操作系统安全

第一部分 引言

中国科学院大学
网络空间安全学院
2017.03.09

目录

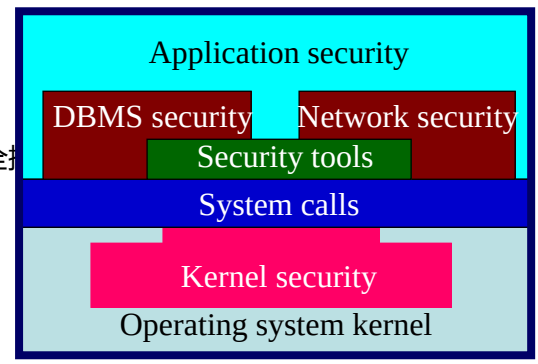
1. 操作系统安全的意义
2. 操作系统安全威胁
3. 操作系统安全发展趋势
4. 课程设计与要求

目录

1. 操作系统安全的意义
2. 操作系统安全威胁
3. 操作系统安全发展趋势
4. 课程设计与要求

安全为什么从OS开始(1)

- OS
  - 最底层软件
  - 完全的硬件访问
  - 负责资源分配、共享和保护
- 如果 OS 有问题 ⇒ 上层的安全



安全为什么从OS开始(2)

"Any and all security features implemented in application software, and any and all security application programs (including firewalls, intrusion detection systems, authentication program, etc.) can be **rendered useless** and of no protective effect **by an attack** which results in **seizure** of control of the computer's **operating system**"

- Randall Sandone, Argus Systems Group, Inc.

"It doesn't matter what else you do if it's all built on untrusted operating systems"

- Steve Kent, Defense Science Board, 5/2000

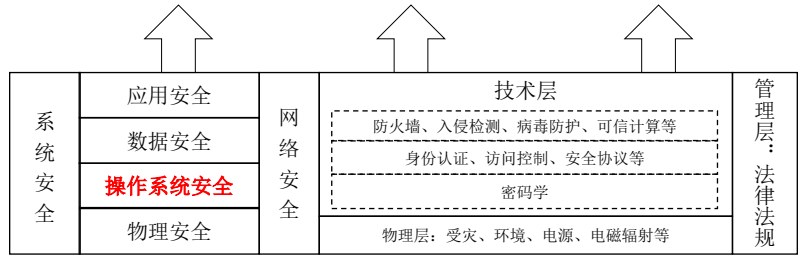
"Current security efforts suffer from the flawed assumption that adequate security can be provided in applications with the existing security mechanisms of mainstream operating systems. In reality, the **need for secure operating systems is growing** in today's computing environment due to substantial increases in connectivity and data sharing. The **threats** posed by the modern computing environment **cannot be addressed without secure operating systems**. Any security effort which ignores this fact can only result in a 'fortress built upon sand'." - NSA Operating System Security Paper, NISSC, October 1998

操作系统安全的重要性

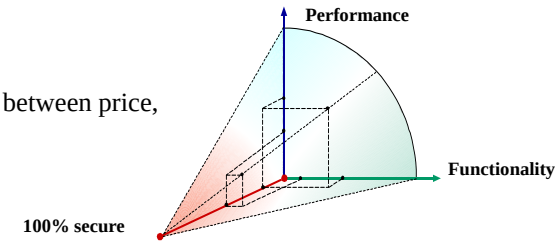
- 操作系统安全是整个系统安全的基础。
- 任何想象中的、脱离操作系统的应用软件的高安全性，就如同幻想在沙滩上建立坚不可摧的堡垒一样，毫无根基可言。
- 没有安全的操作系统支持，网络安全也毫无根基可言。

构建整个信息系统的安全

安全目标： 机密性、完整性、可用性



Find the balanced solution between price, function,and performance



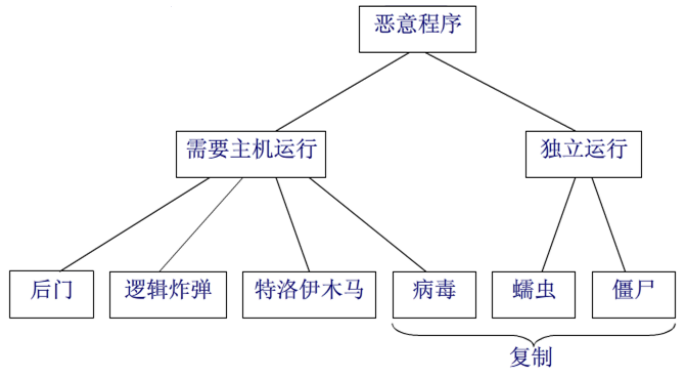
目录

1. 操作系统安全的意义
2. 操作系统安全威胁
3. 操作系统安全发展趋势
4. 课程设计与要求

操作系统面临的安全威胁

恶意软件分类

- 对计算机系统来说，最复杂的威胁是由那些利用计算机系统漏洞的程序带来的。对这类威胁的一般术语是恶意软件（Malware）。
- Malware是专门设计用来制造破坏或用尽目标计算机资源的软件，它尝尝隐藏在合法软件中或伪装成合法软件。
- 在某些情况下， 它通过电子邮件或感染的软盘、U盘将自己传播到其他计算机中。



问题

- 大家遇到的操作系统安全威胁是什么？

Rootkit攻击

- Rootkit源于UNIX系统中的超级用户帐号，UNIX系统是Rootkit工具最初的攻击目标。获取系统管理员权限。
- Rootkit是特洛伊木马后门工具，通过修改现有的操作系统软件，使攻击者获得访问权并隐藏在计算机中
- 关键：隐藏攻击者在系统中的存在，其包括多种掩饰攻击者在系统中存在的功能：如进程，文件，注册表，服务，端口等隐藏

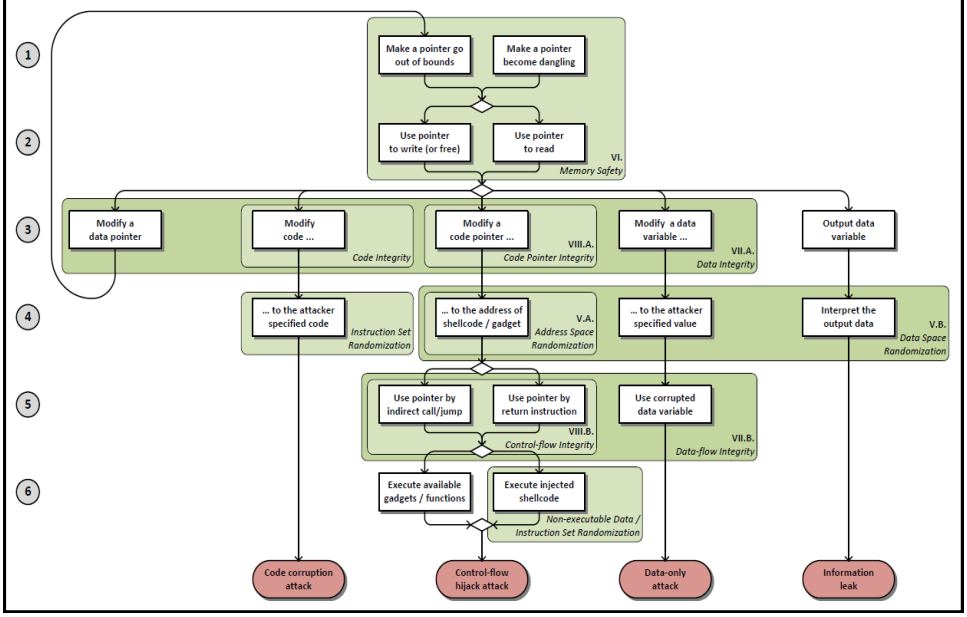
Rootkit攻击

- 按操作系统分类
  - Unix RootKit
  - Windows Rootkit
- 按资源层次分类
  - 用户级Rootkit：不深入系统内核，通常在用户层进行相关操作
  - 内核级Rootkit：深入系统内核，改变系统内核数据结构，控制内核本身
    - 修改现有的操作系统软件（内核本身），从而使攻击者获得一台计算机的访问权并潜伏在其中，比用户模式RootKit更彻底、更高效
    - LRK
    - URK
  - 硬件级

Rootkit攻击：常见功能

- 文件和目录隐藏
- 进程、服务、注册表隐藏
- 网络端口隐藏
- 混合模式隐藏（隐藏网络接口混合状态）
- 执行改变方向（内核后门）
- 设备截取和控制
  - 如底层键盘截获
- 日志擦除

内存攻击



系统漏洞

- 定义
  - “漏洞”一词的本义是指小孔或缝隙，引申为用来表达说话、做事存在不严密的地方。
  - 在计算机系统中，漏洞特指系统中存在的弱点或缺陷，也叫系统脆弱性。
- 由来
  - 范围：计算机系统硬件、软件、协议层面。
  - 设计与实现过程中或系统安全策略上存在的缺陷和不足。
- 危害：破坏系统的安全性
  - 非法用户可利用漏洞获得计算机系统的权限。
  - 在未经授权的情况下访问或提高其访问权限。

漏洞的分类

- 按照漏洞的形成原因分
  - 程序逻辑结构漏洞
  - 程序设计错误漏洞
  - 开放式协议造成的漏洞
  - 人为因素造成的漏洞。

分类说明

- **程序逻辑结构漏洞有可能是程序员在编写程序时，因为程序的逻辑设计不合理或者错误而造成的。**这类漏洞最典型的例子要数微软的Windows 2000用户登录的中文输入法漏洞。非授权人员可以通过登录界面的输入法的帮助文件绕过Windows的用户名和密码验证而取得计算机的最高访问权限。这类漏洞也有可能使合法的程序用途被黑客利用去做不正当的事。
- **程序设计错误漏洞是程序员在编写程序时由于技术上的疏忽而造成的。**这类漏洞最典型的例子是缓冲区溢出漏洞，它也是被黑客利用得最多的一种类型的漏洞。
- **开放式协议造成的漏洞是因为在互联网上用户之间的通信普遍采用TCP/IP协议。**TCP/IP协议的最初设计者在设计通信协议时只考虑到了协议的实用性，而没有考虑到协议的安全性，所以在TCP/IP协议中存在着很多漏洞。比如说，利用TCP/IP协议的开放性和透明性嗅探网络数据包，窃取数据包里面的用户口令和密码等信息; TCP协议三次握手的潜在缺陷导致的拒绝服务攻击等。
- **人为因素造成的漏洞可能是整个网络系统中存在的最大安全隐患。**网络管理员或者网络用户都拥有相应的权限，他们利用这些权限进行非法操作是可能的，隐患是存在的。如操作口令被泄露、磁盘上的机密文件被人利用及未将临时文件删除导致重要信息被窃取，这些都可能使内部网络遭受严重破坏。

漏洞的分类

- 按照漏洞被人掌握的情况分
  - 已知漏洞
  - 未知漏洞
  - 0day漏洞

分类说明

- **已知漏洞是指已经被人们发现，并被人们广为传播的公开漏洞。**这类漏洞的特点是漏洞形成的原因和利用方法已经被众多的安全组织、黑客和黑客组织所掌握。安全组织或厂商按照公布的漏洞形成原因和利用方法，在他们的安全防护产品或安全服务项目中加入针对相应类型漏洞的防护方法。黑客和黑客组织利用公布的漏洞形成原因，写出专门的具有针对性的漏洞利用程序文件，并能绕过安全防护软件。

- **未知的漏洞则是指那些已经存在但还没有被发现的漏洞。**这类漏洞的特征是虽然它们没有被发现，但它们在客观上已经存在了，它们带给计算机网络安全的威胁是隐蔽性的，如果它们哪一天被黑客有意或无意地找出来后，就会对计算机网络安全构成巨大的威胁。所以软件开发商、安全组织、黑客和黑客组织都在努力地发现漏洞，可以说谁先发现了漏洞，谁就可以掌握主动权。

- **0day漏洞是指已经被发掘出来，但还没有大范围传播开的漏洞。**也就是说，这类漏洞有可能掌握在极少数人的手里。黑客有可能在这类漏洞的信息还没有大范围地被传播开的时候，利用这段时间差攻击他们想要攻击的目标机器，因为绝大多数用户还没有获取到相关的漏洞信息，也无从防御，所以黑客要想得手还是很容易的。

问题

- Bug Vs. 漏洞

APT简述

- APT即高级可持续性威胁（Advanced Persistent Threat），也称为定向威胁，
- 是指某组织对某一特定对象展开的持续有效的攻击活动和威胁。
- 这种攻击活动具有极强的隐蔽性和针对性，通常会运用受感染的各种介质、供应链和社会工程学等多种手段实施先进的、持久的且有效的威胁和攻击。

APT攻击的A分析

- 技术上的高级
  - 0DAY漏洞
  - 0DAY木马
  - 通道加密
- 投入上的高级
  - 全面信息的收集与获取
  - 针对的目标和工作分工
  - 多种手段的结合：社工+物理

APT特征

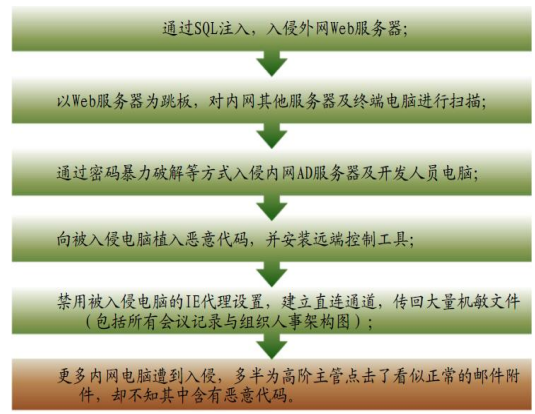
- APT不是一种新的攻击手法，因此也无法通过阻止一次攻击就让问题消失。
- 黑客个人的行为一般不能构成APT攻击，因为通常情况下没有足够的资源来开展这种先进且复杂的攻击活动。
- 这种攻击不会追求短期的收益或单纯的破坏，而是以步步为营的渗透入侵策略，低调隐蔽的攻击每一个特定目标，不做其他多于的活动来打草惊蛇。

案列：极光行动（2009-2010）

- 极光行动（Operation Aurora）是2009年12月中旬可能源自中国的一场网络攻击，其名称“Aurora”（意为极光、欧若拉）来自攻击者电脑上恶意文件所在路径的一部分。
- 2010年1月12日，Google在它的官方博客上披露了遭到该攻击的时间。此外还有20多家公司也遭受了类似的攻击（部分来源显示超过34家）

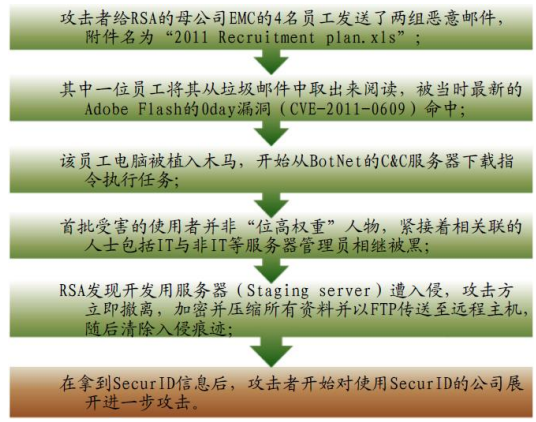
案例：夜龙攻击（2007-2011）

- 据美国《华尔街日报》2011.2.10报道，美国网络安全公司McAfee发表报告称，5家西方跨国能源公司遭到“来自中国”的黑客“有组织、隐蔽、有针对性”的攻击。
- 超过千兆字节的敏感文件被窃，包括油气田操作的机密信息、项目融资与投标文件等。
- McAfee的报告称这场网络间谍行动代号为“夜龙”（Night Dragon），最早可能开始于2007年，目前攻击行动仍在持续。



案例：RSA SecurID窃取攻击（2011）

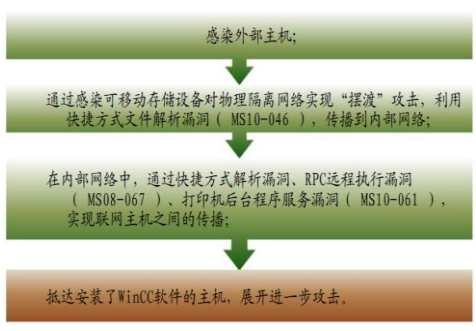
- 2011年3月，EMC公司下属的RSA公司遭受入侵，部分SecurID技术及客户资料被窃取。其后果导致很多使用SecurID作为认证凭据建立VPN网络的公司受到攻击，重要资料被窃取。

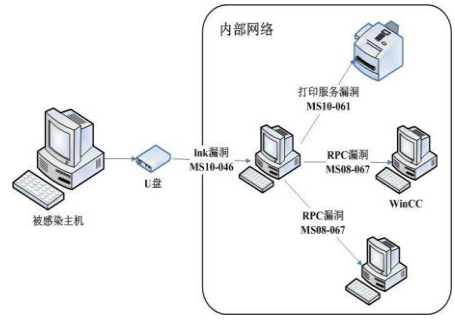


案例：震网攻击（2010）

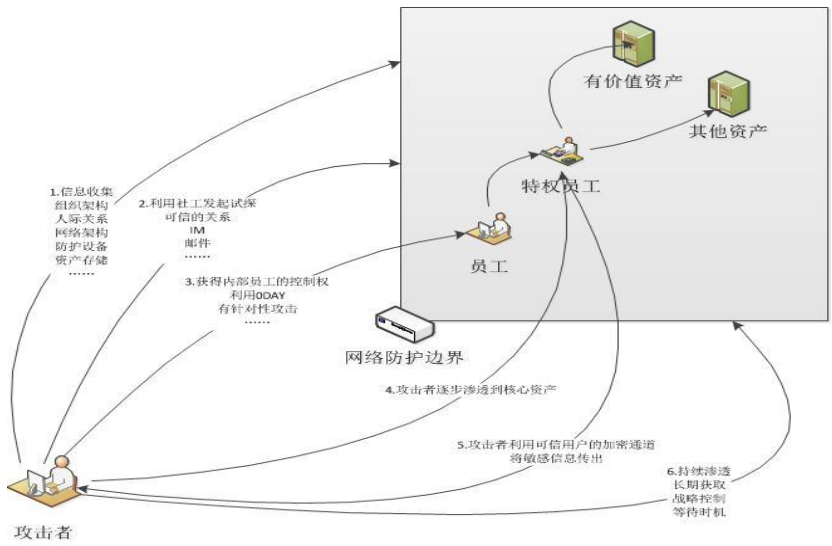
- 超级工厂病毒（Stuxnet）在2010年7月开始爆发。它利用了微软操作系统中至少4个漏洞，其中有3个全新的0day漏洞，为衍生的驱动程序使用有效的数字签名，通过一套完整的入侵和传播流程，突破工业专用局域网的物理限制，利用WinCC系统的2个漏洞，对其开展攻击。

- 它是第一个直接破坏现实世界中工业基础设施的恶意代码。据赛门铁克公司的统计，目前全球已有约45000个网络被该蠕虫感染，其中60%的受害主机位于伊朗境内。伊朗政府已经确认该国的布什尔核电站遭到Stuxnet的攻击。





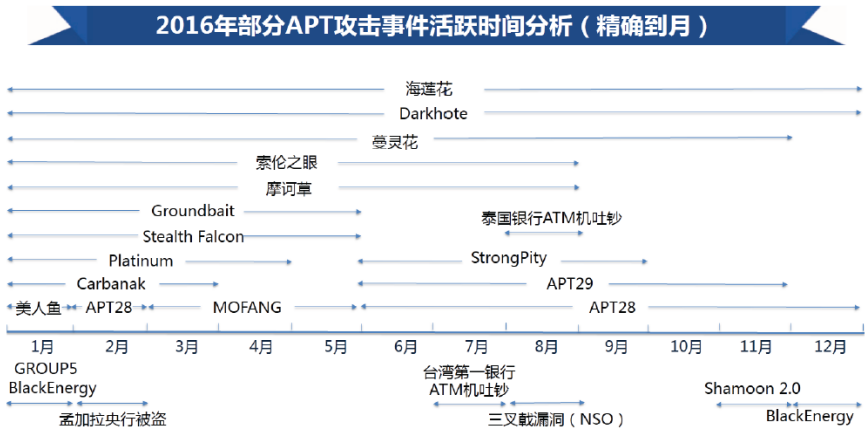
APT攻击步骤



APT攻击目标（2015-2016）

- 针对工业系统的破坏
  - 乌克兰大规模停电事件
  - 伊斯兰教大赦之夜Shamoon2.0攻击
- 针对金融系统的攻击
  - 孟加拉国央行被盗事件
  - 台湾第一银行ATM机吐钞事件
- 针对地缘政治的影响
  - 美国大选DNC邮件泄露事件
  - 方程式组织工具的泄露事件-可能首要针对中国

活跃的APT攻击事件



危险的网络军火交易

- 网络军火分类
  - 专用木马程序及配套的控制工具
  - 安全漏洞和漏洞的利用工具
- 网络军火商
  - 意大利的Hacking Team
  - 英国的Gamma
  - 以色列的NSO

APT攻击的特点与趋势

- 网络空间已经成为大国博弈的新战场
  - 如果说震网病毒事件、乌克兰停电事件表现出了“弱小国家”的基础设施在面临网络空间非常规打击时的脆弱性，那么2016年DNC邮件泄漏直接影响美国大选结果的事件，则充分说明：即便是当今世界唯一的超级大国，在网络攻击面前也同样脆弱，其政治、经济、社会心态等各个方面都有可能受到网络攻击的深远影响。
- 针对基础设施的破坏性攻击日益活跃
  - 自从震网病毒被发现至今，针对基础设施进行破坏的网络攻击活动就一直没有停止。，2015末-2016年以来，一系列针对基础设施的破坏性攻击被曝光，而且尤以工业系统和金融系统遭受的攻击最为严重：乌克兰停电事件，沙特Shamoon2.0事件，孟加拉央行被窃事件，台湾第一银行及泰国邮政储蓄银行ATM被窃事件，都属于非常典型的破坏性攻击

APT攻击的特点与趋势

- 针对特定个人的移动端攻击显著增加
  - 2016年，针对移动终端的APT攻击显得更加活跃：摩诃草、蔓灵花等针对中国发动攻击的APT组织都被发现使用了移动端专用木马程序，其中即有适配安卓系统的，也有适配苹果系统。
  - 2016年最具影响力的针对移动终端的APT攻击非三叉戟漏洞事件莫属。
- 一带一路与军民融合仍将是攻击焦点
  - 2016年的全年监测显示：“一带一路”、“军民融合”等战略方向，仍然是众多APT组织关注的焦点，相关组织主要包括海莲花、摩诃草、蔓灵花、APT-C-05、APT-C-12、APT-C-17等。而这一趋势在2017年，乃至未来几年都仍将持续。
  - 事实上，如“一带一路”等超大型国家系统工程，往往是多学科，多领域的合作工程，也是众多高新技术集中应用的工程，因此具有很高的攻击价值。同时，一旦这些国家级系统工程涉及到边疆地区建设，沿海工程建设，外交外贸等领域时，又必然会在政治、军事和经济层面引发周边相关国家的关注，从而进一步引发相关国家APT组织的攻击活动。
  - “军民融合”项目，则是攻击组织窥探军事情报的重要突破口。因为一旦军事技术或项目转为民用，其安防级别往往就会大幅下降，这也就可能给攻击者的窃密活动留出了可乘之机。而反过来，当民用技术转为军用时，通过攻击民用机构，就有可能实现对军事系统的渗透。这就是为什么军民融合项目会特别受到境外APT组织关注。所以，军民融合项目，更需要特别注意网络安全建设。

问题

- 高级攻击与你个人有关系吗？
  - 勒索病毒：邮件-主机信息收集-公私钥-勒索

目录

1. 操作系统安全的意义
2. 操作系统安全威胁
3. 操作系统安全发展趋势
4. 课程设计与要求

操作系统安全技术路线

- 安全操作系统
- 软件安全
  - 软件脆弱性分析
  - 恶意软件挖掘
  - 软件确保
- 新路线：主动防御
  - 软硬件协同
  - 可信计算
  - 拟态计算

操作系统安全和安全操作系统

- 操作系统安全和安全操作系统
  - 操作系统安全是从各种不同角度分析操作系统安全性
  - 安全操作系统是按照特定安全目标设计实现的操作系统，和相应的安全等级挂钩

安全操作系统目标

- 标识系统中的用户和进行身份鉴别
- 依据系统安全策略对用户的操作进行访问控制，防止用户和外来入侵者对计算机资源的非法访问
- 监督系统运行的安全性
- 保证系统自身的安全和完整性

安全操作系统定义

- 安全操作系统定义
  - 是一个其访问执行满足引用监视器概念的操作系统
  - 引用监视器概念定义了任何系统能安全地执行一个强制保护系统的充要属性

安全操作系统的发展

- 2001年中科院石文昌博士将安全操作系统的发展分为奠基时期、食谱时期、多政策时期和动态政策时期
- 2004年卿斯汉教授在《操作系统安全》中也对其发展进行了概述。

安全操作系统的发展：奠基时期

- 1969 年，C. Weissman研究的Adept-50 是历史上的第一个分时安全操作系统。
- 同年，B.W. Lampson 通过形式化表示方法运用主体（subject）、客体（object）和访问矩阵（access matrix）的思想第一次对访问控制问题进行了抽象。
- 1970 年，W.H. Ware 对多渠道访问的资源共享的计算机系统引起的安全问题进行了研究，提出了多级安全系统和need-to-know原则的实现

安全操作系统的发展：奠基时期

- 1972年，J.P. Anderson提出了参照监视器（reference monitor）、访问验证机制（reference validation mechanism）、安全内核（security kernel）和安全建模（modeling）等重要思想。
- 1973 年，B.W. Lampson 提出了隐通道（covert channel）；同年，D.E. Bell 和L.J. LaPadula 提出了简称BLP 模型。

安全操作系统的发展：奠基时期

- 1975 年，J.H. Saltzer 和M.D. Schroeder 以保护机制的体系结构为中心，重点考察了权能（capability）实现结构和访问控制表（access control list）实现结构，给出了信息保护机制的八条设计原则。
- 1976 年，M.A. Harrison、W.L. Ruzzo 和J.D. Ullman 提出了操作系统保护的第一个基本理论——HRU 理论。

安全操作系统的发展：奠基时期

- 1967-1979年典型的安全操作系统研究有：
  - Multics
  - Mitre 安全核
  - UCLA 数据安全Unix
  - KSOS（Kernelized Secure Operating System）
  - PSOS等

安全操作系统的发展：食谱时期

- 1983 年，美国国防部颁布了历史上第一个计算机安全评价标准TCSEC橙皮书（Trusted Computer System Evaluation Criteria)。
- 1984 年，AXIOM 技术公司的S. Kramer开发了基于Unix的实验型安全操作系统LINUS IV ，达到B2级；
- 1986 ，IBM 公司的V.D. Gligor 等设计了基于SCO Xenix的安全Xenix 系统，基于安全注意键（secure attention key，SAK）实现了可信通路（Trusted path）。

安全操作系统的发展：食谱时期

- 1987年美国Trusted Information Systems公司开发了B3级的Tmach(Trusted Mach)；
- 1988 年，AT&T Bell 实验室的。System V/MLS；
- 1989年，加拿大多伦多大学的安全TUNIS；
- 1990 年，TRW 公司的ASOS 系统；
- 1991年，UNIX国际组织的UNIX SVR4.1ES，符合B2级；

安全操作系统的发展：多政策时期

- 1991年，英、德、法、荷四国制定了信息技术安全评定标准ITSEC；
- 1992～93年，美国国家安全局NSA和安全计算公司SCC设计实现了分布式可信Mach操作系统DTMach；
- 1993年，美国国防部推出的新的安全体系结构DGSA；
- 1997年，DTOS（Distributed Trusted Operating System）项目
- 1999年，EROS(Extremely reliable operating system），基于权能的高性能微内核实时安全操作系统；
- 其他：92 年访问控制程序（ACP）和93 年看守员（custodian）两种范型思想；

安全操作系统的发展：动态政策时期

- 2001年，Flask的实现，SELinux操作系统；多级安全（MLS）政策、类型裁决（TE）政策、基于标识的访问控制（IBAC）政策和基于角色的访问控制（RBAC）政策
- 其他在研项目
  - Honeywell的STOP、Gemini的GEMSOS、DEC的VMM、HP和Data General等公司的安全操作系统；

国内安全操作系统的发展

- 1993 年，国防科技大学SUNIX 病毒防御模型；
- 1999 年，中科院软件研究所从红旗Linux；
- 2000年，中科院计算技术研究所的LIDS，南京大学的SoftOS，中科院信息安全技术工程研究中心的SecLinux；
- 2001年，海军计算技术研究所安全增强系统Unix SVR4.2/SE

国内安全操作系统的发展

- UNIX 类国产操作系统COSIX V2.0 的安全子系统；
- 96年，军用计算机安全评估准则GJB2646-96；
- 99年，国家技术监督局的国家标准GB17859-1999(计算机信息系统安全保护等级划分准则)，2001年强制执行；
- 2000年，安胜安全操作系统V1.0；
- 2001年，GB/T18336-2001(信息技术安全性评估准则)

问题

- 不同场景下的操作系统安全？
  - Windows桌面机
  - 党政军用计算机
  - 移动智能终端
  - IoT设备（泛在设备）

软件安全：漏洞检测

- 漏洞库
  - Common Vulnerabilities and Exposures（CVE）
  - 国家安全漏洞库
- 漏洞检测工具
  - 微软公司提供的MBSA(Microsoft Baseline Security Analyzer)工具，它可以对Windows系统的漏洞进行扫描，分析系统安全配置并形成相关的报表。MBSA工具可以检查Windows系统的漏洞、弱口令、IIS漏洞、SQL漏洞及检测安全升级等。
  - 金山毒霸漏洞扫描工具。
  - 360安全卫士漏洞扫描工具。
  - 瑞星漏洞扫描工具。
  - X-scan漏洞扫描工具。

软件安全：恶意软件分析

- 源代码
- 二进制代码
- 同源代码分析

软件确保

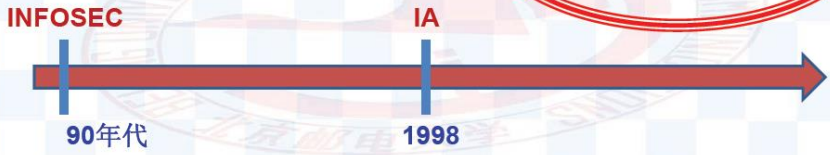
从信息确保、软件确保到系统确保
——寻找可信信息系统之路

方滨兴, 2010年1月8日

信息确保

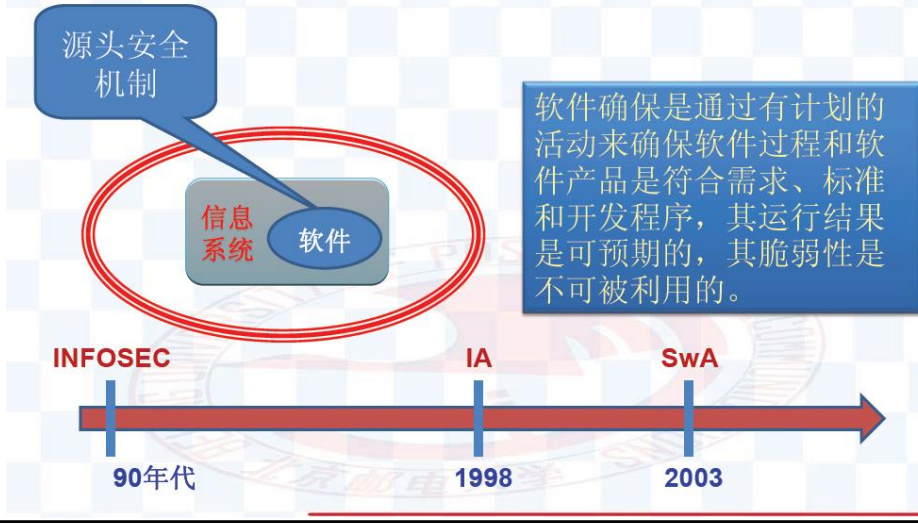
信息确保的基本理念

信息确保：
用于保护信息和信息系统的机密性、可鉴别性、完整性、可用性以及不可抵赖性的信息操作。通过施加额外的防护、检测和响应等安全服务能力来保障信息系统的安全使用



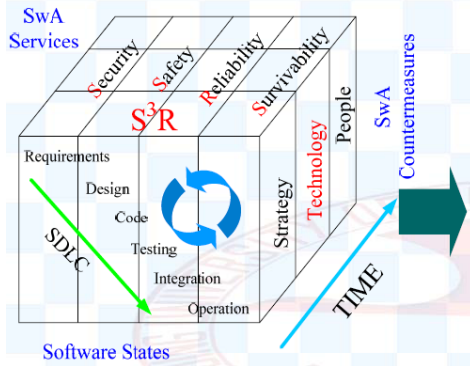
软件确保

软件确保的基本理念



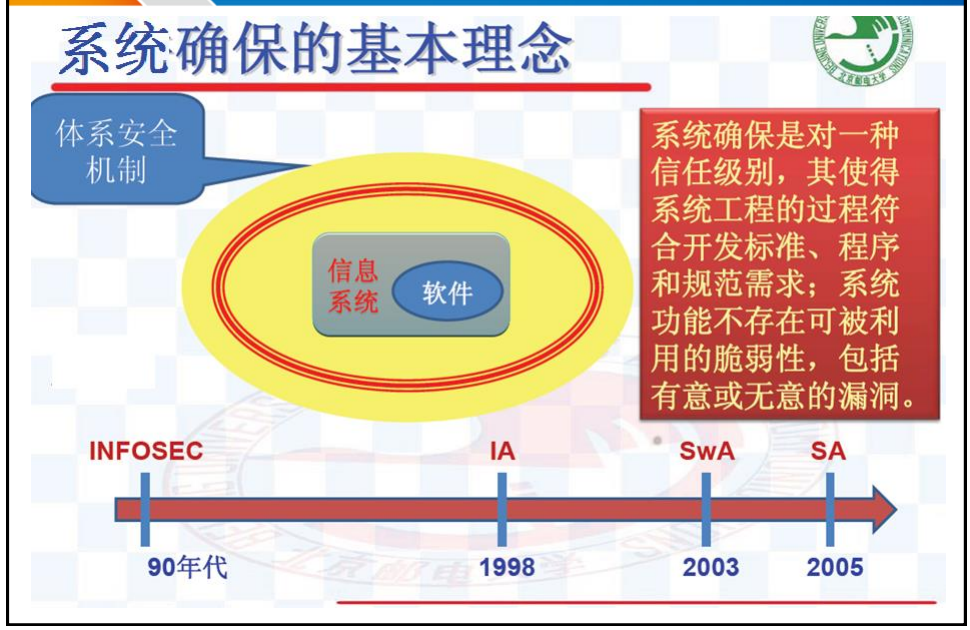
软件确保

软件确保的主要研究内容



- S³R 软件开发过程
- S³R 需求与策略
- Software 风险管理
- S³R 结构与设计
- 代码分析
- S³R 测试
- Software组装(Assembly)与集成
- 认证(Certification)与鉴定(accreditation)
- 建模
- S³R 测量与分析
- S³R的程序设计语言与分析

系统确保



软件确保与系统确保的融合

系统确保：建立可信的信息系统

- **软件确保：系统安全的根基所在**
  - 软件确保核心属性度量、预测方法，软件确保需求工程 方法，满足软件确保核心属性的软件开发过程方法 、设计与构造方法 、验证与测试方法。
- **等级保护：外壳型安全手段，支持“恶人假定”**
  - 安全检测、风险评估、访问控制、鉴别技术、安全服务
- **可信基与可靠平台：贯穿内外的安全链**
  - 可信计算、安全协议、安全策略、系统恢复、应急响应
- **可信网络基础设施：支撑交互的安全基础**
  - 安全防御体系、入侵监测、跨域协同、认证技术、可信终端 、服务的可生存性 、网络的可控性

问题：系统确保的难度?

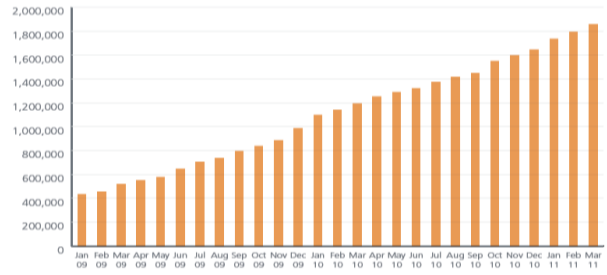
- 软件确保 vs 遗留资产
- 动态开放网络环境

主动防御技术体系

- 软硬件协同
- 可信计算
- 拟态计算

隐蔽性恶意软件数量剧增

- Rootkits数量稳步增长
  - 42例，2007
  - 200万例，2011



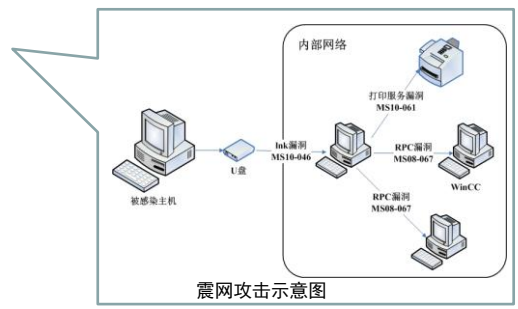
（来自美国迈克菲公司研究报告）

- APT攻击更加盛行，网络窃密风险加大
  - 我国遭受境外的网络攻击持续增多，2011年控制近890万主机（500万，2010）
  - 木马和僵尸网络活动越发猖獗，2011年比2010年增加78.5%

（摘自《2011年我国互联网网络安全态势综述》，国家互联网应急中心）

APT威胁影响巨大

- 网络攻击
  - 震网病毒：针对伊朗核设施
  - Shady RAT攻击：针对重要政府、组织和企业



震网攻击示意图

火焰病毒攻击示意图

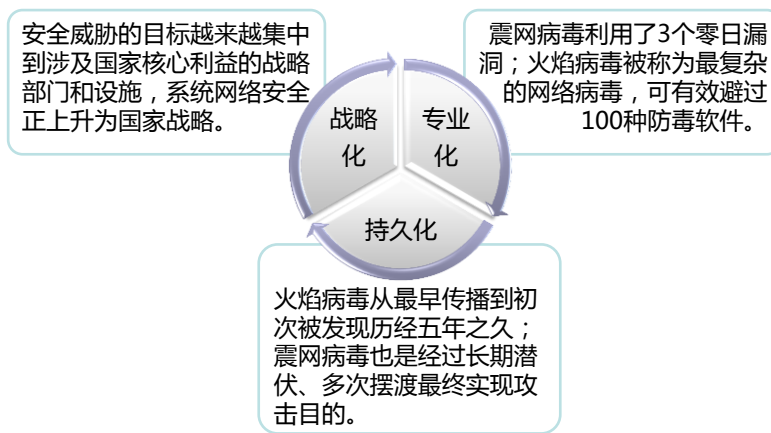
- 数据窃密
  - 火焰病毒：针对中东高官
  - Google极光攻击：针对google公司
  - 夜龙攻击：针对主要能源公司

政府主导APT攻击

- 美国已将网络安全上升为国家战略
  - 战略演变
    - 主题：基础设施保护；重点：全面防御（克林顿政府）
    - 主题：网络反恐；重点：攻防结合（小布什政府）
    - 主题：网络战；重点：攻击为主、网络威慑（奥巴马政府）
  - 假想威胁
    - “网络珍珠港”
    - “电子9.11事件”
  - 战略目标：全球制网权

计算机系统安全威胁新特点

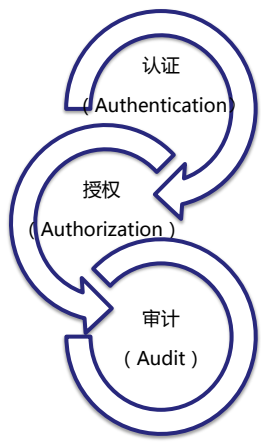
- “战略化、专业化、持久化”



“外壳式”安全防护力不从心

- 病毒样本不胜其多，反病毒程序消耗计算机资源不胜其负
  - 病毒样本库更新太慢
  - “特征”匹配很容易被“躲避”
- 攻击变化太快，现有的黑名单机制总是更新不及时
  - Advanced Malware，
  - Zero-Day，
  - Targeted APT Attacks
- 业务越来越复杂，新应用太多，用户位置多变，终端多样化并且很难规范化
  - 端口失效 - > Applications
  - IP地址失效 -> Users
  - 数据包层面的检查失效 -> Content

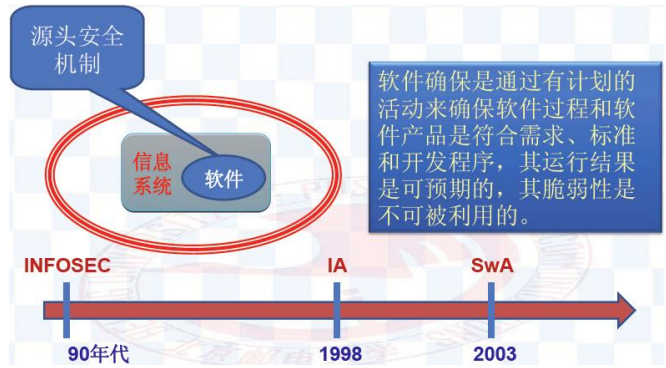
“AAA”技术需要深化



- “AAA”技术已广泛应用到了信息安全领域
- 主要在软件层面实现
- 广泛使用的口令认证机制安全强度严重不足
- 安全操作系统中强制访问和安全审计功能需要权衡性能和易用性

系统安全防护的根本解决方法

- 软件确保



（摘自方滨兴院士报告《从信息确保、软件确保到系统确保——寻找可信信息系统之路》）

- 从系统结构角度构筑主动安全防御体系

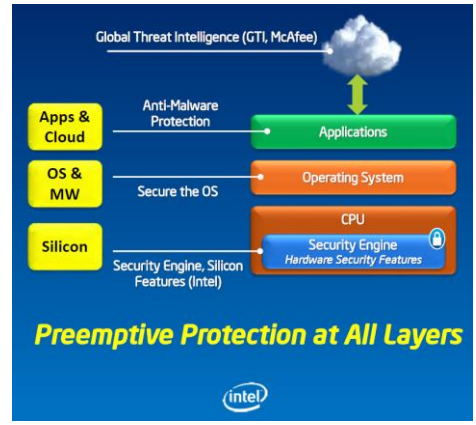
从系统结构解决计算机安全是未来重要研究方向

- “未来的可信任计算机系统至少应该为机密性、完整性、可用性等关键安全属性提供内建的支持……我们希望能促进软件安全领域与硬件体系架构领域的协作，从而通过软硬件协同设计实现安全”

---Professor Ruby B. Lee,2012 ACM SIGSAC 主题演讲

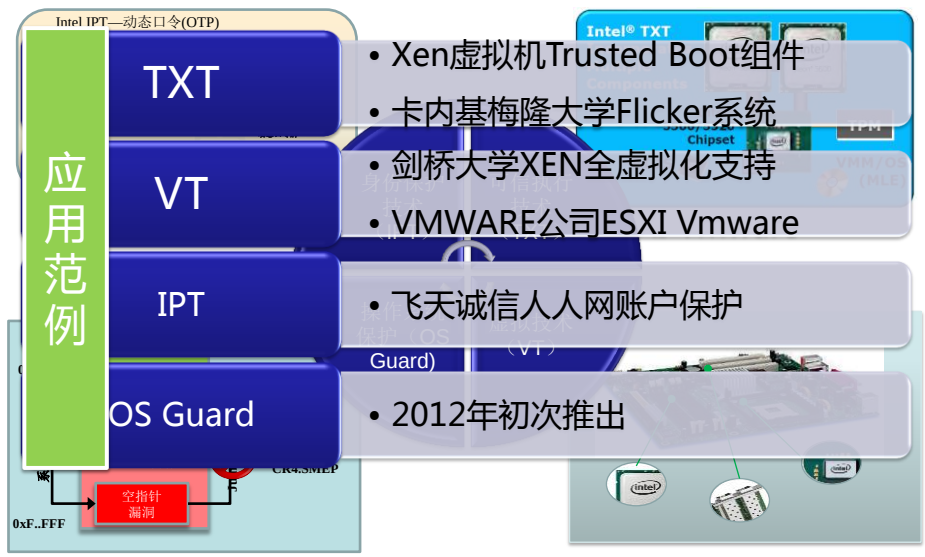
Intel安全战略

- “安全与性能同等重要”
  --- IDF2012
- 全层次安全（*Security at all Levels*）
  - 关键安全模块迁移至硬件
  - 系统固件和启动过程的安全
  - 为安全进程提供隔离执行环境
  - 基于硬件的恶意代码检测和修复



（来自Intel公司研究报告）

Intel处理器安全特性



ARM处理器安全特性

- TrustZone
  - 硬件：提供代码隔离的安全执行环境
- VE

应用范例

TrustZone

- 2012年三星GALAXY SIII欧洲版成为第一款采用Trust Zone支持安全在线支付、电子邮件等应用的手机产品

VE

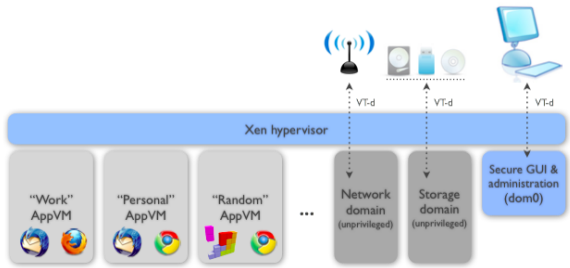
- 2011年12月Xen邮件列表公布完成了VE技术的概念原理验证开发

可信计算平台

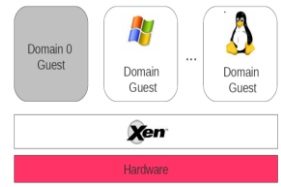
- 国际可信计算组织TCG
  - 历史简介：1999年前身TCPA成立，2003改组为TCG
  - 核心理念：专注于从计算平台体系结构上增强其安全性
  - 目的：基于硬件安全模块实现平台的完整性、身份认证和数据保护
  - 最新进展：
    - 下一代TPM标准推出
    - 成立虚拟可信计算工作组
    - 成立嵌入式可信计算平台工作组
- 自主可信计算
  - 中国可信计算联盟
  - 中国可信计算工作组

基于虚拟机隔离的安全操作系统Qubes

- 综合利用Intel VT与TXT技术、基于Xen虚拟机监控器，著名黑客组织ITL推出了面向桌面的Qubes安全操作系统
  - 实现了内核组件隔离，有效避免传统单内核体系架构带来的安全隐患
  - 实现了统一的用户交互界面，将基于虚拟机的安全操作系统成功应用于个人桌面应用环境
  - 有效提高系统安全性、可靠性



Qubes安全操作系统

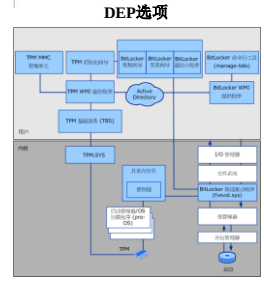


Xen虚拟机监控器架构

Windows7安全特性

- DEP（数据执行保护）机制：
  - 基于CPU不可执行位
  - 阻止恶意程序从堆和栈执行代码
- BitLocker驱动器加密：
  - 基于TPM安全芯片实现系统启动时的完整性验证
  - 基于TPM安全芯片实现加密密钥的安全存储
- ASLR（地址空间布局随机化）
  - 关键操作系统模块内存地址随机化
  - 防止攻击者实施恶意代码注入
- 提供“沙箱”运行环境

DEP选项



BitLocker体系架构

小结

- 软硬件协同的系统防护体系的研究和探索已成为重要趋势

系统层次 \ 平台类型	移动/嵌入式设备	桌面计算机	服务器
CPU	ARM TrustZone、ARM VE、Intel VT-x	Intel VT-x、Intel TXT、Intel IPT、Intel OS Guard	Intel VT-x、Intel TXT、Intel IPT、Intel OS Guard
外围芯片	Intel VT-d、TCG MTM	Intel VT-d、TPM、NX bit	Intel VT-d、TPM、NX bit
操作系统	Xen-ARM、OKL4	Qubes、BitVisor、Windows7	Xen、NOVA、KVM

问题

- 你了解哪些软硬件协同安全的实例？

目录

1. 操作系统安全的意义
2. 操作系统安全威胁
3. 操作系统安全发展趋势
4. 课程设计与要求

课程目标

- 理解操作系统中的安全问题、原理和发展趋势
- 掌握安全操作系统的基本概念、安全机制、安全模型和安全体系结构等基础理论
- 掌握操作系统高级攻击原理和防御体系
- 掌握以操作系统为核心的虚拟化软件栈的安全风险和防御体系
- 了解操作系统安全研究的技术前沿

课程要求

- 课程形式
  - 主课：十次课
    - 基础知识：1~5讲
    - 前沿技术：6~10讲
  - 实验课：三次课
    - 原理实验
    - 前沿讨论
- 成绩评定
  主课（50%）、实验课（50%）

课程内容-1 引言

1.1操作系统安全的意义

1.2 操作系统安全威胁

1.2.1恶意软件

1.2.2 Rootkit

1.2.3 APT

1.2.4 内存攻击

1.2.5 系统漏洞

1.3 操作系统安全发展趋势

1.3.1 安全操作系统

1.3.2 软件确保

1.3.3 硬件协助安全

课程内容-2 操作系统安全理论

- 2.1 操作系统安全理论概述
  - 2.1.1 操作系统安全基本概念
  - 2.1.2 操作系统安全设计
- 2.2 操作系统安全模型
  - 2.2.1 访问控制模型和信息流模型
  - 2.2.2 安全策略模型
- 2.3 操作系统安全机制
  - 2.3.1标识与鉴别
  - 2.3.2访问控制
  - 2.3.3 安全审计
  - 2.3.4 可信路径
- 2.4 操作系统安全体系结构
  - 2.4.1 Flask安全体系结构和LSM框架
  - 2.4.2 权能机制
  - 2.4.3 可信计算3.0
- 2.5 操作系统安全测评
  - 2.5.1 TCSEC和CC
  - 2.5.2等级保护
- 2.6 操作系统安全理论实验设计
  - 2.6.1 Set-UID与权能结合
  - 2.6.2基于权能的三权分立
  - 2.6.3 RBAC访问控制

课程内容-3 内存安全保护

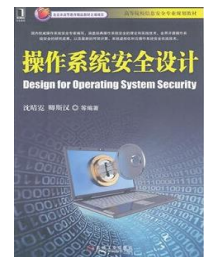
- 3.1 内存攻击原理及防护
  - 3.1.1 内存威胁分类
  - 3.1.2 内存攻击原理
  - 3.1.3 内存安全机制
- 3.2 Rootkit攻击原理及检测
  - 3.2.1 Rootkit攻击原理
  - 3.2.2 Rootkit检测技术
- 3.3 可信隔离架构
  - 3.3.1 Intel SGX
  - 3.3.2 AMD SME SEV
  - 3.3.3 ARM TrustZone
- 3.4 内存安全保护实验设计
  - 3.4.1缓冲区溢出与数据执行保护
  - 3.4.2 跨运行级提权
  - 3.4.3 Rootkit检测

课程内容-4 虚拟化安全

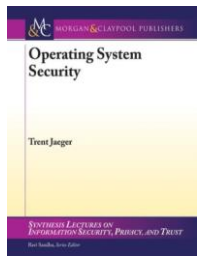
- 4.1 虚拟机安全
  - 4.1.1 虚拟机技术原理
  - 4.1.2 虚拟机安全风险
  - 4.1.3 虚拟化软件栈安全
- 4.2 虚拟机安全防护进阶
  - 4.2.1 基于处理器的安全防护
  - 4.2.2 基于额外硬件的安全防护
  - 4.2.3 同层防护
- 4.3 容器安全
  - 4.3.1 容器技术原理
  - 4.3.2 LXC
  - 4.3.3 Docker
  - 4.3.4 容器安全机制
- 4.4 前沿技术讨论
  - 经典论文3~6篇

参考文献

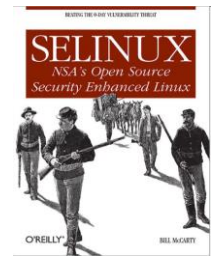
卿斯汉等．操作系统安全（第2版）清华大学出版社. 2011



沈晴霓、卿斯汉等. 操作系统安全设计. 机械工业出版社. 2013



Operating System Security by Trent Jaeger，MORGAN & CLAYPOOL PUBLISHERS，2008



SELinux NSA‘s Open Source Security Enhanced Linux，By Bill McCarty，October 2004

联系方式

- 助教：李艳昭
- 邮箱：liyanzhao@iie.ac.cn
- 手机：156-5226-6861

谢谢！

中国科学院大学
University of Chinese Academy of Sciences