信息安全导论 | Introduction to Information Security

目录

1. 信息化发展与信息安全

1.1. 举例说明信息化的意义及对我国的重要影响

1. 工业控制系统，国家关键基础设施。
2. 最活跃的生产力要素，国家综合实力和国际竞争力的关键要素。

1.2. 解释信息疆域的概念

1. 不止传统疆域概念的“海，陆，空”。
2. 国家“信息疆域”不仅包括互联网，同样也包括已经广泛使用的诸如金融、电力、电信、运输、能源、军事、统计等国家安全关键系统的信息网络。

1.3. 信息安全的三个基本属性

1. 保密性，信息访问要经过授权，保护个人信息隐私。
2. 完整性，防止对信息进行不适当的修改或破坏，包括确保信息的不可否认性和真实性。
3. 可用性，确保可以及时可靠地访问和使用信息。

1.4. 描述信息安全概念的演变过程

1. 三种最基本的安全需求：保密性，完整性，可用性。
2. 信息技术进步，经历通信保密，计算机安全和信息系统安全，信息保障阶段。
3. 时至今日，信息安全问题的解决，除技术因素外，还需考虑政治，经济，文化等因素。

1.5. 为什么要提出新的信息安全观？新的信息安全观与以前的信息安全概念有什么区别吗？

1. 传统的信息安全概念已经不足以概括人们对信息安全的需求。
2. 以前信息安全概念偏向于技术因素。新的信息安全观相比以前，除技术因素外，还需考虑政治，经济，文化等因素。更加全方位。

1.6. 为什么说信息安全是非传统安全？它有哪些特点？

1. 因为随着信息化的推进，传统信息安全不足以概括人们对信息安全的需求。
2. 信息安全的特点：威胁多元性、攻防非对称性，影响广泛性、后果严重性、事件突发性。

1.7. 我国信息安全保障工作的总体要求和主要原则分别是什么？

1. 总体要求：坚持积极防御，综合防范。
2. 主要原则：立足国情，以我为主，坚持管理与技术并重；正确处理安全与发展的关系，以安全保发展，在发展中求安全。

1.8. 我国信息安全保障基础性工作有哪些？

1. 实行安全等级保护
2. 开展信息安全风险评估
3. 加强密码技术应用，建设网络信任体系
4. 高度重视应急处理工作
5. 加强技术研发，推进产业发展
6. 加强法制建设和标准化建设
7. 加快人才培养，增强全民意识

1.9. 我国的国家信息化发展战略对信息安全保障工作提出了哪些要求？

1. 全面加强国家信息安全保障体系建设
2. 坚持积极防御，综合防范

3. 主动应对信息安全挑战

1.10. 其他

- 信息化定义：充分利用信息技术，开发信息资源，促进信息交流和知识共享，提高经济增长质量，推动经济社会发展转型的历史进程。
- 保密性：指信息不被泄露给非授权的用户、实体或过程，或被其利用的特性。
  - 常见保密技术：防侦听、防辐射、信息加密、物理保密、信息隐形。
  - 特点：
    1. 包括信息内容保密和信息状态保密两个方面；
    2. 信息的存储和处理中的保密往往被人忽略。
- 完整性：指信息未经授权不能进行更改的特性，保证信息不受各种异常事件的破坏。
  - 常见完整性保护方法：协议（TCP 协议）、检错和纠错编码方法（CRC、RAID5）、密码校验和方法等。
- 可用性：指信息可被授权实体访问并按需求使用的特性。（确保可以及时可靠地访问和使用信息）
  - 影响可用性的几个方面：硬件可用性、软件可用性、人员可用性、环境可用性。
- 信息安全概念的演变：
  - 通信保密
    1. 主要解决如何在远程通信中拒绝非授权用户的信息访问以及确保通信的真实性
    2. 技术重点是通过密码技术解决通信保密问题，保证数据的保密性和完整性
  - 计算机安全和信息系统安全
    1. 20 世纪 80 年代，随着计算机的应用，数据的完整性和可用性开始走上历史舞台；
    2. 20 世纪 90 年代，随着信息网络的发展，信息系统安全开始成为信息安全的核心内容
  - 信息保障
    1. 保护和防御信息及信息系统，确保其可用性、完整性、保密性、鉴别、不可否认性等特性。
    2. 这包括在信息系统中融入保护、检测、反应功能，并提供信息系统的恢复功能。
- 信息安全的非传统安全特点：传统安全威胁从发生、发展直至造成后果，往往需要较长时间，而信息安全事件具有攻击的便利性和可能的的巨大收益的特点。
  1. 威胁的多元性
  2. 攻防的非对称性
  3. 影响的广泛性
  4. 后果的严重性
  5. 事件的突发性

2. 信息安全基础

2.1. 信息系统安全要素的基本概念

使命、资产、资产价值、威胁、脆弱性、事件、风险、残余风险、安全需求

1. 威胁：分为人为威胁和自然威胁两种。由威胁源所实施的、导致安全事件发生的行为成为攻击。
2. 脆弱性：信息系统的脆弱性是安全风险产生的内因，威胁是安全风险产生的外因。

2.2. 信息系统保护的实质

- 风险管理

2.3. 信息系统安全保护的直接目的

- 控制安全风险

2.4. 安全风险产生的内因

- 信息系统的脆弱性

2.5. 安全风险产生的外因

- 威胁

2.6. 威胁、脆弱性、风险三者的关系

- 威胁针对信息系统的脆弱性发起攻击，对系统产生安全风险

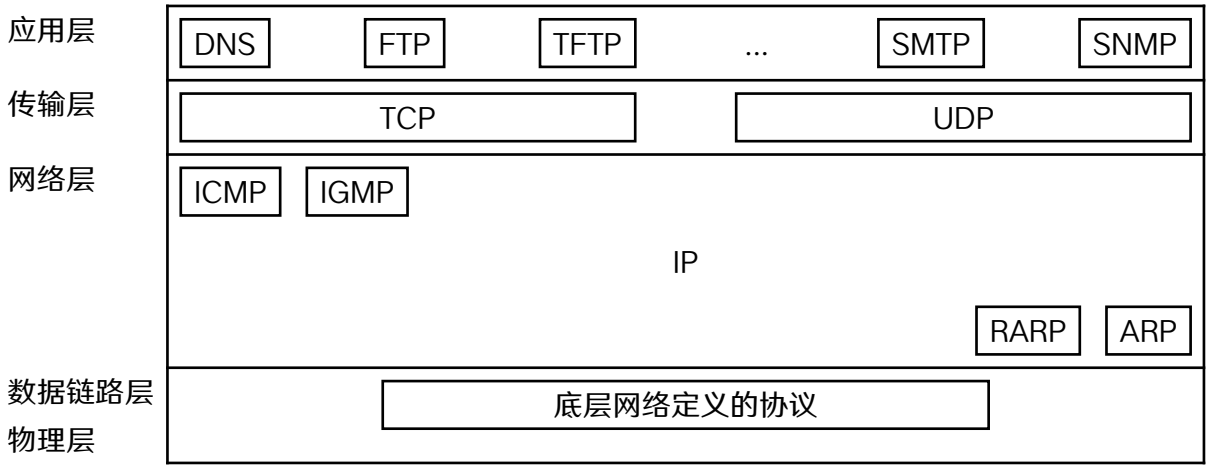
2.7. 常见的风险控制措施

1. 风险降低
2. 风险承受
3. 风险规避
4. 风险转移

2.8. ISO/OSI（开放互联系统）

7	应用层	直接为用户的应用进程提供服务，比如电子邮件、远程登录等
6	表示层	解决传输数据的语法表示
5	会话层	两个通信应用进程之间建立、维持通信，同步交互，对传输数据进行管理
4	传输层	提供端到端的通信，实现报文段的传输
3	网络层	将源端发出的分组按照路由规则传输到目的端，实现主机到主机的传输
2	数据链路层	两个相邻节点无差错的传输数据帧，实现可靠传输
1	物理层	提供在物理介质上透明传输比特流所需的各种功能

2.9. TCP/IP 协议族



每一层的作用:

- 物理层和数据链路层：由底层网络定义，TCP/IP 协议族没有定义任何特定的协议。底层网络可以包括局域网、城域网和广域网。
- 网络层：
  - IP 协议（一种主机到主机的协议）是网络层主要协议，提供一种不可靠的、尽最大努力交付的服务。IP 协议之上可以有多个传输协议。
  - 还包括一些其他协议：
    - ARP（地址转换协议）和 RARP（逆地址转换协议）：实现 IP 地址与物理地址的相互转换
    - ICMP（互联网控制报文协议）：实现网络层的差错报告和查询报告

  - IGMP（互联网组管理协议）：用于多播路由器和实现多播站点之间进行群组成员关系的通信
  - RIP（路由信息协议）和 OSPF（开放最短路径优先协议）：用于动态生成路由表
- 传输层：包括 TCP（传输控制协议）和 UDP（用户数据报协议），他们都使用相同的网络层 IP 协议
  - TCP：提供面向连接的、可靠的传输服务
  - UDP：提供无连接服务，不能保证数据报传输的可靠性
  - TCP 和 UDP 协议实现进程到进程间的通信，被称为端到端的协议。TCP 用于需要可靠传输的情况，UDP 用于实时性要求高的情况
- 应用层：包含了各种直接针对用户需求的协议，每个应用层协议都是为了解决某一类应用问题而设计的
  - DNS（域名服务系统）：实现域名与 IP 地址的对应
  - FTP（文件传输协议）：实现传输文件功能
  - SMTP（简单邮件传送协议）：实现电子邮件的发送
  - SNMP（简单网络管理协议）：实现网络管理的需要

2.10. 概述信息系统安全要素，并说明各要素之间的关系

1. 威胁、脆弱性、风险等等。
2. 信息系统的脆弱性是安全风险产生的内因；威胁是安全风险产生的外因。威胁针对信息系统的脆弱性发起攻击，对系统产生安全风险。

2.11. OSI 开放互联安全体系中包含哪些安全服务？

五种安全服务：

1. 鉴别：提供对通信对等实体和数据来源的鉴别
2. 访问控制：针对资源的不同类型，不同权限的访问
3. 数据保密性：对数据提供保护使之不被非授权的泄露
4. 数据完整性：防止被非授权的实体进行非法修改
5. 抗抵赖性：防止对方实体、进程抵赖

具体而言：

1. 鉴别：鉴别服务提供了通信中针对实体或数据来源的鉴别。利用密码来验证用户身份是常用方法。
   - 对等实体鉴别
   - 数据源鉴别
2. 访问控制：
   - 访问控制保证了所有主体实现授权访问
   - 访问控制决策控制着哪些主体在何种条件下，为了什么目的，可以访问哪些目标
3. 保密性：
   - 保证信息不泄露或不暴露给那些未授权掌握这一信息的实体。
   - 分为连接保密性、无连接保密性、选择字段保密性和通信业务流保密性。
4. 完整性：用于防止对抗信息在存储、传输等处理过程中受到非授权的修改。
5. 抗抵赖性：
   - 有数据原发证明的抗抵赖：为数据的接收者提供数据来源的证据，使发送方无法抵赖。
   - 有交付证明的抗抵赖：为数据的发送者提供数据交付证据，使接收方无法抵赖。

2.12. OSI 开放互联安全体系中包含哪些安全机制？

八种安全机制：

1. 鉴别交换机制

2. 访问控制机制
3. 路由选择控制机制
4. 加密机制
5. 通信业务填充机制
6. 数据完整性机制
   - 单个数据单元或字段的完整性：主要采用校验数据方式来保证，例如 MD5 值
   - 数据单元流完整性：主要通过通讯协议来实现（TCP）
7. 数字签名机制
8. 公证机制

2.13. 安全服务和安全机制的关系

- 一类安全服务可以通过某种安全机制单独提供，也可以通过多种安全机制联合提供
- 一种安全机制可以提供一类或者多类安全服务
- 最适合配置安全服务的是物理层、网络层、传输层及应用层

2.14. 信息安全保障体系由哪些部分组成？

1. 一个确保
   1. 确保国家基础信息网络和重要信息系统安全，创建安全健康的网络环境
2. 四个层面
   1. 信息安全法制体系
   2. 信息安全组织管理体系
   3. 信息安全技术保障体系
   4. 信息安全平台及安全基础设施
3. 两个支撑
   1. 信息安全经费保障体系
   2. 信息安全人才保障体系

2.15. 常用安全技术

1. 防火墙技术
2. IDS（入侵检测系统）
3. IPS（入侵防御系统）
4. 恶意代码防护
5. 恶意代码处置

2.16. 传统的信息安全保护思路存在哪些弊端？

- 比较被动。

2.17. 概述积极防御的信息安全保护技术保护框架的主要内容

1. 以可信的应用操作平台为核心
2. 辅以安全的共享服务资源边界保护
3. 安全保护的网络通信和安全管理中心

2.18. “两个中心”支持下的三重信息安全技术保护框架与 OSI 开发互联系统的区别是什么？

- 保护框架是针对应用安全进行逐级防护。OSI 是网络通信的体系架构。二者功能上存在很大的区别。

2.19. 防火墙

- 作用：起边界保护的作用，阻断来自外部的攻击对内部网络的入侵，保护内部网络安全。凡是能有效阻止网络非法链接的方式，均可称作防火墙。
- 分类：
  - 按实现技术不同：
    1. 包过滤防护墙
    2. 状态检测防火墙
    3. 代理服务防火墙
  - 按实现形态不同：
    1. 软件防火墙
    2. 硬件防火墙
- 功能：
  1. 基础功能：
     1. 包过滤：利用设置的过滤条件，检测通过网络包的特征来决定放行或者阻止。通过包过滤，防火墙可以实现阻挡攻击，禁止外部/内部访问某些站点，限制每个 IP 流量和连接数。
     2. 包的透明转发：防火墙一般架设在内部网络的最外端，也就是服务器->防火墙->客户机的地理布局。客户机对服务器的访问请求与服务器反馈给用户的信息，都需要经过防火墙的转发。（很多防火墙具备网关的能力）
     3. 阻挡外部攻击：如果用户发送的信息是防火墙设置所不允许的，防火墙将立刻阻断该访问，避免其进入防火墙之后的服务器。
     4. 记录攻击：防火墙可以把攻击行为记录下来供管理员分析。但出于效率考虑，当前一般记录攻击的工作交给 IDS 完成。
  2. 访问控制
  3. 内容控制
  4. 安全日志
  5. 集中管理
  6. 其他：流量控制、NAT（网络地址转换）、VPN（虚拟专用网）
- 不足：

1. 可以阻断攻击，但不能消灭攻击源
2. 不能抵抗最新的未设置策略的攻击漏洞
3. 并发连接数的限制容易造成拥塞或者溢出
4. 无法阻止服务器合法开放的端口
5. 一般不阻止内部网络的攻击
6. 防火墙本身可能存在 BUG
7. 一般防火墙不检测病毒

2.20. IDS（入侵检测系统）和 IPS（入侵防御系统）

IDS：

- 作用：监视受保护系统或网络的状态，可以发现正在进行或已发生的攻击
- 分类：
  1. 基于主机的入侵检测系统
  2. 基于网络的入侵检测系统
- 功能：
  1. 监视用户和系统活动
  2. 发现入侵行为
  3. 记录和报警

IDS（入侵检测系统）定义：通过对计算机网络中的若干关键点收集信息并对其进行分析，从中发现网络或系统中是否有违反安全策略的行为和被攻击的迹象。它可以将得到的数据进行分析，并得出有用的结果。

IPS（入侵防御系统）定义：IPS 技术可以深度感知并检测流经的数据流量，对恶意报文进行丢弃以阻断攻击，对滥用报文进行限流以保护网络带宽资源。

IDS 与 IPS 的区别：

1. 采取的行为不同
   - IDS 存在于网络之外起到报警的作用，而不是在网络前面起到防御的作用。
   - IPS 如果检测到攻击，会立即阻止该恶意通信。
2. 检测攻击的方法不同
   - IDS 采用旁路侦听的方式监控网络包，对网络中传输的数据不做任何处理，不会减慢网络的传输速度。
   - IPS 将检查入网的数据包，确定这种数据包的真正用途，然后决定是否允许这种数据包进入网络，可能会减低网络传输速度。
3. 关注的内容不同
   - IDS 注重的是网络安全状况的监管，主要是被动记录攻击信息。
   - IPS 关注的是对入侵行为的控制，倾向于主动防护。与防火墙类产品、IPS 可以实施的安全策略不同，入侵防御系统可以实施深层防御安全策略，即可以在应用层检测出攻击并予以阻断，这是防火墙和 IDS 做不到的。
4. 部署的地理位置不同
   - IDS 需要部署在网络内部的中心点。
   - IPS 部署在防火墙和网络设备之间，确保所有来自外部的数据必须串行通过 IPS，IPS 实时分析网络数据，发现攻击行为立即予以阻断，保证来自外部的攻击数据不能通过网络边界进入网络。

2.21. 防火墙、IDS、IPS

- 防火墙：大门警卫（只查证件）
- IDS：监控录像观察人员（发现异常报警但无法阻止破坏行为）
- IPS：厂区内必经路口的固定检查点内保人员（发现异常行为的场内人员马上处理）

2.22. 恶意代码可以分为哪些常见类型

1. 病毒
   - 定义：病毒是一种靠修改其他程序来插入或进行自身拷贝，从而感染其他程序的一段程序。
   - 特点：采用文件寄生，具有传染性、潜伏性、多态性和破坏性。
2. 蠕虫
   - 定义：利用操作系统或应用程序漏洞，通过网络通信功能将自身从一个节点发送到另一个节点并启动运行的程序。
   - 蠕虫和病毒的比较
     - 相同点：传播性、隐蔽性、破坏性。
     - 不同点：蠕虫不利用文件寄生，对网络造成拒绝服务，与黑客技术相结合
     - 破坏性：蠕虫通过互联网可以在短时间内蔓延全球，消耗内存和网络带宽，造成网络瘫痪。
3. 特洛伊木马
   - 定义：隐藏在合法程序中的非法代码，在用户不知情的情况下执行
   - 木马与病毒的区别：
     1. 木马不具备传染性，不能复制自身；
     2. 通过将自身伪装，欺骗用户执行；

3. 木马程序主要分为服务器（被控）和控制器程序（主控），感染了木马程序的计算机，用户在不知情的情况下完全受控；
   - 木马程序的危害：木马主要以窃取用户相关信息为目的。
4. 逻辑炸弹
   - 定义：在特定逻辑条件满足时实施破坏的计算机程序，该程序不会传播；
   - 用途：主要在军事方面使用，强调破坏作用本身。

2.23. 恶意代码处置

1. 检测阶段
   - 传统的检测技术：“特征码”检测技术，缺点是：滞后于威胁的出现，误报率高。
   - 先进检测技术：启发式检测法、基于行为的检测法等。
2. 反应阶段：定位恶意代码存储位置、辨别具体的恶意代码、删除存在的恶意代码并纠正恶意代码造成的后果等；
3. 恢复阶段：找出事件根源并彻底清除，恢复被破坏的数据。

3. 密码技术与应用

3.1. 密码学的特点

- 密码学是信息安全的基础：身份认证、信息存储、加密传输、信息的完整性、信息的不可抵赖性等。
- 密码学是建立在一定的数学基础上的。
- 密码技术的实现和应用也多种多样 。

3.2. 密码技术概述

- 研究目的：以研究秘密通信为目的而产生的学科；
- 研究内容：对传输信息采用何种秘密的变换，从而达到防止攻击者截取传输信息的还原；
- 两个分支
  - 密码编码学：主要目的是寻求保证信息保密性或可认证性的方法；
  - 密码分析学：主要目的是研究加密信息的破译或信息的伪造。

3.3. 密码体制分类

- 单钥密码体制（对称密码体制）
  - 特点：加密和解密采用同一密钥，通讯双方均需获得该密钥；
  - 安全性依赖：
    1. 加密算法尽可能强大，保证只有获得密钥才能够解密密文；
    2. 加密安全性的根本保证在于密钥的秘密性，即使算法泄密，密钥仍然可以保证加密过程；
  - 缺点：
    - 密钥数量多：n 个用户的网络，需要 n(n-1)/2 个密钥（任意两个用户之间均有一个唯一的密钥）
    - 密钥的分发和保存复杂；
  - 加密方式：
    - 流密码：按照明文信息逐字符加密；
    - 分组密码：将明文分组，逐组进行加密。
- 双钥密码体制（非对称密码体制）
  - 特点：加密密钥（公钥）与解密密钥（私钥）是不同的，公钥和私钥不能相互计算，而且公钥是公开的；

- 优缺点：
  - 密钥管理简单：每个用户均有一对属于自己的公钥和私钥，其中公钥公开
  - 双钥密码体制计算非常复杂，实现速度远低于单钥密码体制
- 工作过程：发送者必须使用接收者的公钥对数据进行加密，一旦加密，只有接收方用其私钥才能加以解密。
- 应用：
  - 双钥密码体制通常用来加密核心数据，单钥密码体制通常用来加密大量的明文数据；
  - 混合加密系统：采用单钥密码体制加密文件，采用双钥密码体制加密“加密文件”的密钥。

3.4. 分组密码

- 分组密码是将明文信息划分为等长的组，各组采用相同的密钥进行加密运算，得到密文。
- 特点:
  - 加密密钥和解密密钥相同
  - 分组密码的安全性依赖于密钥的保密，而不是加密或解密算法的保密；
- 常见分组密码算法：DES 算法、IDEA 算法等

3.4.1. DES 算法

- 特点
  - 明文按照 64 位进行分组，不足部分填充为 64 位
  - 密钥长度为 64 位，实际使用 56 位
  - 密文长度大部分也为 64 位
- 安全性
  - 密钥的个数为2^{56}个，绝大部分情况下可以保证明文的安全；
  - 随着高速计算机的出现，最大的问题是密钥长度过短；
- 三重 DES 算法
  - 三重 DES 算法核心是为了解决 DES 密钥过短的问题，使用两个密钥执行三次 DES 算法
  - 三次操作中之所以使用2个密钥的原因在于112位长度的密钥安全性已经足够,采用168位密钥的话，系统效率过低。

3.5. 公钥密码

- 单钥密码体制的缺点
  - 在任何密文传输之前，通信双方必须使用一个安全通道来协商加密密钥；
  - 无法提供一种数字化签名的方法；
- 公钥密码体制的优点：较好的解决了上边存在的两个问题。

3.5.1. RSA 算法

- RSA 算法是一种公钥算法，78 年由 Ron Rivest、Adi Shamirh 和 LenAdleman 在（美国麻省理工学院）开发的，是第一个既能用于数据加密也能用于数字签名的算法，是目前最有影响力的公钥加密算法；
- 缺点
  - 产生密钥较麻烦，不能实现一次加密一个密钥对；
  - 运算比较耗时，软件实现的 RSA 算法用时大约是 DES 算法的一千倍；
- 用途：一般用于数字签名、密钥管理和数字证书方面，极少用于数据加密；
- 安全性：随着高速计算机的出现，要求至少采用 1024 位长度的密钥，CA 中心要求采用 2048 位密钥。

3.6. 散列函数

- 又称为杂凑函数、哈希函数；

- 将任意长度的信息 M 映射成为较短的、固定长度的一段数据 h(M)，当 M 的一位或几位发生变化时，h 均会发生变化；
- 只能实现单向运算
- 应用：是数据完整性和身份认证的核心技术。
- 常见算法：MD5、SHA-1
- 数字指纹
  - 散列函数的单向特性和输出数据的长度固定的特性，可以生成消息或数据块的“数字指纹”。
  - 散列函数会为任何一段二进制数据生成散列值，该散列值唯一对应该段二进制数据。
- 散列函数的分类
  - 强无碰撞的散列函数
  - 弱无碰撞的散列函数

3.6.1. MD5

- 特点：散列值长度为 128 位；
- 应用：
  - 对信息产生摘要：例如下载的文件包含一个 MD5 值，可以防止下载过程中出现错误；
  - 登录认证：在 Unix 系统中，系统保存的不是用户密码，而是密码经过 MD5 算法经哈希运算后得到的 MD5 值。用户登录时，系统把用户输入的密码进行 MD5 哈希运算后，再将结果与保存的口令 MD5 值比较。
- 安全性：2004 年，已被山东大学王小云教授破解。

3.6.2. SHA-1

- 介绍：SHA (Secure Hash Algorithm，译作安全散列算法) 是美国国家安全局 (NSA) 设计，美国国家标准与技术研究院 (NIST) 发布的一系列密码散列函数。1995 年，SHA-1 散列算法发布；
- 特点：散列值长度为 160 位；
- 安全性：2005 年，已被山东大学王小云教授破解，需要2^{69}的计算复杂度。

3.7. 相关方面的应用

3.7.1. 数字签名

3.7.1.1. 数字签名

- 数字签名必须实现的目标
  1. 接收者能够核实发送者对报文的签名；
  2. 发送者事后不能抵赖对报文的签名；
  3. 接收者不能伪造对公文的签名。
- 基本原理：发送者针对整个消息采用私钥加密
- 缺点
  - 对称密码体制的特点，对整个消息进行签名（私钥加密），速度过慢；
  - 发送者计算信息的摘要(指纹),加密只是针对消息的摘要(指纹)进行私钥加密,而消息本身采用明文传输。接收方收到信息后，利用公钥得到摘要并依据消息重新计算摘要，如果二者相同,说明数据在发送的过程中没有被篡改。但是因为公钥的公开性,消息本身可能被截获。
- 解决方法：采用数字信封技术。

3.7.1.2. 数字信封

- 基本原理
  - 原文用对称密钥加密传输之前，先将对称密钥用收方公钥加密作为电子信封发送给对方。对方收到后用自己的私钥解密信封，取出对称密钥，再接收密文并解密。

  - 对称+非对称，一起使用
- 特点
  - 发送方签名只针对散列值，而不针对消息本身；
  - 消息本身采用单钥密码体制，实现了高速的加密；
- 优点
  - 针对散列值签名，既保证了传输过程中数据的完整性，也保证了系统运行效率；
  - 信息采用对称算法加密在网络中传输，既保证了信息的安全性，系统效率得到保证；

3.7.1.3. 两台计算机 A 和 B 传送数据的几种方式

1. 利用对称密钥加密传送
   - 优点：加密速度快；
   - 缺点：无法安全传送密钥；
2. 利用非对称密钥传送
   - 优点：可以实现完全保密；
   - 缺点：速度很慢；
3. 利用非对称密钥传送对称密钥（数据传输密钥）
   - 优点：既提高了传输速度，也保证了安全性；
   - 缺点：无法实现数字签名；
4. 全部原文均采用接收方公钥加密，接收方采用私钥解密
   - 优点：实现了数字签名；
   - 缺点：加密速度慢；
5. 生成原文的散列值，发送方仅对散列值采用自己的私钥加密
   - 优点：实现了数字签名；
   - 缺点：原文采用明文发送；
6. 数字信封
   - 过程：
     1. A 生成原文 M 的散列值 MD，A 用自己的私钥加密 MD，得到数字签名 DS；
     2. A 用对称密钥K_{AB}将原文 M 和 DS 加密，得到 E；
     3. A 用 B 的公钥加密K_{AB}，生成数字信封并和 E 一起发送给 A；B 解密得到原文 M 和 DS；
     4. B 根据 M 计算其散列值并与 DS 比较，如相同，则 M 没有修改过。
   - 优点：敏感数据传输过程中不会被修改；缺点：数据可能遭受“中间人”攻击。
   - 数字信封=（密文对称密钥加密+摘要签名对称密钥加密）私钥加密

3.7.1.4. C/S 网络环境中保护用户口令安全的用户身份认证方法

- 用户身份认证过程的实质，是获得注册时存放在服务器的私钥和公钥
- 认证过程
  1. 服务器新建用户时，服务器生成服务器密钥对$(K_{\text{PUB-S}}, K_{\text{PRI-S}})$和用户密钥对$(K_{\text{PUB-U}}, K_{\text{PRI-U}})$。
  2. 密钥除用户私钥外采用明文保存在服务器中，用户私钥$K_{\text{PRI-U}}$加密后保存，即利用 DES 算法，以用户口令D_{PW}为密钥加密后存放在服务器中，结果如下，记为结果 1:

$$[K_{\text{PRI-U}}]D_{\text{PW}}$$

  3. 客户机登录时，客户机将用户的账户名传送给服务器；
  4. 服务器将自己的公钥$K_{\text{PUB-S}}$和用户的公钥$K_{\text{PUB-U}}$及用户私钥加密后的结果 1 传送给客户机；
  5. 客户机采用 DES 加密算法，以用户口令为密钥，解密结果 1，如果口令正确，则得到用户私钥$K_{\text{PRI-U}}$，客户机得到服务器的公钥以及自己的公钥和私钥；

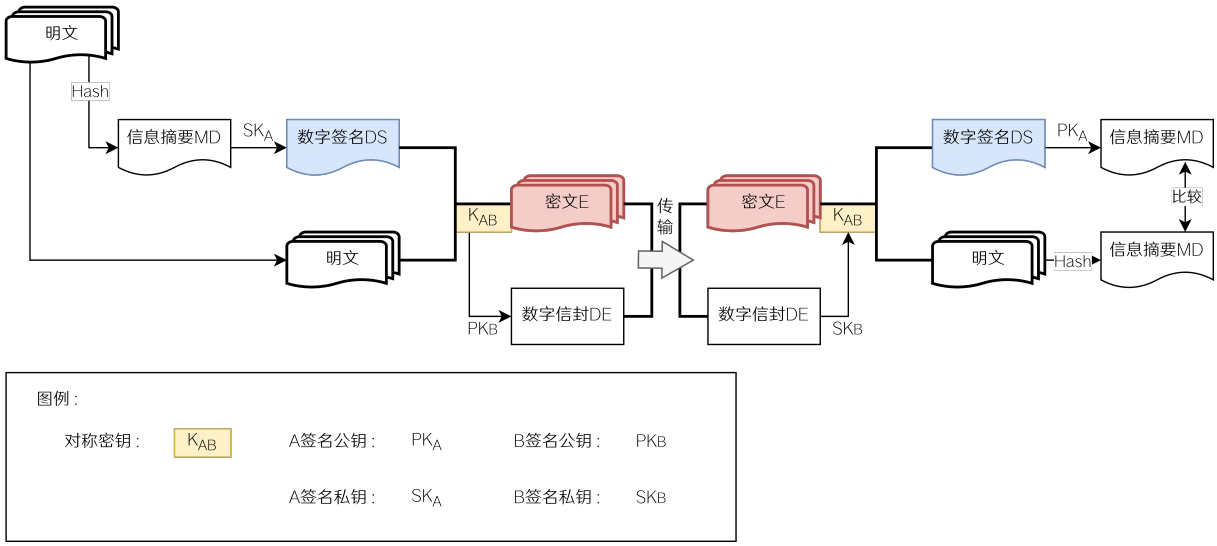
6. 客户机将自己的公钥在网络中公布，利用自己的私钥实现与服务器以及其他用户的加密数据传输。

- 优点：在整个认证过程中，没有在网络中传输用户的口令。

3.7.1.5. 数字签名和电子签名的关系

- 电子签名的定义：能够在电子文件中识别双方交易人的真实身份，保证交易的安全性和真实性以及不可抵赖性，起到与手写签名或者盖章同等作用的签名的电子技术手段，称为电子签名。
- 分类
  - 数字签名：电子签名包含数字签名
  - 手写签名和图章的模式识别
  - 生物识别技术

3.7.1.6. 数字信封示意



3.7.2. 公钥基础设施（Public Key Infrastructure，PKI）

3.7.2.1. 公钥密码系统存在的问题（“中间人”攻击）

- 攻击描述：
  - 网络中 A、B 进行通讯时，如果 C 能够截获公钥的交换，C 就可以截获 A 和 B 之间的所有通讯。（C 向 A 发送自己的公钥，但故意伪装成 B 的公钥；C 再向 B 发送自己的公钥，故意伪装成 A 的公钥）。攻击者 C 处在 A,B 之间，称之为中间人攻击
  - 产生该问题的实质是 A 无法确定他得到的公钥是否真的属于 B
- 解决方法：解决方法是建立一种遍布整个网络的信任机制，其目的在于每个经过认证的公钥均是可信的

3.7.2.2. PKI 的组成和功能

- CA（认证机构）
  - 功能：
    - CA 使用自己的私钥对申请方提交的证书申请进行签名，来保证证书数据的完整性。CA 的私钥是高度保密的，而公钥是在网络中公开的；
    - CA 给用户公钥进行签名，保证公钥均是可信的，使攻击者无法替换
  - 数字证书的颁发过程：
    1. 用户获得认证中心的公钥；
    2. 用户生成自己的密钥对（保证私钥不在网络中传输），并将公钥和部分个人信息采用认证中心公钥加密后传送给认证中心；

3. 认证中心根据提交的信息来核实身份，并检查是否为用户发送而来；
4. 认证中心将用户个人信息和他的公钥信息进行签名后，生成该用户的证书；
5. 证书回传给用户，用户拥有了经过认证且可被他人信任的公钥。

- 证书和证书库
  - CA 存储已签发的证书及公钥，提供给用户查询；
- 密钥备份及恢复系统
  - 保存用户的密钥和证书以备恢复时使用；
- 密钥和证书的更新系统
  - 密钥和证书是有时效性的，到期后，用户需重新申请；
- 证书历史档案
  - 保存用户所有的证书
- 应用接口系统
  - 供用户应用程序编程时调用
- 交叉认证
  - 解决了各个 CA 机构互不关联的问题，从而实现了“交叉认证”

3.7.2.3. 数字证书

- 作用：
  1. 数字证书提供了一种在网上验证身份的方式，用于互联网中的身份认证，一个数字证书对应着一个用户；
  2. 数字证书的作用是证明证书中列出的用户身份经过确认，且合法拥有证书中列出的公开密钥。
- 数据结构
  1. 数字证书是一个经证书授权中心（第三方）数字签名的包含公开密钥拥有者信息以及公开密钥的文件。最简单的证书包含一个公开密钥、名称以及证书授权中心的数字签名。
  2. 授权中心数字签名（身份认证方公钥+身份信息）
- 优点：使用了数字证书，即使您发送的信息在网上被他人截获，甚至您丢失了个人的账户、密码等信息，仍可以保证您的账户、资金安全。（因为密码没有丢，可以防止中间人攻击）

3.7.2.4. 数字证书的使用（用户与用户之间建立安全通信）

1. 用户 A 将自己的证书发给 B；
2. B 验证 A 的证书的完整性，用获得的 CA 的公钥解密证书，获得 A 的身份信息和 A 的公钥；
3. B 将自己的证书发给 A，重复上述过程，获得 A 的公钥；
4. 双方相互验证通过，建立安全连接。

总结：相互验证经过认证的公钥。保证通信双方都是真实的。

3.7.2.5. SSL 协议

- SSL（Secure Socket Layer)中文含义是“安全套接字协议”，是一种使用公钥和私钥技术组合的安全网络通讯协议，可以在客户端和服务器之间建立安全通道，端口为 443；
- SSL 协议工作在应用程序协议（如 http、Telnet、NMTP、FTP 等）和 TCP/IP 协议之间，其作用包括以下几个方面：
  1. 为 TCP/IP 连接提供数据加密
  2. 服务器认证
  3. 消息完整性
  4. 客户机认证
- SSL 可以通过数字证书实现通信双方实体认证；
- SSL 协议应用（网上银行数字证书工作流程）

1. 客户机浏览器发送一个连接请求给安全服务器；
2. 服务器将自己的证书，以及同证书相关的信息发送给客户浏览器；
3. 客户机浏览器检查接收的服务器证书的合法性，如通过，则认可服务器的身份，执行后续部分；
4. 服务器要求客户发送客户证书，接收后检查客户的证书，如通过，则建立连接；（双方彼此获得了对方经过认证的公钥）
5. 客户机浏览器通知服务器自己所能支持的对称密码算法，并发送给服务器；
6. 服务器选择一种加密程度最高的对称密码算法，并用客户机的公钥加密后发送给客户机；
7. 浏览器针对服务器选定的算法，生成通话密钥，用服务器公钥加密后发送给服务器；
8. 服务器接受浏览器发送过来的加密后的密钥，采用私钥解密后获得通话密钥；
9. 客户机和服务器采用对称加密方式完成后续通讯。

总结：首先通信双方通过证书进行身份验证。之后。采用私钥发送对称密钥。最后，两方采用对称密钥通信

3.8. 其他

3.8.1. 密码学经过了几个发展阶段
- 基于直觉和经验进行密码分析
- 香农发表《保密系统的信息理论》，密码学建立
- Diffe 和 Hellman 发表《密码学新方向》，开创了公钥密码学

3.8.2. DES 密码体制中 f 函数如何将 32bit 扩展成 48bit
- 将 32bit 分成 8 组，每一组 4bit 数据
- 将每一组 4bit 数据加上它左右两边的数据后，就扩展成了 6bit
- 6*8=48bit

3.8.3. 概述公钥密码产生的原因以及优势和不足
- 产生的原因：
  1. 单钥密码体制的缺陷：在密文传输之前，通信双方必须使用一个安全渠道协商加密密钥。
  2. 如何为数字化的信息提供一种签名方法。
- 优势：
  1. 无需传输加密密钥
  2. 可以提供数字签名
- 不足：计算开销大

3.8.4. RAS 密码体系中，如果密文 C=10,它对应的公钥 e=5，n=35，请问明文是什么？

$$c = m^e (\mathrm{mod}\, n)$$

根据公钥和密文无法解密出明文

3.8.5. 为了加强 RSA 的安全性，对 p 和 q 的选取有什么要求
- n 如果被破解了，RSA 的安全性就不能被保障。
- 攻击者可以通过 n 分解出 p，q 和 d。
- 因此 p 和 q 应选取尽可能复杂的大素数。

3.8.6. 散列函数为什么可以用于消息的完整性验证？
- 散列函数将任意长的信息映射为较短的，固定长度的值。这个值称为消息摘要
- 它提供一种错误检测的能力。即改变消息中的任何一个 bit，消息摘要都会发生变化

3.8.7. 数字签名的基本原理是什么？

- 它的实现基础是非对称加密算法
- 发送者使用自己的私钥对信息进行签名。这样，只有对应的公钥才能进行解密。而私钥只有发送者采用用户，因此确定签名来自于发送者

3.8.8. 数字信封的应用过程

- 要求同时实现数字签名和加密。这种情况下使用“数字信封”
- 将原文用对称密钥加密，将对称密钥用接收方的公钥发送给对方。对方收到电子邮件，用自己的私钥解密信封，取出对称密钥解得原文。

3.8.9. PKI 应用在一种什么样的信任环境下？它由哪些部分组成

- PKI 应用在一种需要有第三方信任机制的场景下，它可以防止中间人攻击。确保通信双方的身份不存在问题
- 组成：
  1. 认证机构（CA）
  2. 证书和证书库
  3. 密钥备份与恢复系统
  4. 密钥和证书更新系统
  5. 证书历史档案
  6. 应用接口
  7. 交叉认证

4. 信息系统安全

4.1. BLP 模型

BLP 模型（Bell-LaPadula Model）是计算机安全领域中最经典的强制访问控制模型之一，由 David Bell 和 Leonard LaPadula 在 1973 年提出。它主要用于保护机密性，特别适用于军事和政府系统。BLP 模型的主要内容：BLP 模型的核心目标是确保信息不会从高安全级别泄露到低安全级别，即维护信息机密性（Confidentiality）。其主要内容包括以下几个方面：

1. 安全级别：BLP 模型定义了一个安全等级体系，通常包括：
   - 主体（Subject）：主动访问系统资源的实体，如用户或进程。
   - 客体（Object）：被访问的资源，如文件、数据库等。
   - 安全级别（Security Level）：每个主体和客体都被赋予一个安全级别，通常由两个部分组成：
     - 机密性级别（Confidentiality Level）：如 公开（Public）、机密（Confidential）、秘密（Secret）、绝密（Top Secret）等。
     - 类别（Categories）：用于细化访问控制，例如不同部门（军事、外交等）或项目分类。
2. 访问控制规则：BLP 模型通过两条主要规则和一条可选规则来约束信息流动：
   1. 简单安全规则（Simple Security Property，简称 ss-property
      - “不读上”（No Read Up, NRU）：
        - 主体只能读取小于或等于自己安全级别的客体，不能读取高于自己安全级别的客体。
        - 例如，一个机密级别的用户不能读取绝密级别的文件，以防止机密信息泄露。
   2. 属性（Star Property，简称 *-property）
      - “不写下”（No Write Down, NWD）：
        - 主体只能写入大于或等于自己安全级别的客体，不能写入低级别的客体。
        - 例如，一个绝密级别的用户不能向机密级别的文件写入内容，以防止高等级的信息泄露到低等级。

3. 强制完整性约束（Strong Tranquility Property）
    - 主体和客体的安全级别在系统运行时不能随意更改，以防止绕过访问控制策略。
3. 访问模式:BLP 模型还定义了不同的访问模式，例如：
    - 读（Read）：主体读取客体内容。
    - 写（Write）：主体向客体写入数据。
    - 执行（Execute）：主体可以运行客体（如程序）。
    - 创建（Create）：主体创建新的客体。
4. BLP 模型的应用：BLP 模型适用于高机密环境，如：
    - 军事系统：防止机密信息泄露给低级别人员。
    - 政府机构：确保不同级别的人员只能访问他们有权限的信息。
    - 保密数据库：控制不同用户对数据的访问级别。
5. 5. BLP 模型的局限性：
    - 无法保证完整性：BLP 模型关注的是机密性，而不考虑数据的完整性。例如，它不能防止低权限用户向高权限文件写入垃圾数据（这可以通过 Biba 完整性模型补充）。
    - 不能灵活适应动态需求：安全级别不可更改，可能会导致一些实际应用的限制。
    - 缺乏对权限最小化原则的支持：用户可能拥有过多的权限，导致安全风险。

- 总结：BLP 模型是一种基于安全级别的强制访问控制模型，核心思想是“不读上，不写下”，用于防止高机密信息泄露到低级别实体。它广泛应用于军事、政府、保密数据库等需要严格机密性的场景，但在数据完整性、权限管理等方面存在局限性，需要与其他模型（如 Biba 模型）配合使用。

TL;DR：BLP 将非等级类别和等级分类组合起来，实现一个单向的信息流，合法的信息流从低安全级别的主客体流到高安全级别的主客体，逆向的信息流动是禁止的。

4.2. Biba 模型

Biba 模型是由 Kenneth J. Biba 在 1977 年提出的强制访问控制（MAC, Mandatory Access Control）模型，旨在保护系统的数据完整性。与专注于机密性的 BLP 模型（Bell-LaPadula Model）相反，Biba 模型的核心目标是防止低完整性数据污染高完整性数据，即防止未经授权的修改或破坏。

1. 安全级别：在 Biba 模型中，每个主体（Subject）和客体（Object）都被赋予一个完整性级别（Integrity Level），类似于 BLP 模型的机密性级别。完整性级别代表数据的可信度和修改权限：
    - 完整性高（High Integrity）：数据可信，不应受到低级别数据的污染。
    - 完整性低（Low Integrity）：数据不太可信，可能受到非授权修改的影响。
    - 完整性中等（Medium Integrity）：介于高低之间的数据。
2. 访问控制规则：Biba 模型采用两条主要规则，防止低完整性数据影响高完整性数据：
    1. 简单完整性规则（Simple Integrity Property）
        - “不读下”（No Read Down, NRD）
        - 主体只能读取大于或等于自己完整性级别的客体，不能读取低完整性级别的客体。
        - 目的：防止高完整性主体读取低完整性数据，避免数据污染。
        - 示例：一个系统管理员（高完整性）不能从普通用户的文件（低完整性）读取数据，以防止受污染的信息影响高完整性决策。
    2. 完整性规则（Star Integrity Property, *-property）
        - “不写上”（No Write Up, NWU）
        - 主体只能写入小于或等于自己完整性级别的客体，不能写入高完整性级别的客体。
        - 目的：防止低完整性主体修改高完整性数据，保证数据可信度。
        - 示例:普通用户(低完整性)不能修改系统配置文件(高完整性),防止篡改系统关键数据。

3. Biba 模型的访问模式：Biba 模型的访问控制方式可以适用于多种场景：
   - 只读访问：高完整性主体可以查看低完整性数据，但不能写入。
   - 只写访问：低完整性主体只能写入同级或更低级的数据，防止污染高完整性数据。
   - 读写限制：某些关键系统数据只能被高完整性用户读取和修改，低完整性用户无法接触。
4. Biba 模型的应用：Biba 模型主要用于保护数据完整性，适用于以下场景：
   - 操作系统完整性保护
     - 防止普通用户修改系统关键文件（如 `/etc/passwd`、注册表等）。
     - 只允许可信进程修改核心系统数据。
   - 金融系统
     - 低权限用户不能修改银行交易记录，以防止篡改数据。
   - 医疗系统
     - 普通医生可以查看患者记录，但不能修改重要医疗数据。
   - 工业控制系统
     - 低级别进程不能修改关键控制指令，确保设备安全。
5. Biba 模型的局限性：
   - 无法保证机密性
     - Biba 模型不限制信息泄露，高完整性用户可以读取低完整性数据，可能会泄露敏感信息。
     - 解决方案：可以结合 BLP 模型使用，兼顾机密性和完整性。
   - 应用场景受限
     - 主要适用于防篡改场景，对普通数据访问控制作用不大。
   - 灵活性较低
     - 在某些情况下，严格的“不读下、不写上”规则可能会影响正常业务流程。
6. 变种 Biba 模型：为了增强模型的适用性，Biba 模型的变种包括：
   1. 低水位线模型（Low Watermark Model）：允许高完整性主体读取低完整性数据，但读取后会降低其完整性级别，以减少污染影响。
   2. 高水位线模型（Ring Model）：允许主体读取低完整性数据，但不能直接修改高完整性数据，而是通过可信代理进行修改。
7. 结论：Biba 模型是一个强制访问控制模型，核心规则是不读下、不写上，用于防止低完整性数据污染高完整性数据。它适用于操作系统、金融、医疗和工控等对数据完整性要求高的场景，但不适用于机密性控制。Biba 模型通常与 BLP 模型结合使用，以同时实现机密性和完整性的安全保障。

TL;DR：Biba 模型是一个针对完整性安全需求的模型。其访问规则为：主体完整性级别低于客体完整性级别，主体可以读客体；反之，主体可以写客体。

4.3. BLP 模型和 Biba 模型的区别

特性	BLP 模型	Biba 模型
目标	保护数据机密性（防泄露）	保护数据完整性（防篡改）
核心规则	不读上、不写下	不读下、不写上
适用场景	军事、政府、保密数据库	关键系统、金融、医疗、工控
信息流方向	高 → 低 受限	低 → 高 受限
缺陷	不能防止数据篡改	不能防止数据泄露

- BLP 是针对保密性安全需求的模型
- Biba 是针对数据完整性需求的模型

4.4. 自主访问控制和强访问控制的区别

- 自主访问控制将访问的权限交给访问对象也就是客体的拥有者来决定。它具有控制权
- 强访问控制通过设定好的系统机制来控制对客体的访问，个人用户不能改变这种控制，因此又叫基于规则的访问控制

4.5. 信息系统和信息安全概念

- 现代信息系统组成：现代信息系统是指以计算机为信息处理工具，以网络为信息传输手段的系统
- 信息系统安全的基础：操作系统和数据库的安全，是信息系统安全的基础

4.6. 安全操作系统

- 特点
  1. 安全思路是从加强操作系统自身的安全功能和安全保障出发；
  2. 最终目标是保障其上应用的安全乃至最终信息系统的安全；
  3. 安全机制在操作系统层面实施，并为应用层的安全提供底层服务；
- 信息系统安全性实施的两个层面
  - 应用层层面提供安全性：仅能防止从本应用中发起的非法资源访问行为，不能控制其他程序发起的攻击行为
  - 操作系统层面提供安全性
    1. 安全机制对所有的应用均可生效；
    2. 可以为特定应用程序提供资源封装（防止某应用访问特定资源，例如程序自身之外的内存）和自身保护。

4.7. 安全操作系统的基本概念

- 访问控制机制是安全操作系统的核心内容；
- 安全策略是 OS 一系列安全需求的规范；
- 访问控制的两种类型
  - 自主访问控制：确认身份的个人用户，可以对所属的资源访问控制机制来制定规则。
  - 强制访问控制：个人用户不能更改，仅能由系统管理员完成修改的访问控制机制。
- 可信计算基(TCB):计算机系统内保护装置的总体,包括硬件、固件、软件和负责执行安全策略的组合体。它建立了一个基本的保护环境并提供了一个可信计算系统所要求的附加用户服务。

4.8. 安全操作系统主要安全技术

- 内存保护：进程只能访问自己的存储空间
- 客体重用：
  - 当主体获得对一个已被释放的客体（存储空间）的访问权时，当前主体不能获得原主体活动所产生的任何信息。网络环境下,客体的表现形式一般为报文的缓冲区
  - 不能利用之前的信息
- 身份标识和认证：标示用户的身份，认证用户使用系统的合法性
- 访问控制：控制主体访问客体的行为，防止出现非法访问
- 可信路径：实现用户与可信软件之间、可信软件与可信软件之间的可信通信
- 加密支持：提供信息加密和解密以及密钥的管理
- 特权管理：实现特权的合理划分、使用和管理，以便支持最小特权原则
- 安全审计：通过日志记录、管理和报告安全相关行为信息
- 多安全策略的灵活支持：根据应用需要，支持多种安全策略的灵活配置和选择

- 隐蔽信道处理：防止利用非法隐蔽通信途径泄露信息
- 完整性和可用性保护：防止系统或数据遭受非法篡改

4.9. 身份标示与认证的基本方法

- 合法用户身份的标识和认证是 OS 安全控制的基础
- 身份标识的基本方法
  - 定义：为用户建立能够确定其身份状况的信息的过程称为对用户进行身份标识（新建用户）
  - 方法
    1. OS 为每个合法用户建立一个账户，每个账户存放一组管理信息；
    2. 所有帐户的管理信息构成系统的用户账户信息数据库；
    3. OS 使用账户名来建立用户与账户之间的连接，并要求账户名具有唯一性；
    4. OS 自动为用户生成唯一对应的内部身份标识(UID)，并建立账户名与内部身份标识的对应关系。
    5. OS 将具有相同性质的多个用户组成一个用户组，使用组名和组标识号（GID）来表示用户组。
- 身份认证的基本方法
  - 定义：确认用户的合法身份的过程称为对用户进行身份认证。
  - 方法
    1. 用户使用 OS 之前首先提供账户名和密码；
    2. OS 检索账户信息数据库，存在该账户则执行后续操作，否则退出；
    3. OS 检索到账户信息后，将用户输入的口令与账户信息中的口令进行对比，如果相同，则身份确认成功，否则退出。

4.10. 口令信息的管理方法

- 口令信息的维护与应用
  - 明文存放口令信息；
    - 缺点：无法保证口令的安全
  - 用确定的算法针对口令进行加密运算，然后存放口令的密文。身份认证时，系统解密口令并与用户输入口令明文进行对比
    - 优点：可抵御口令猜测攻击；
    - 缺点：攻击者用非法获取的密文猜测明文的方法去破解系统的全部口令
  - 用确定的算法针对口令进行某种运算，然后存放运算的结果。身份认证时，系统将输入的口令进行相同的运算，针对存放结果进行对比
    - 优点：攻击者即使获得账户信息数据库存储的密码，仍然很难破解口令。攻击者无法获得明文消息。
    - 一般采用此种办法
  - 口令作为加密运算中的密钥存在（而不是明文），用户登录时输入的口令执行相同的运算，并将结果与用户账户数据库中的结果对照
- 密码和口令的区别
  - 当你输入一串字符，如果不经过任何处理直接送到服务器来验证，它一定不是密码，只是一个口令
  - 如果输进去的字符，通过密码运算得出另外一个结果，那么这个结果可以验证你是否是合法的用户时，这个口令就变成了密码
- 口令管理中的撒盐措施
  - 定义：在 OS 口令管理中，给口令拌入随机数的过程称为给口令撒盐，拌入的随机数称为口令的盐值。

- 实现方式：给原始口令加入盐值，再把加盐后得到的结果进行加密运算，将加密运算后得到的结果作为要保存和使用的口令信息。
- 优点：撒盐措施实质是增加了密码的长度和随机性，从而有效防范字典攻击。
- 字典攻击：攻击者根据口令信息的加密算法，计算出所有口令的加密结果，然后将用户口令对应的加密结果与事先计算的结果进行对比，如相同，则获得了用户的口令。

- 口令信息与账户信息的分离
  - 为了减少用户得到加密后口令的概率，应该将口令字段信息从账户信息数据库中分离并单独存放，且只允许具有特定权限的用户方可查看。

4.11. 面向网络的身份认证

- 认证信息的网络化管理
  - 网络中身份认证的特点
    1. 用户希望实现“一次登录，整个网络为其提供服务”的登录方式；
    2. 网络中每台主机均存储整个网络的用户身份认证信息是不现实的；
  - 服务器集中身份认证
    1. 网络中所有用户身份认证信息集中存放在网络服务器中，客户机不负责身份认证；
    2. 缺点：安全性能较差
  - 混合身份认证
    1. 客户机和服务器均维护本地的身份认证数据库，共同完成身份认证的工作；
    2. 客户机在本地账户记录信息中增加“+”或“-”前缀。“+”前缀表示由服务器认证，“-”前缀表示由客户机认证。
- 认证信息的加密传输
  - 条件：
    1. 服务器密钥$(K_{\text{PUB-S}}, K_{\text{PRI-S}})$和用户密钥$(K_{\text{PUB-U}}, K_{\text{PRI-U}})$保存在服务器中；
    2. 用户的私钥$K_{\text{PRI-U}}$采用 DES 算法，采用口令D_{PW}为密钥加密后存放，结果如下，记为结果 1：

$$[K_{\text{PRI-U}}]D_{\text{PW}}$$

  - 传输过程：
    1. 客户机将账户名传送给服务器；
    2. 服务器将自己的公钥$K_{\text{PUB-S}}$和用户的公钥$K_{\text{PUB-U}}$及结果 1 传送给客户机；
    3. 客户机采用 DES 加密算法，以用户口令为密钥，解密结果 1，如果口令正确，则得到用户私钥$K_{\text{PRI-U}}$，客户机得到自己的公钥和私钥；
    4. 客户机的服务器均获得了自己的公钥和私钥，采用双钥密码体制交换会话密钥，采用单钥体制传输数据。
- 面向服务的再度认证
  - 在一个网络环境中，用户必须拥有服务认可才能获得特定的服务，请给出用户身份认证和服务认证的一种方法，要求不能在网络中传输明文的或加密过的用户口令，而且，不能使用双钥密码算法。
    - 一种可行的方法是使用挑战认证（Challenge-Response Authentication）。具体步骤如下：
      1. 用户向服务器发送身份认证请求。
      2. 服务器随机生成一个 challenge，将 challenge 发送给用户。
      3. 用户使用 hash 函数计算 challenge 加上用户的口令的散列值，并将散列值发送回服务器。
      4. 服务器也使用同样的方法计算 challenge 加上用户口令的散列值。

5. 如果服务器计算得到的散列值和用户计算得到的散列值相同，则认为用户通过了身份认证，可以提供相应的服务。
   - 这种方法的主要原理是，通过挑战和响应的方式，消除了明文口令的传输和储存，同时也不需要使用双钥密码算法。另外，可以使用安全的 hash 函数和随机数生成器来确保认证的安全性。

4.12. 基于权限位的访问控制

- 访问权限的定义与表示
  - 用户对文件的三种基本操作：r（读）、w（写）、x（执行）权限
  - 计算机中的权限表示方式
    - 采用二进制位来表示。一个二进制位表示一种操作，0 为不允许，1 为允许
    - 例如：001：只能够执行，不能读也不能写
- 用户的划分与访问控制
  - 文件的“属主”：文件的创建者称为文件的属主，一个文件只有一个属主
  - 用户的“属组”
    - 一个用户可以隶属多个用户组，可确定一个用户组作为用户的主用户组；
    - 属主所隶属的主用户组称作该文件的“属组”。
  - 系统中用户的分类（针对一个文件）
    - 属主类：只包括一个用户（文件的创建者）
    - 属组类：文件属主所属主用户组中的其他用户
    - 其余类：除去以上两类的其他用户
  - “属主/属组/其他”式访问控制（基于权限位的访问控制）
    - OS 为每个文件均设立 3 组，每组 3 位，共计 9 位二进制位，3 组分别对应着“属主/属组/其他”三类用户的访问权限。
    - 优点:当一个用户试图访问一个文件时,只要我们为该文件定义了三类用户对它的访问权限，就一定能找到与该用户匹配的访问权限，从而控制该用户对该文件的访问。
- 访问控制算法
  - 首先确定用户是“属主”、“属组”和“其余”中的哪类用户，然后根据为该类用户分配的权限进行判定。

4.13. 进程的有效身份和权限

- 进程与文件和用户的关系
  - 用户启动的进程，是代表用户进行工作的，该进程是用户的化身
  - 进程对应的程序文件 F，启动该进程的用户要对文件 F 具有执行权限
  - 启动进程的用户对某个文件的访问权限作为该进程对该文件的访问权限
- 进程权限判断使用的数据结构
  - 用户属性：用户标识和用户组标识
  - 文件属性：文件属主、文件属组和访问权限位串
- 进程权限判断过程
  - 案例：进程 P 请求对文件 F 的访问
  - 条件：
    1. 进程 P 的用户标示 lup，用户组标示 lgp
    2. 文件 F 的属主 luf，属组 lgf 和访问权限位串 S1S2S3（S1：属主类权限位串；S2：属组类权限位串；S3：其他类权限位串）
  - 判定过程
    1. lup=luf 时，检查 S1 是否有相应权限，然后根据结果决定操作并结束判定（进程 P 创建者是文件 F 的属主）

2. Igp=Igf 时，检查 S2 是否有相应权限，然后根据结果决定操作并结束判定（进程 P 创建者与文件 F 的属组是同一个组）
3. 检查 S3 是否有相应权限，然后根据结果决定操作并结束判定

- 总结：进程 P 请求对文件 F 的访问，根据用户的身份，结合访问权限位串，判定权限。

4.14. 基于访问控制列表（ACL，Access Control List）的访问控制

- 基于权限位访问控制的缺点
  1. 只能区分三类用户，粒度太粗
  2. 针对给定的一个文件，同类用户无法分配相互独立的访问权限
- 基于 ACL 的访问控制优点
  1. 可以实现细粒度用户权限划分
  2. 针对一个给定的文件，可以为任意个数的用户分配相互独立的访问权限
- ACL 的表示方法
  - 实现 ACL 功能的数据结构
    1. 属主、属组和其余的结构扩展为属主、指定用户、属组、指定组、其余五类用户
       - “指定用户”类可以包含任意个数的相互独立的用户
       - “指定组”可以包含任意个数的相互独立的用户组
    2. 每个文件具有一张表，用于存放 ACL 信息（ACL 表）
       - “指定用户”类的每个用户占一行
       - “指定组”类的每个用户组占一行
       - 其他每类用户占一行
    3. 设立“mask”行，表示该文件所有用户权限的最大值（最大权限是什么）
- 基于 ACL 的访问判定
  - 访问判定需考虑的问题
    1. ACL 表中如果存在 mask 行，首先将用户权限和 mask 行权限进行“逻辑与”运算，再将得到的结果进行后续判定
    2. 需要考虑一个用户同时与多个组关联

4.15. 加密文件系统

- 访问控制机制的局限性
  1. 访问控制机制的针对信息的保护作用必须在 OS 的保护下才能够生效，信息一旦离开 OS 环境，保护则完全失效
  2. 例如通过移动介质拷贝的文件
- 加密文件系统的定义：在 OS 中，具有对文件进行加密和解密功能的文件系统称为加密文件系统
- 加密文件系统的应用方法
  - 基本概念
    1. 文件系统：OS 中针对文件进行组织和管理的数据结构，称为文件系统
    2. 目录树结构：由目录和文件构成的倒立的树形层次结构
    3. 根目录，子目录
    4. 面向单个文件的加密方法：实现对任意给定的一个文件的加密
    5. 面向文件系统的加密方法：实现对任意给定的一个文件系统中的所有文件的加密
  - Unix 文件系统特点
    1. 一个文件系统 A 可以安装到另一个文件系统 B 的目录中，也就是说，一个子目录结构对应的可能是一个新的文件系统，优点在于可以实现对给定的任意一个子目录树的所有文件进行加密

2. 允许将任意一个目录定义为加密文件系统的安装点。加密机制自动地对通过加密安装点写入到文件系统的文件进行加密，自动对通过加密安装点从文件系统读出的文件进行解密

- 加密文件系统的基本原理
  - 堆叠式文件系统 :在现有的文件系统之上叠加一层新的机制 ,从而为文件系统增加新的功能，例如加密/解密功能
  - eCryptfs 加密文件系统工作原理
    - eCryptfs 加密文件系统简介
      1. eCryptfs 是一个 Linux 操作系统中利用堆叠式文件系统技术实现的加密文件系统
      2. eCryptfs 加密文件系统堆叠在 ext3 等现有文件系统之上的
    - 工作过程 : 相比正常文件调用 , eCryptfs 加密文件系统增加了一个调用 eCryptfs 加密模块，读取文件内容并解密的过程
- 加密算法的加密密钥
  - 相关概念 :
    1. FEK（File Encryption Key，文件加密密钥）
       - 作用：FEK 用于实际加密文件的内容，每个文件都有一个唯一的 FEK。
       - 生成方式：由操作系统（OS）随机生成。
       - 存储方式 : FEK 不直接存储在文件系统中 , 而是经过加密后存储在文件的头部。具体来说，FEK 会被 FEKEK（文件加密密钥的加密密钥）加密后，形成 EFEK，然后存储在文件头信息中。
    2. EFEK（Encrypted File Encryption Key，加密的文件加密密钥）
       - 作用：EFEK 是 加密后的 FEK，它存储在加密文件的元数据（文件头）中。
       - 生成方式：通过 FEK 使用 FEKEK 加密得到 EFEK。
       - 存储方式：EFEK 存放在每个加密文件的头信息中。
       - 解密方式：当用户访问文件时，系统会使用 FEKEK 解密 EFEK，从而得到 FEK，再用 FEK 进行实际的文件解密。
    3. FEKEK（File Encryption Key Encryption Key，文件加密密钥的加密密钥）
       - 作用 : FEKEK 负责加密 FEK , 使得即使文件的 EFEK 被窃取 , 也无法直接得到 FEK。
       - 生成方式 : 基于用户输入的口令（Passphrase）生成 : 当用户挂载 eCryptfs 目录时，系统会根据用户提供的口令派生出 FEKEK。
       - 作用范围：一个加密目录只有一个 FEKEK，该目录下所有文件共享同一个 FEKEK。
       - 解密方式 : 当用户挂载 eCryptfs 时，系统会使用用户的口令生成 FEKEK，并用它解密 EFEK，进而恢复 FEK，并最终解密文件数据。

    - 总结 :
      1. FEK：唯一的文件加密密钥，每个文件一个。
      2. EFEK：FEK 经过 FEKEK 加密后的密钥，存储在文件头。
      3. FEKEK：加密目录的全局加密密钥，所有文件共享，基于用户输入的口令生成。
    - 示例：假设你有一个 eCryptfs 加密的文件 secret.txt :
      - 文件加密密钥（FEK）：随机生成，例如 abc123
      - 文件加密密钥的加密密钥（FEKEK）：从用户口令派生，例如 xyz789
      - 加密的文件加密密钥（EFEK）：
        - FEK 通过 FEKEK 加密，得到 EFEK，例如 def456
        - EFEK 存储在 secret.txt 头部
      - 加密文件内容：用 FEK (abc123) 加密 secret.txt 的内容
      - 当用户挂载 eCryptfs 并输入正确的口令，系统计算出 FEKEK (xyz789)，解密 EFEK (def456)，恢复 FEK (abc123)，最终解密 secret.txt 的内容。

- 文件的加密与解密（eCryptfs 加密文件系统+ext3 低层文件系统）
  1. 建立新文件
     1. eCryptfs 层接收来自 VFS 的明文文件内容，并把文件内容划分为数据块；
     2. 如果为新文件，eCryptfs 层为文件随机生成加密密钥 FEK，并调用 OS 内核的加密功能函数，利用该密钥实现针对明文的逐块加密，并将加密后的数据块按照原来的顺序组合在一起；
     3. eCryptfs 层将 FEK 利用 FEKEK（文件加密密钥的加密密钥）作为密钥进行加密，得到 EFEK（经过加密的文件加密密钥），继而将 EFEK 等信息作为加密文件的头信息存放在加密文件中；头信息的作用是描述文件内容的加密方法，文件的加密密钥经过加密后作为头信息的一部分嵌入到文件内容中，随文件一起保存和传输；
     4. 将加密后的文件内容（头信息+加密后文件内容）交给 ext3 层，并最终存放在磁盘中。

     - 总结：跟 TCP/IP 中的数据报等方式一样，增加头部信息。主题是加密文件，头部是加密文件信息
  2. 读文件：逆序执行

  - eCryptfs 加密过程总结：
    - 文件加密：
      1. OS 生成随机 FEK
      2. 用 FEKEK 对 FEK 进行加密，生成 EFEK，存入文件头部
      3. 用 FEK 对文件内容进行实际加密
    - 文件解密：
      1. 用户提供口令，系统生成 FEKEK
      2. 用 FEKEK 解密文件头部的 EFEK，得到 FEK
      3. 用 FEK 解密文件内容
- 文件加密密钥的加密密钥（FEKEK）生成方法（基于口令）
  1. 安装加密文件系统时，将安装口令参与特定运算，运算结果作为 FEKEK；
  2. 将 FEKEK 保存在 OS 内核的密钥环中，作为该加密文件系统中所有文件的 FEKEK；
  3. 解密时，OS 首先验证用户输入的访问口令，再决定其是否能够从 OS 内核的密钥环中读取 FEKEK，并执行解密操作。

  - 特点：用户必须提供正确的口令，才能执行加密文件的解密过程
  - 缺点：操作系统需要保存依据口令生成的 FEKEK

4.16. 安全数据库

- 数据库系统组成部分
  - 数据库
  - DBMS（数据库管理系统）
- DBMS 的主要功能
  - 数据库定义：数据库结构
  - 数据库操纵：数据的访问及更新
  - 数据控制：数据安全性控制、数据完整性控制及并发控制
- DBMS 的作用
  1. 提供数据共享，集中统一管理数据
  2. 减少数据冗余
  3. 简化应用程序对数据的访问
  4. 解决数据一致性问题：不同的应用程序访问同一数据，须是相同的结果
  5. 保证数据独立性问题：数据独立于应用程序

4.17. BIOS 简介

- BIOS 全称：基本输入输出系统
- 主要功能
  1. 在 OS 系统启动之前负责完成硬件系统的自检和 OS 引导
  2. 管理系统中的硬件并对外提供编程接口
     - INT 10H 中断：显示中断
     - INT 13H 中断：硬盘中断
- BIOS 的结构：BIOS 的 Boot Block 是 BIOS 中一段特定的区域，包含有用于引导的最小指令集，正常的 BIOS 升级操作不能消除这段信息。开机时，计算机会首先执行根区程序，由它检测 BIOS 文件的完整性。如 BIOS 完好无损时，就会正常引导自检；如发现 BIOS 程序受到破坏就会启用根区程序，但这时程序能够驱动的只有软驱、ISA 显卡、键盘等最“原始”的设备，连硬盘都不能识别。假如这时主板插有 ISA 显卡和 DOS 启动盘的话一样可以引导系统进入 DOS 模式。

4.18. 计算机引导过程

- 加电：通电之后，电扇开始运转，电源指示灯变亮
- 启动 BIOS 引导程序：CPU 是从内存地址 FFFF0H 处开始执行指令的，该地址为系统 BIOS 的地址范围内，放在这里的是一条跳转指令，跳到系统 BIOS 中真正的启动代码处
- 开机自检
  1. 系统 BIOS 的启动代码首先要做的事情就是进行 POST（Power - On Self Test，加电后自检），POST 的主要任务是检测系统中一些关键设备是否存在和能否正常工作，例如内存和显卡等设备。由于 POST 是最早进行的检测过程，此时显卡还没有初始化，如果系统 BIOS 在进行 POST 的过程中发现了一些致命错误，例如没有找到内存或者内存有问题（此时只会检查 640K 常规内存），那么系统 BIOS 就会直接控制喇叭发声来报告错误。正常情况下，POST 过程进行得非常快。
  2. POST 结束之后，系统 BIOS 将查找显卡的 BIOS 并调用它的初始化代码，由显卡 BIOS 来初始化显卡，此时多数显卡都会在屏幕上显示出一些初始化信息，介绍生产厂商、图形芯片类型等内容。系统 BIOS 接着会查找其他设备的 BIOS 程序，找到之后同样要调用这些 BIOS 内部的初始化代码来初始化相关的设备（例如光驱）。
  3. 查找完所有其他设备的 BIOS 之后，系统 BIOS 将显示出它自己的启动画面，其中包括系统 BIOS 的类型、序列号和版本号等内容。然后检测和显示 CPU 的类型和工作频率，然后开始测试所有的 RAM，并同时在屏幕上显示内存测试的进度。
  4. 内存测试通过之后，系统 BIOS 将开始检测系统中安装的一些标准硬件设备，包括硬盘、CD - ROM、串口、并口、软驱等设备，另外绝大多数较新版本的系统 BIOS 在这一过程中还要自动检测和设置内存的定时参数、硬盘参数和访问模式等。
  5. 标准设备检测完毕后，系统 BIOS 内部支持即插即用的代码将开始检测和配置系统中安装的即插即用设备，每找到一个设备之后，系统 BIOS 都会在屏幕上显示出设备的名称和型号等信息，同时为该设备分配中断、DMA 通道和 I/O 端口等资源。
  6. 经过上面几步，所有硬件都已经检测配置完毕，多数系统 BIOS 会重新清屏并在屏幕上方显示出一个表格，其中概略地列出了系统中安装的各种标准硬件设备，以及它们使用的资源和一些相关工作参数。
  7. 接下来系统 BIOS 将根据 CMOS 配置更新 ESCD（Extended System Configuration Data，扩展系统配置数据），ESCD 是系统 BIOS 用来与操作系统交换硬件配置信息的一种手段。通常 ESCD 数据只在系统硬件配置发生改变后才会更新，所以不是每次启动机器时我们都能够看到“Update ESCD… Success”这样的信息。
- 加载操作系统

1. 在 POST 成功之后，系统 BIOS 的启动代码将进行它的最后一项工作，即根据用户指定的启动顺序从软盘、硬盘或光驱启动，定位并加载操作系统文件。
2. 如果是通过硬盘引导，系统将首先执行位于硬盘 0 面 0 道 1 扇区的主引导记录并分析分区表，找到引导分区的第一个扇区，继而执行位于引导分区的第一个扇区中的操作系统引导记录。该记录首先查找两个操作系统文件：Io.sys 和 Msdos.sys。如果这两个文件不存在，引导过程会显示如下信息：“Non-system disk or disk error”或者“Invalid system disk”。
3. 然后，微处理器将试图加载另外一个操作系统文件 Command.com。如果加载失败（Command.com 文件不存在或版本不对），会出现故障信息：“Bad or missing command interpreter”，而且你发出的所有命令计算机都无法执行，因为 Command.com 是专门用来解释这些命令的。

4.19. 其他

4.19.1. BLP 模型中，安全标志为什么是非等级类别与等级分类的组合？

- 一个人也许拥有很高的密级，但他不一定允许查阅部署于他工作范围内的其他部门的低密级信息，这就是著名的“应需可知”原则。

4.19.2. 为什么说安全操作系统是信息安全的基础？

- 因为操作系统的功能是管理信息系统内部的资源。如果操作系统不安全，信息系统处在危险之中。

4.19.3. 什么是可信计算基？

- 他是一个集成平台，是系统内保护装置的总体。包含硬件、固件、软件、和负责执行安全策略的组合体
- 是安全操作系统自身安全的基石

4.19.4. 安全操作系统主要有哪些安全技术

- 身份鉴别
- 标志
- 审计
- 自主访问控制
- 强制访问控制
- 客体重用
- 可信路径
- 隐通道分析
- 形式化分析与验证

4.19.5. 概述客体重用和可信路径的概念

- 客体重用：计算机系统控制资源分配，当一个资源被释放，操作系统会允许下一个用户或程序访问这个资源。但是，已被释放的资源可能残留上次使用的信息，如果一个对某客体没有授权的用户通过资源申请获取了该课题曾经使用过的资源，就可能获取这些信息。这种攻击叫客体重用。
- 可信路径：可信路径为用户和可信计算基之间提供一条可信任的通信途径，保护通信数据免遭修改和泄露。

4.19.6. 安全数据库的威胁包括哪些方面？其安全需求包含哪些？

安全数据库的威胁或者侵犯大致可以分为以下几类：

- 安全数据库的威胁
  - 偶然的、无意的侵犯和破坏，自然事故之类

    - 硬件或者软件的故障导致数据丢失
    - 人为的失误，操作人员输入错误
    - 蓄意的破坏，比如操作人员滥用权限
    - 病毒破坏系统
- 安全需求：4 种
    - 完整性。数据的完整性不能被破坏
    - 保密性。保护敏感信息不会被泄露给未授权的用户
    - 可用性。当系统授权给用户，系统要保证用户正常访问
    - 可追究性。追踪谁修改了信息。

4.19.7. 安全数据库主要使用了哪些技术？

- 身份认证。合法的用户才能进入系统
- 访问控制。确保用户只能访问符合他权限的数据、
- 视图机制。把用户能看的给他看
- 审计机制。监督非法授权访问的情况
- 攻击检测。及时发现系统漏洞
- 数据加密。对重要数据进行加密
- 系统安全恢复。对遭受的破坏的数据库恢复

4.19.8. 骨干网的安全要求有哪些？

- 访问控制。区别用户和管理员的权限
- 鉴别。鉴别从其他网络设备发送来的通信，确定他们的来源。如路由信息。
- 可用性。这是网络最重要的要求。
- 保密性。像一些路由信息，指令信息需要被保护
- 完整性。保证通信双方之间的信息完整性
- 不可否认性。谁做出了改变要承认。

4.19.9. 骨干网面临着哪些安全威胁？

可用性安全威胁可分为三种：

- 可用宽带损耗。宽带是有限的，黑客的攻击可能减少带宽。
- 网络通信信道破坏。黑客可以攻击网络信道。破坏通信双方的通信。
- 网络基础设施损坏。黑客攻击导致设施失控，无法提供设备服务。

4.19.10. 针对骨干网的攻击有哪些？安全措施是什么？

- 被动攻击。它可以监测并收集网络中传输的信息。像窃听信道
    - 措施：使用加密信道。对信息加密传输
- 主动攻击。如拒绝服务攻击，中间人攻击等
    - 措施：建立完整的防范机制。
- 内部人员攻击。内部人员有意或者无意的破坏了网络的可用性。
    - 措施：设置严格的访问控制机制。

5. 可信计算技术

5.1. 可信计算概述

- “可信”的定义：一个实体在实现给定目标时，若其行为总是如同预期，则该实体是可信的。
- “可信计算”的概念（微软）：是一种可以随时获得的可靠安全的计算，使人类信任计算机，就像使用电力系统、电话那样自由和安全。

5.2. 可信计算 TCG 规范

- 基本思想：首先构建一个信任根，再建立一条信任链，从信任根开始到硬件平台，到操作系统，再到应用，一级认证一级，一级信任一级，把这种信任扩展到整个计算机系统，从而确保整个计算机系统的可信，也解决了 PC 结构所引起的安全问题。
- 可信计算平台信任根的组成部分
  - 可信度量根（RTM）：RTM 是指一个能够进行完整性度量的计算引擎，是度量计算系统的起点
  - 可信存储根（RTS）：RTS 是指一个能够可靠进行安全存储的计算引擎
  - 可信报告根（RTR）：RTR 是一个能够可靠报告 RTS 所保存信息的计算引擎
- 完整性度量：任何想要获得平台控制权的实体，在获得控制权之前首先要被度量
- 完整性存储：实体完整性的度量值被可信平台模块（TPM）保存，该过程的度量结果同时被存入内存和硬盘
- 完整性报告：可信平台模块对外提供其保护区域中的完整性度量值、日志中的度量事件和相关证书，咨询的一方可以通过完整性报告判断平台的状态
- 信任链的建立：从可信度量根为起点，建立的过程包含了完整性的度量和存储

5.3. 可信平台模块（TPM）

- 介绍：是一个具备多种密码支持部件、安全功能部件和存储部件的片上系统，由 CPU、存储器、I/O、密码运算器、随机数产生器和嵌入式操作系统等部件组成。
- TPM 使用的加密算法：
  - 输出长度为 160 比特的 SHA-1 杂凑（散列）算法
  - 长度为 2048 位的 RSA 算法（非对称加密算法）
- TCG 密钥体系
  - 签名密钥（Signing Key)
  - 存储密钥（SK，Storage Key）
  - 平台身份认证密钥（AIK）
  - 签署密钥（EK）
  - 绑定密钥（BK）
  - 继承密钥：由 TPM 外部产生
  - 验证密钥：用于保护传输会话的对称密钥。

5.4. 可信计算平台体系结构

- 可信计算平台的定义：是构建在计算机系统中并用来实现可信计算功能的支撑系统，构造以密码技术为基础，TPM 为信任根，可信主板为平台，可信基础支撑软件为核心，可信网络为纽带的体系结构。
- 可信计算平台的组成
  - 可信平台控制模块（TPCM）
    - 以密码算法、密码协议、密钥管理等技术为基础。
    - 物理形式为集成在主板上的硬件芯片
    - 为可信计算提供完整性度量、可信存储根、可信报告及密码服务等功能
    - 作为信任度量的起点
  - 可信平台主板
    - 实现信任链的建立和维护
    - 建立基于 TPCM 的静态信任链
    - 信任链的范围从 TPCM 开始，一直到 OS 内核运行之前
  - 可信基础支撑软件（可信操作系统）
    - 实现信任链向应用的扩展
    - 保障信任链在软件系统的传递，从而保证系统软件的可信性

    - 为应用开发提供必要的标准编程接口
    - 管理可信计算平台的可信资源
  - 可信网络连接架构
    - 实现网络连接
    - 连入网络前对自身进行度量，满足网络要求方可连入
    - 度量自身要连入的服务器，符合安全要求方可连接

5.5. 可信平台控制模块（TPCM）

主要功能：

1. 完整性度量
   - TPCM 上电，并进行初始化
   - 针对位于 Boot ROM 中的 EMM1 模块进行度量（散列运算），并存储度量结果
   - 执行 EMM1 模块，为后续度量操作做好准备，平台开始上电
   - 执行后续操作
2. 完整性存储
   - PCR 是 TPCM 内部用于存储平台完整性度量值的存储单元
   - PCR 具有覆盖存储和递加存储两种存储方式
   - 递加存储的对象可以是一个部件，也可以是一组顺序启动的部件
3. 完整性报告：TPCM 可向外部实体提供完整性度量值报告，所报告的度量值作为判断可信计算平台可信性的依据

5.6. 可信平台主板

体系结构：

- 概述：可信计算平台主板是由 TPCM 和其他通用部件组成的，以 TPCM 自主可信根为核心部件实现完整性度量和存储机制，并实现平台可信引导功能
- TPCM 的启动：TPCM 先于计算机的其它部件启动，包括 CPU

5.7. 可信基础支撑软件（TBSS）

- 可信基础支撑软件的作用
  - 保障信任链在软件系统的传递
  - 为应用开发提供必要的标准编程接口
  - 管理可信计算平台的可信资源
- 软件框架
  - 组成
    - 可信软件基（TSB）
    - 可信基础支撑软件系统服务（TSS）
    - 可信基础支撑软件应用服务（TAS）
- 各模块的作用
  - 可信软件基：包含在 OS 内核中，其作用包括：
    - 完成 OS 内核之上的系统软件与用户应用软件的完整性度量，实现 OS 之上的信任链传递
    - 对外提供系统完整性报告
  - 可信基础支撑软件系统服务：为应用程序提供 TPCM 的证书、密钥、密码功能和完整性数据管理四类接口
  - 可信基础支撑软件应用服务：为用户提供完整性保护、可信认证、数据保护三类应用服务接口

5.8. 可信网络连接（TNC）

- 终端在接入网络之前，对其进行用户身份认证、平台身份认证和平台完整性度量，只有满足安全策略的终端才能被允许接入网络中，目的是使信任链从终端扩展到网络，将单个终端的可信状态扩展到互联网络

5.9. 其他

5.9.1. 如何理解可信计算概念？可信计算和传统的信息安全保护机制的不同点是什么？

- 可信计算的概念：一个实体在实现给定的目标时，若其行为总能符合预期，那么该实体则是可信的
- 不同点：
  - 传统的信息安全保护机制类似于头痛医头，脚痛医脚。哪儿出现问题了，就加强对哪的防护
  - 可信计算是全方位立体式，从设备一启动就开始

5.9.2. 概述可信计算平台的体系结构和主要功能

- 可信平台控制模块：是可信应用的核心控制模块。为可信应用提供物理上的三个根功能：可信度量根、可信存储根、可信报告根。
- 可信平台主板：可信平台控制模块是安装在可信主板上。它主要通过可信度量建立信任链。
- 可信基础支撑软件：向可信计算平台上层应用提供保密性、完整性、身份认证等接口。
- 可信网络连接：控制对外界网络的网络请求，例如可信报告。

5.9.3. 密码算法与可信密码支撑平台的关系

密码算法是可信密码支撑平台的基础，为可信平台实现安全功能提供密码支持。

5.9.4. 概述可信平台控制模块三大功能

- 完整性度量。使用杂凑算法对被度量的对象计算杂凑值。
- 完整性存储。实现对内部数据的安全存储。
- 完整性报告。TPCM 可以向外部实体提供完整性度量值的报告。可以作为判定可信平台可信的依据。

5.9.5. 描述信任链的建立过程

- 信任链的建立从开机到操作系统内核装载完成。
- 信任链以 TPCM 作为信任根，一层一层度量完整值。信任链往上传递。实现信任传递与扩展。

5.9.6. 概述可信基础支撑软件的三个层次

- 可信软件基 TSB。对操作系统内核之上的系统软件和用户应用软件进行度量。
- 可信基础支撑软件系统服务。处于系统服务层，向应用程序提供密码数据等系统服务接口。
- 可信基础支撑软件应用服务。处于应用服务层，向用户提供完整性保护、可信认证、数据保护等应用服务接口。

5.9.7. 概述可信网络连接中三个实体完成的主要功能

- 访问请求者。发出访问请求，请求接入网络
- 访问控制器。控制访问请求者的访问。完成访问请求者的身份鉴别和可信平台评估
- 策略管理器。负责制定可信平台评估策略。辅助访问控制器完成身份鉴别和可信平台评估

5.9.8. 举例说明可信计算的应用

可以应用到咱们学校的信息网络领域。

学校的网络分为内网和外网，可以使用可信计算平台解决安全管理问题。比如其中的网络可信连接，在访问时需要向学校提供他的可信报告，可以用来对访问请求者的身份进行鉴别。同时也可以防止学生落入钓鱼网站的陷阱。