**实验2 UNIX系统中权限的管理**

**一、chroot的配置和root 的capability使用**

1. 利用chroot工具来虚拟化管理

1) 实现bash或ps的配置使用；

2)利用chroot实现SSH服务或FTP服务的虚拟化隔离；

3)chroot后如何降低权限，利用实验一中编制的程序检查权限的合理性；

4)在chroot之前没有采用cd xx目录，会对系统有何影响，编制程序分析其影响。

1. 修改系统内核，配置capability的能力位，实现几种能力位的设置可验证。

样例：redhat 2.4下的能力

1）函数说明

getcap可以获得程序文件所具有的能力(CAP).

getpcaps可以获得进程所具有的能力(CAP).

setcap可以设置程序文件的能力(CAP).

注:

1)cap\_chown=eip是将chown的能力以cap\_effective(e),cap\_inheritable(i),cap\_permitted(p)三种位图的方式授权给相关的程序文件.

2)如果改变文件名,则能力保留到新文件.

3)用setcap -r /bin/chown可以删除掉文件的能力.

4)重新用setcap授权将覆盖之前的能力.

能力位：CAP\_SYS\_NICE 23(允许提升优先级,设置其它进程的优先级)

{

对于普通用户程序的NICE优先级,不能超过ulimit对它的限制,如下:

nice -n -5 ls

nice: cannot set niceness: Permission denied

而CAP\_SYS\_NICE可以帮助普通用户设置一个想要的一个任意优先级.

setcap cap\_sys\_nice=eip /usr/bin/nice

切换到普通用户,指定优先级,如下:

nice -n -5 ls

log mnt mount.c mounttest pacct psacct psacct.c reboot1 reboot1.c test

[root@localhost zy]# setcap cap\_sys\_nice=eip /home/tttt/test

[root@localhost zy]# getcap /home/tttt/test

/home/tttt/test = cap\_sys\_nice+eip

}

1. 实验要求：

　a)实现３种能力位的授权和查看，并分析授权前和授权后的差异；

b)系统启动时关闭某能力位，对系统的应用和安全性有何影响，以具体能力位为例说明

c)比较能力位和chroot管理权限的差异

1. **passwd程序功能的仿制实现**

1、passwd程序功能描述

在Linux中，passwd程序是可信任的，修改存储经过加密的密码的影子密码文件（/etc/shadow），passwd程序执行它自己内部的安全策略，允许普通用户修改属于他们自己的密码，同时允许root修改所有密码。为了执行这个受信任的作业，passwd程序需要有移动和重新创建shadow文件的能力，在标准Linux中，它有这个特权，因为passwd程序可执行文件在执行时被加上了setuid位，它作为root用户（它能访问所有文件）允许，然而，许多程序都可以作为root允许（实际上，所有程序都有可能作为root允许）。这就意味着任何程序（当以root身份运行时）都有可能能够修改shadow文件。

2、实验要求

自己编制文件和程序，仿制passwd程序修改/etc/shadow的功能，包括：

1. 自己设置一个类/etc/shadow文件aaa，该文件中约定好内容格式，和读取该文件的程序相配合，文件中包括超级用户及其内容、普通用户及其内容
2. 编制程序使得：Root用户能够读取和修改aaa文件中所有用户的内容

普通用户仅能够读取和修改aaa文件中属于自己用户的内容

1. 普通用户能以root身份执行所编制的类passwd程序

**三、编制实验报告，回答上面问题，给出源代码、分析过程和实验结果**