第九章 安全脆弱性、威胁和对策

9.1 评估和缓解安全脆弱性

9.1 硬件

- 处理器
- 执行类型
  i. 多任务处理： 同时处理两个或更多任务
  ii. 多处理： 利用多个处理器完成一个应用程序的处理能力
  iii. 多程序设计：通过操作系统对单个处理器上的两个任务进行协调，从而模拟两个任务同时执行的情况
    - 多程序和多任务处理的差距： 多程序通常用于大规模系统使用，多任务处理在个人计算机操作系统中使用，多任务通常由操作系统协调，多程序要求特别编写的软件
  iv. 多线程处理：单个进程中执行多个并发线程，多线程的优势是降低多个线程之间转换的开销
- 处理类型： 许多安全要求较高的系统控制着被分配不同安全级别的信息的处理任务
  - 单一状态：使用策略机制来管理不同安全级别的信息
  - 多态： 多台系统能够实现更高的安全级别，通过使用特定的安全机制组织信息跨越不同的安全级别
- 保护机制
  - 保护环，保护环将操作通中的代码和组件表示为同心环，越进入环内部，特权级别越高
    - 本质：保护环的本质在于优先权、特权和内存分割
    - 安全性：操作系统和应用程序隔离，搞特权操作系统组件和地特权操作通组件之间有严格的界限
  - 进程状态：也被叫做操作状态，指进程可能再其中运行的各种执行形式
  - 安全模式：部署安全模式之前，必须存在三种特定元素：
    - 分层的MAC环境
    - 对能够访问计算机控制台的主机完全物理控制
    - 对能够进入计算机控制台所在房间的主体的完全物理控制
  - 安全模式的分类：
    - 专有模式：对系统所处理全部信息的额“知其所需”权限，等于没有知其所需。
    - 高级系统模式： 专有模式和高级系统模式的差异，高级系统模式中不必对系统处理信息的知其所需权限（只需对系统处理的部分信息具有知其所需权限）

        - 分割模式：分割模式的系统用户不必批准访问系统中的全部信息
        - 多级模式：多级模式暴露出最高的风险级别
    - 安全许可、知其所需、处理多许可级别数据PDMCL

表 9.1　安全模式的比较

模式	安全许可	知其所需	PDMCL
专用模式	相同	无	无
系统高级模式	相同	是	无
分隔模式	相同	是	是
多级模式	不同	是	是

- 操作模式
    - 处理器支持两种操作模式：用户模式和特权模式
        - 用户模式：只允许执行其整个指令集中的部分指令，为了防止执行设计很差的代码以及无意识的滥用代码而意外损坏系统
        - 特权模式(监管模式、系统模式、内核模式)：在CPU上执行的进程授予方位很广的特权

9.1.2 存储器

存储器：计算机为了保持信息使用的便捷所需的存储位置。

- 只读存储器：能够读取但是不能够修改的存储器，子类型如下
    - 可编程只读存储器：PROM芯片内容没有在工厂被烧入，允许终端用户稍后烧入内容
    - 可擦除可编程只读存储器(EPROM):当使用特殊的紫外线光照射时，可以擦除芯片上的内容，之后用户可将新信息烧入EPROM
    - 电可擦除可编程只读存储器（EEPROM）：使用送到芯片引脚上的电压进行擦除，
    - 闪存：非易时性存储媒介，可进行电子擦除和重写，闪存可以以块或页的方式进行擦写
- 随机存储器（RAM）：可读可写的存储器，当电源关闭时，数据会消失，RAM有以下类型：
    - 实际的存储器：计算机中可用的最大RAM存储资源
    - 高速缓存RAM：将数据从速度较慢的设备取出并暂时存储到高性能的设备上，高速缓存能提高系统的性能
- 寄存器：CPU的核心部分提供可直接访问的存储位置
- 存储器寻址：五种不同类型的寻址
    - 寄存器寻址：使用寄存器地址去访问寄存器的内容
    - 立即寻址：引用某些数据的一种方法，这些数据作为指令的一部分提供给CPU使用
    - 直接寻址：访问的存储器位置的实际地址会提供给CPU

    - 间接寻址：作为指令的一部分提供给CPU存储器，并不包括CPU用作操作的真实数值
    - 基址+偏移量寻址：使用存储在某个CPU寄存器中的数值作为开始计算的基址，然后将CPU指令提供的偏移量与基址相加，并计算得到存储位置取出操作数
- 辅助存储器：指磁性/光学介质或包含CPU不能立刻获得数据的其他存储设备 辅助存储器比实际存储器便宜，而且可以存储大量信息，硬盘、软盘和光学介质都可以当做辅助存储器
- 存储器安全问题： 围绕存储的最重要的安全问题是：计算机使用过程一定要控制那些人对存储器中的数据进行访问

9.1.3 存储设备

- 主存储设备与辅助存储设备
    - 主存储设备：计算机用于保存运行时CPU容易获得必要信息的RAM
    - 辅助存储设备：由磁性介质和光学介质组成
- 易失性存储设备和非易失性存储设备
    - 衡量存储设备在电源被切断时丢失数据的可能性的方法
- 随机存取与顺序存取
    - 随机存储设备允许操作系统通过某些寻址系统从设备内的任何位置立刻读取数据
    - 顺序存储设备要求到达指定位置之前读取该位置之前物理存储的所有数据

9.1.4 存储介质的安全性

- 数据剩磁：数据诶删除后仍可能保留在辅助存储设备上，净化能解决这个问题
- 净化对固态硬盘无效
- 辅助性存储设备容易被盗
- 对存储在辅助存储设备上的数据访问，是计算机安全专家面对的最紧要的问题之一

9.1.5 输入和输出设备

输入和输出设备未基本的、原始的外围设备，存在安全风险

- 显示器
    - TEMPEST的技术会危机显示器上锁显示数据的安全性
    - 肩窥和相机的长焦镜头是最大风险
- 打印机
    - 忘记取出打印出的敏感信息
    - 磁盘保留着无期限的打印拷贝
- 鼠标/键盘
    - TEPEST技术的监控
- 调制解调器

    - 处于商业原因必须使用调制解调器，否则在组织的安全策略中重点考虑禁止使用调制解调器
- 输入/输出结构
    - 存储映射I/O：映射存储位置的访问应当由操作系统居间调停，并且得到正确的授权和访问控制
    - 中断（IRQ）：只有操作系统能够在足够高的特权级别间接访问IRQ，以防止篡改或意外的错误配置
    - 直接内存访问（DMA）：

9.1.6 固件

固件：描述在ROM芯片中存储的软件，很少更改

- BIOS：基本输入输出系统，使用UEFI（统一可扩展固件接口）取代传统系统的BIOS
- 设备固件：

9.2 基于客户端

applet:代码对象被从服务器发送至客户端以便执行某些操作

- 优势
    - 处理压力转移到客户端，从而防止服务器资源
    - 可以更快的响应对输入数据的修改
    - 正确编程的applet中，可以维护参悟数据的安全和隐私性
- 安全问题：applet准许远程系统向本地系统发送执行代码
- Java applet和Active X控件
    - Java applet: 优点：可以在操作系统间共享，不需要进行修改，缺点：沙盒通过JAVA减少了恶意事件的种类，但还是存在很多广泛利用的漏洞
    - Active X控件：具有操作系统的全部访问权限，能执行很多特权操作，微弱后期可能要淘汰Active X
- 本地缓存：暂时存储在客户端上的任意内容，用于将来重新使用
    - ARP缓存投毒、DNS缓存投毒、临时互联网文件或互联网文件缓存
    - 缓解本地缓安全方法：
        - 安装操作系统和应用程序补丁
        - 安装主机入侵检测系统和网络入侵检测工具定期审计日志

9.3 基于服务端

服务端关注数据流控制

9.4 数据库安全

没有数据库安全性方面的努力，业务任务可以被终端，保密信息被泄露

9.4.1 聚合

- 聚合：将一个或多个表中记录组合在一起，生成可能有用的信息
- 防止方法：严格控制对聚合函数的访问并充分估计可能展示给未授权个体的潜在信息

9.4.2 推理

- 推理：利用几个非敏感信息片段的组合，从而获得对应属于更高级分类的信息的访问能力
- 解决办法：使用混淆，对个人用户特权保持警惕

9.4.3 数据挖掘和数据仓库

- 数据仓库包含大量潜在的敏感信息，容易受到聚合和推理攻击
- 数据挖掘技术开发基于异常的入侵检测系统的基准时，可以当安全工具使用

9.4.4 数据分析

数据分析：对原始数据进行检查，提取有用的信息

9.4.5 大规模并行数据系统

并行数据系统或并行计算系统

9.5 分布式系统

- 分布式系统结构容易出现想不到脆弱性
- 通信设备会提供不期望的分布式环境入口

9.5.1 云计算

- 云计算存在问题：隐私问题、合规性困难、采用开放标准以及基于云计算的数据是否安全
- 云的概念
  - 平台即服务(PaaS)：提供计算平台和软件解决方案作为虚拟的或基于云的服务，提供操作系统和完整的解决方案
  - 软件即服务(SaaS)：提供对特定软件应用或套件的按需在线访问而不需要本地安装，SaaS可以实现订阅服务，付费服务或免费服务
  - 基础设施即服务(IaaS)：不但提供安全操作的解决方案，还提供完全的外包选择

9.5.2 网格计算

- 并行分布处理的一种形式，松散的把大量处理节点组合在一起，为实现某个处理目标而工作

9.5.3 点对点(P2P)

点对点技术是网络和分布式应用程序的解决方案，用于在点对点实体间共享任务和工作负载，网格计算的主要去呗是没有中央管理系统，并且提供服务是实时的

9.6 工业控制系统

- 集散控制系统（DCS）：负责从单个地点的大型网络环境中收集数据和实施控制，采用模拟或数字系统
- 有效的单用途或专用图数字计算机（PLC）：用于各种工业化机电自动化管理与操作
- SCADA（数据采集与监控系统）：

9.7 评估和缓解基于Web系统的脆弱性

- 脆弱性包括XML和SAML，以及开放式WEB应用程序安全项目(OWASP)中讨论的问题

9.8 评估和缓解移动系统的脆弱性

- 恶意内部员工可以通过外部不同类型的存储设备把恶意代码进入内部
- 移动设备同行包含敏感数据

9.8.1 设备安全

- 全设备加密：
- 远程擦除：远程擦除可以远程的删除设备上的所有数据甚至配置设置，应数据恢复工具可以恢复数据，配合加密使用
- 锁定：多次尝试失败账户或设备禁用知道管理员清楚锁定标志
- 锁屏：防止有人随便拾起并能使用你的手机或移动设备
- GPS：通过GPS跟踪以跟踪丢失的设备
- 应用控制：限制设备上的应用，也可以被用来强制安装特定的应用或执行某些应用的设置
- 存储分割：人为的在存储介质上划分不同类型或数值的数据
- 资产跟踪：
- 库存控制：
- 移动设备管理：解决员工使用移动设备访问公司资源的挑战性任务
- 设备访问控制：减少对移动设备的未授权访问，如强制锁屏配置和防止用户禁用该功能

- 可移动存储：
- 关闭不使用的功能，删除那些对业务任务和个人使用无关的应用程序和禁用其他功能

9.8.2 应用安全

应用安全：专注设备上使用的应用程序和功能

- 秘钥管理 大多数密码系统问题都出在秘钥管理而不是算法上，一旦创建秘钥，尽量减少暴露损失和风险的方式进行存储
- 凭证管理：中心位置的凭证存储被称为凭证管理 凭证管理解决方案提供了一种方法来安全的存储大量的凭证集
- 认证：谨慎的结合设备认证和设备加密，以组织通过连接电缆访问存储的信息
- 地理标记： GPS
- 加密：加密能提供对未授权数据访问的保密机制
- 应用白名单： 禁止未授权软件能够被执行的安全选项

9.8.3 BYOD关注点：

- BYOD： 允许员工在工作中携带自己的个人移动设备并使用这些设备连接公司网络业务或互联网
- BYOD带来的问题：
  - 数据所有权：提供数据隔离、分割、支持业务数据处理且不影响个人数据
  - 所有权支持
  - 补丁管理
  - 反病毒管理
  - 取证
  - 隐私
  - 在线/不在线
  - 遵守公司策略
  - 用户接受
  - 架构/基础设施考虑
  - 法律问题
  - 可接受策略
  - 机载摄像头/视频

9.9 评估和缓解嵌入式设备和物联网系统的脆弱性

- 嵌入式系统通过计算机实现一个更大系统的一部分

9.9.1 嵌入式系统和静态数据示例

- 支持网络功能的设备是那些本身有网络功能的编写或非便携设备
- 网络物理系统指的是一种计算手段来控制物理世界中某样东西的设备
- 网络物理系统、嵌入式系统和具备网络功能的设备的一种新扩展是物联网（IoT）
- 大型机是高端计算机系统并用于执行高度复杂的计算和提供大容量的数据处理
- 现代大型机更灵活
- 车辆计算系统把偶偶用于见识发送机性能和优化制动、转向及悬挂的组件

9.9.2 安全方法

- 嵌入式系统或静态系统往往缺乏安全性且难于升级或安装补丁，安全治理的方法：
  - 网络分隔：涉及控制网络设备之间的流量，隔离完成，传输仅限于分隔网络的设备之间
  - 安全层：当不同级别分类或灵敏度不同的设备被分在一组时，就存在安全层，从而对不同级别的分组进行隔离，逻辑隔离要求数据包使用不同的分类标签，物理隔离需要实现不同安全级别网络之间的网络分割和空间隔离
  - 应用防火墙：设备、服务器插件、虚拟服务或系统过滤器，定义服务和所有用户之间严格的通信规则
  - 网络防火墙：为一般网络过滤而设计的装置，目的是提供全网的广泛保护
  - 手动升级：在静态环境下以确保只实施测试和授权更改，使用自动更新系统将允许未检测的更新引进未知的安全降级
- 固件版本控制：固件版本控制优先保持稳定的操作平台，同时尽量减少危险暴露和停机时间
- 包装：封闭或包装其他东西，控制信道可以是特定的包装器，包括完整性和认证功能
- 控制冗余和多样性：深度防御以同心圆或平面层方式使用多层访问控制，通过冗余和多样性的安全控制，避免单一安全功能失效的陷阱

9.10 基本安全保护机制

安全机制分两技术机制和策略机制两部分

- 技术机制：系统设计人员针对系统建立的控制措施
  - 分层法：实现与用于操作系统的环模型类似的结构，并且能够应用于每一个操作系统进程，层与层之间的通信只能使用定义良好的特定接口，以提供必要的安全性
  - 抽象：对象的用户没有必要知道对象的工作细节，只要使用正确的语法和作为结果返回的数据烈性，抽象的另一方式是引入安全组
  - 数据隐藏：多级安全的重要特征，能够确保某个安全级别的数据对于运行在不同安全级别的进程来说是不可见的
  - 进程隔离:要求操作系统为每个进程的指令和数据提供不同的内存空间，进程隔离的优点：组织未经授权的数据访问、保护进程的完整性

    - 硬件分隔：用于组织对属于不同进程/安全级别的信息的访问，硬件隔离使通过无控制来实现要求，进程隔离通过操作系统强加的逻辑进程隔离

9.10.2 安全策略与计算机体系结构

- 安全策略指导组织红日常的安全操作、过程和措施
- 安全策略的角色是告知和指导某些特殊系统的设计、开发、实现、测试和维护
- 组织信息从较高安全级别流向较低安全级别的策略被陈伟多级安全策略

9.10.3 策略机制

- 最小化特权原则
- 特权分离，职责分离可以被视为针对管理员的最小特权的应用
- 可问责性：支持可问责性需要用户活动记录，可靠的审计和监控系统，灵活的授权系统和完美的身份认证系统

9.11 常见的缺陷和安全问题

安全模型和体系结构的目的是尽可能多的解决已知的缺陷

9.11.1 隐蔽通道

- 隐蔽通道是用于传递信息的方法，违反、绕过或会比了安全策略而不被发现的一种方法
- 隐蔽通道类型：
    - 时间隐蔽通道:改变系统组件的性能或改变资源的时间安排来穿搭信息，难以检测
    - 存储隐蔽通道:将数据写入其他进程可以读到的公共存储区域来传达信息
- 针对隐蔽通道活动的最佳防护措施时实现审计和分析日志文件

9.11.2 基于设计或编码缺陷的攻击和安全问题

较差的设计方法、可以的实现应用和措施、不充分的测试，都可能导致特定的攻击

1. 初始化和失败状态：可信恢复能够保证安全控制失效的情况下不发生任何访问活动，甚至在系统恢复阶段
2. 输入和参数检查：试图将恶意入侵或代码作为程序输入部分时导致的攻击类型被称为缓冲区溢出，缓冲区溢出脆弱性的责任是编程人员
3. 维护钩子和特权程序：维护钩子程序是只有系统开发人员才知道的系统入口点，也称为后门，可通过监控审计日志发现未经授权的管理员访问后门的行为
    - 系统脆弱性是程序在执行过程中安全级别被提高的情况
4. 增量攻击：攻击形式以缓慢的、渐进的增量方式发生
    - 攻击者获得系统的权限并且在存储、处理、输入、输出或事物处理期间对数据进行细小的、随机的或增量的改变时，就会发生数据欺骗，数据欺骗是修改数据的

方法，视为主动攻击
    - salami攻击，系统化的消减账户或财务就中的资产

9.11.3 编程

- 编程的最大权限，缓冲区溢出
- 所有的程序必须经过完整的测试以遵从安全模型

9.11.4 计时、状态改变和通信中断

- 攻击者可以根据任务执行的可预测性来开发攻击程序
- 当资源的状态或整个系统发生改变时，攻击者可以试图在两中已知的状态之间采取行动

9.11.5 技术和过程完整性

9.11.6 电磁辐射

- 消除电磁辐射拦截最容易的方法：通过电缆屏蔽或放入导管来降低辐射