特殊权限和acl

特殊权限

SUID:

运行某程序时，相应进程的属主是程序文件自身的属主，而不是
用户本身了,只对二进制程序有效,执行者对于程序需要有x权限
例子：passwd命令 需要在/etc/shadow中写入密码
ls -l /bin/cat
ls -l /etc/shadow
chmod u+s file （如果本身具有x，为s，否则为S）

SGID：

对于文件：运行某程序时，相应进程的属组是程序文件自身的属组，而不是用户本身的基本组

chmod g+s file

例子：locate命令 需要访问/var/lib/mlocate/mlocate.db文件

对于目录：

用户对此目录有rx权限可以进入目录
用户进入此目录后，有效用户组会变成该目录的用户组
若用户在此目录有w权限，则用户创建的文件用户组与该目录用户组相同

例如：

一个团队想在linux某个目录下协同工作来做一个项目，那么每个团队成员都得对这个目录下的所有文件具有rwx权限。

于是我们首先新建一个用户组，再新建几个账号，每个账号的用户组都加入刚才新建的那个用户组。

再新建工作目录，权限设为770，把目录的用户组加上上一步新建的用户组。

到这里为止，我们思考下会有什么问题?

现在账号A新建一个文件，新建文件的拥有者和用户组都会是A！重要的是其他用户都无法访问这个文件!

所以我们需要给这个目录加入SGID权限，之后任意一个用户创建的文件，文件用户组都会是这个目录的用户组。万事OK!

develop team , hadoop hbase hive

/tmp/project　三个用户可以编辑彼此之间创建的文件

#useradd hadoop hbase hive

#mkdir /tmp/project

#groupadd developteam

#chown -R :developteam /tmp/project

#chmod -R 770 /tmp/project

#usermod -a -G developteam hadoop

#chmod g+s /tmp/project

Sticky (BIT):

只针对目录有效,当用户对目录拥有wx权限时，用户在该目录创建的文件或目录，只有自己与root才可以删除。

在一个公共目录，每个人都可以创建文件，删除自己的文件，

但是不能删除别人的文件（冒险位，粘贴位）

例子：chmod o+t dir

SUID是4　SGID是2　SBIT是1

chmod 4755 filename

第一个7代表的就是这三个特殊命令，后面的755是普通权限。上面的命令把filename这个文件加入了SUID权限

文件系统访问控制列表

FACL:filesystem access control list

利用文件的扩展属性，保存了额外的访问控制权限

getfacl 查看

setfacl 设置

语法：setfacl [-bkRd] [-m|-x acl 参数] 目标文件名

选项与参数：

-m:设置后续的acl参数，不可与-x一起使用

-x: 删除后续的acl参数，不可与-m一起使用

-b:删除所有的acl参数

-k:删除默认的acl参数

-R:递归设置acl参数

-d:设置默认acl参数，只对目录有效

setfacl -m m:rw inittab

-m设定，可以设定到用户或者是组上

u:uid:perm

g:gid:perm

例子:

#mkdir /backup

#cd /backup

#cp /etc/inittab ./

#getfacl inittab

#setfacl -m u:redhat:rw inittab

owner>facl,user> group > facl group>

所有权限都不能超过mask的权限

setfacl -m m:rwx [filename or directory_name]

-x取消

setfacl -x u:uid file_name

为目录设定默认访问控制列表:

d:u:uid:perm file_name

mount -o acl /dev/myvg1/mylv1 /mnt

dumpe2fs -h /dev/myvg1/mylv1(查看是否支持ACL)

tune2fs -o

例子: 授权一个用户读权限

setfacl -m u:lisa:r file

Revoking write access from all groups and all named users (using the effective rights mask)

撤销所有的组和用户的写权限（使用有效的正确mask）

```
setfacl -m m::rx file
```

Removing a named group entry from a file’ s ACL

移除一个组的ACL权限

```
setfacl -x g:staff file
```

Copying the ACL of one file to another

复制一个文件的ACL到另一个文件

```
getfacl file1 | setfacl --set-file=- file2
```

Copying the access ACL into the Default ACL

复制访问的目录的ACL作为目录的默认ACL

```
getfacl --access dir | setfacl -d -M- dir
```