

## 实验内容

- SetUID与权能
- 基于PAM的用户权能分配
- RBAC访问控制

- 1.解释 “passwd” , “sudo” , “ping” 等命令为什么需要setuid位, 去掉s位试运行, 添加权能试运行。
- 2.指出每个权能对应的系统调用, 简要解释功能
- 3.查找你Linux发行版系统(Ubuntu/centos等)中所有设置了setuid位的程序, 指出其应该有的权能
- 4.实现一个程序其满足以下的功能:
  - (1)能够永久的删除其子进程的某个权能。
  - (2)能暂时性的删除其子进程的某个权能。
  - (3)能让上面被暂时性删除的权能重新获得。

# 基于PAM的用户权能分配

1. 指出每个权能对应的系统调用，简要解释功能
2. 基于PAM用户权限设置系统
  - (1) 在某用户**登录**时，规定其只具有某几种权能。
  - (2) 例如，用户A登录，其只具有修改网络相关的权能。
  - (3) Hint：比如，按照权能execve变换规则，根据用户名，登陆前设置cap\_net\_raw，然后设置相应的ping程序的文件权能
3. 参考资料：
  - (1) man capability: 介绍权能
  - (2) libcap2, libcap-dev : 操纵权能的工具和库
    - (1) apt-cache search libcap
  - (3) PAM手册: 一个劫持登录程序的接口

1.实现一个LSM (Linux Security Module) 安全模块, 使得linux具备简单RBAC安全功能。

1.用户可以承担角色; 角色对应权限

2. 功能点:

(1)用户、客体、操作 (Linux系统中已具备, 简单起见, 可以只针对部分操作)

(2)权限: 自定义

(1)或采用Linux Capabilities

(2)或仅针对文件/网络/...访问的权限

(3)或简单输出一句话

(3)角色: 自定义

(1)角色创建、删除、更新用户等

(4)开关: 开的时候该安全功能发挥作用, 关闭时不发挥作用

# RBAC访问控制-实现例子

- 1.先定义好策略（策略可以存放在文件中，每次需要判断时内核访问该文件；或者采用虚拟文件系统，直接写入内核）
  - 1.用户与角色：1-1关系
  - 2.角色与权限：1-n关系
- 2.粗粒度实现
  - 1.某角色可以进行文件删除操作，那么承担该角色的用户就可以执行文件删除操作，就可以在文件删除对应的钩子函数中作判断。。这种实现只需要考虑钩子函数即可。
- 3.细粒度的实现：
  - 1.某角色可以执行带有XX标签的文件，...。这种实现就还需要考虑LSM的客体域了
- 4.对于角色的维护可以写一个用户层程序也可以不写