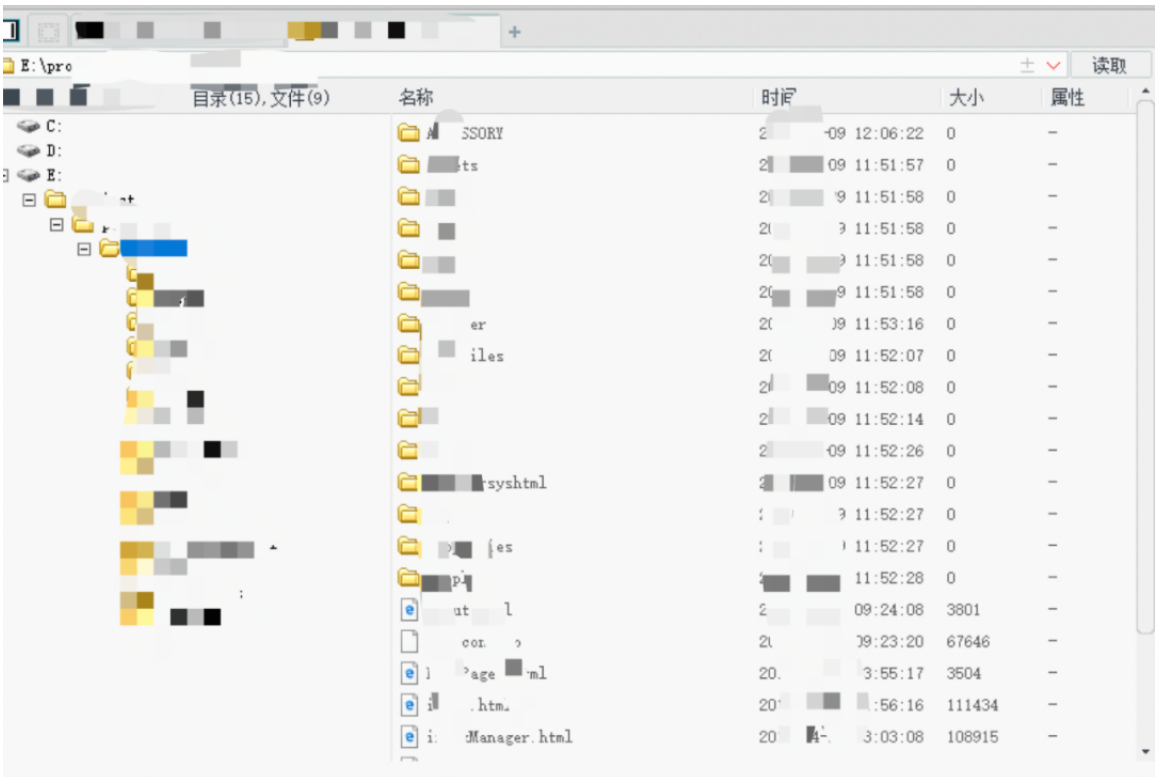
通过之前收集到浏览器中存储的部分密码，尝试登录，发现密码xx成功登陆其后台……



按照基本操作来，优先去找能拿权限的点，经过测试发现在后台的一个添加xx功能处存在任意文件上传，上传shell,可以看到成功上传，并返回了文件名。



对url跟文件名进行拼接，确认后使用菜刀连接。



这个时候已经通过找到的web服务拿到了部分机器的权限，但这样一个一个的拿，效率不那么高，想起了最开始的时候收集的mssql数据库的密码，换着走另一条弱口令的爆破路。

弱口令的沦陷

收集到了1433端口mssql数据库的密码，单独针对1433端口对已知的网段进行了扫描，整理出一批内网开放1433服务的列表，随之通过sa用户，跟单口令进行爆破，成果如下：

10.10.	SQLServer	1433	sa		11.00.3460	0
10.10.	SQLServer	1433			10.50.1600	0
10.10.	SQLServer	1433			10.50.1600	0
10.10.	SQLServer	1433			10.50.1600	0
10.100.	SQLServer	1433			10.50.1600	9
10.100.	SQLServer	1433			10.50.1600	7
10.100.	SQLServer	1433			10.00.1600	11
10.100.	SQLServer	1433			10.00.2531	14
10.100.	SQLServer	1433			10.00.2531	14
10.100.	SQLServer	1433			10.50.1600	4
10.100.	SQLServer	1433			10.50.1600	7

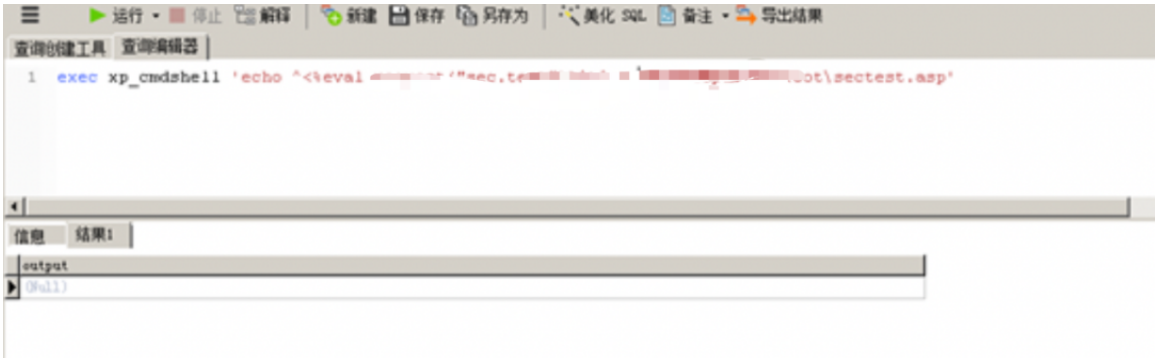
根据之前抓到的服务器密码，老方法，依旧使用口令进行爆破，这里获取到部分服务器成果如下：

10	SMB	445	administrator		258
10	SMB	445	administrator		192
10.	SMB	445	administrator		165
10.	SMB	445	administrator		140
10.	SMB	445	administrator		172
10.	SMB	445	administrator		200

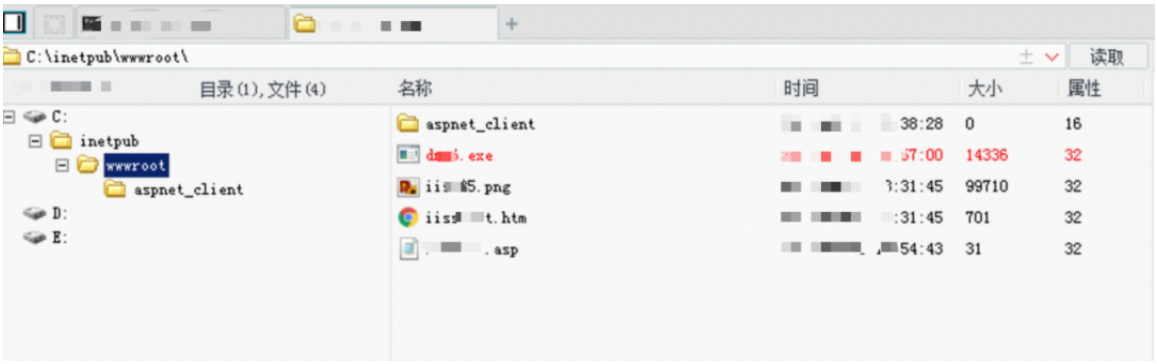
mssql写shell

这里直接用navicat连接上mssql数据库，测试exec xp_cmdshell

’systeminfo’发现可以直接执行命令，这里直接通过系统命令写入一句话。



随后用菜刀连接。



dns加密隧道控制

对获取到的目标进行控制，使用Cobalt Strike，生成后门，然后扔到目前已经拿到权限的所有目标机器上去，相关上线主机如下：

清理痕迹

到这一步的时候，因为这次任务只是一个普通的评估任务，也没有说要拿到指定的靶标，且全程都在跟客户进行沟通，此时已经达到了初步的效果，因此被‘叫停’。

叫停之后，我们对上传的shell，dns.exe、tunnel.jsp等上传过的后门及其他文件进行了一一的清除，且什么地方执行的关键操作均在报告中体现后，提交报告，随之本次任务正式结束。

总结

本次任务是一个寻常的评估任务，过程当中也都是一些常规的操作。一是目标本身的安全建设并不是那么的成熟，突破点多，且进入内网后，通过信息收集找到密码，跟一些常规的密码就直接跑了一批服务或机器的权限。二是进入的区域是DMZ区，并没有过多的深入。

操作很基础，希望各位大佬不要见笑，希望通过一次又一次的项目，积累经验，努力的提升技术，毕竟渗透很快乐，技术厉害了就是双倍的快乐！

精选留言

用户设置不下载评论