确保未经授权的用户不能存取某些文件

文件系统的安全性

我们只考虑如何确保未经授权的用户，不能存取某些文件

文件保护机制

- 用于提供安全性、特定的操作系统机制
- 对拥有权限的用户，应该让其进行相应操作，否则，应禁止
- 防止其他用户冒充对文件进行操作

实现：
* 用户身份验证
* 访问控制

用户是谁？
用户拥有什么？
用户知道什么？

虹膜等等，来确保这个用户是他本人 那么对于访问控制呢

文件的访问控制

访问控制

主动控制：访问控制表

- ✓ 每个文件一个
- ✓ 记录用户ID和访问权限
- ✓ 用户可以是一组用户
- ✓ 文件可以是一组文件

能力表(权限表)

- ✓ 每个用户一个
- ✓ 记录文件名及访问权
- ✓ 用户可以是一组用户
- ✓ 文件可以是一组文

这是对于文件的一个访问控制的一个基本的一个设计 下面我们来介绍一下

UNIX的文件访问控制

采用文件的二级存取控制
审查用户的身份、审查操作的合法性

第一级：对访问者的识别
对用户分类：
- ✓ 文件主（owner）
- ✓ 文件主的同组用户（group）
- ✓ 其他用户（other）

第二级：对操作权限的识别
对操作分类：
- ✓ 读操作（r）
- ✓ 写操作（w）
- ✓ 执行操作（x）
- ✓ 不能执行任何操作（-）

例子：rwx rwx rwx

chmod 711 file1 或 chmod 755 file2

0:00 下面我们介绍一下文件系统的安全性 文件系统的安全，涉及到了方方面面，在这里 我们只考虑如何确保未经授权的用户，不能存取某些文件 我们这里头 介绍的文件保护机制呢，主要要满足两个要求 第一个要求呢，是对拥有权限的用户，应该能够让他进行相应的操作 否则就要制止，禁止他进行相应的操作 第二个要求呢，是防止其他的用户 来冒充用户本人呢，对文件进行相应的操作 因此在实现的时候，我们要考虑到用户身份的验证 和访问控制。 用户身份呢 也就是说这个用户他是谁？他拥有什么？ 他知道些什么？通常我们可以用口令、 密码 或者是让他回答一些问题来确定这个用户的身份 还可以采用一些物理鉴定的手段，比如说指纹 虹膜等等，来确保这个用户是他本人 那么对于访问控制呢 在一个文件系统当中 可能提供不同的访问控制手段 我们左边这个呢，实际上是一种 主动的访问控制的方式，也是一个访问控制表 这个访问控制表里呢，是每个文件有这样一张表 这张表里记录了哪些 用户可以对这个文件进行什么样的操作 也就是用户的 ID 和访问权限 如果这个用户对这个文件没有任何权限，这张表里就不会出现 这个用户的 ID，所以通过检查就知道，这不是一个可以访问这个文件的用户 那么对于用户来讲，其实可以是一组用户，我们可以把用户 分成若干组，按组来规定他的访问的权限 那么文件呢，也可以按一组文件来设置这样一个访问控制表 右边这种情况呢 是指的一种权限表 这个权限表呢 指的是每个用户有这样一纸表 刚才是每个文件一张表，这个是每个用户一张表 这张表里记录了这个用户可以访问的一些文件 可以访读这个文件，还可以写这个文件，记录了这样一个权限 那么同样道理，用户也可以是一组用户，文件也可以是一组文件 这是对于文件的一个访问控制的一个基本的一个设计 下面我们来介绍一下 UNIX 当中的文件访问控制 在 UNIX 文件系统当中，它采用的是二级的存取控制 那么首先要审查用户的身份 然后要审查操作的合法性 那么第一级呢 是对访问者的这个身份的识别 它对这个用户的分类呢 分成三种，一种是文件的创建者，文件的主人 owner 一种呢 是文件主的 同组用户，他们可能在同一个项目组里头，把他们划定为同组用户，就是 group 还有一类用户就是跟他们没有关系的，其他用户 other 这是第一级对访问者的这个身份的一个识别 第二级呢 是对操作权限的一个识别 那么操作权限呢 分成了读写 执行和不允许执行任何操作这样几类 好，既有身份识别，又有 权限的这个控制，那么把它组合在一起呢 在 UNIX 中呢 提供了这样一个手段，提供了这样一个 三位组，三组三位 那么第一组呢 代表的是 文件主可以进行什么样的操作，最左边的第一组 第二组代表的是文件的同组的用户可以进行什么样的操作 那么第三组是

对其他的用户可以执行什么样的操作 那么操作呢 是读、写、 执行，分别代表一位 那么如果是可读，可写，可执行 那么就对应的是每一位都有是设置就 111 ， 111 我们知道是 7 那么这样的话呢，我们就可以通过命令行 对这个文件的访问控制进行相应的设置了 这里举了两个例子，第一个例子呢 是改变这个 文件 file1 的一个访问控制的这样一个状态 那么改变了是用 711 来设置 那么 7 就表示文件主人 那么可读、可写、 可执行，那么中间这个 1 就表示文件的同组用户 那么他呢只能够执行，因为最后这个是执行 X 是执行 那么其他用户也可以执行这个文件 第二个例子呢 是chmod 755 对一个 file 2 进行相应的权限设置 那么 7 是说文件的主人 文件创建者可读、 可写、 可执行这个文件 而文件的同组用户和其他用户 对这个文件都可以读，或者执行，因为 5 呢 是 101 ，所以读或者执行 当然我们可以不同的组合来对这个 文件的访问控制进行相应的设置