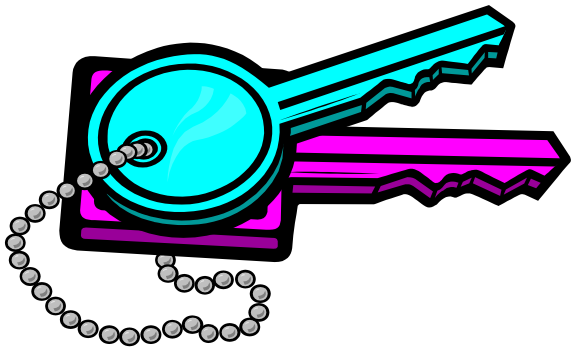
第四章
系统安全



信息安全概论

安全服务 (X.800)

- 身份认证
  - 带照片的身份证
- 访问控制(防止对资源的非授权使用)
  - 钥匙开锁，演唱会门票
- 数据保密性
  - 密封的信件
- 数据完整性
  - 不可擦除墨水，门上的封条
- 不可否认性
  - 手写签名/指纹

大纲

1. 访问控制模型
2. 访问控制实现
3. 授权、信任模型& 审计
4. TCSEC标准

1 访问控制模型

- **访问控制**
  - 主体依据某些控制策略，获得不同的权限对客体进行的不同授权访问
- **访问控制模型**
  - 从主体、客体、控制策略三者关系的角度出发，描述安全系统，建立安全模型的方法

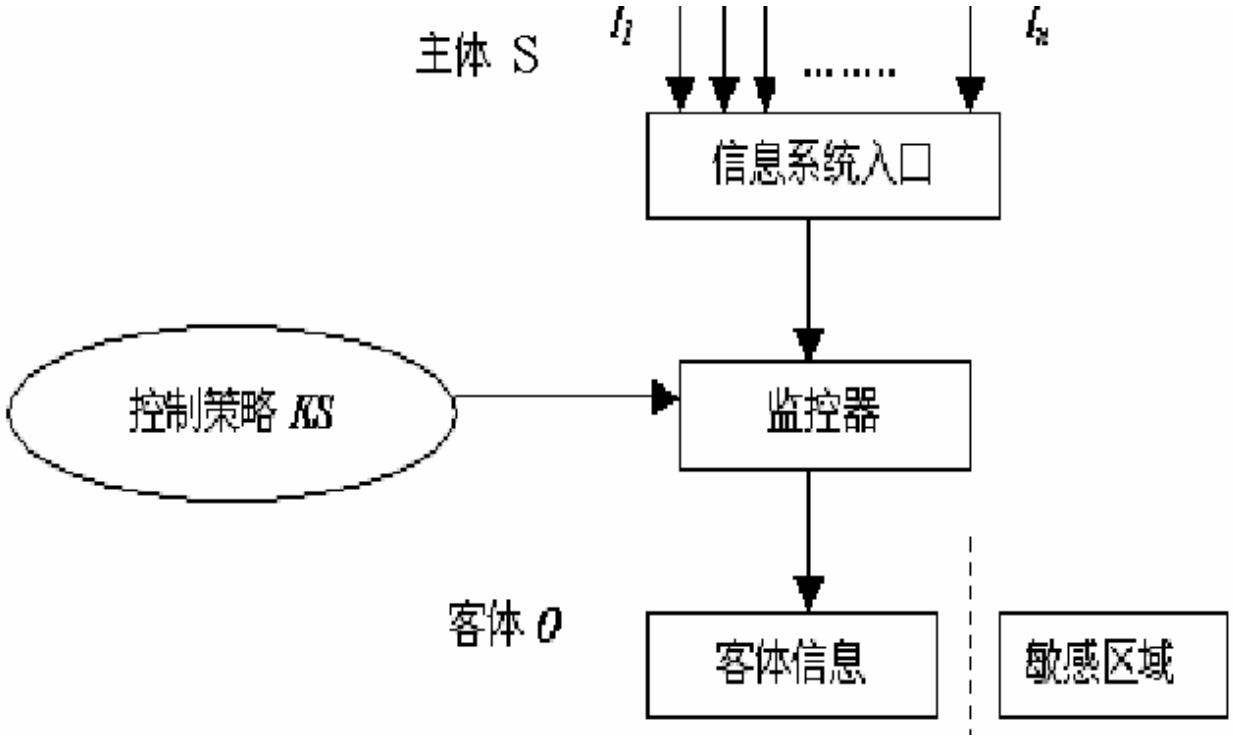
访问控制目标

- 对主体和客体提供安全防护
- 确保不会有非授权者使用合法或敏感信息
- 确保授权者能够正确使用信息资源
- 实现安全的分级管理

访问控制三要素

- 主体Subject
  - 对客体提出访问请求的主动实体，也称为用户或授权访问者
- 客体Object
  - 接受主体访问的被动实体
- 控制策略Control Policy
  - 主体对客体的访问规则集合

实现框架



多级安全系统

- 将敏感信息和普通信息相互隔离的系统
- 层次型
  - 绝密Top Secret
  - 秘密Secret
  - 机密Confidential
  - 限制Restricted
  - 无级别Unclassified
- 无层次型
  - 给出客体接收访问时允许的规则和主体

访问控制三步骤

- 身份认证：主客体间的双向认证
- 控制策略的制定和实现
- 审计

控制策略

设定适当的规则集合，确保授权用户以多级安全访问信息，防止敏感信息被泄露给非授权用户

- 制定——创建用户& 文件共享
- 实现——用户访问行为

审计

- 审计是访问控制的重要内容与补充，可以对用户使用何种信息资源、使用的时间、以及如何使用进行记录与监控
- 审计的意义在于客体对其自身安全的监控，便于查漏补缺，追踪异常事件，从而达到威慑和追踪非授权者的目的；也可以避免合法用户滥用职权（不能被控制策略限制的不安全行为）

访问控制模型

主体、客体和访问策略三者之间关系的实现构成了不同的访问控制模型，是访问控制实现的基础

- 自主访问控制Discretionary Access Control (DAC)
- 强制访问控制Mandatory Access Control (MAC)
  - ✓BLP
  - ✓Biba
- 基于角色的访问控制Role-based Access Control (RBAC)

自主访问控制DAC

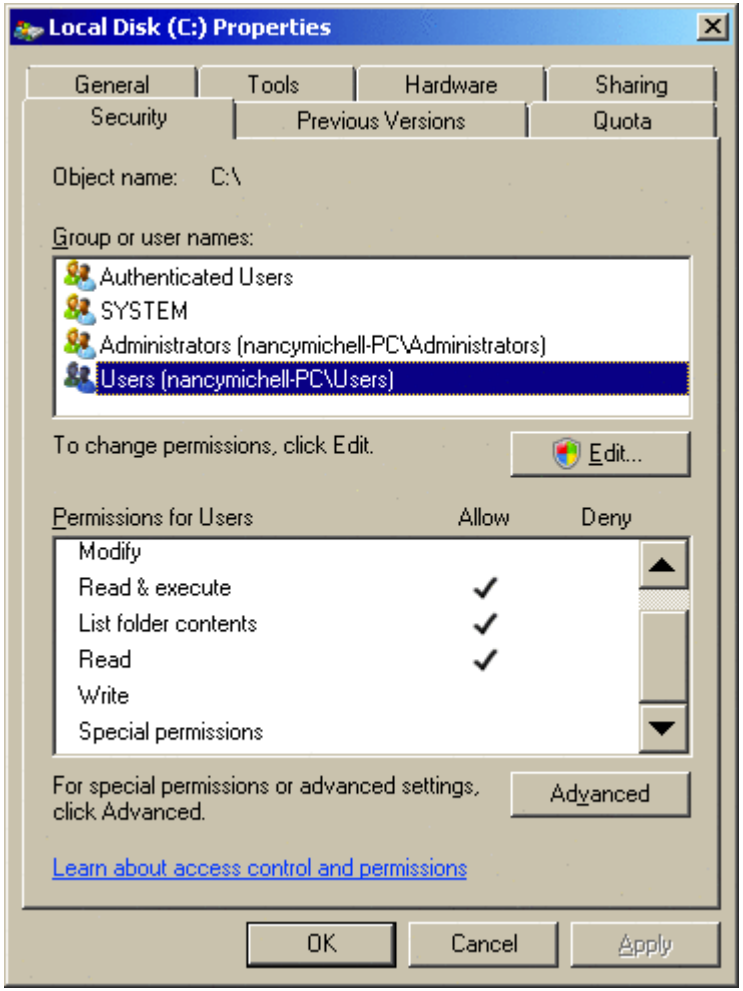
- 属于无层次型多级安全

  允许合法用户根据访问策略以**用户**或**用户组**的身份访问客体，同时阻止非授权用户访问客体；某些用户还可以**自主地**把自己所拥有的客体的访问权限**授予他人**

- 访问权限的赋予和回收由管理员实现，也可由文件创建者自身实现
- 访问控制由访问控制矩阵和访问控制列表来实现
- 方便灵活，广泛应用于商业和工业
- 任意授权，安全防护相对较低

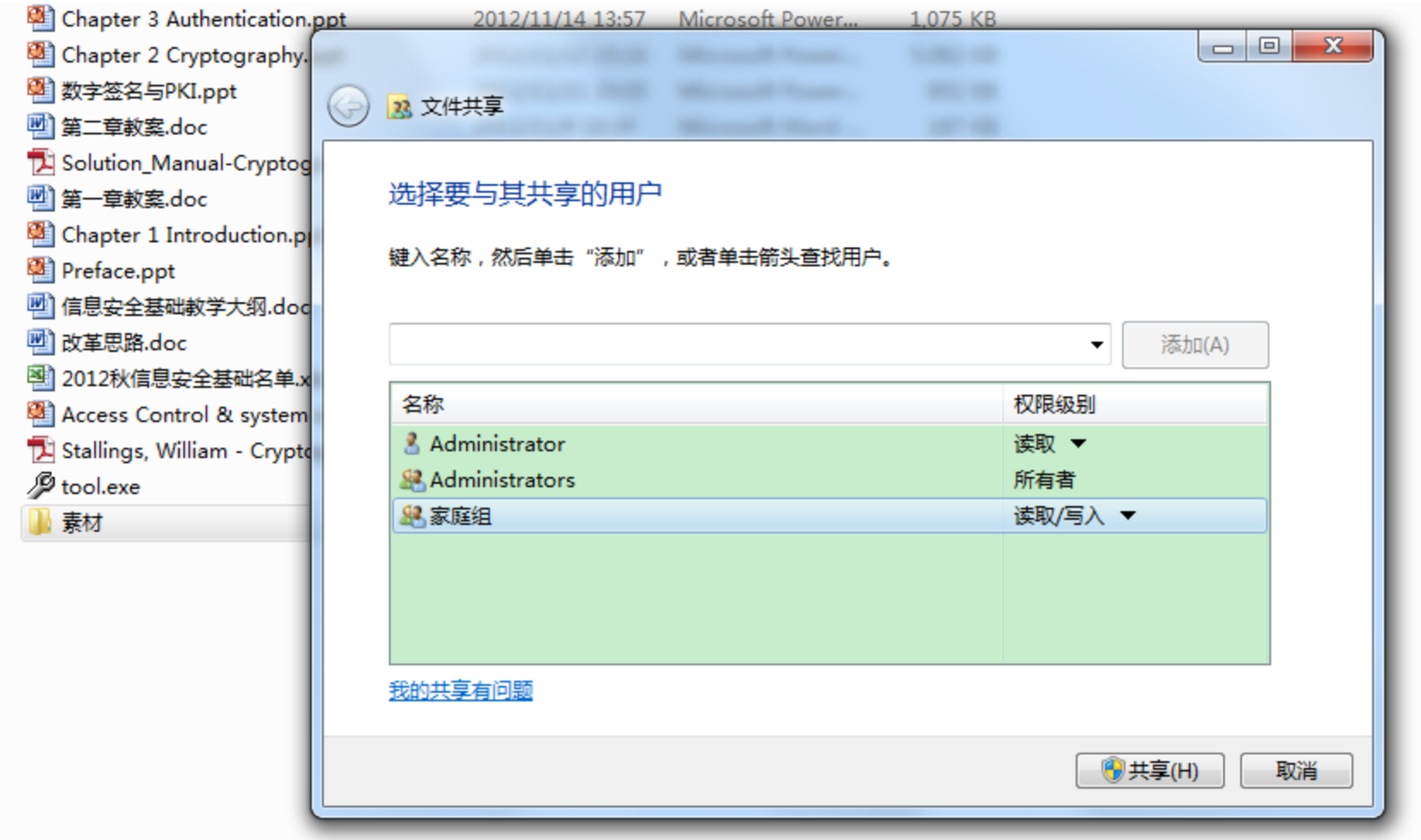
	计算机管理员	受限用户
安装程序和硬件	√	
进行系统范围的更改	√	
访问和读取所有非私人的文件	√	
创建与删除用户帐户	√	
更改其他人的用户帐户	√	
更改自己的帐户名或类型	√	
更改自己的图片	√	√
创建、更改或者删除自己的密码	√	√

Windows XP Users



Windows Server 2008 Users

Windows 7/10 Users



File Control Policy of Windows 7

强制访问控制MAC

- 属于层次型多级安全

  系统事先给主体和客体分配不同的安全级别；在实施访问控制时，系统先对主体和客体的**安全级别**进行**比较**，再决定主体能否访问该客体

安全等级的比较

主体和客体可能分属不同的安全级别SC，SC构成一个偏序关系(TS>S>C>R>U)

四种访问方式

- 下读 (Read down, rd)
  - 主体安全级别高于客体安全级别时允许的读操作
- 上读 (Read up, ru)
  - 主体安全级别低于客体安全级别时允许的读操作
- 下写 (Write down, wd)
  - 主体安全级别高于客体安全级别时允许的执行或写操作
- 上写 (Write up, wu)
  - 主体安全级别低于客体安全级别时允许的执行或写操作

- MAC实现了信息的单向流通
  - Bell-LaPadula模型只允许下读上写，可以有效地防止机密信息向下级泄露
  - Biba模型只允许上读下写，可以有效地保护数据的完整性

BLP模型(1976)

- 主要应用于军事系统
- 维护系统的保密性，有效地防止信息泄露
  - 下读，当且仅当SC(s)≥SC(o)
  - 上写，当且仅当SC(s)<SC(o)

- 信息流向：**低→高**
- 有效防止低级用户和进程访问安全级别比他高的信息资源
- 安全级别高的用户和进程也不能向比他安全级别低的用户和进程写入数据
- 缺点：使非法、越权篡改成为可能

Biba模型(1977)

- 信息流向：高→低
- 用户只能向比自己安全级别低的客体写入信息，从而防止非法用户创建安全级别高的客体信息，避免越权篡改等行为的产生
- 维护完整性，但忽略了保密性
  - 上读，当且仅当SC(s) ≤ SC(o)
  - 下写，当且仅当SC(s) SC(o)
- 缺点：信息有可能从高级别主体向低级别客体流动，造成敏感信息泄露

DAC&MAC的区别

- 前者是无层次型；后者是层次型
- 前者的授权基于用户，所以用户的访问权限可以自由的赋予他人；后者的授权基于安全级别，访问权限是严格的按照安全级别来执行的，仅能由安全管理员去修改安全级别

基于角色的访问控制模型RBAC

- 基本思想：将访问许可权分配给一定的角色，用户通过饰演不同的角色获得角色所拥有的访问许可权
- 角色（Role）是指一个可以完成一定事务的命名组，不同的角色通过不同的事务来执行各自的功能

- RBAC从控制主体的角度出发，根据管理中相对稳定的职权和责任来划分角色，将访问权限与角色联系，通过给用户分配合适的角色，让用户与访问权限相联系。角色成为访问控制中访问主体和受控对象之间的一座桥梁

- 角色和组的主要区别在于
  - 用户属于组是相对固定的；而用户能被指派到哪些角色则受时间、地点、事件等诸多因素影响，变动比较频繁

- RBAC是实施面向企业的安全策略的一种有效的访问控制方式
- 只有系统管理员有权定义和分配角色。用户与客体无直接联系，他只有通过角色才享有该角色所对应的权限，从而访问相应的客体

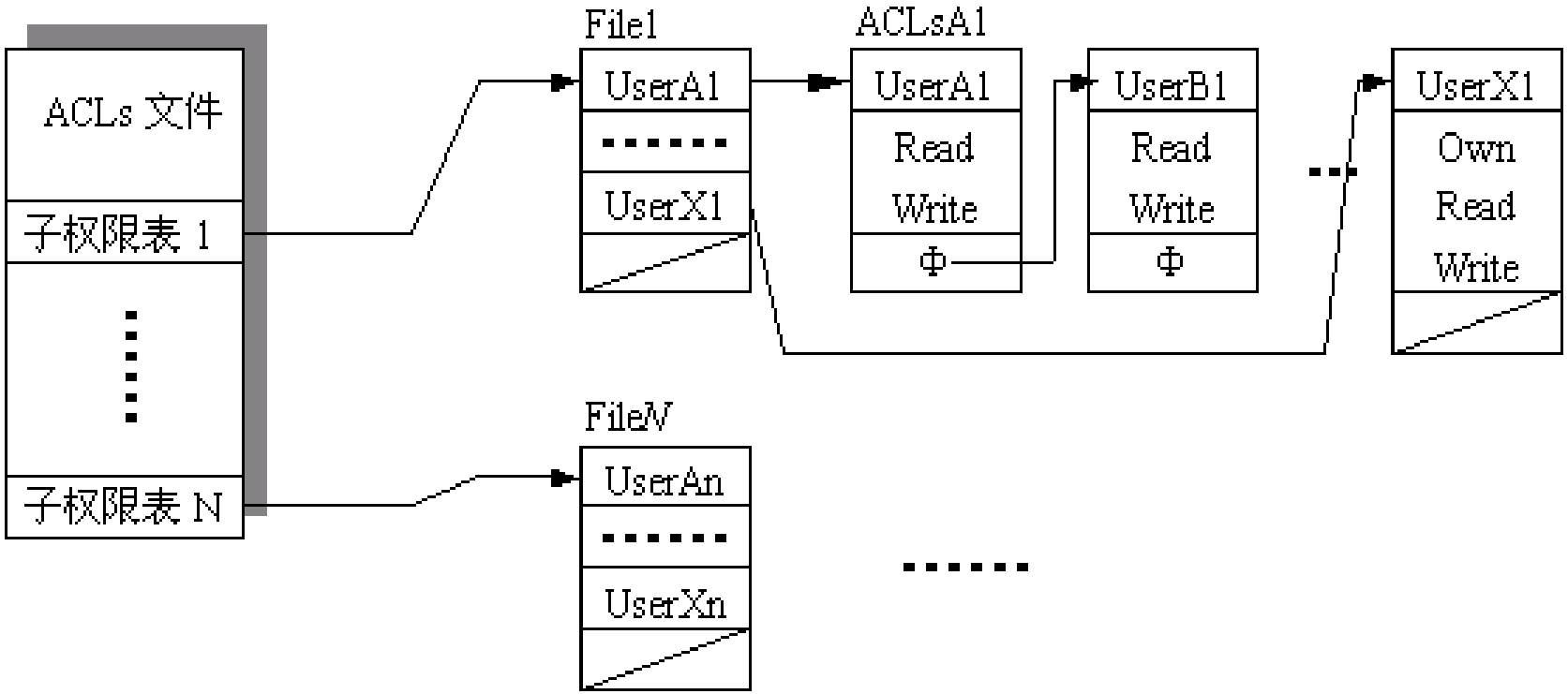
- RBAC与DAC的区别
  - ✓ 用户不能自主的将访问权限授予别的用户
- RBAC与MAC的区别
  - ✓ RBAC属于无层次安全级别，MAC属于层次型安全级别

2 访问控制实现

- 访问权限
  - 读
  - 写/执行
  - 管理: 修改访问控制规则或安全级别的能力

访问控制列表

- 以文件为中心



访问控制矩阵

- 包含着所有主客体及其关系的矩阵

权限 / 授权用户	文件 1	文件 2	……	文件 N
授权用户 A	Own R W			
授权用户 B		R		
……				
授权用户 N	R W			Own R W

安全标签列表

- 用于MAC模型中主客体的安全级别集合

用户	安全级别
UserA	S
UserB	C
……	……
UserX	TS

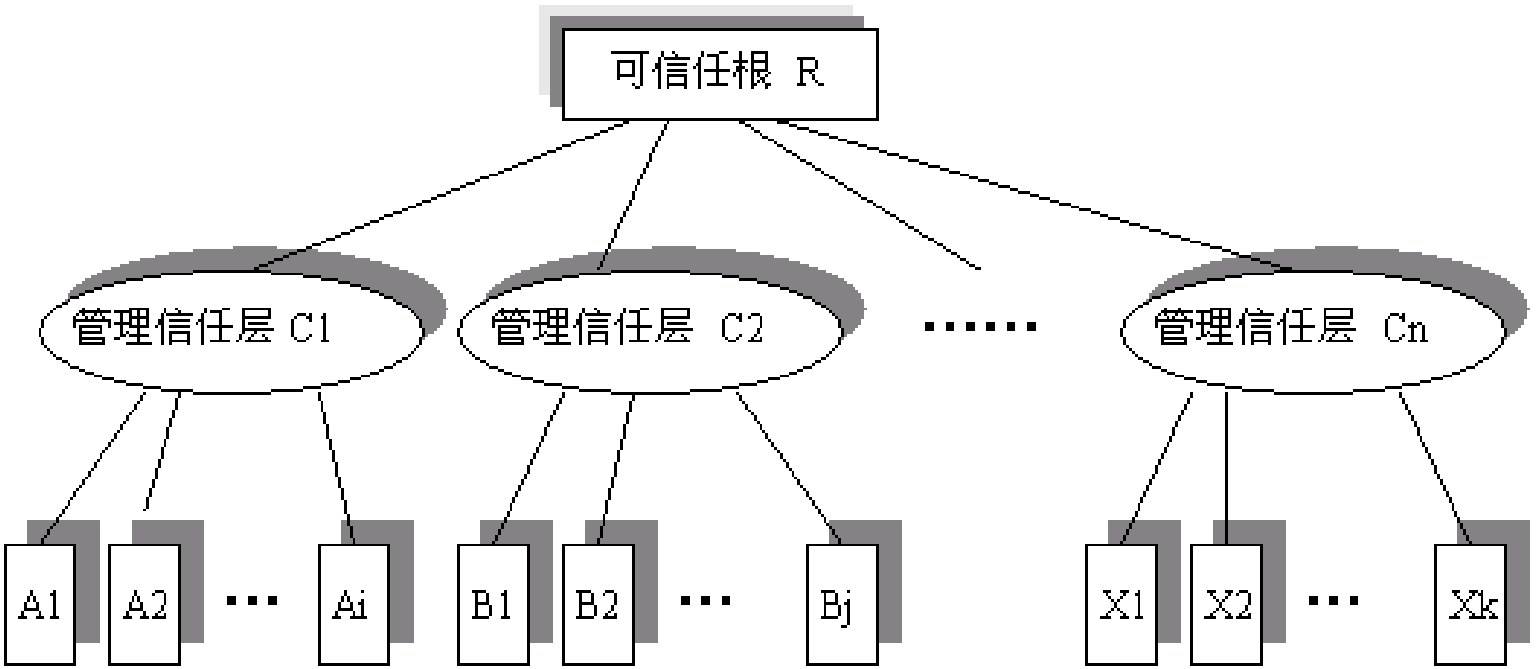
文件	安全级别
File1	S
File2	TS
……	……
FileN	C

3 授权、信任模型&审计

- 授权是客体赋予主体某些访问权限的过程，是实现访问控制的前提
- 授权基于信任关系，需要建立信任模型对这种关系进行描述
- 信任模型
  - 建立和管理信任关系的框架
  - 如果主体行为能满足客体的期望，则称客体信任主体
  - 信任域是所有主客体间建立信任关系的范畴集合

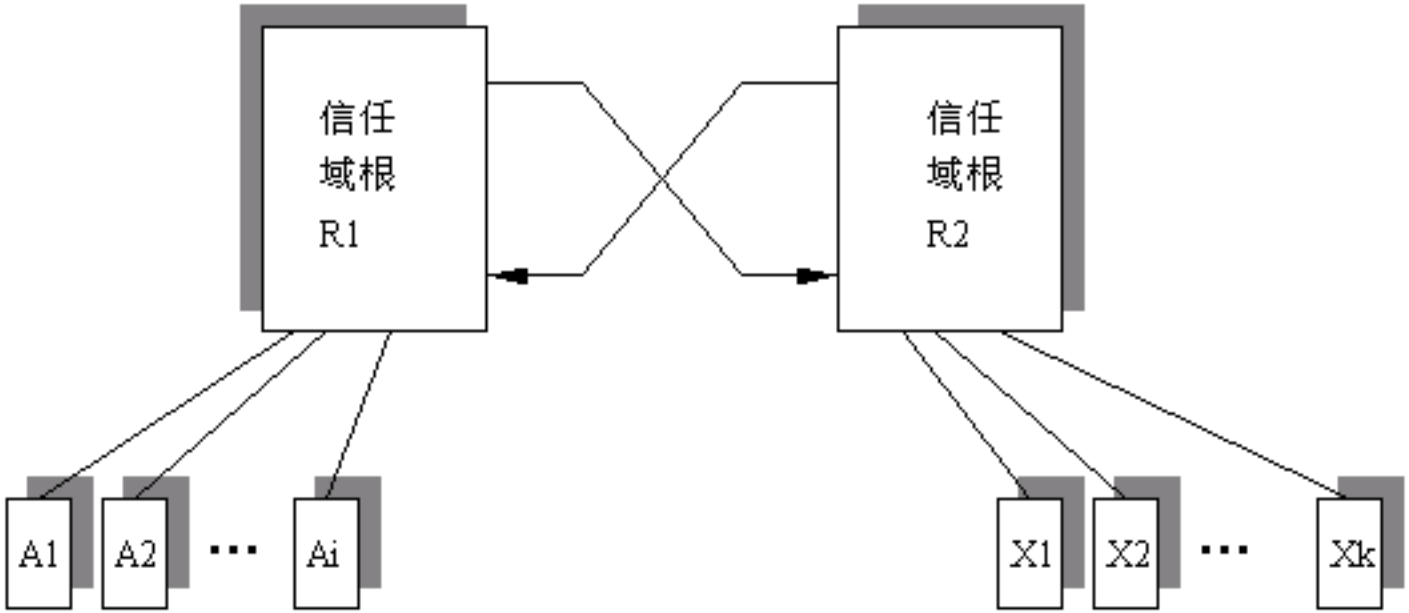
- 信任模型
  - 层次
  - 对等
  - 网络

- 层次型



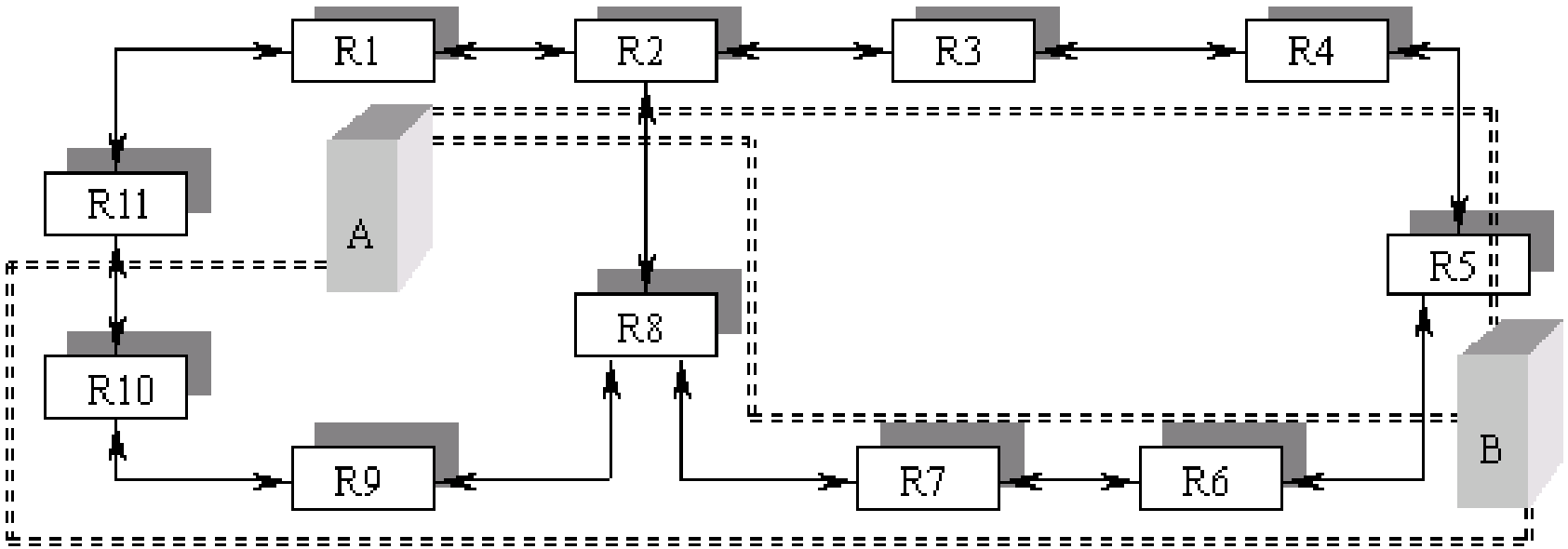
- 所有的节点都基于同一个可信根
- 优点：结构简单，管理易于实现
- 缺点：一旦信任根出现问题，则所有的信任无法建立
- 适用于孤立的，层次型的企业，不适用于多个有边界交叉的情况

- 对等型



- 两个或两个以上对等的信任域间的交互信任
- 适用于多个动态变化的信任机构

- 网络型



- 对等信任模型的扩充
- 利用信任链的传递建立了信任域间的间接信任关系
- 信任度随着信任链的长度递增而急速衰减

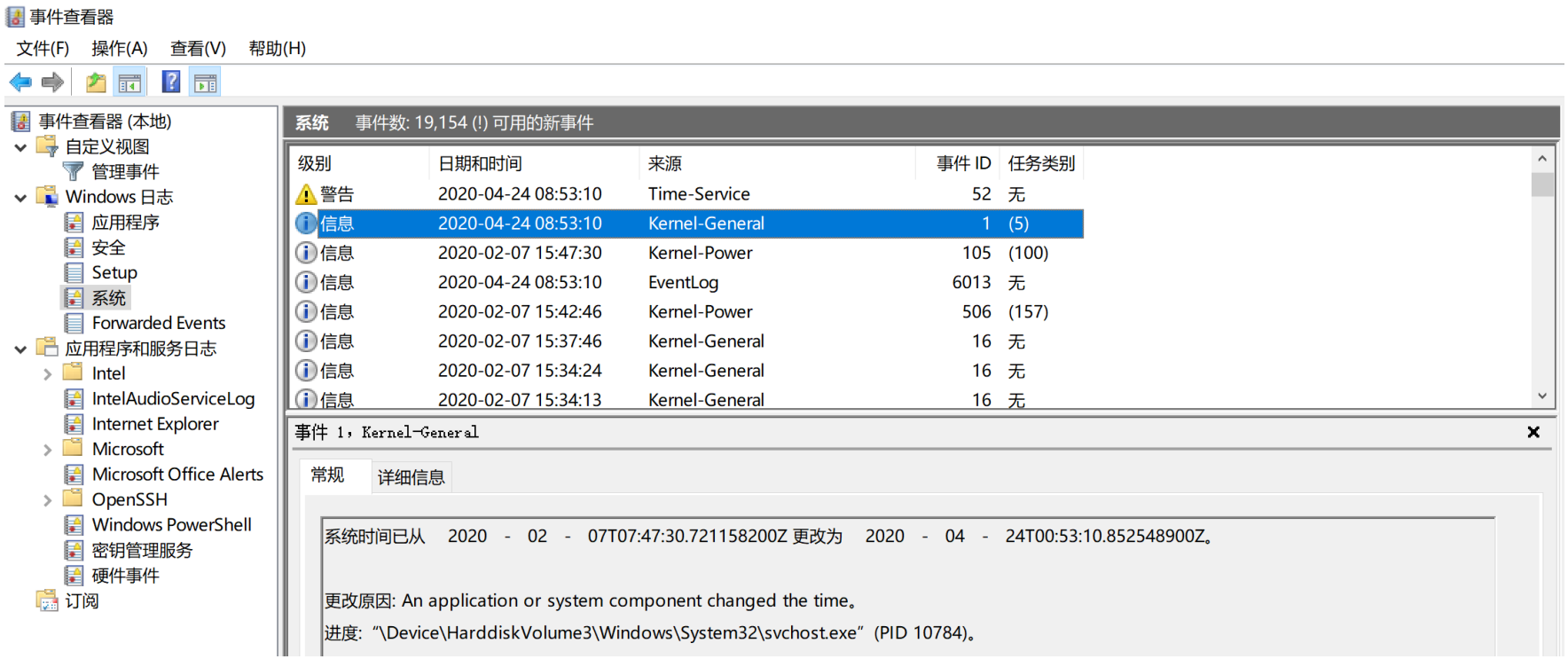
审计

- 审计是对访问控制的必要补充，是实现系统安全的最后一道防线，处于系统最高层
- 追踪和记录系统行为和用户行为
  - 系统行为包括OS和应用程序的活动
  - 用户行为包括用户在OS和应用程序中的活动
- 审计有助于保护系统和资源免受非授权篡改，能在一定程度上防止合法用户的滥用职权，同时还能提供对数据恢复的帮助

审计内容

- 个人职能
- 事件重建
- 入侵检测
- 故障分析

Windows审计机制



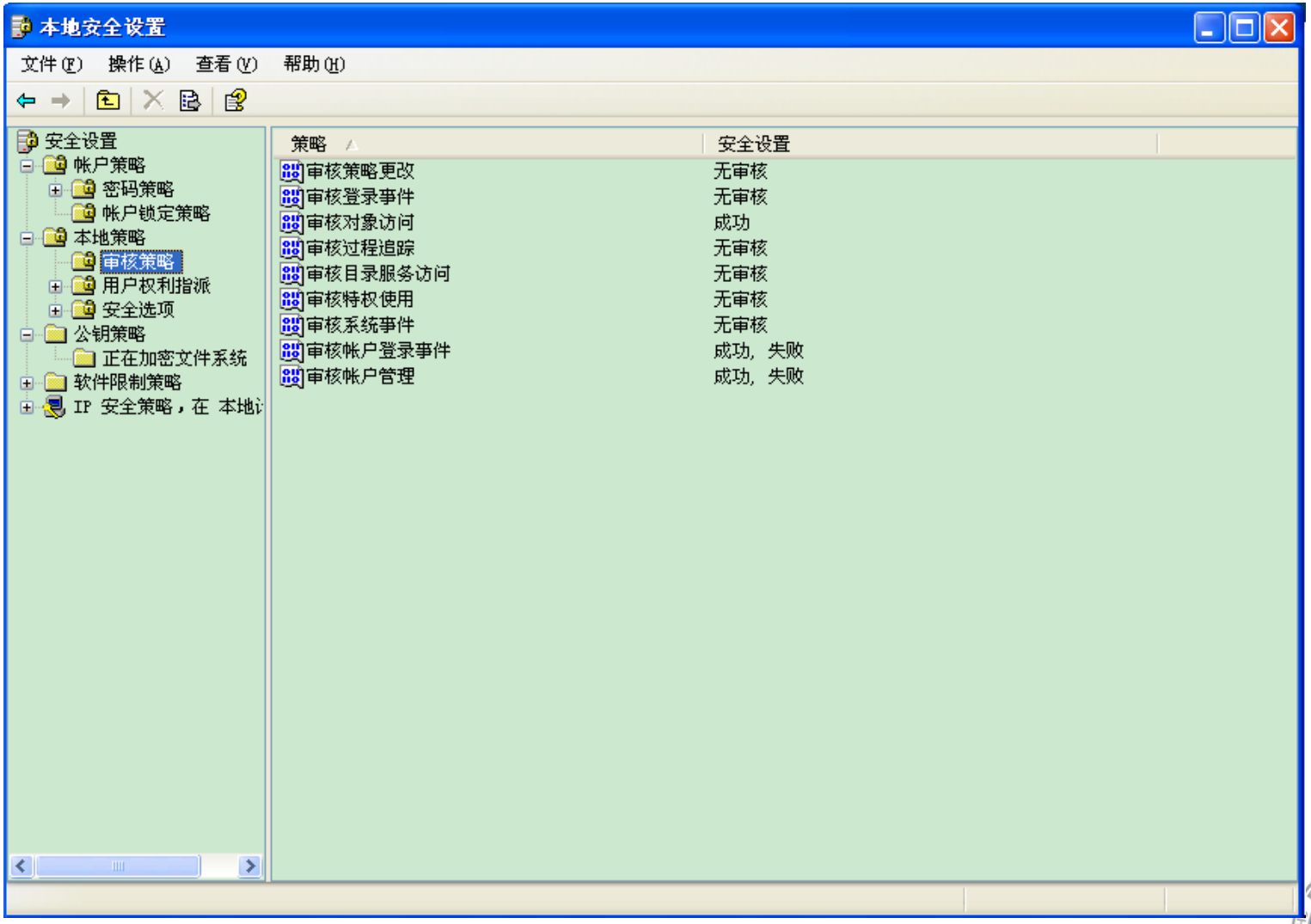
控制面板-系统和安全-管理工具-事件查看器

Reference:

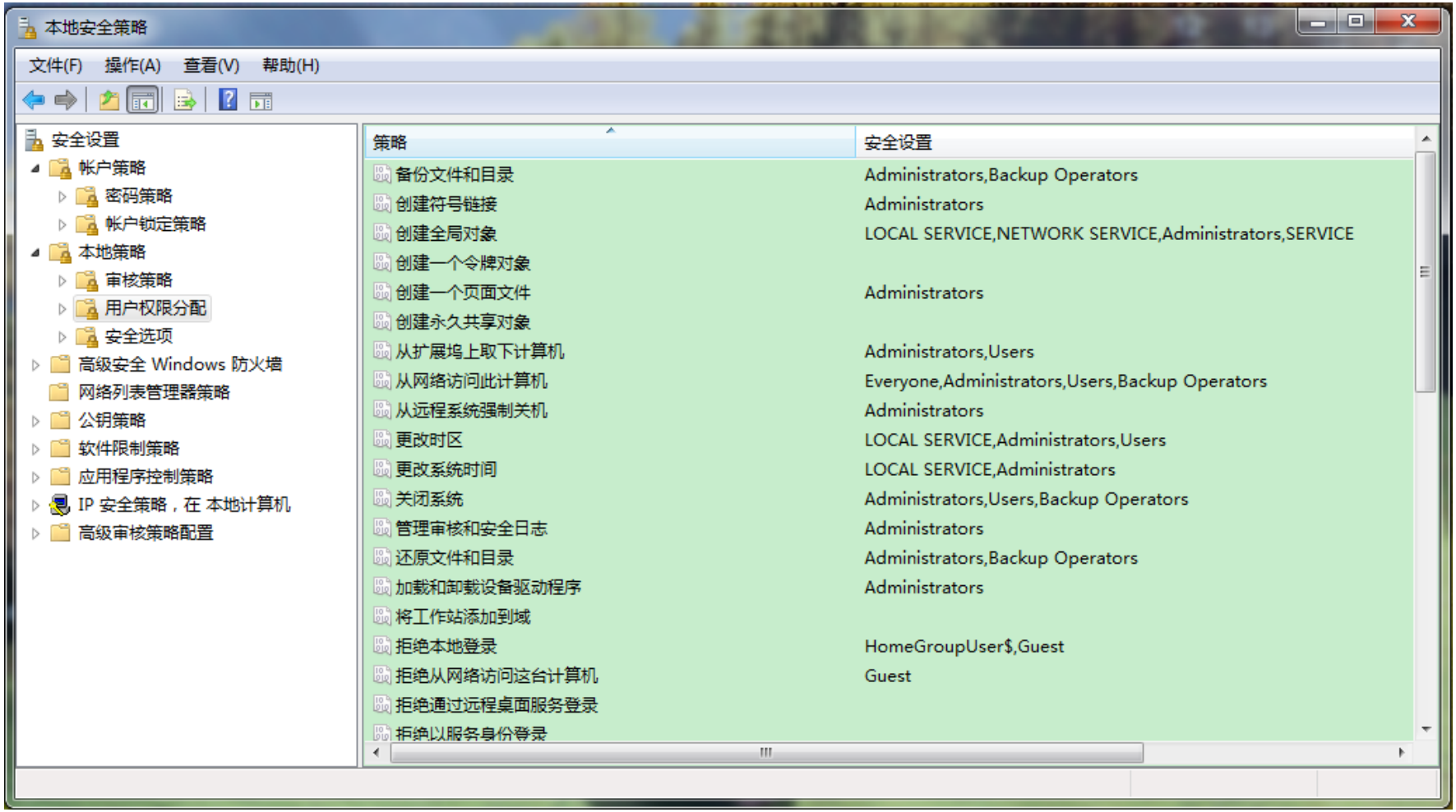
http: //support.microsoft.com

http://www.eventid.net

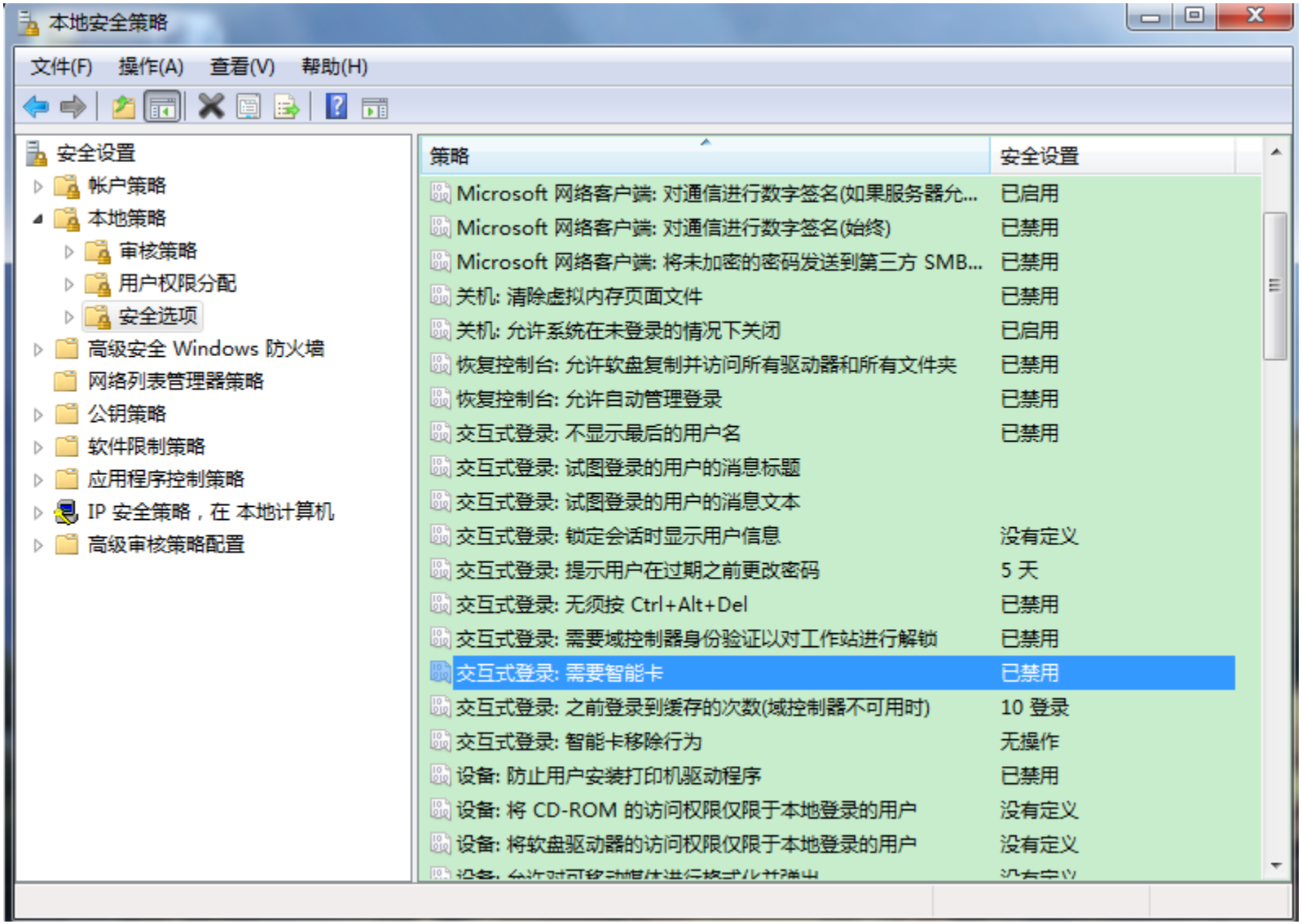
Windows审计机制



Windows审计机制

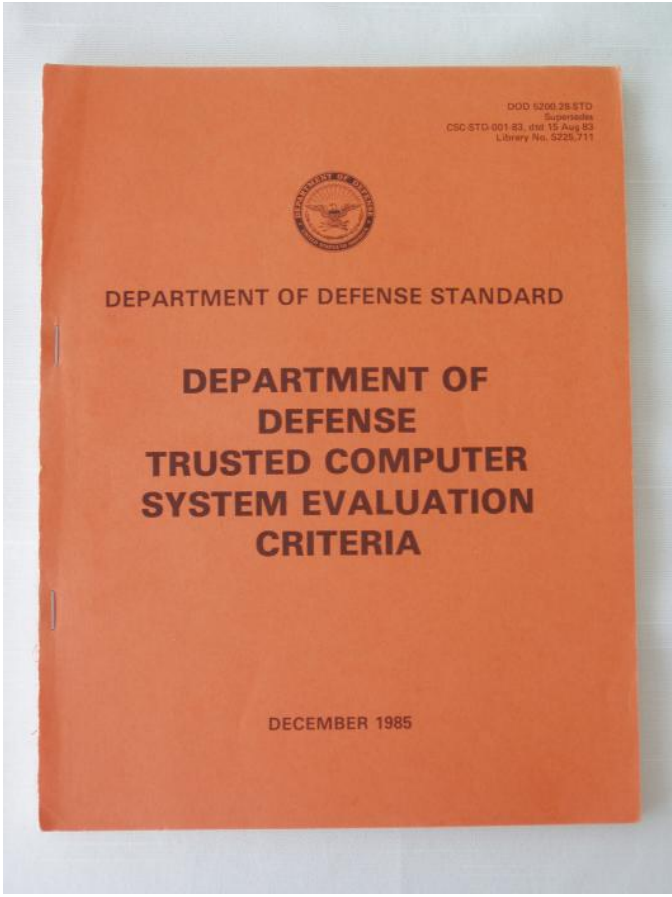


Windows审计机制



4 TCSEC标准

- 美国橘皮书
  - ✓ 1983, 美国国防部制定了评估计算机安全有效性的基本标准----TCSEC (可信计算机系统评估标准,Trusted Computer System Evaluation Criteria)
  - ✓ 面向敏感或机密信息的处理、存储和恢复，用于评估、分类和选择计算机OS



分类

- D
- C
  - C1
  - C2
- B
  - B1
  - B2
  - B3
- A
  - A1
  - 超A1

- D---最低保护(DOS)

  没有访问控制甚至没有用户身份认证

- C---自主保护
  - C1 — 自主安全保护(早期Windows)
    - 只有认证
    - 用户和数据分离
    - DAC直接作用于OS的根，C1级所有文件具有相同机密性
  - C2 — 受控访问保护(当前Windows)
    - 更细粒度的DAC
    - 每个用户有独立的DAC机制，支持多级安全
    - 审计机制
    - 资源隔离

- B---强制保护
  - B1 — 标识安全保护
    - 安全策略模型的非形式化描述
    - 在部分主体和客体上实现MAC
  - B2 — 结构保护
    - 明确定义的形式化模型
    - 对所有主体和客体实现DAC和MAC
    - 分析隐蔽存储信道Covert storage channels
  - B3 —安全域保护
    - 专用硬件TCB(Trusted Computing Base)加强域的安全
    - 满足参考监视器的要求
    - 要求最小化系统设计的复杂度，删除对于安全策略不必要代码
    - 自动化的实时入侵检测和响应功能
    - 可信系统恢复功能
    - 分析隐蔽时间信道

- A---验证保护
  - 形式化的设计与验证
  - 形式化的管理和分配过程

2021

Thank you…