信息安全导论

第六章 操作系统安全

黄 玮

温故

- 信息安全基础
- 密码学基础理论
- 密码学典型应用

温故

- 信息安全基本属性
  - —机密性/完整性/可用性
- 信息安全基本概念
  - —资产/风险/漏洞/威胁/攻击/影响
  - —安全策略/安全机制
  - —安全模型：STRIDE
  - —安全标准：CVSS

温故

- 加密
  - —分组密码/流密码
  - —单密钥体系/双密钥体系
- 完整性
  - —散列算法
  - —数字签名

温故

- 认证
  - —数字标识
  - —双因素认证
- 不可抵赖性
  - —数字证书/PKI

温故

- 信息安全实践
  - —PGP
    - 公钥加密与解密/数字签名/私钥备份/公钥发布
  - —Truecrypt
  - —Windows文件系统加密
    - 私钥的备份

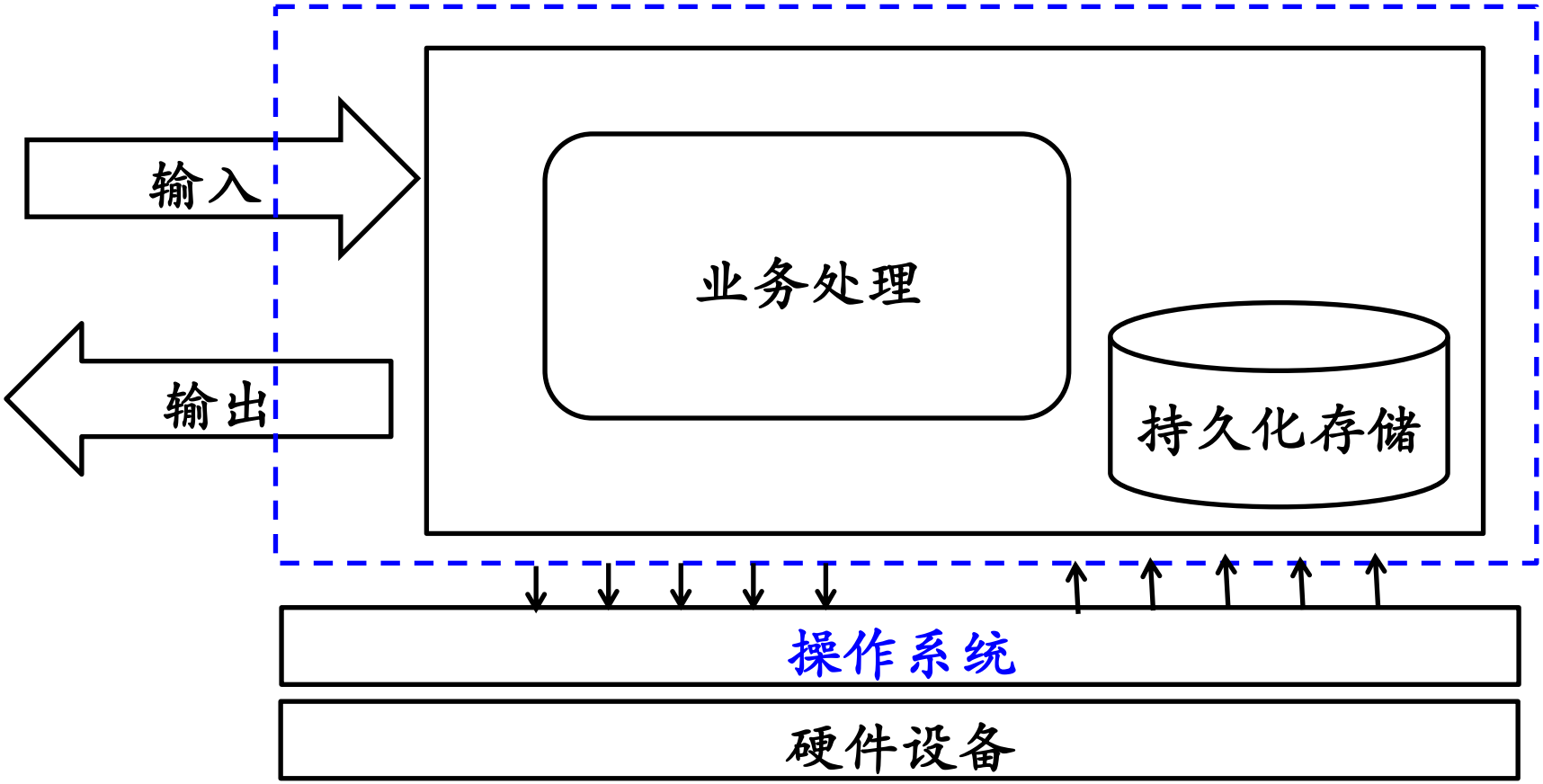
- 安全操作系统
- 主流操作系统的安全策略与安全机制

本章内容提要

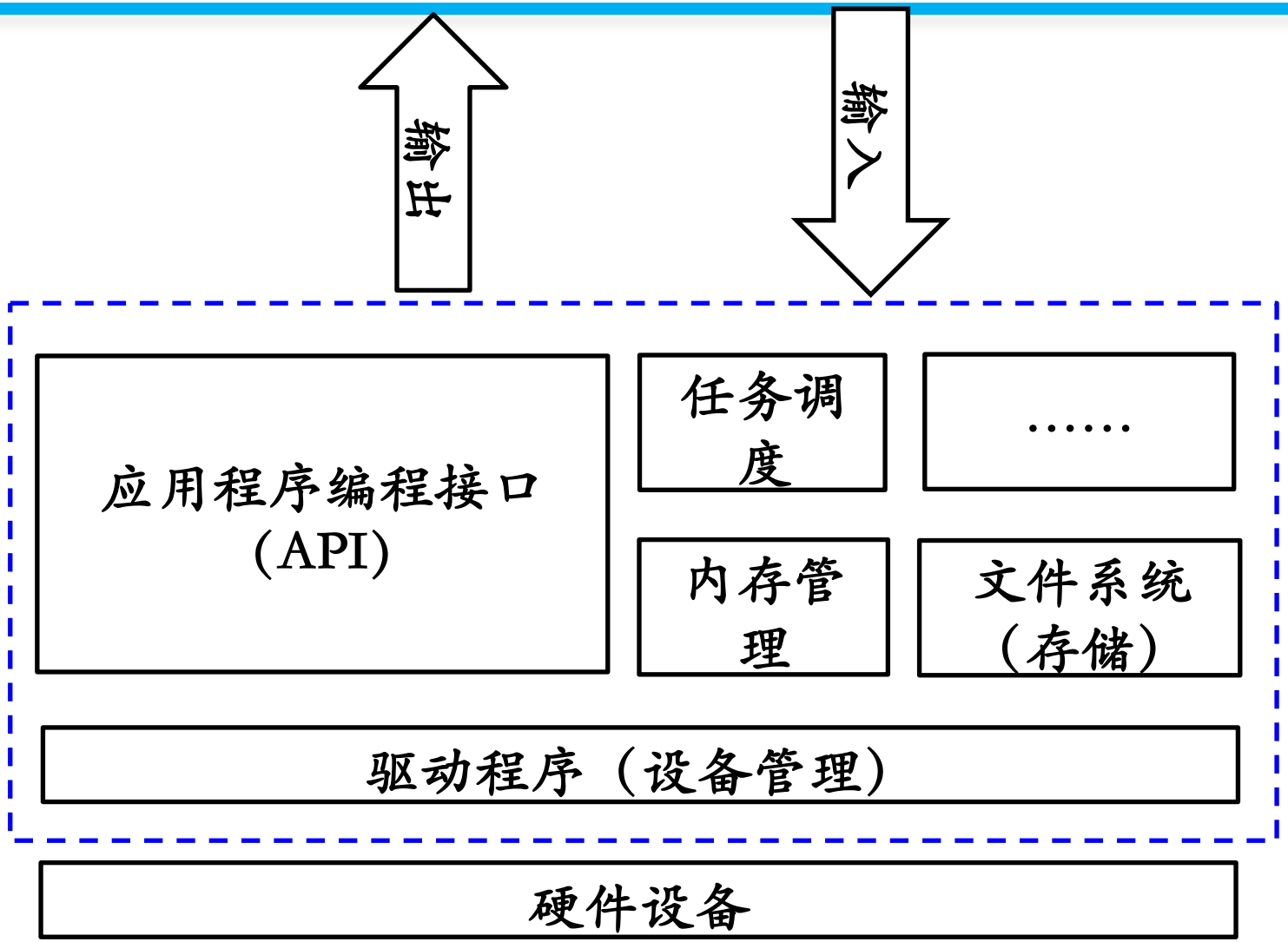
- **信息系统安全模型**
- 安全操作系统
- 主流操作系统安全模型
- 主流操作系统安全机制

信息系统是什么

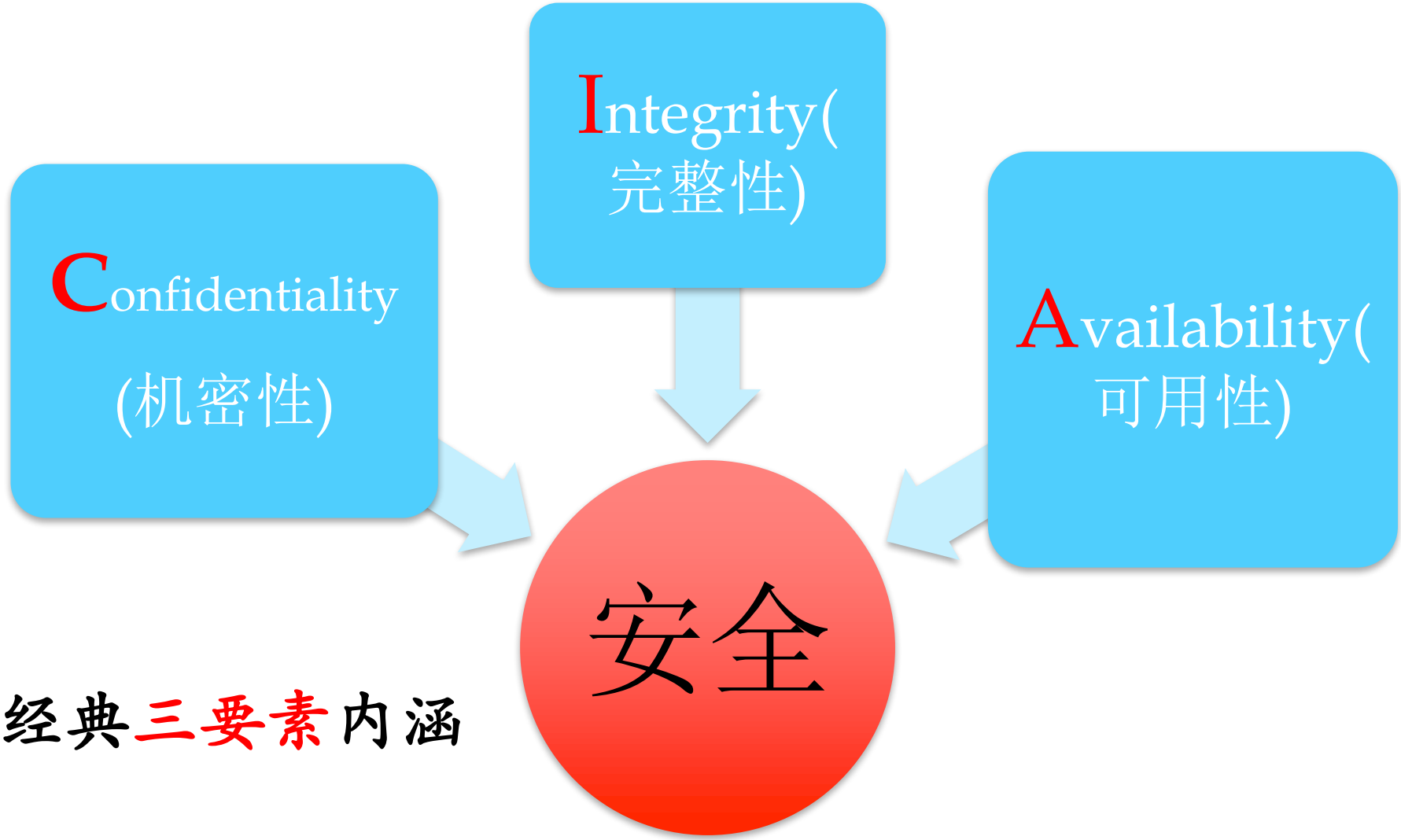
- 典型信息系统的组成与结构



操作系统



安全是什么

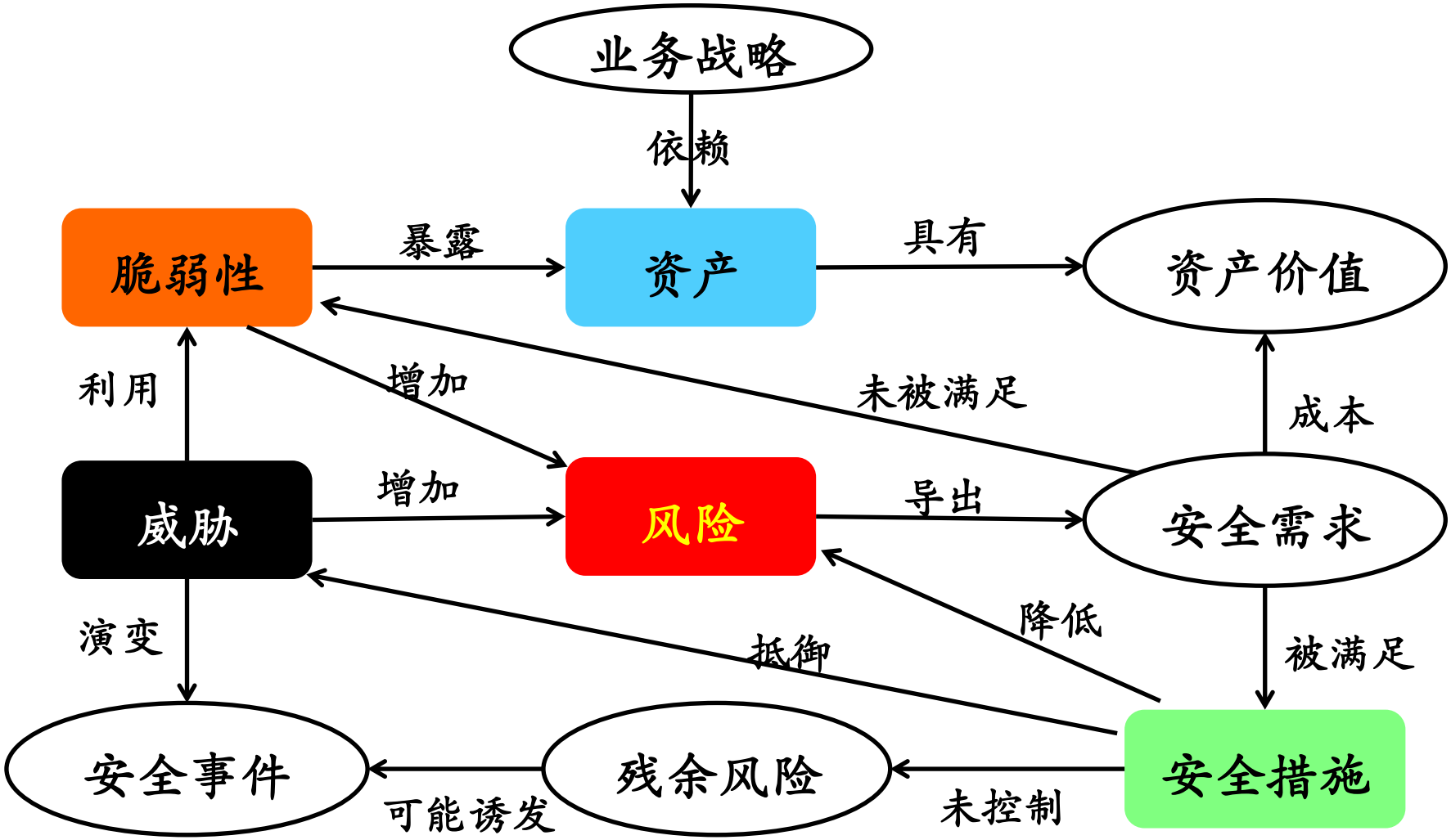


经典三要素内涵

基本概念

- 安全策略 (Security Policy)
  - —声明
  - —哪些能做，哪些不能做
  - —哪些行为允许，哪些行为禁止
- 安全机制 (Security Mechanism)
  - —方法/工具/手段
  - —实现安全策略

我国国家标准给出的信息系统各要素间关系



信息系统安全模型概述

- 信息系统安全策略模型
- 访问控制是信息安全模型的实现机制
  - —安全策略：信息安全模型
  - —安全机制：信息系统访问控制

访问控制的基本概念

- 主体
  - —主动的实体，是访问的发起者，它造成了信息的流动和系统状态的改变，主体通常包括人、进程和设备等
- 客体
  - —包含或接受信息的被动实体，客体在信息流动中的地位是被动的，客体通常包括文件、设备、信号量和网络节点等
- 访问
  - —是使信息在主体和客体之间流动的一种交互方式
    - 读、写、执行等

访问控制的基本概念

- 授权访问
  - —主体访问客体的允许，授权访问对每一对主体和客体来说是给定的
- 安全访问策略
  - —一套规则，可用于确定一个主体是否对客体拥有访问能力
  - —主体对客体的操作行为集和约束条件集
- 访问控制的三要素
  - —主体、客体、安全访问策略

访问控制模型

- 访问控制的三个基本面
  - ——认证
    - 身份认证：客体对主体的识别认证
    - 客体和主体的身份可以随着时间、应用场景而改变
  - ——(访问控制)策略实现：访问授权
    - 授权主体对客体可以正常访问
    - 非授权主体对客体无法访问
  - ——访问审计
    - 记录访问历史，实现不可抵赖性

BLP安全策略模型

概述

- Bell-LaPadula
  - —第一个有数学基础的访问控制模型
  - —Bell和LaPadula在1973年提出
  - —Denning于1975年给出一个基于格的严格数学描述
  - —结合强制访问控制和自主访问控制

BLP强制访问控制模型

- 军队保密需求催生了BLP强制访问控制模型
  - —军事文档分保密等级
    - 绝密、机密、秘密、公开
  - —高密级信息禁止流入低密级文档
- 应需可知（need-to-know）原则
  - —非等级（安全）类别：范畴
    - 对主客体所处部门的一种描述
  - —高密级用户未经允许，不能访问低密级文档

BLP强制访问控制模型

- 简单安全条件
  - ——一个主体可以读客体的条件
    - 仅当主体保密级别不低于客体的保密级别
    - 且主体安全级别中的非等级类别包含了客体安全级别中的全部非等级类别
- *-属性（星属性）
  - ——一个主体可以写另一个客体的的条件
    - 仅当主体保密级别不高于客体保密级别
    - 且主体安全级别中的非等级类别包含于客体安全级别中的非等级类别

BLP强制访问控制模型

- BLP本质上是一种信息流策略
  - —在系统中，信息可以被认为是从一个客体流到另一个客体
  - —客体间的信息流动可以看作是通过主体的访问（例如：读写操作）来实现
  - —如果系统中的所有访问都遵循BLP模型的两条规则
    - 信息将不会从高安全级别的客体流到低安全级别的客体

BLP的局限性

- 合法信息流从低安全级别主体/客体流向高安全级别的主体/客体
  - —逆向信息流被禁止
- 实际信息系统中，经常有逆向信息流的需求
  - —上级向下级发布通知、下达命令
  - —工程实现逆向信息流的一般方法
    - 划定可信主体，授予特权

BIBA安全策略模型

概述

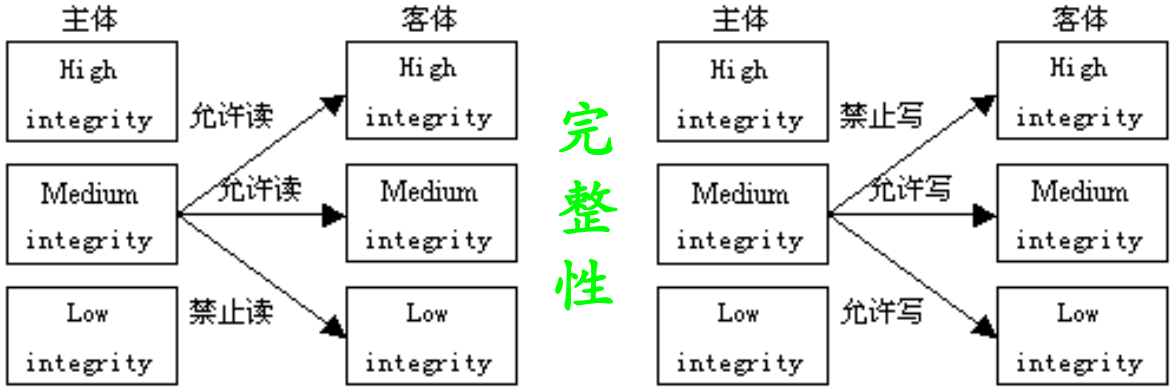
- Biba模型是一个针对完整性安全需求的模型
- 1977年Biba对系统的完整性提出了3种策略
  - —其中一种策略是BLP模型在数学上的对偶
    - BLP：上写下读 Biba：上读下写
- Biba模型引入了“信任”概念——完整性等级
  - —程序执行是影响完整性的一个重要行为

Biba模型与BLP模型的对比

BLP模型

上写下读

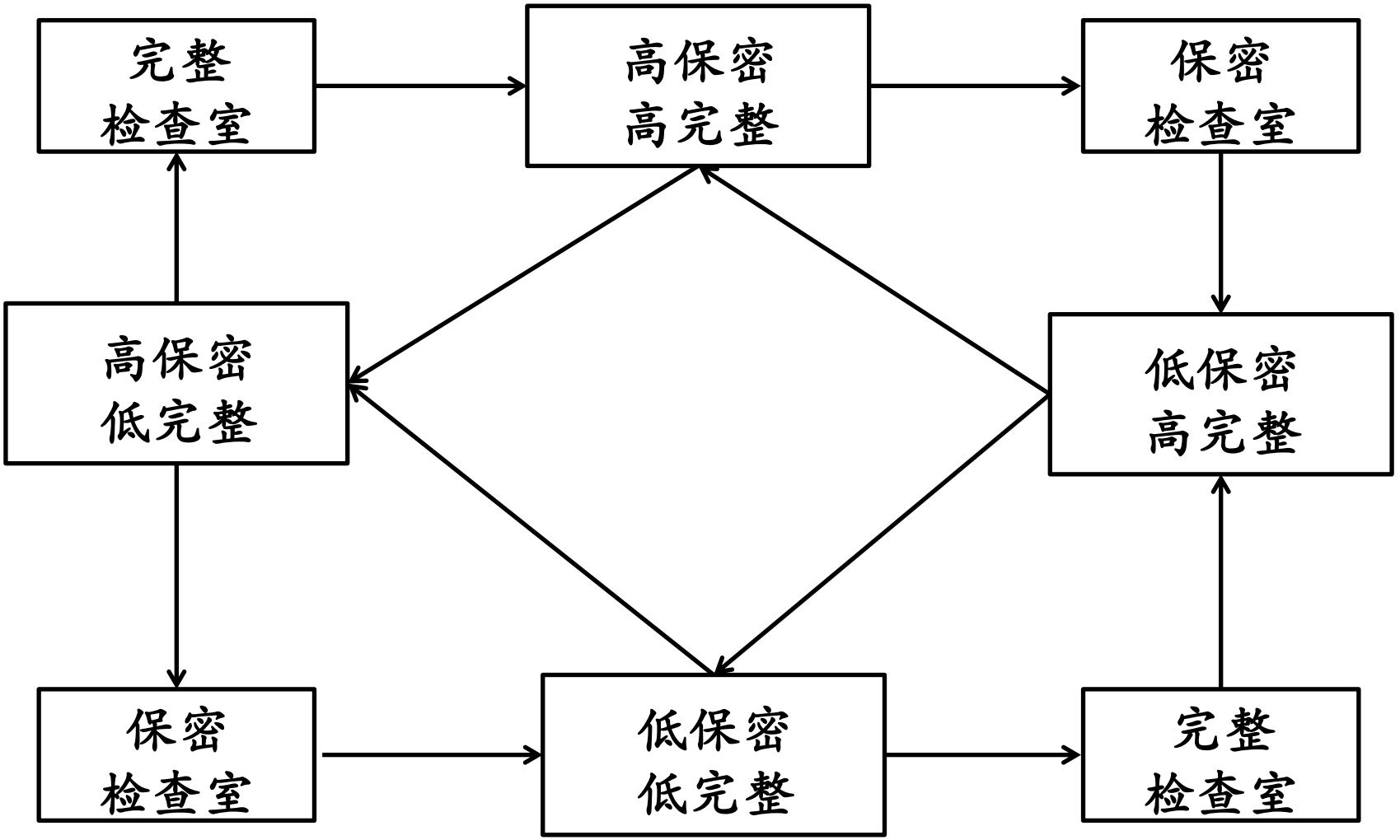
Biba模型



上读下写

二维安全策略模型

二维安全策略模型的信息流向



安全策略模型面临的问题和挑战

- 信息流安全理论和现有体系结构的融合问题
  - ——当前操作系统和软件工程并不是按照信息流策略设计的
- 信息流模型需要避免绝对的无干扰限制
  - ——安全假设过于严格，脱离现实
- 信息流模型需要能够解释和管理复杂的安全策略

本章内容提要

- 信息系统安全模型
- 安全操作系统
- 主流操作系统安全模型
- 主流操作系统安全机制

安全操作系统与TPM

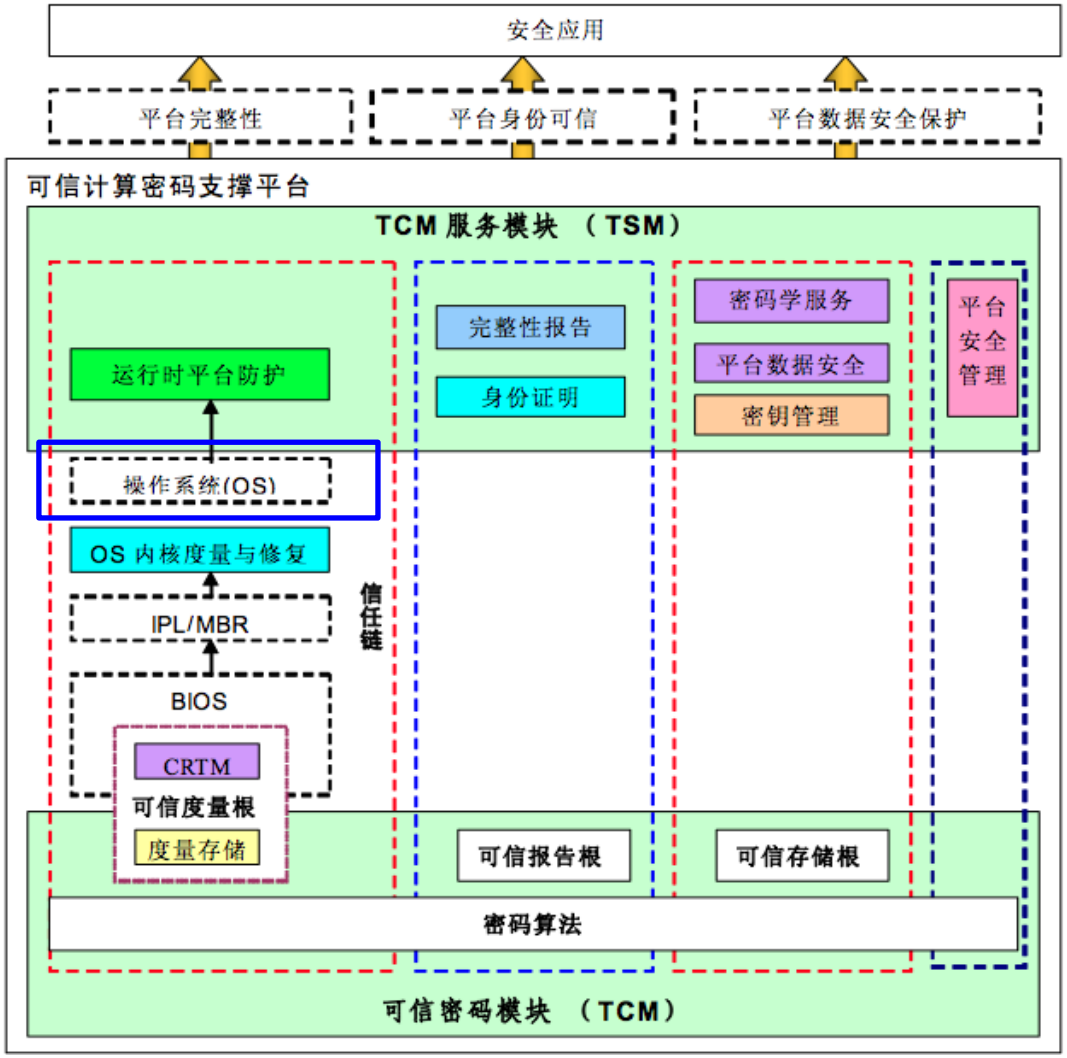
- TPM解决的是硬件可信问题

  —硬件信任根

- 安全操作系统解决的是软件可信问题

  —软件信任根

- 可信计算是安全操作系统的安全基石



安全操作系统简史——国际篇（1/2）

- 1967年，Adept-50
- 1969年，访问控制矩阵模型
- 1972年，引用监控机、引用验证机制、安全核、安全建模
- 1973年，隐蔽通道，BLP模型
- 1975年，信息保护机制的八条设计原则
- 1976年，操作系统保护的第一个基本理论——HRU理论
- 1979年，基于安全核的计算机安全系统设计方法，TCB的定义
- 典型开源安全操作系统
  —L4、MINIX、OpenBSD

TCSEC

- TCSEC，1983，美国国防部发表
  - —《可信计算机系统评估准则》
  - —计算机安全等级定义：4类7级
    - 无保护级：D
    - 自主保护级：C1、C2
    - 强制保护级：B1、B2、B3
    - 验证保护级：A1
  - —TCSEC的评估目标只涉及了保密性，而没有涉及完整性和可用性的评估
  - —于2000年被废止

安全操作系统简史——中国篇（2/2）

- GB 17859-1999《计算机信息系统安全保护等级划分准则》
  - —以TCSEC为基础
  - —五个安全等级
- 典型系统
  - —SUNIX、COSIXV2.0安全子系统、LIDS安全操作系统、SoftOS、安胜操作系统、麒麟操作系统等

安全操作系统基本概念

- 机密性
  - —敏感信息仅允许授权用户访问
    - 禁止所有未经授权的访问
- 完整性
  - —数据或资源的可信度
  - —实现完整性的两大机制：预防和检测
- 可用性
  - —信息或资源的预期使用能力
    - 如何区分破坏、滥用和超负荷?

访问控制策略

- 自主访问控制
  - —DAC: Discretionary Access Control
  - —基于用户身份
- 强制访问控制
  - —MAC: Mandatory Access Control
  - —基于操作系统的内置授权

自主访问控制

- 特点
  - —已授权主体可以访问客体
  - —非授权主体无法访问客体
  - —访问授权可以自主分配（授权和取消授权）
    - A可以访问文件α，则A可以授权B也能访问文件α
- 实现方式举例
  - —访问控制列表(ACL：Access Control List)
  - —访问控制矩阵
  - —面向过程的访问控制

访问控制矩阵

访问控制矩阵示例

某系统中有2个进程和2个文件

访问权限集合：{读、写、执行、追加、属主}

	文件A	文件B	进程A	进程B
进程A	读、写、属主	读	读、写、执行、属主	写
进程B	追加	写、属主	读	读、写、执行、属主

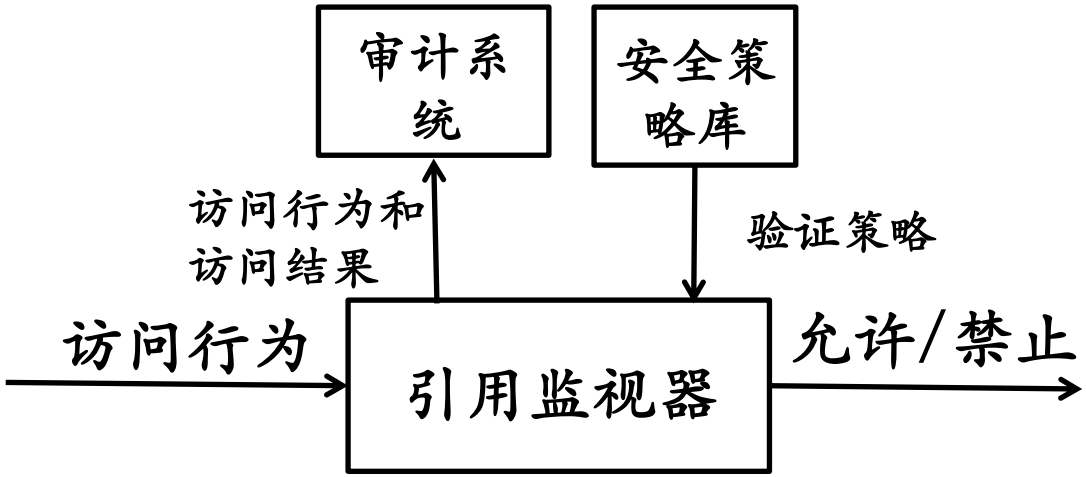
- **属主：绝大多数现代操作系统，属主权限的拥有主体可以对所拥有的权限自行分配**

强制访问控制

- 特点
  - —(操作)系统对访问主体和受控对象(客体)实行强制访问控制
  - —多级访问控制策略
  - —(操作)系统预先分配好主客体安全级别：安全标签
  - —主体访问客体时先进行安全级别属性比较，再决定访问主体能否访问该受控对象(客体)

引用监视器模型

- 引用监视器接收安全策略作为输入
  - —所有访问行为接收裁决
  - —所有访问行为和裁决结果写入审计记录
- 引用监视器依赖于验证机制
  - —独立机制并且具有自我防护或防篡改功能
  - —不可旁路
  - —足够小



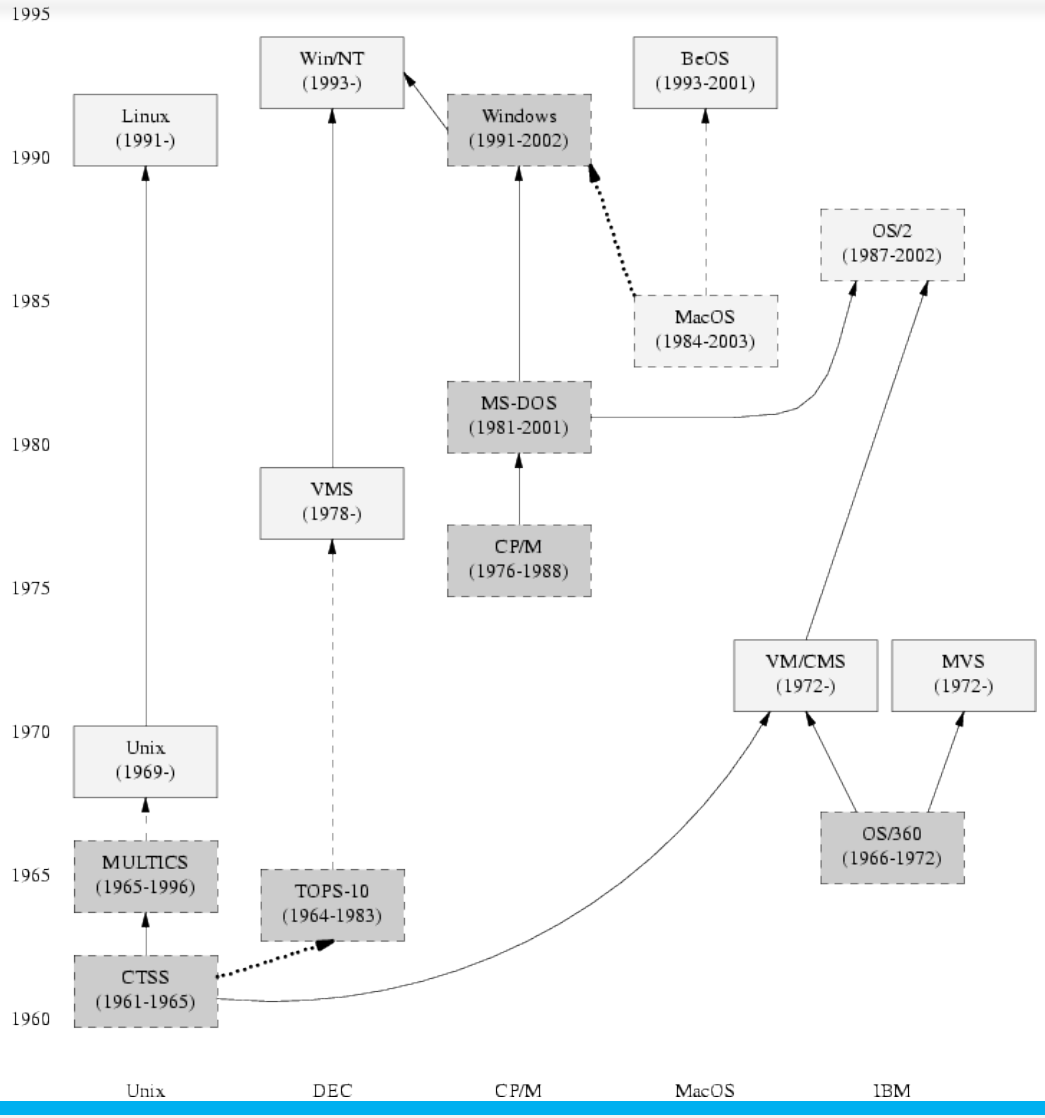
安全操作系统主要安全技术

- 身份鉴别
- 标志（数字标识）
- 审计
- 自主访问控制
- 强制访问控制
- 客体重用
- 可信路径
- 隐蔽通道分析
- 形式化分析与验证

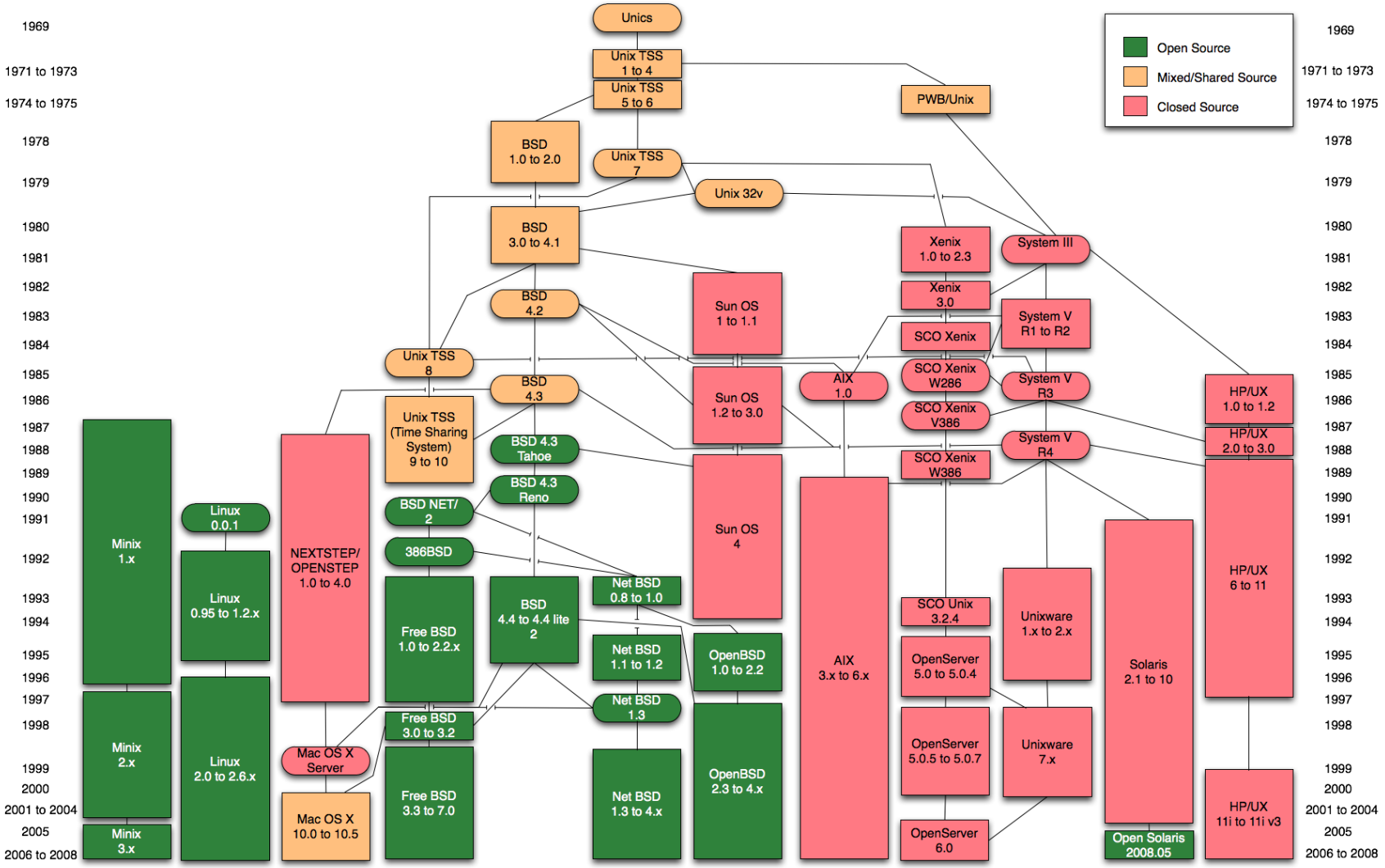
本章内容提要

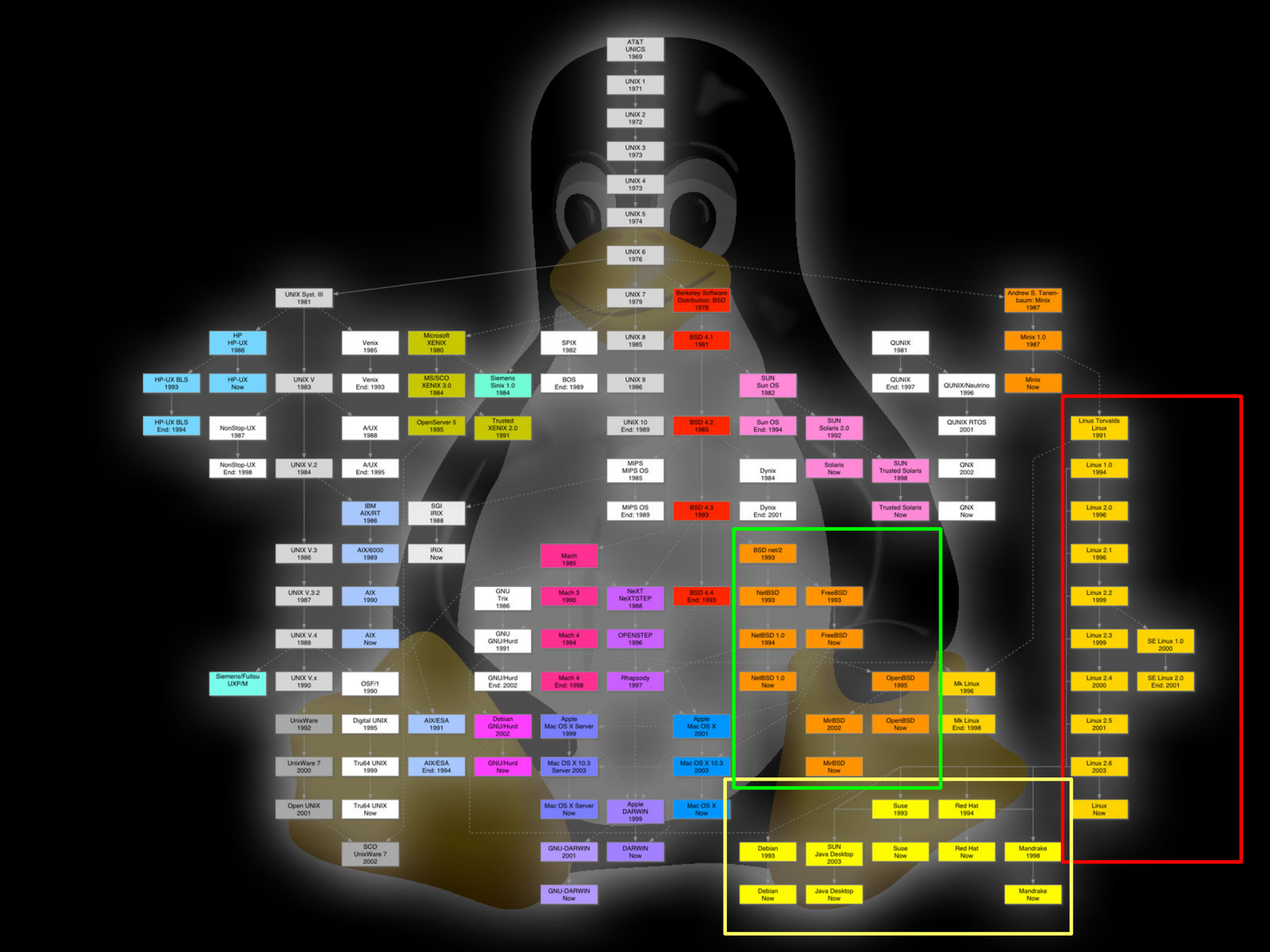
- 信息系统安全模型
- 安全操作系统
- 主流操作系统安全模型
- 主流操作系统安全机制

近代操作系统简史


1995
1990
1985
1980
1975
1970
1965
1960
Win/NT
(1993-)
BeOS
(1993-2001)
Linux
(1991-)
Windows
(1991-2002)
OS/2
(1987-2002)
MacOS
(1984-2003)
MS-DOS
(1981-2001)
VMS
(1978-)
CP/M
(1976-1988)
VM/CMS
(1972-)
MVS
(1972-)
Unix
(1969-)
OS/360
(1966-1972)
MULTICS
(1965-1996)
TOPS-10
(1964-1983)
CTSS
(1961-1965)
Unix
DEC
CP/M
MacOS
IBM

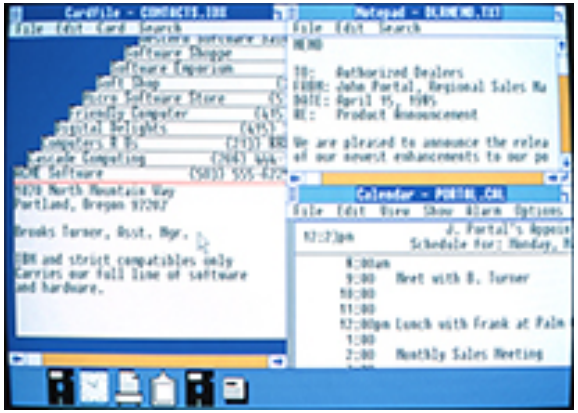
Unix系统简史——源代码授权协议角度

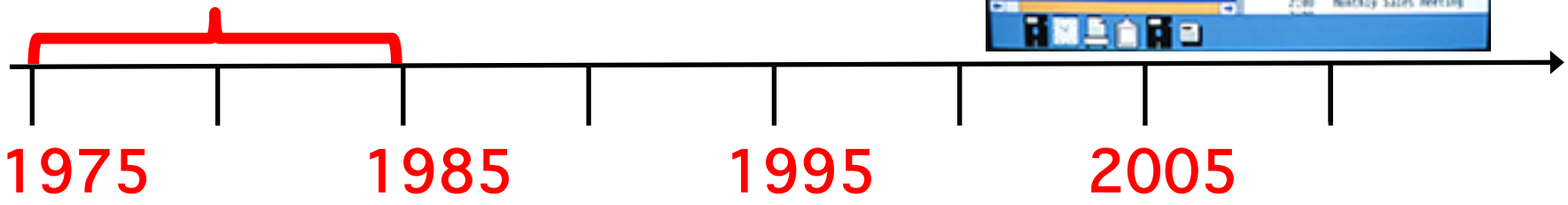



AT&T UNICS 1969
UNIX 1 1971
UNIX 2 1972
UNIX 3 1973
UNIX 4 1973
UNIX 5 1974
UNIX 6 1976
UNIX Syst. III 1981
UNIX 7 1979
Berkeley Software Distribution: BSD 1978
Andrew S. Tanen-baum: Minix 1987
HP HP-UX 1986
Venix 1985
Microsoft XENIX 1980
SPIX 1982
UNIX 8 1985
BSD 4.1 1981
QUNIX 1981
Minix 1.0 1987
HP-UX BLS 1993
HP-UX Now
UNIX V 1983
Venix End: 1993
MS/SCO XENIX 3.0 1984
Siemens Sinix 1.0 1984
BOS End: 1989
UNIX 9 1986
SUN Sun OS 1982
QUNIX End: 1997
QUNIX/Neutrino 1996
Minix Now
HP-UX BLS End: 1994
NonStop-UX 1987
A/UX 1988
OpenServer 5 1995
Trusted XENIX 2.0 1991
UNIX 10 End: 1989
BSD 4.2 1985
Sun OS End: 1994
SUN Solaris 2.0 1992
QUNIX RTOS 2001
Linus Torvalds Linux 1991
NonStop-UX End: 1998
UNIX V.2 1984
A/UX End: 1995
MIPS MIPS OS 1985
Dynix 1984
Solaris Now
SUN Trusted Solaris 1998
QNX 2002
Linux 1.0 1994
IBM AIX/RT 1986
SGI IRIX 1988
MIPS OS End: 1989
BSD 4.3 1993
Dynix End: 2001
Trusted Solaris Now
QNX Now
Linux 2.0 1996
UNIX V.3 1986
AIX/6000 1989
IRIX Now
Mach 1985
BSD net/2 1993
Linux 2.1 1996
UNIX V.3.2 1987
AIX 1990
GNU Trix 1986
Mach 3 1990
NeXT NeXTSTEP 1988
BSD 4.4 End: 1993
NetBSD 1993
FreeBSD 1993
Linux 2.2 1999
UNIX V.4 1988
AIX Now
GNU GNU/Hurd 1991
Mach 4 1994
OPENSTEP 1996
NetBSD 1.0 1994
FreeBSD Now
Linux 2.3 1999
SE Linux 1.0 2000
Siemens/Fuitsu UXP/M
UNIX V.x 1990
OSF/1 1990
GNU/Hurd End: 2002
Mach 4 End: 1998
Rhapsody 1997
NetBSD 1.0 Now
OpenBSD 1995
Mk Linux 1996
Linux 2.4 2000
SE Linux 2.0 End: 2001
UnixWare 1992
Digital UNIX 1995
AIX/ESA 1991
Debian GNU/Hurd 2002
Apple Mac OS X Server 1999
Apple Mac OS X 2001
MirBSD 2002
OpenBSD Now
Mk Linux End: 1998
Linux 2.5 2001
UnixWare 7 2000
Tru64 UNIX 1999
AIX/ESA End: 1994
GNU/Hurd Now
Mac OS X 10.3 Server 2003
Mac OS X 10.3 2003
MirBSD Now
Linux 2.6 2003
Open UNIX 2001
Tru64 UNIX Now
Mac OS X Server Now
Apple DARWIN 1999
Mac OS X Now
Suse 1993
Red Hat 1994
Linux Now
SCO UnixWare 7 2002
GNU-DARWIN 2001
DARWIN Now
Debian 1993
SUN Java Desktop 2003
Suse Now
Red Hat Now
Mandrake 1998
GNU-DARWIN Now
Debian Now
Java Desktop Now
Mandrake Now

Windows的历史（1/4）

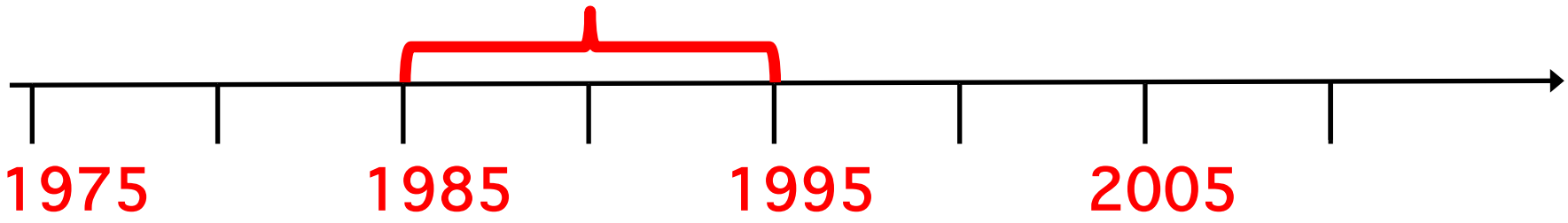
- Microsoft的起步
  - —创始人：Bill Gates & Paul Allen
  - —1980年6月：Steve Ballmer受雇负责公司运营
  - —1981年推出运行MS-DOS的IBM PC
    - C:\ 开始流行
- Windows 1.0
  - —1982-1985 （研发历史3年）





Windows的历史（2/4）

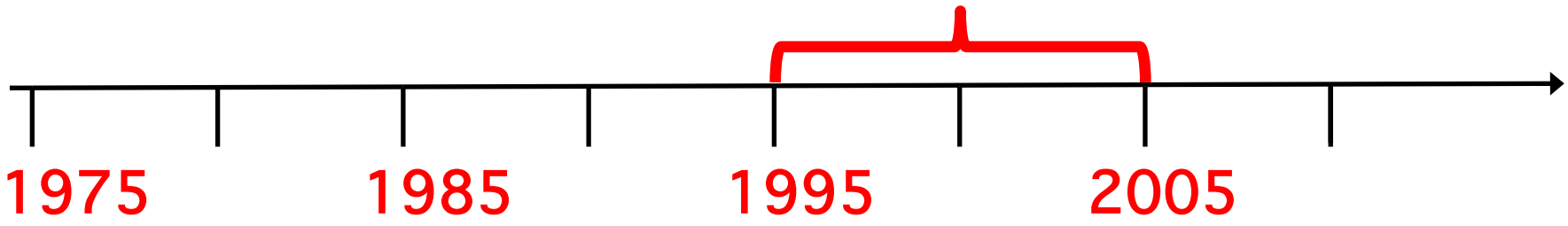
- 1987 – 1992：Windows 2.0 – 2.11
  —窗口更多，速度更快
- 1990 – 1994：Windows 3.0 – Windows NT
  —实现图形效果
- 1995 – 2001: Windows 95
  —个人电脑和 Internet 蓬勃发展



Windows的历史（3/4）

- 1998 – 2000： Windows 98, Windows 2000, Windows Me
- 2001 – 2005： Windows XP

—稳定、易用且快速



Windows的历史（4/4）

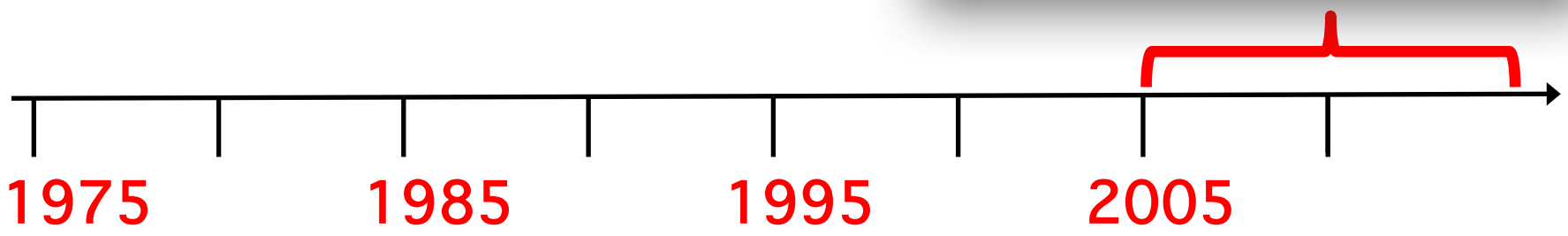
- 2006 – 2008: Windows Vista

  ——安全智能

- 2009 – 2012： Windows 7

- 2012.10.23 ： Windows 8

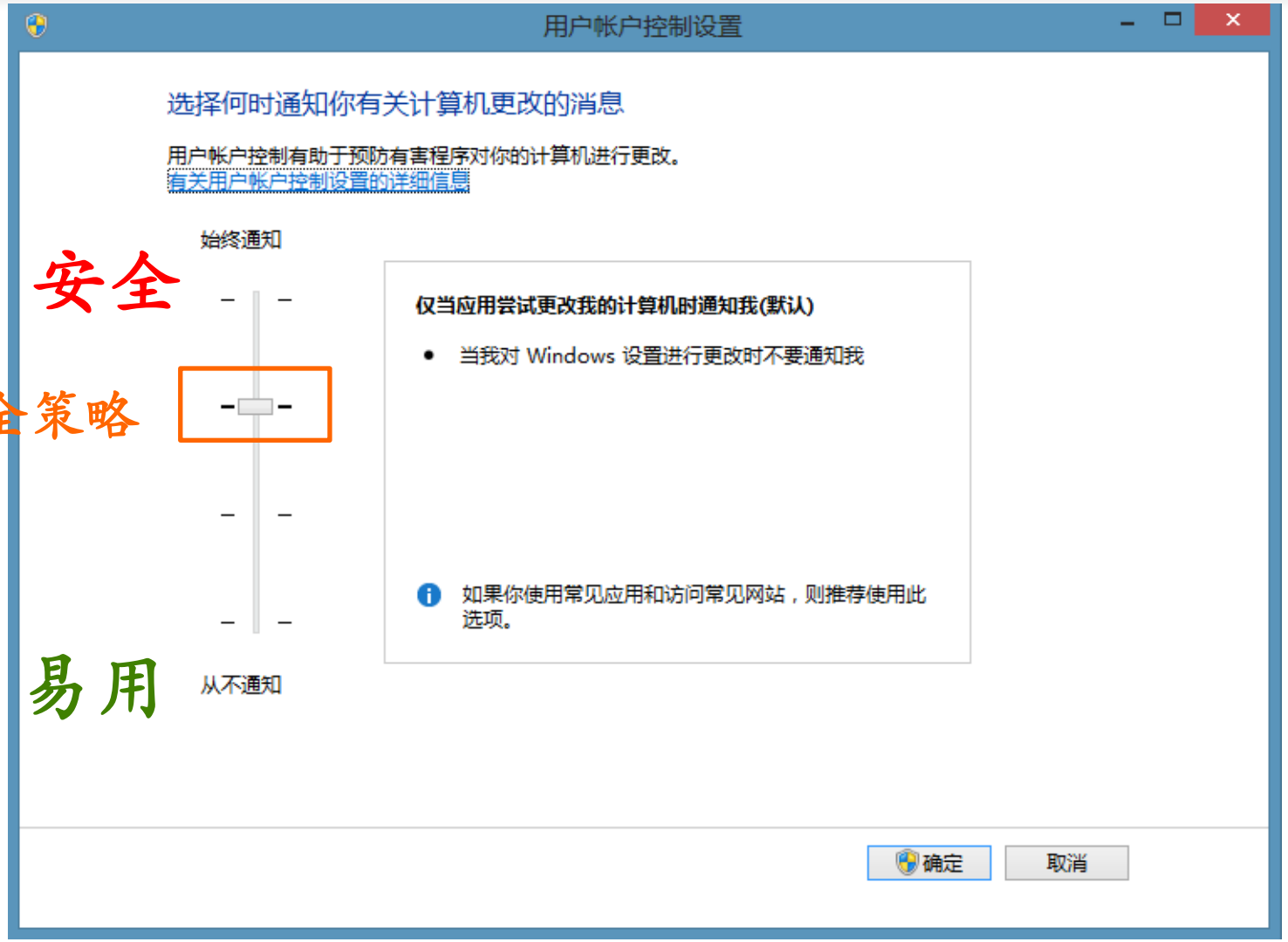
  ——移动化



主流操作系统的安全策略设计

- 多帐户（用户）隔离
  - —最小化授权
    - – 特权隔离
- 默认安全
  - —默认安装、默认配置的安全合规性
- 数据加密
  - —存储和处理

安全与易用性之间的关系



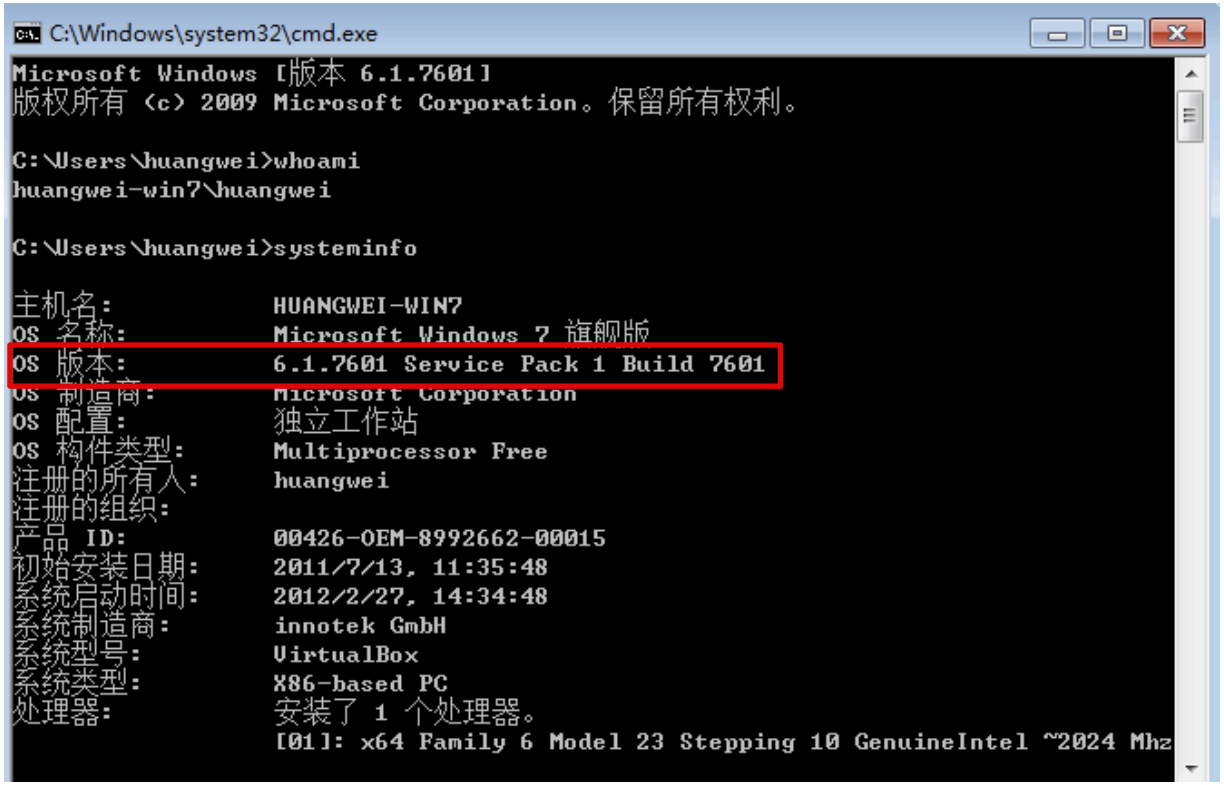
本章内容提要

- 信息系统安全模型
- 安全操作系统
- 主流操作系统安全模型
- **主流操作系统安全机制**

WINDOWS

Windows系统的访问控制机制

- 查看当前Windows系统版本



从组策略编辑器的更新看Windows安全机制的演进

Windows XP

Windows 8

访问控制机制

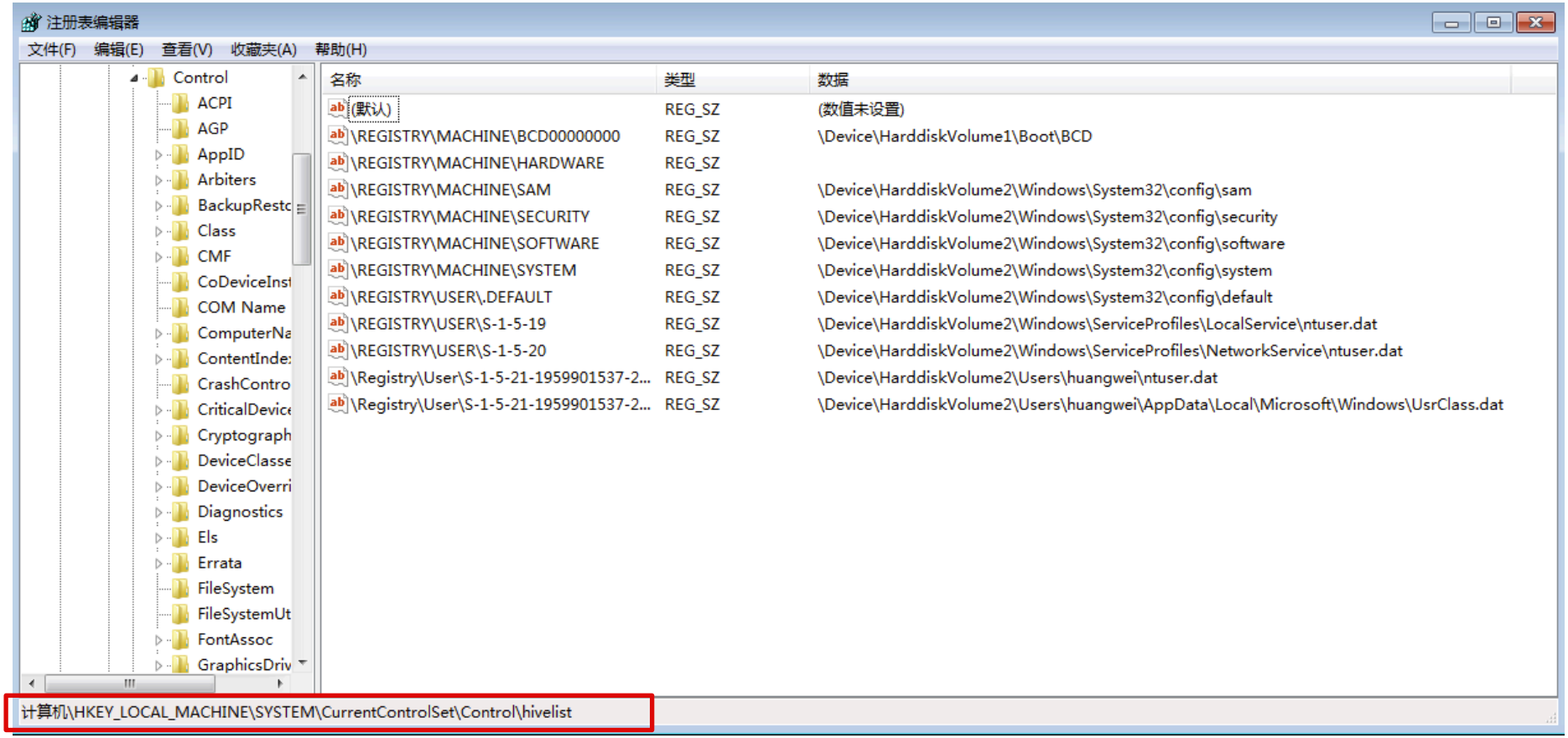
- 主体
  - —帐户/用户组
- 客体
  - —文件/文件夹/注册表
- 访问控制策略
  - —DACL
  - —组策略（编辑器）
    - – gpedit.msc

“隐秘”的注册表机制

- Windows注册表的作用
  - —Windows配置和控制方面关键角色
  - —系统全局配置的存储仓库
  - —每个用户配置信息的存储仓库
- 注册表管理工具
  - —regedit.exe
- Windows系统攻防必争之地
  - —恶意代码实现随系统启动时加载
  - —恶意代码拦截和篡改系统关键调用/文件关联/系统默认设置等

“隐秘”的注册表机制

注册表的存储



HKLM\SYSTEM\CurrentControlSet\Control\hivelist

系统启动时加载机制

- **系统启动时加载相关的注册表项（win7）**
  - HKLM\System\CurrentControlSet\Control\Terminal Server\Wds\rdpwd\StartupPrograms
  - HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\Userinit
  - HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\VmApplet
  - HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\Shell
  - HKLM\SYSTEM\CurrentControlSet\Control\SafeBoot\AlternateShell
  - HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
  - HKLM\SOFTWARE\Microsoft\Active Setup\Installed Components
  - HKLM\Software\Microsoft\Windows NT\CurrentVersion\Windows\IconServiceLib
  - HKCU\Software\Microsoft\Windows\CurrentVersion\Run
  - …

- **系统启动时加载的其他方法**
  - **计划任务**
  - **开始菜单->启动**
    - Windows NT 6.1, 6.0：%SystemDrive%\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup\
    - Windows NT 5.2, 5.1, 5,0：%SystemDrive%\Documents And Settings\All Users\Start Menu\Programs\StartUp\

恶意代码绕过Windows安全机制实现启动(1/2)

- 开机自动运行
  - —系统启动时加载
- 随其他应用程序启动时后台加载
  - —修改默认的文件关联
    - 文件后缀名注册的默认打开应用程序
    - URI篡改注册

      thunder:// tencent:// telnet://
  - —PE文件执行劫持

    HKLM\Software\Microsoft\Windows NT\CurrentVersion\Image File Execution Options

恶意代码绕过Windows安全机制实现启动(2/2)

- 应用程序安装劫持
  - —系统下载目录中放置被篡改的msiexec.exe
- 快捷方式的启动参数篡改
- 应用程序恶意捆绑

微软的SysInernals工具集

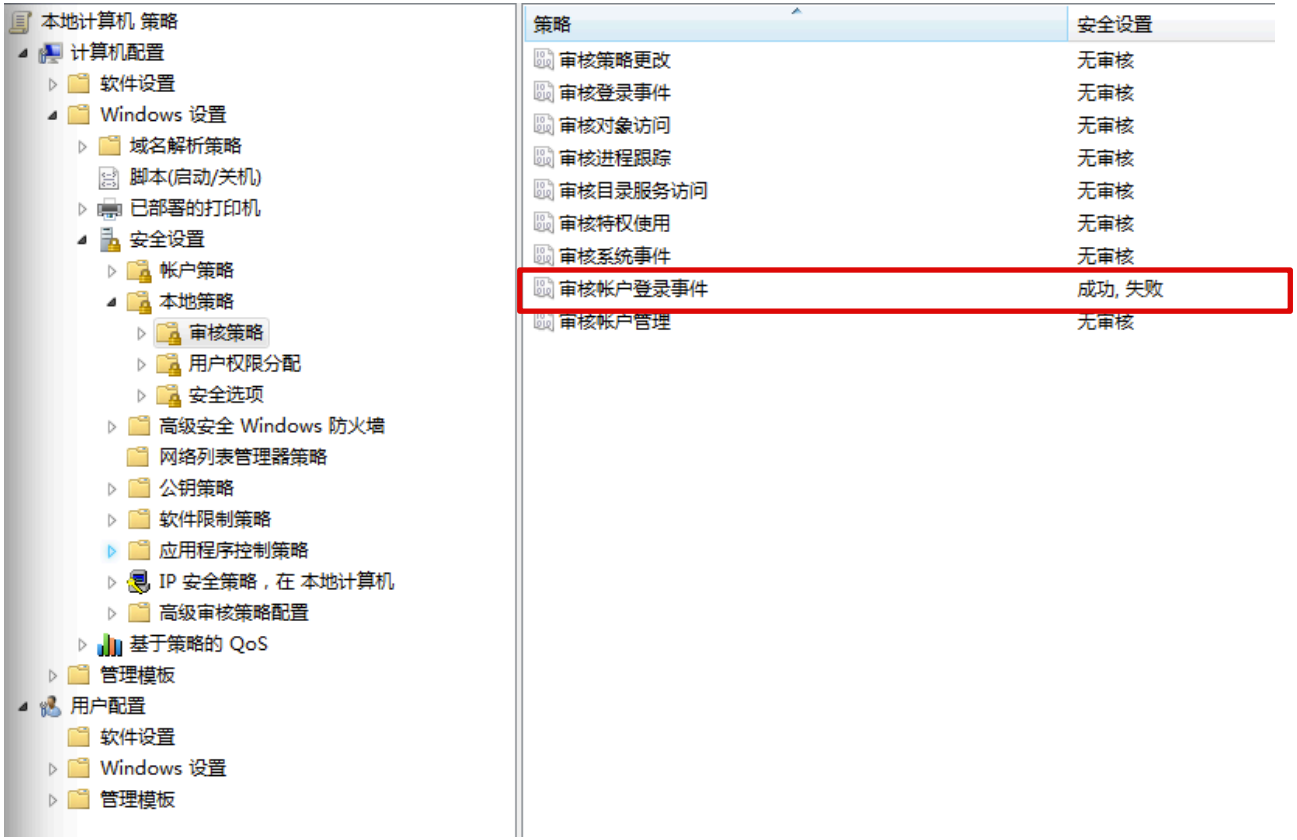
- 文件和磁盘工具
- 网络工具
- 进程安全
- 安全工具
- 系统信息工具
- 杂项工具

下载链接：http://technet.microsoft.com/en-us/sysinternals/

内置的安全审计

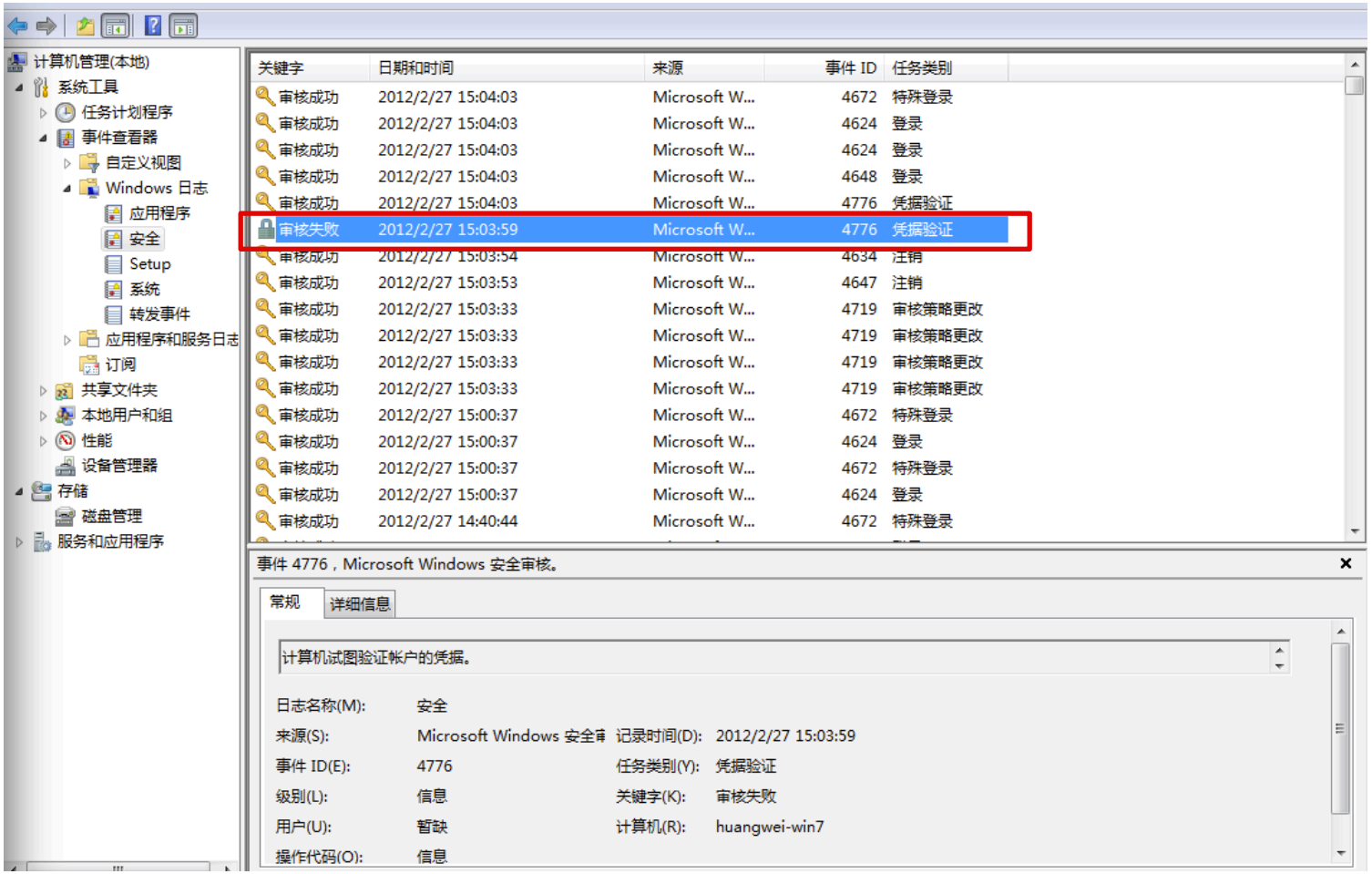
- 审计策略配置

  —gpedit.msc



内置的安全审计

- 审计日志查看



Windows的其他系统安全机制

- Windows安全中心
  - —防火墙
  - —自动更新
  - —防病毒软件
- Internet选项
- DEP：数据执行保护
- ASLR：内存空间随机化

Windows的其他系统安全机制

- UAC（用户帐户控制）
  - —它要求用户在执行可能会影响计算机运行的操作或执行更改影响其他用户的设置的操作之前，提供权限（确认）或管理员密码
- IPSec
  - —IP加密和验证策略
  - —本地安全配置/IP安全策略
- WFP（Windows文件保护机制）
  - —防止Windows系统文件被恶意替换
  - —驱动程序签名及验证机制

Windows的其他系统安全机制

- EFS（加密文件系统）
  - —Windows XP
  - —系统级文件/文件夹加密（防止物理硬盘被盗后的数据泄密）
  - —一旦加密密钥丢失则无法恢复和访问数据

Windows的其他系统安全机制

- VHD
  - —Microsoft Virtual Hard Disk format
  - —微软专有的虚拟磁盘格式
  - —类似虚拟光驱的使用方法
- BitLocker
  - —Windows Vista/7
  - —磁盘数据加密

EFS VS. BitLocker

EFS	BitLocker
用于对个人文件和文件夹逐个加密，它不对某个驱动器的整个内容进行加密	用于对操作系统驱动器、固定数据驱动器和可移动数据驱动器上的所有个人文件和系统文件进行加密
将根据与其关联的用户帐户来加密文件。如果计算机具有多个用户或组，则每个用户或组都可以单独加密各自的文件	并不依赖于与文件相关联的各个用户帐户
并不需要（或不使用）任何特殊硬件	使用受信任的平台模块(TPM)，该模块是许多计算机中一种支持高级安全功能的特殊微芯片，用于加密操作系统驱动器
不必具有管理员身份	必须是管理员才能在安装了 Windows 的驱动器和固定数据驱动器上打开或关闭 BitLocker 加密

ANDROID（演示）

演示提纲

- 多用户系统
  - —进程权限隔离机制
  - —最小化授权
- 软件的代码签名检测机制
- USB调试模式
  - —普通用户，务必禁止USB调试模式
    - 一旦手机被root，手机上的所有机密数据可被随意访问
- 重要数据避免存储在SD卡上
  - —不受Android文件系统访问权限控制

IOS（演示）

演示提纲

- 普通用户可体验到的iOS区别于Android的几点重要安全机制
  - —禁止第三方程序访问所有通话历史（iOS 4+）
  - —禁止第三方程序访问短信（iOS 5+）
  - —第三方程序访问通讯录、日历、照片需要用户明确授权（iOS 6+）
  - —GPS定位功能需要用户明确授权（iOS 4+）
  - —限制后台联网程序种类（音乐、VoIP，iOS 4+）
  - —限制后台静默运行程序（音乐、GPS 、VoIP 、消息推送守护程序以及周边配件附属的程序，iOS 4+）
- 越狱系统的默认SSH密码问题
  - —同一局域网/WIFI接入点用户可以借助SSH远程访问
- 软件的代码签名检测机制
  - —越狱手机有被植入高危恶意代码的风险
    - 自签名+高权限代码

Android系统安全使用科普

- Android
  - —尽可能只安装来自Google Play的APP
    - 国内安卓手机要避免从小市场安装APP
  - —关闭USB调试模式
  - —Root后手机安装Super User（授权管理）
  - —安装系统安全软件
    - 精细化关闭应用程序的非必要授权请求

iOS系统安全使用科普

- iOS
  - ——避免越狱
  - ——如果越狱
    - 关闭SSH或修改默认密码
    - 避免添加小众cydia源
    - 避免安装小众APP

参考文献

① 《Unix操作系统发展大事记》http://www.techcn.com.cn/index.php?doc-view-112413

② John R. Michener，理解 Windows 文件和注册表权限 http://msdn.microsoft.com/en-us/magazine/cc982153.aspx

③ Windows的历史 http://windows.microsoft.com/zh-CN/windows/history

课后小实验

- 学习使用Windows的组策略编辑器，实现以下安全机制
  - —口令连续输入错误3次，锁定帐户30分钟或由管理员解锁
  - —禁止未登录系统用户关闭计算机