用户权限与文件权限

1 用户权限和配置命令

2 文件权限归属

3 文件的特殊权限

4 文件的隐藏属性

5 文件访问控制列表

用户权限

用户权限

管理员UID为0：系统的管理员用户。

系统用户UID为1～999： Linux系统为了避免因某个服务程序出现漏洞而被黑客提权至整台服务器，默认服务程序会有独立的系统用户负责运行，进而有效控制被破坏范围。

普通用户UID从1000开始：是由管理员创建的用于日常工作的用户。

useradd

useradd命令用于创建新的用户，格式为“useradd [选项] 用户名”。

```
[root@localhost ~]# useradd -d /home/eagle -u 8888 -s /bin/bash
eagle
[root@localhost ~]# id eagle
uid=8888(eagle) gid=8888(eagle) 组=8888(eagle)
```

useradd

参数	作用
-d	指定用户的家目录（默认为/home/username）
-e	账户的到期时间，格式为YYYY-MM-DD.
-u	指定该用户的默认UID
-g	指定一个初始的用户基本组（必须已存在）
-G	指定一个或多个扩展用户组
-N	不创建与用户同名的基本用户组
-s	指定该用户的默认Shell解释器

groupadd

groupadd命令用于创建用户组，格式为“groupadd [选项] 群组名”

```
[root@localhost ~]# groupadd work
```

usermod

usermod命令用于修改用户的属性，格式为“usermod [选项] 用户名”

```
[root@localhost ~]# usermod -G root eagle
[root@localhost ~]# id eagle
uid=8888(eagle) gid=8888(eagle) 组=8888(eagle),0(root)
```

usermod

参数	作用
-c	填写用户账户的备注信息
-d -m	参数-m与参数-d连用，可重新指定用户的家目录并自动把旧的数据转移过去
-e	账户的到期时间，格式为YYYY-MM-DD
-g	变更所属用户组
-L	锁定用户禁止其登录系统
-U	解锁用户，允许其登录系统
-s	变更默认终端
-u	修改用户的UID

passwd

passwd命令用于修改用户密码、过期时间、认证信息等，格式为“passwd [选项] [用户名]”

```
[root@localhost ~]# passwd eagle
更改用户 eagle 的密码 。
新的 密码:
重新输入新的 密码:
passwd: 所有的身份验证令牌已经成功更新。
[root@localhost ~]# passwd -l eagle
锁定用户 eagle 的密码 。
passwd: 操作成功
[root@localhost ~]# passwd -S eagle
eagle LK 2018-08-07 0 99999 7 -1 (密码已被锁定。)
[root@localhost ~]# passwd -u eagle
解锁用户 eagle 的密码。
passwd: 操作成功
[root@localhost ~]# passwd -S eagle
eagle PS 2018-08-07 0 99999 7 -1 (密码已设置，使用 SHA512 算法。)
```

passwd

参数	作用
-l	锁定用户，禁止其登录
-u	解除锁定，允许用户登录
--stdin	允许通过标准输入修改用户密码，如echo "NewPassWord" | passwd --stdin Username
-d	使该用户可用空密码登录系统
-e	强制用户在下次登录时修改密码
-S	显示用户的密码是否被锁定，以及密码所采用的加密算法名称

userdel

userdel命令用于删除用户，格式为“userdel [选项] 用户名”

```
[root@localhost ~]# userdel -rf eagle
```

参数	作用
-f	强制删除用户
-r	同时删除用户及用户家目录

文件权限归属

文件权限归属

Linux系统使用了不同的字符来加以区分，常见的字符如下所示。

- -：普通文件。
- d：目录文件。
- l：软链接文件。
- b：块设备文件。
- c：字符设备文件。
- p：管道文件。

```
[root@localhost ~]# ls -l anaconda-ks.cfg
-rw-------. 1 root root 929 5月  18 2016 anaconda-ks.cfg
```

文件权限归属

权限分配	文件所有者			文件所属组			其他用户		
权限项	读	写	执行	读	写	执行	读	写	执行
字符表示	r	w	x	r	w	x	r	w	x
数字表示	4	2	1	4	2	1	4	2	1

文件特殊权限

SUID

让二进制程序的执行者临时拥有所属者的权限（仅对拥有执行权限的二进制程序有效）

```
[root@localhost ~]# ls -l /etc/shadow
----------. 1 root root 644 8月   8 04:25 /etc/shadow
[root@localhost ~]# ls -l /bin/passwd
-rwsr-xr-x. 1 root root 27832 6月  10 2014 /bin/passwd
[root@localhost ~]# chmod u+s anaconda-ks.cfg
[root@localhost ~]# ls -l anaconda-ks.cfg
-rwS------. 1 root root 929 5月  18 2016 anaconda-ks.cfg
```

SGID

SGID主要实现如下两种功能：

- 让执行者临时拥有属组的权限（对拥有执行权限的二进制程序进行设置）；
- 在某个目录中创建的文件自动继承该目录的用户组（只可以对目录进行设置）。

SGID

```
[root@localhost ~]# mkdir test
[root@localhost ~]# ls -ald test
drwxr-xr-x. 2 root root 6 8月   8 19:38 test
[root@localhost ~]# chmod -Rf 777 test/
[root@localhost ~]# chmod -Rf g+s test
[root@localhost ~]# ls -ald test
drwxrwsrwx. 2 root root 6 8月   8 19:38 test
```

```
[root@localhost ~]# su eagle
[eagle@localhost tmp]$ cd test/
[eagle@localhost test]$ echo hello > test.txt
[eagle@localhost test]$ ls -al test
ls: 无法访问test: 没有那个文件或目录
[eagle@localhost test]$ ls -al test.txt
-rw-rw-r--. 1 eagle root 6 8月   8 19:46 test.txt
```

chmod

用来设置文件或目录的权限，格式为“chmod [参数] 权限 文件或目录名称”。

```
[root@localhost ~]# ll test
-rw-r--r-- 1 root root 0 8月  10 17:45 test
[root@localhost ~]# chmod 245 test
[root@localhost ~]# ll test
--w-r--r-x 1 root root 0 8月  10 17:45 test
```

chown

设置文件或目录的所有者和所属组，其格式为“chown [参数] 所有者:所属组 文件或目录名称”。

```
[root@localhost test]# ls -l
总用量 4
-rw-rw-r--. 1 eagle root 6 8月   8 19:46 test.txt
[root@localhost test]# chown root:root test.txt
[root@localhost test]# ls -l
总用量 4
-rw-rw-r--. 1 root root 6 8月   8 19:46 test.txt
```

SBIT

当对某个目录设置了SBIT粘滞位权限后，那么该目录中的文件就只能被其所有者执行删除操作。

当目录被设置SBIT特殊权限位后，文件的其他人权限部分的x执行权限就会被替换成t或者T，原本有x执行权限则会写成t，原本没有x执行权限则会被写成T。

```
[root@localhost tmp]# cd test
[root@localhost test]# chmod -R o+t test.txt
[root@localhost test]# ls -l
总用量 4
-rwxrwxrwt. 1 root root 6 8月   8 19:46 test.txt
[root@localhost test]# su eagle
[eagle@localhost test]$ rm -f test.txt
rm: 无法删除"test.txt": 不允许的操作
```

文件的隐藏属性

chattr

chattr命令用于设置文件的隐藏权限，格式为“chattr [参数] 文件”。

```
[root@localhost test]# echo hello > test.txt
[root@localhost test]# chattr +a test.txt
[root@localhost test]# rm -f test.txt
rm: 无法删除"test.txt": 不允许的操作
[root@localhost test]# ls -l
总用量 4
-rw-r--r--. 1 root root 6 8月   8 20:10 test.txt
```

参数	作用
i	无法对文件进行修改；若对目录设置了该参数，则仅能修改其中的子文件内容而不能新建或删除文件
a	仅允许补充（追加）内容，无法覆盖/删除内容（Append Only）
S	文件内容在变更后立即同步到硬盘（sync）
s	彻底从硬盘中删除，不可恢复（用0填充原文件所在硬盘区域）
b	不再修改文件或目录的存取时间
D	检查压缩文件中的错误
d	使用dump命令备份时忽略本文件/目录
c	默认将文件或目录进行压缩
u	当删除该文件后依然保留其在硬盘中的数据，方便日后恢复
t	让文件系统支持尾部合并（tail-merging）
X	可以直接访问压缩文件中的内容

lsattr

lsattr命令用于显示文件的隐藏权限，格式为“lsattr [参数] 文件”。

```
[root@localhost test]# chattr +a test.txt
[root@localhost test]# lsattr test.txt
-----a---------- test.txt
[root@localhost test]# chattr -a test.txt
[root@localhost test]# lsattr test.txt
---------------- test.txt
```

文件的扩展属性

setfacl

setfacl命令用于管理文件的ACL规则，格式为“setfacl [参数] 文件名称”。

```
[root@localhost eagle]# su eagle
[eagle@localhost ~]$ cd /root
bash: cd: /root: 权限不够
[eagle@localhost ~]$ su root
密码:
[root@localhost eagle]# setfacl -Rm u:eagle:rwx /root
[root@localhost eagle]# su eagle
[eagle@localhost ~]$ cd /root
[eagle@localhost root]$ ls
test.txt
[eagle@localhost root]$ cat test.txt
hello
[root@localhost ~]# ls -l test.txt
-rw-rwxr--+ 1 root root 6 8月   8 20:19 test.txt
```

getfacl

getfacl命令用于显示文件上设置的ACL信息，格式为“getfacl 文件名称”。

```
[root@localhost ~]# getfacl /root
getfacl: Removing leading '/' from absolute path names
# file: root
# owner: root
# group: root
user::r-x
user:eagle:rwx
group::r-x
mask::rwx
other::---
```

su

su命令可以解决切换用户身份的需求，使得当前用户在不退出登录的情况下，顺畅地切换到其他用户

su命令与用户名之间有一个减号（-），这意味着完全切换到新的用户，即把环境变量信息也变更为新用户的相应信息，而不是保留原始的信息。强烈建议在切换用户身份时添加这个减号（-）。

```
[root@localhost ~]# su - eagle
上一次登录：三 8月  8 20:19:59 CST 2018pts/0 上
[eagle@localhost ~]$ id
uid=1000(eagle) gid=1000(eagle) 组=1000(eagle) 环境
=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[eagle@localhost ~]$ su root
密码：
```

sudo

sudo命令用于给普通用户提供额外的权限来完成原本root管理员才能完成的任务，格式为“sudo [参数] 命令名称”。

sudo命令具有如下功能：

- 限制用户执行指定的命令：
- 记录用户执行的每一条命令；
- 配置文件（/etc/sudoers）提供集中的用户管理、权限与主机等参数；
- 验证密码的后5分钟内（默认值）无须再让用户再次验证密码。

参数	作用
-h	列出帮助信息
-l	列出当前用户可执行的命令
-u用户名或UID值	以指定的用户身份执行命令
-k	清空密码的有效时间，下次执行sudo时需要再次进行密码验证
-b	在后台执行指定的命令
-p	更改询问密码的提示语

使用visudo命令配置sudo命令的配置文件

```
[root@localhost ~]# visudo
 96 ##
 97 ## Allow root to run any commands anywhere
 98 root ALL=(ALL) ALL
 99 eagle ALL=(ALL) ALL
```

```
[eagle@localhost ~]$ sudo -l
[sudo] password for eagle:
对不起，用户 eagle 不能在 localhost 上运行 sudo。
[eagle@localhost ~]$ su
密码：
[root@localhost eagle]# visudo
[root@localhost eagle]# su eagle
[eagle@localhost ~]$ sudo -l
[sudo] password for eagle:
匹配此主机上 eagle 的默认条目：
    requiretty, !visiblepw, always_set_home, env_reset, env_keep="COLORS DISPLAY H
OSTNAME HISTSIZE INPUTRC KDEDIR LS_COLORS",
    env_keep+="MAIL PS1 PS2 QTDIR USERNAME LANG LC_ADDRESS LC_CTYPE", env_k
eep+="LC_COLLATE LC_IDENTIFICATION LC_MEASUREMENT
    LC_MESSAGES", env_keep+="LC_MONETARY LC_NAME LC_NUMERIC LC_PAPER LC_
TELEPHONE", env_keep+="LC_TIME LC_ALL LANGUAGE LINGUAS
    _XKB_CHARSET XAUTHORITY", secure_path=/sbin\:/bin\:/usr/sbin\:/usr/bin
用户 eagle 可以在该主机上运行以下命令：
    (ALL) ALL
[eagle@localhost ~]$
```

ALL权限太大了，如果只是想赋予某种命令的权限
可以取消每次都要输入密码

```
[eagle@localhost ~]$ exit
logout
[root@localhost ~]# whereis poweroff
poweroff: /usr/sbin/poweroff /usr/share/man/man8/poweroff.8.gz
[root@linuxprobe ~]# visudo
……省略部分文件内容……
 96 ##
 97 ## Allow root to run any commands anywhere
 98 root ALL=(ALL) ALL
 99 eagle ALL=NOPASSWD: /usr/sbin/poweroff
……省略部分文件内容……
[eagle@localhost ~]$ sudo cat /root/test.txt
hello
```