第7讲 Linux文件系统安全配置

本讲主要内容

- 1 Linux 本地文件系统安全设置
- 2 Linux 服务器文件系统运维

1 Linux 本地文件系统安全设置

在本节，我们主要介绍：

- 使用 ls 命令查看文件和目录详情
- 使用chmod命令改变文件许可权限
- 使用访问控制列表
- 使用mv命令处理文件
- 安装配置基本LDAP服务器

1.1 使用 ls 命令查看文件和目录详情

这部分内容较为简单，大部分同学都应该了解。

命令格式：

```
ls [参数 -alrtAFR] [name...]
```

参数说明：

- -a 显示所有文件及目录 (ls内定将文件名或目录名称开头为"."的视为隐藏档，不会列出)
- -l 除文件名称外，亦将文件型态、权限、拥有者、文件大小等资讯详细列出
- -r 将文件以相反次序显示(原定依英文字母次序)
- -t 将文件依建立时间之先后次序列出
- -A 同 -a ，但不列出 "." (目前目录) 及 ".." (父目录)
- -F 在列出的文件名称后加一符号，例如：
  - 可执行文件后会加 "*"
  - 目录则加 "/"
- -R 若目录下有文件，则以下之文件亦皆依序列出

1.2 使用chmod命令改变文件许可权限

Linux/Unix 的文件访问权限分为三级 :

- 文件拥有者（owner）
- owner所在组
- 其他人

利用 chmod 可以控制文件的访问权限。

命令格式：

```
chmod [-cfvR] [--help] [--version] mode file...
```

参数说明:

mode : 权限设定字串，格式如下 :

[ugoa...][[+-=][rwxX]...][,...]

- u 表示该文件的拥有者;
- g 表示与该文件的拥有者属于同一个群体(group)者;
- o 表示其他以外的人;
- a 表示这三者皆是。
- "+" 表示增加权限;
- "-" 表示取消权限;
- "=" 表示唯一设定权限。
- r 表示可读取
- w 表示可写入
- x 表示可执行
- X 表示只有当该文件是个子目录或者该文件已经被设定过为可执行。

其他参数说明：

- -c : 若该文件权限确实已经更改，才显示其更改动作
- -f : 若该文件权限无法被更改也不要显示错误讯息
- -v : 显示权限变更的详细资料
- -R : 对目前目录下的所有文件与子目录进行相同的权限变更(即以递回的方式逐个变更)
- --help : 显示辅助说明
- --version : 显示版本

用ls命令所得到文件访问权限的表示法的格式是类似这样的： `-rwxr-xr-x` 。

这种表示法一共有十位：9 8 7 6 5 4 3 2 1 0 ：

- 第9位表示文件类型,可以为p、d、l、s、c、b和-
  - p表示命名管道文件
  - d表示目录文件
  - l表示符号连接文件
  - -表示普通文件
  - s表示socket文件
  - c表示字符设备文件
  - b表示块设备文件
- 第8-6位表示文件所有者的权限，其形式为rwx
  - r表示可读，可以读出文件的内容；
  - w表示可写，可以修改文件的内容；
  - x表示可执行，可运行这个程序；
  - 没有权限的位置用-表示
- 第5-3位表示同组用户的权限，其形式为rwx
- 第2-0位表示其他用户的权限，其形式为rwx

人们也常以 `chmod abc 文件名或目录名` 这种模式形式给文件或目录赋予权限。其中a,b,c各为一个数字，分别表示User、Group、及Other的权限。

数字的确定方法：

- 读取权限r=4
- 写权限w=2
- 执行权限x=1
- 若要rwx属性则4+2+1=7；
- 若要rw-属性则4+2=6；
- 若要r-x属性则4+1=5。

所以，命令 `chmod 755 test.txt` 的含义是将test.txt的权限设置为：u=rwx，g=r-x，o=r-x。

文件除了读写执行权限外，还有几个特殊权限：

- suid，文件执行身份为所属目录身份，而非执行文件的用户身份。
  - 设置suid属性命令 `chmod u+s text`
  - suid数字表示为4
- sgid，以文件所属组身份执行，如果是目录则目录中创建的任意新文件的所属组与该目录的所属组相同。
  - 设置命令： `chmod g+s test`
  - 数字表示为2

- sticky，对目录拥有写入权限的用户仅可以删除其拥有的文件，不能删除其它用户所拥有的文件。
  - 设置命令： `chmod o+t test`
  - 数字表示为1

1.3 使用访问控制列表

使用chmod命令对文件的基本访问权限进行设置是不够的，还需要配置ACL。

基础ACL通过两条命令管理：

- setfacl用于增加或者修改ACL;
- getfacl用于显示分配完的ACL。

ACL能够为特定文件定义访问权限。

使用ACL之前需要验证是否启动了ACL。验证命令如下：

```
getfacl <文件名>
```

如果显示没有安装，可执行下列命令安装：

```
sudo apt install acl
```

1.4 使用mv命令处理文件

当希望将一个文件从一个目录移动到另一个目录，同时不建立副本，则可使用mv命令或move命令。

命令语法：

```
mv [options] source dest
mv [options] source... directory
```

参数说明：

- -i: 若指定目录已有同名文件，则先询问是否覆盖旧文件;
- -f: 在mv操作要覆盖某已有的目标文件时不给任何指示;

1.5 安装配置基本LDAP服务器

LDAP是一种对文件或目录访问权限集中管理的协议。主要用于集中授权。

LDAP类似于一个数据库，但包含了很多描述性、基于属性的信息。

2 Linux 服务器文件系统运维

在本节，我们主要讲解以下内容：

- 锁定系统重要文件
- 文件系统权限检查
- /tmp, /var/tmp，/dev/shm 安全设定

2.1 锁定系统重要文件

系统运维人员可能常会遇到通过root用户都不能修改或者删除某个文件的情况。

产生这种情况的原因是该文件被锁定了。

在Linux下锁定文件的命令是 `chattr` ，通过这个命令可以修改的文件系统下的属性：

- ext2
- ext3
- ext4

在修改属性前，一般要使用 `lsattr` 命令查看文件属性，格式如下：

```
lsattr [-adlRvV] 文件或目录
```

参数说明：

- -a，列出目录中的所有文件，包括以“.”开头的文件；
- -d，显示指定目录的属性；
- -R，以递归的方式列出目录下所有文件、子目录、属性；
- -v，显示文件或目录版本。

修改属性的命令如下：

```
chattr [选项] [-v version] [mode] 文件或目录
```

选项：

- -R，递归操作
- -V，显示详情
- -f，强制，忽略错误

mode部分用来控制文件的属性，说明：

- “+” 表示给文件或目录添加属性；

- “-” 表示移除文件或目录拥有的某些属性；
- “=” 表示给文件或目录设定一些属性；
- a：即Append Only，系统只允许在这个文件之后追加数据，不允许任何进程覆盖或截断这个文件。如果目录具有这个属性，系统将只允许在这个目录下建立和修改文件，而不允许删除任何文件。
- b：不更新文件或目录的最后存取时间。
- c：即compress，设定文件是否经压缩后再存储。读取时需要经过自动解压操作。
- i：即Immutable，系统不允许对这个文件进行任何的修改。如果目录具有这个属性，那么任何的进程只能修改目录之下的文件，不允许建立和删除文件。
- s：彻底删除文件，不可恢复，因为是从磁盘上删除，然后用0填充文件所在区域。
- u：当一个应用程序请求删除这个文件，系统会保留其数据块以便以后能够恢复删除这个文件，用来防止意外删除文件或目录。
- d：当dump程序执行时，该文件或目录不会被dump备份。
- A：即Atime，告诉系统不要修改对这个文件的最后访问时间。
- S：即Sync，一旦应用程序对这个文件执行了写操作，使系统立刻把修改的结果写到磁盘，可以有效地避免数据流失。
- D: 检查压缩文件中的错误。
- t:文件系统支持尾部合并（tail-merging）。
- X：可以直接访问压缩文件的内容。

特权用户可以对权限内文件进行增删改，为了防止误删等误操作，锁定或限制文件读写属性是一道安全防线。

所以管理员需要对重要目录和文件进行锁定保护，可以使用 `chattr -R +i` 命令。

常见需要禁止修改的目录、文件有：

- chattr -R +i /bin /boot /lib /sbin
- chattr -R +i /usr/bin /usr/include /usr/lib /usr/sbin
- chattr +i /etc/passwd
- chattr +i /etc/shadow
- chattr +i /etc/hosts
- chattr +i /etc/resolv.conf
- chattr +i /etc/fstab
- chattr +i /etc/sudoers

需要设置为仅可增加内容的文件或目录有：

- chattr +a /var/log/messages
- chattr +a /var/log/wtmp

为文件加锁，能够提高服务器安全性，也会带来一些问题和不便。

- 在软件安装、升级时，需要关闭一些目录（例如/etc下文件或目录）的immutable属性和appendonly属性。
- 日志文件设置appendonly属性会阻止日志内容轮替。
- 不适合使用chattr保护的文件或目录有：
  - 根目录/
  - /dev/
  - /tmp/
  - /var/

2.2 文件权限检查和修改

不正确的权限设置直接威胁系统的安全，因此运维人员应该能及时发现这些不正确的权限设置，并立刻修正，防患于未然。

下面列举几种查找系统不安全权限的方法：

查找系统中任何用户都有写权限的文件或目录

- 查找文件

```
find / -type f -perm -2 -o -perm -20 | xargs ls -al
```

- 查找目录

```
find / -type d -perm -2 -o -20 |xargs ls -ld
```

说明：

- -type b/d/c/p/l/f 可用于查找块设备、目录、字符设备、管道、符号链接、普通文件。
- perm ，指按执行权限来查找。
  - linux“读写执行无”四类权限，R，W，X，-对应数字4，2，1，0，用3位数字分别表示所有者、所在组、其它用户对文件的权限。-2表示所有人有“写”权限，-20表示所在组有写权限。
- -o表示或者的意思；-a表示而且的意思；-not 表示相反的意思。
- |xargs 表示后续要执行命令

查找系统中含有“s”位的程序

含“s”位权限的程序对系统安全威胁很大，通过查找系统中所有具有“s”位权限的程序，可以把某些不必要的“s”位权限的程序去掉，防止用户滥用权限或提升权限。

```
find / -type f -perm -4000 -o -perm -2000 -print | xargs ls -al
```

上述命令中的“-2000”，表示sgid权限，“-4000”，表示suid权限。

检查系统中所有suid及sgid文件

```
find / -user root -perm -2000 -print -exec md5sum {} \;
```

```
find / -user root -perm -4000 -print -exec md5sum {} \;
```

把检查结果存放到文件里，有助于后期对比排查。

说明：对于find查询到满足前序条件的文件或目录，执行md5sum，{} \；表示按查找结果一项一下进行。注意：{}和\中间有空格，后面有分号。

检查系统中没有属主的文件

没有属主的“孤儿”文件比较危险，常被黑客所利用，所以可以在找到后考虑删除或修改属主。

```
find / -nouser -o -nogroup
```

find命令说明

find命令格式：

```
find <指定目录> [选项] [找到后动作]
```

或：

```
find path -option [-print ] [ -exec -ok command ] {} \
```

- path：要查找的目录路径。
- print：表示将结果输出到标准输出。
- exec：对匹配的文件执行该参数所给出的shell命令。
  - 形式为command {} ;，注意{}与;之间有空格
- ok：与exec作用相同，区别在于，在执行命令之前，都会给出提示，让用户确认是否执行
- |xargs 与exec作用相同，起承接作用。区别在于 |xargs 主要用于承接删除操作，而 -exec 都可用如复制、移动、重命名等。
- options ：表示查找方式。

常用选项：

按时间查找

- -amin<分钟>：查找在指定时间曾被存取过的文件或目录，单位以分钟计算；
- -anewer<参考文件或目录>：查找其存取时间较指定文件或目录的存取时间更接近现在的文件或目录；
- -atime<24小时数>：查找在指定时间曾被存取过的文件或目录，单位以24小时计算；
- -cmin<分钟>：查找在指定时间之时被更改过的文件或目录；

- -cnewer<参考文件或目录>查找其更改时间较指定文件或目录的更改时间更接近现在的文件或目录；
- -ctime<24小时数>：查找在指定时间之时被更改的文件或目录，单位以24小时计算；
- -daystart：从本日开始计算时间；
- -mmin<分钟>：查找在指定时间曾被更改过的文件或目录，单位以分钟计算；
- -mtime<24小时数>：查找在指定时间曾被更改过的文件或目录，单位以24小时计算；
- -used<日数>：查找文件或目录被更改之后在指定时间曾被存取过的文件或目录，单位以日计算；

按名字查找

- -name<范本样式>：指定字符串作为寻找文件或目录的范本样式；
- -ilname<范本样式>：此参数的效果和指定“-lname”参数类似，但忽略字符大小写的差别；
- -iname<范本样式>：此参数的效果和指定“-name”参数类似，但忽略字符大小写的差别；
- -prune：不寻找字符串作为寻找文件或目录的范本样式;
- -regex<范本样式>：指定字符串作为寻找文件或目录的范本样式；

按文件目录查找

- -depth：从指定目录下最深层的子目录开始查找；
- -path<范本样式>：指定字符串作为寻找目录的范本样式；
- -ipath<范本样式>：此参数的效果和指定“-path”参数类似，但忽略字符大小写的差别；
- -maxdepth<目录层级>：设置最大目录层级；
- -mindepth<目录层级>：设置最小目录层级；

按文件大小查找

- -expty：寻找文件大小为0 Byte的文件，或目录下没有任何子目录或文件的空目录；
- -size<文件大小>：查找符合指定的文件大小的文件；

按用户、组查找

- -gid<群组识别码>：查找符合指定之群组识别码的文件或目录；
- -group<群组名称>：查找符合指定之群组名称的文件或目录；
- -uid<用户识别码>：查找符合指定的用户识别码的文件或目录；
- -user<拥有者名称>：查找符和指定的拥有者名称的文件或目录；

按权限查找

- -perm<权限数值>：查找符合指定的权限数值的文件或目录；

按文件类型查找

- -type<文件类型>：只寻找符合指定的文件类型的文件；

- -xtype<文件类型>：此参数的效果和指定“-type”参数类似，差别在于它针对符号连接检查

后续操作

- -exec<执行指令>：假设find指令的回传值为True，就执行该指令；
- -false：将find指令的回传值皆设为False；
- -fls<列表文件>：此参数的效果和指定“-ls”参数类似，但会把结果保存为指定的列表文件；
- -ok<执行指令>：此参数的效果和指定“-exec”类似，但在执行指令之前会先询问用户，若回答“y”或“Y”，则放弃执行命令；

查找结果的输出

- -fprint<列表文件>：此参数的效果和指定“-print”参数类似，但会把结果保存成指定的列表文件；
- -fprint0<列表文件>：此参数的效果和指定“-print0”参数类似，但会把结果保存成指定的列表文件；
- -fprintf<列表文件><输出格式>：此参数的效果和指定“-printf”参数类似，但会把结果保存成指定的列表文件；
- -print：假设find指令的回传值为Ture，就将文件或目录名称列出到标准输出。格式为每列一个名称，每个名称前皆有“./”字符串；
- -print0：假设find指令的回传值为Ture，就将文件或目录名称列出到标准输出。格式为全部的名称皆在同一行；
- -printf<输出格式>：假设find指令的回传值为Ture，就将文件或目录名称列出到标准输出。格式可以自行指定

其它

- -follow：排除符号连接；
- -fstype<文件系统类型>：只寻找该文件系统类型下的文件或目录；
- -inum<inode编号>：查找符合指定的inode编号的文件或目录；
- -iregex<范本样式>：此参数的效果和指定“-regexe”参数类似，但忽略字符大小写的差别；
- -links<连接数目>：查找符合指定的硬连接数目的文件或目录；
- -help或——help：在线帮助；
- -true：将find指令的回传值皆设为True；
- -version或——version：显示版本信息；
- -xdev：将范围局限在先行的文件系统中；

2.3 /tmp、/var/tmp、/dev/shm 安全设定

在Linux中，用来存放临时文件主要有两个目录或分区，分别是：

- /tmp
- /var/tmp

它们的特点是：所有用户可以读写、可以执行，这是一个安全隐患。

攻击者可以将恶意代码存放于这些目录中。我们需要对这写临时目录进行特殊设定。

/dev/shm 是Linux一个共享内存设备，Linux启动时会默认加载到/dev/shm，被加载的/dev/shm使用的是tmpfs文件系统。

而tmpfs是一个内存文件系统，存储到tmpfs文件系统的数据会完全驻留在RAM中，这样通过/dev/shm就可以直接操作系统内存，这十分危险。

对/tmp的安全设置

首先要弄清/tmp是一个独立磁盘分区，还是一个根目录下的文件夹。

若/tmp是一个独立磁盘分区，则：

- 修改/etc/fstab文件中/tmp分区对应的挂载属性，加上三个选项即可：nosuid、noexec、nodev。
- 修改后的/tmp分区挂载属性类似如下： `LABEL=/tmp /tmp ext3 rw,nosuid,noexec,nodev 0 0`
- 说明：nosuid、noexec、nodev选项表示不允许任何suid程序，在这个分区内不能执行任何脚本程序，以及不存在设备文件。
- 挂载属性设置完毕后，重新挂载/tmp分区（使用mount 命令），保证设置生效。
- 命令使用格式： `mount [-fnrsvw] [-t vfstype] [-o options] device dir`

若/tmp是一个根目录下的文件夹，设置将比较复杂。具体如下：

- 创建一个loopback文件系统，利用Linux内核的loopback特性将文件系统挂载到/tmp下；
- 然后，在挂载时指定限制加载选项即可。

例如：

(1)先运行下列命令

```
dd if=/dev/zero of=/dev/tmpfs bs=1M count=10000
mke2fs -j /dev/tmpfs
cp -av /tmp /tmp.old
mount -o loop, noexec, nosuid, rw /dev/tmpfs  /tmp
chmod 1777 /tmp
mv -f /tmp.old/* /tmp/
rm -rf /tmp.old
```

(2)编辑/etc/fstab，添加以下内容，以便系统在启动时自动加载loopback文件系统

```
/dev/tmpfs  /tmp  ext3  loop, nosuid, noexec, rw 0 0
```

为了验证挂载时指定限制加载选项是否生效，可以在/tmp分区创建一个shell文件，具体操作如下：

```
ls -al | grep shell

pwd

./shell-test.sh
```

可以看到，虽然文件有可执行属性，但/tmp文件夹下已经不可以执行任何文件了。

对/var/tmp的安全设置

- 若/var/tmp是独立分区，安装/tmp的设置方法是修改/etc/fstab文件即可；
- 若/var/tmp是/var下的一个目录，那么可以将/var/tmp目录下所有数据移动到/tmp分区下，然后在/var下做一个指向/tmp的软连接即可。例如：
  - `mv /var/tmp/* /tmp`
  - `ln -s /tmp /var/tmp`

对/dev/shm的安全设置

由于/dev/shm是一个共享内存设备，因此也可以通过修改/etc/fstab文件设置而事先。

默认情况下，/dev/shm通过defaults选项来加载，这样保证其安全性是不够的，需要修改/dev/shm的挂载属性，操作如下：

```
tmpfs  /dev/shm  tmpfs  defaults, nosuid, noexec, rw 0 0
```

通过这种方式，限制了任何suid程序，同时也限制了/dev/shm的可执行权限，系统安全性进一步提升。

补充说明：mount命令与/etc/fstab文件

在linux中常常用mount命令把硬盘分区或者光盘挂载到文件系统中。/etc/fstab就是在开机引导的时候自动挂载到linux的文件系统。

mount命令

使用格式：

```
mount [-fnrsvw] [-t vfstype] [-o options] device dir
```

device：指明要挂载的设备，可以是：

- 设备文件：例如/dev/sda5

- 卷标：-L 'LABEL', 例如 -L 'MYDATA'
- UUID, -U 'UUID'：例如 -U '0c50523c-43f1-45e7-85c0-a126711d406e'
- 伪文件系统名称：proc, sysfs, devtmpfs, configfs

dir：挂载点，需要预先建立，如果内容不为空则原文件被隐藏。

常用命令选项：

- -t vsftype：指定要挂载的设备上的文件系统类型；
- -r: readonly，只读挂载；
- -w: read and write, 读写挂载；
- -n: 不更新/etc/mtab；
- -a：自动挂载所有支持自动挂载的设备；(定义在了/etc/fstab文件中，且挂载选项中有“自动挂载”功能)
- -L 'LABEL': 以卷标指定挂载设备；
- -U 'UUID': 以UUID指定要挂载的设备；
- -B, --bind: 绑定目录到另一个目录上；

/etc/fstab

/etc/fstab就是在开机引导的时候自动挂载到linux的文件系统。使用下列命令可查看已挂载的内容：

```
cat /proc/mounts
```

在linux中/etc/fstab的数据项如下所示：

```
/dev/device mountpoint type rules 0 order
```

说明：

- /dev/device就是需要挂载的设备；
- mountpoint 就是挂载点。/、 /home、 /swap、/dev、都是系统安装时分区的默认挂载点。
- type 是指文件系统类形。
  - ext2
  - ext3
  - ext4
- rules 是指挂载时的规则。下面列举几个常用的：
  - auto 开机自动挂载
  - default 按照大多数永久文件系统的缺省值设置挂载定义
  - noauto 开机不自动挂载
  - nouser 只有超级用户可以挂载
  - ro 按只读权限挂载
  - rw 按可读可写权限挂载
  - user 任何用户都可以挂载

- 0 是指dump(系统备份工具）。
  - 这一项为0，就表示从不备份。
  - 如果上次用dump备份，将显示备份至今的天数。
- order 指fsck（启动时fsck检查的顺序）。
  - 为0就表示不检查
  - /分区永远都是1
  - 其它的分区只能从2开始
  - 当数字相同就同时检查（但不能有两1）。

修改了/etc/fstab后，一定要重新引导系统才会有效。