习题1

1.从操作系统安全的角度，如何理解计算机恶意代码、病毒、特洛伊木马之间的关系?

2.从操作系统面临的安全威胁看，密码服务与操作系统安全功能之间的关系如何?

3.操作系统的通用安全需求主要包括哪些?请简要描述这些需求的含义和通用机制。

4.你认为Linux自由软件的广泛流行对我国安全操作系统的研究与开发有什么影响?

5.我国GB 17859—1999将计算机信息系统安全保护能力划分为哪几级?请比较各级之间的主要差别。

6.国际通用准则CC与美国国防部可信计算机系统评测准则相比较，主要做了什么改进?

7.可否用CC准则的保护轮廓PP定义书来对应编写美国TCSEC相应评价级的安全要求?

1. 恶意代码、病毒、特洛伊木马之间的关系：

恶意代码是一个广义的概念，指的是所有设计用来实施恶意活动的软件。病毒和特洛伊木马是恶意代码的两种具体形式。病毒能够自我复制并感染其他文件或系统，而特洛伊木马则通常伪装成合法软件，诱导用户安装，以便执行未经授权的、有害的操作。两者都是操作系统安全的威胁，但病毒更多关注于传播，特洛伊木马则更多关注于掩饰和实施特定的恶意行为。

2. 密码服务与操作系统安全功能的关系：

密码服务是操作系统安全中的一个重要组成部分，它通过加密技术保护数据的机密性、完整性和可用性。密码服务可以用于用户认证、文件加密、网络通信安全等多个方面，是实现访问控制、数据保护和安全通信的关键技术。

3. 操作系统的通用安全需求主要包括：

- 认证：确定并验证用户的身份。

- 授权和访问控制：控制用户对系统资源的访问权限。

- 审计：记录系统活动和用户行为，以便于事后分析和追踪。

- 完整性：保护数据和系统不被未授权的修改。

- 可用性：确保系统和服务正常运行，抵御拒绝服务攻击。

- 机密性：确保信息不被未授权的用户访问。

4. Linux自由软件的影响：

Linux自由软件的广泛流行为我国安全操作系统的研究与开发提供了一个开放的、成熟的平台和丰富的资源。一方面，基于Linux的开源特性，研究人员可以深入了解操作系统的内部机制，有利于安全研究和特性的改进。另一方面，国内外的广泛应用和社区支持也为我国安全操作系统的推广和应用提供了有利环境。

5. 我国GB 17859—1999安全保护能力划分级别：

我国GB 17859—1999将计算机信息系统安全保护能力划分为五级，即D、C、B、A、S级。D级最低，主要针对一般数据处理环境；C级提供一定的安全保护措施；B级要求具有可信路径和较为完善的安全措施；A级提供更高级别的安全保护，需要全面的安全机制和严格的安全策略；S级则是最高级别，适用于国家安全等级别的信息系统，要求非常严格的安全机制和策略。

6. 国际通用准则CC与美国国防部TCSEC（橙皮书）的比较：

CC（Common Criteria）是在TCSEC（也称橙皮书）等多个国家标准的基础上发展起来的国际标准。CC主要改进包括：

- 引入了保护轮廓（Protection Profiles, PP）和安全目标（Security Targets, ST），提供了更灵活的方法来定义产品的安全需求。

- 支持对产品功能和保证需求进行独立的评估。

- 拥有更广泛的国际认可，便于不同国家之间的安全产品评估和认证工作。

7. 使用CC准则的保护轮廓PP来对应TCSEC的评价级：

理论上是可行的，保护轮廓（PP）可以被设计来满足特定的TCSEC评价级别的安全要求。通过这种方式，可以将TCSEC的评价级别映射到CC的评估体系中，从而使得基于TCSEC的旧有评价标准与CC的新评价体系兼容。

习题2

1.安全操作系统的安全功能与安全保证之间有怎样的关系?

2.从操作系统安全的角度如何区分可信软件与不可信软件?

3.在操作系统中哪些实体既可以是主体又可以为客体?请举例说明。

4.如何从安全策略、安全模型和系统安全功能设计之间的关系上，来验证安全内核的安全?

5.为什么要将可信的管理人员列入可信计算基的范畴?这与在讨论操作系统安全时，首先应该相信数据的拥有者不会将该数据泄露给不应看到它的其他用户，或对它进行不适当的修改有什么关联性?

1. 安全操作系统的安全功能与安全保证之间的关系：

安全功能是指操作系统中实现特定安全需求的功能性组件或机制，比如身份验证、访问控制、审计等。安全保证是指确信这些安全功能有效并按预期工作的一系列措施和评估活动，包括正式的验证、评估和测试过程。安全功能是实现安全目标的手段，而安全保证则是验证这些功能正确实现并且可靠的过程。两者相辅相成，安全功能需要安全保证来证明其有效性，而安全保证的目的是确保安全功能的正确实施和操作。

2. 可信软件与不可信软件的区分：

从操作系统安全角度来看，可信软件指的是那些经过验证，能够按照预期安全策略执行操作，不会损害系统安全的软件。不可信软件则可能包含安全漏洞、恶意代码或无法保证按照既定安全策略运行的软件。区分二者的方法通常涉及软件来源的可信度、是否经过安全审核、是否有良好的安全记录和社区支持等。

3. 在操作系统中既可以是主体又可以为客体的实体：

在操作系统中，进程通常既可以作为主体（Subject）也可以作为客体（Object）。例如，一个进程（主体）可以请求读取另一个进程的内存空间（客体），同时，后者也可以作为主体来请求访问文件系统中的文件（客体）。此外，线程、用户和角色等也可以根据上下文既是主体又是客体。

4. 验证安全内核的安全：

要验证安全内核的安全，可以从安全策略、安全模型和系统安全功能设计之间的关系上进行。首先，安全策略定义了系统的安全目标和规则。然后，安全模型是安全策略的形式化表示，用于说明系统如何实施这些安全策略。最后，系统安全功能设计实现了这些模型。验证安全内核的安全需要证明系统安全功能设计正确实现了安全模型，并且确实执行了安全策略。

5. 可信管理人员与操作系统安全的关联性：

将可信的管理人员列入可信计算基（TCB）的范畴是因为人员是实施安全策略的关键因素，他们的行为直接影响系统安全。如果管理人员不可信，即使系统技术上安全也可能因为管理上的失误或恶意行为而受到威胁。这与操作系统安全讨论中的数据所有者信任问题有关；系统需要相信数据所有者不会泄露或不当修改数据，因为操作系统安全机制的有效性部分取决于用户和管理人员的可信行为。如果数据所有者不可信，那么即便操作系统具有强大的安全功能，也无法保障数据的安全性。

习题3

1.Linux系统中运行状态分为用户态和核心態两种，所有的I/O指令只能通过系统调用陷入核心态才能使用。因此基于Linux内核中的程序，可以据此对用户的请求进行完备的访问控制。为什么?

2.在安全操作系统中，对于用户的标识与鉴别需要注意哪些问題?请设计一个用户登录的模拟流程。

3.自主访问控制与强制访问控制是安全操作系统常用的两种访问控制机制，请分別简述两种访问控制的基本内容以及它们之间的异同点。

4.在自主访问控制中常有几种表达访问控制信息的方式，分别简述它们的主要内容并且分析各自的优缺点。

5.为什么在实现了强制访问控制的不同系统中，访问控制的主/客体范疇、控制规则可能会有所不同?

6.在一个安全操作系统中，特权的设置与访问控制机制的关系是怎样的?

7.在一个没有提供可信通路的系统中，如何模拟实现一个特洛伊木马?请给出其流程，并说明如何通过可信通路机制来限制它。

8.在一个安全操作系统中，审计日志空间滿了以后怎么办?请给出几种可行的设计思路。

9.何谓隐蔽通道?隐蔽通道主要分为哪几类?它们的判定规则是什么?

10.当前主流的隐蔽通道分析技术有哪些?请具体给出SRM分析法的主要步骤。

11.请比较隐蔽通道帶宽计算的两种方法在假设前提和计算结果上有何不同。

12.簡述隐蔽通道的处理技术，并分析系统审计在隐蔽通道处理中的作用和优缺点。

1. Linux系统的用户态和核心态：

在Linux系统中，用户态与核心态是两种不同的CPU运行模式。用户态是普通用户程序的运行模式，权限有限，无法直接执行I/O操作和影响系统级的操作。核心态（也称内核态）是操作系统的运行模式，拥有执行任何CPU指令和访问所有硬件资源的权限。所有的I/O指令需要通过系统调用，由操作系统内核代为执行。这样的设计允许内核对用户请求进行完备的访问控制，因为任何对硬件资源的操作都必须通过内核的控制和检查。

2. 安全操作系统中用户的标识与鉴别注意事项与登录流程设计：

用户标识与鉴别需要注意的问题包括：确保用户的唯一性和可追溯性、密码或认证信息的安全存储和传输、防范密码猜测和暴力攻击、多因素认证等。

用户登录模拟流程设计：

- 用户输入用户名。

- 系统根据用户名查找用户相关的安全凭证信息。

- 用户输入密码或使用其他认证因素（如指纹、智能卡等）。

- 系统验证输入的认证信息与存储的凭证是否匹配。

- 如果匹配，授予用户访问权限并记录登录事件；如果不匹配，则拒绝访问。

3. 自主访问控制（DAC）与强制访问控制（MAC）：

DAC基于用户的身份和自主决定的策略来控制访问权限。用户可以控制自己的资源，并决定谁能访问这些资源。MAC则是基于中央策略，由系统根据安全标签（如密级）来决定访问权限，无法被单个用户随意改变。DAC更灵活，但安全性较低；MAC安全性较高，但不够灵活。

4. 表达DAC信息的方式：

- 访问控制列表（ACL）：列出了哪些用户/组可以对对象执行哪些操作。优点是直观、灵活；缺点是管理复杂，随着用户和对象数量增加而变得难以管理。

- 基于角色的访问控制（RBAC）：通过定义角色，并将权限分配给角色来简化管理。优点是提高了管理效率；缺点是可能不够灵活，难以精细控制。

5. 强制访问控制的差异性：

不同系统中MAC的实现可能有所不同，因为不同的系统可能有不同的安全需求、目标和环境。这些差异可能体现在主体和客体的定义、安全标签的设计、访问控制规则的实施等方面。

6. 特权与访问控制机制的关系：

在安全操作系统中，特权是指执行某些受限操作的能力。访问控制机制通过限制哪些主体拥有对特定对象的访问权限来实施安全策略。特权设置是访问控制机制的一部分，用于控制对敏感操作的访问。

7. 模拟实现特洛伊木马和可信通路限制：

在没有可信通路的系统中，可以通过伪装成正常软件或注入恶意代码到合法程序中来实现特洛伊木马。流程可能包括用户下载并执行看似合法的软件，该软件实际上进行了未授权的操作如窃取信息。通过可信通路机制，如要求所有敏感操作都通过安全认证的路径进行，可以限制特洛伊木马的行为，因为它无法绕过这些安全路径。

8. 审计日志空间满后的处理：

- 自动覆盖最旧的日志记录。

- 停止记录新事件直到空间被释放。

- 自动扩展日志存储空间。

- 报警并通知管理员手动处理。

9. 隐蔽通道及其分类：

隐蔽通道是系统中未被设计为传输信息的通道，但可以被恶意利用来传输信息。主要分为存储型隐蔽通道和时间型隐蔽通道。存储型隐蔽通道通过改变系统状态（如文件权限）来传递信息；时间型隐蔽通道通过改变事件的时间特性（如响应时间）来传递信息。判定规则包括是否存在未授权的信息流和是否能够被有效监测。

10. 隐蔽通道分析技术和SRM分析法：

当前主流的隐蔽通道分析技术包括静态分析和动态监测等。SRM分析法的主要步骤包括：

- 定义系统资源矩阵（System Resource Matrix）。

- 分析资源矩阵中的信息流。

- 识别潜在的隐蔽通道。

- 评估隐蔽通道的带宽。

11. 隐蔽通道带宽计算的方法差异：

两种方法的假设前提和计算结果可能不同，例如一种方法可能假设攻击者有无限时间进行传输，而另一种方法可能考虑实际的系统使用情况。因此，计算出的带宽可能会有所不同。

12. 隐蔽通道的处理技术与系统审计的作用：

隐蔽通道的处理技术包括消除、限制和监测等。系统审计可以监测和记录系统活动，帮助检测隐蔽通道的使用，但可能无法完全消除隐蔽通道。审计的优点是能够记录证据，缺点是可能增加系统负担并且依赖于审计策略的有效性。

习题4

1.为什么基于Linux内核中的程序，可以据此对用户的请求进行完备的存取控制?

2.在安全操作系统中，对于用户的标识与鉴别需要注意哪些问题?请简述Linux与Windows的标识与鉴别机制。

3.自主存取控制与强制存取控制是安全操作系统常用的两种存取控制机制，请分别简述这两种存取控制的基本内容以及它们之间的异同点。

4.在自主存取控制中常有哪几种表达访问控制信息的方式?简述UNIX/Linux系统中所使用的控制方式，并分析它们各自的优缺点。

5.为什么在实现了强制存取控制的不同系统中，存取控制的主/客体范畴、控制规则可能会有所不同?

6.在一个安全操作系统中，特权的设置与存取控制机制的关系是怎样的?

7.在一个没有提供可信通路的系统中，如何模拟实现一个特洛伊木马?请给出其流程，并说明如何通过可信通路机制来限制它。

8.在一个安全操作系统中，审计日志空间满了以后怎么办?请给出几种可行的设计思路。

9.找一套最新版本的Linux系统，实际测试一下其所提供的安全功能。

1. 基于Linux内核中的程序进行存取控制：

Linux操作系统的核心态具有执行所有CPU指令和访问所有硬件资源的权限，而用户态则权限受限。所有涉及硬件资源或敏感操作的请求都必须通过系统调用，陷入核心态由内核处理。这为内核提供了拦截、验证和控制所有用户请求的机会。Linux内核利用这种机制，结合文件系统的权限设置、进程的权限（如UID/GID）和可选的安全模块（如SELinux或AppArmor），可以实现对用户请求的完备存取控制。

2. 用户的标识与鉴别问题及Linux与Windows机制：

需要注意的问题包括确保用户标识的唯一性、密码或认证信息的安全存储与传输、防范各种认证攻击等。

Linux通常使用用户名和密码进行用户鉴别，还可以配合PAM（Pluggable Authentication Modules）支持更多认证方法，如二因素认证、生物识别等。文件系统权限基于用户ID（UID）和组ID（GID）。

Windows使用Active Directory服务进行用户标识和鉴别，支持多种认证协议，如Kerberos、NTLM等。权限管理基于访问控制列表（ACL）。

3. 自主存取控制（DAC）与强制存取控制（MAC）：

DAC允许用户控制自己的资源，用户可以设置权限决定谁可以访问他们的文件。UNIX/Linux系统中的文件权限（读、写、执行）是一个DAC实例。

MAC是基于系统强制的策略来控制访问，不受用户自主控制。它通常使用安全标签（例如，密级）来决定访问规则。SELinux是一个实现MAC的例子。

它们的主要区别在于谁负责设置访问策略：DAC由用户设定，MAC由系统强制实施。

4. 表达DAC信息的方式：

UNIX/Linux系统中常见的DAC表示方式是使用文件的权限位（rwx），它们定义了文件所有者、所有者所在组和其他用户的访问权限。优点是简单直观；缺点是难以对大量文件进行精细的权限管理。

5. 不同MAC系统中存取控制差异：

不同系统可能有不同的安全需求和目标，因此它们实现MAC时的主体/客体定义、安全策略和控制规则可能会有所差异。例如，军事系统的密级和商业系统的角色基础访问控制可能完全不同。

6. 特权与存取控制机制的关系：

特权是指执行某些特定操作的能力，通常与系统的存取控制机制密切相关。存取控制机制定义了谁可以执行哪些操作，而特权则是对这些操作的直接授权。通常，只有拥有适当特权的进程或用户才能执行敏感操作。

7. 实现特洛伊木马和可信通路限制：

在没有可信通路的系统中，特洛伊木马可以通过伪装成合法软件或通过软件漏洞注入恶意代码。通过可信通路机制，系统可以确保所有敏感的操作和认证过程都在受保护的环境中进行，从而限制特洛伊木马的潜在危害。

8. 审计日志空间满后的处理：

- 自动覆盖最旧的日志。

- 停止记录新事件，并通知管理员。

- 自动将日志归档到第二存储设备。

- 提前监控日志空间并在接近满时通知管理员。

9. 测试Linux系统的安全功能：

需要选择一个最新版本的Linux发行版，如Ubuntu、Fedora或Debian。测试的安全功能可能包括：

- 用户权限和访问控制（文件权限、用户管理、sudo等）。

- 审计日志（如auditd）。

- 防火墙（如ufw或iptables）。

- MAC机制（如SELinux或AppArmor）。

- 安全加固和配置。

- 安全模块和扩展的支持（如PAM）。

具体测试时，可以通过配置不同的安全策略、模拟攻击行为等方式来验证这些功能的有效性和可靠性。

习题5

1.何谓系统安全需求、安全策略以及安全模型?试述它们三者之间的关系。

2.为什么在高等级安全操作系统中强调使用形式化的安全模型?简述一个形式化安全模型的设计步骤。

3.什么是状态机模型?为什么状態机模型在安全模型中得到了成功的应用，而没有在软件开发中得到广泛推广?

4.简述BLP安全模型，談談已有的一些对BLP安全模型的看法。

5.簡述Biba和Clark-Wilson这两种完整性模型，比较分析它们之间的优缺点。

6.简述BN中国墙模型与侵略型中国墙模型设计思想，比较它们在冲突类定义上的区别。

7.RBAC存取控制机制的要点是什么?试述RBAC存取控制机制的主要类型和各自的特点。

8.试述安全信息流模型结构和隐蔽通道分析之间的关系。

1. 系统安全需求、安全策略以及安全模型的关系：

系统安全需求是指系统为了保护信息不受威胁，需要满足的安全目标和条件。安全策略是为了满足这些安全需求，系统应该遵循的一系列规则和实践。安全模型则是安全策略的抽象和形式化表述，用于说明如何通过技术手段实现安全策略。三者之间的关系是层次性的：安全需求定义了目标，安全策略提供了达成目标的方法论，而安全模型则是策略的具体实现框架。

2. 高等级安全操作系统与形式化安全模型：

在高等级安全操作系统中，形式化安全模型的使用能够提供明确、精确和无歧义的安全保证。形式化模型通过数学方法定义系统行为，易于验证和分析，可提高安全性。设计步骤通常包括：

- 明确安全需求；

- 定义系统状态和状态转换；

- 构造形式化表示；

- 证明或验证模型满足安全需求。

3. 状态机模型：

状态机模型是一种数学模型，它描述一个系统的状态以及在输入下如何从一个状态转换到另一个状态。它在安全模型中得到成功应用，因为它能够清晰地描述系统在接收到不同输入时的确切行为，这对于分析和验证系统安全性至关重要。然而，在软件开发中，由于复杂性和灵活性的需求，状态机模型可能不足以覆盖所有的开发场景，因此没有得到广泛推广。

4. BLP安全模型：

BLP（Bell-LaPadula）安全模型主要用于保护机密性，它基于状态机模型，通过定义安全状态和状态转换规则来保证信息流的安全性。它包括简单安全性规则和星形安全性规则，用于控制信息的读取和写入。对BLP模型的看法包括它在机密性保护方面的强大能力，以及它不涉及完整性和可用性等其他安全层面的局限性。

5. Biba与Clark-Wilson完整性模型：

Biba模型是一个基于状态机模型的完整性模型，它通过“不能读取低完整性”和“不能写入高完整性”来保护信息的完整性。Clark-Wilson模型则通过“访问三元组”（用户、程序、数据项）和“完整性验证过程”来实施事务的完整性和一致性。Biba模型的优点是模型简单明了，缺点是过于严格可能限制正常操作。Clark-Wilson模型的优点是更贴近商业应用，缺点是实现复杂。

6. BN中国墙模型与侵略型中国墙模型：

BN中国墙模型是用于解决利益冲突的安全模型，通过动态控制访问规则来防止信息流向可能导致利益冲突的实体。侵略型中国墙模型则是对BN模型的一种扩展，它考虑到了内部攻击和侵略行为。两者在冲突类定义上，BN模型通常基于数据分类进行冲突类的划分，而侵略型模型可能会进一步考虑实体之间的关系和交互。

7. RBAC存取控制机制：

RBAC（Role-Based Access Control）的要点是将权限分配给角色，而不是直接分配给单个用户。用户通过成为角色的成员来获得权限。RBAC的主要类型包括：

- 核心RBAC：基本角色权限管理。

- 分层RBAC：角色之间可以建立继承关系。

- 受限RBAC：对角色的权限做出一定限制。

- 动态RBAC：角色可以根据上下文动态变化。

各类型的特点在于角色与权限之间的管理复杂性和灵活性。

8. 安全信息流模型结构与隐蔽通道分析：

安全信息流模型结构用于描述和分析在系统中信息如何流动和被处理，它包括了系统状态、状态转换和信息流规则。隐蔽通道分析利用这些模型来识别和量化非预期的信息流通道，即隐蔽通道。通过对系统的信息流模型进行分析，可以识别出潜在的隐蔽通道，并采取措施来消除或减少这些通道的影响。

习题6

1.简述安全体系结构对安全操作系统的意义，以及它的描述层次结构。

2.简述安全体系结构的设计原则和设计目标。

3.用一个典型的访问控制请求分析它在GFAC和Flask结构中处理过程的异同。

4.参考SELinux的实现，介绍Flask体系结构在LSM框架中的实现方式。

5.试利用LSM框架实现Chinese Wall模型。

6.简述权能体系结构的优缺点。

1. 安全体系结构对安全操作系统的意义及其描述层次结构：

安全体系结构为安全操作系统提供了一个框架，用于指导如何设计、实施和维护系统的安全机制。这个框架确保系统的所有部分都按照既定的安全标准和策略协同工作，以达到整体的安全性。描述层次结构通常包括以下几个层次：

- 策略层：定义系统的安全策略和规则。

- 机制层：实现策略的具体技术和方法。

- 实现层：具体的软件和硬件实现机制。

2. 安全体系结构的设计原则和设计目标：

安全体系结构的设计原则包括最小权限原则、防御深度、开放设计、安全默认设置、故障隔离和简单性原则等。设计目标通常关注保证系统的机密性、完整性和可用性，同时还需要考虑到系统的可扩展性、灵活性、透明性和易用性。

3. 访问控制请求在GFAC和Flask结构中处理过程的异同：

GFAC（Generalized Framework for Access Control）和Flask（Flux Advanced Security Kernel）都是访问控制框架，但它们的设计思想有所不同。在GFAC中，访问控制请求通常通过一系列模块化的组件进行处理，这些组件可能包括认证、授权和审计等。Flask则更多强调强制访问控制（MAC）和安全策略的灵活性，其通过安全服务器来提供决策支持，操作系统内核负责执行这些决策。两者的不同在于处理流程的松耦合与紧耦合，以及策略决策的灵活度。

4. Flask体系结构在LSM框架中的实现方式：

Linux安全模块（LSM）框架允许安全模块如SELinux（实现了Flask安全体系结构）钩入内核以提供额外的访问控制。在LSM框架中，SELinux通过安全钩子（security hooks）集成到内核中，在系统调用和关键内部操作之前进行安全检查。SELinux的安全服务器组件（称为安全决策管理器）根据预定义的安全策略来处理安全相关的决策。

5. 利用LSM框架实现Chinese Wall模型：

要在LSM框架中实现Chinese Wall模型，首先需要定义冲突利益类和数据对象的关系，然后在LSM框架中注册相应的安全钩子。当用户尝试访问数据对象时，安全模块将根据Chinese Wall策略检查用户之前的访问历史，以确保用户没有违反利益冲突规则。如果违反，访问将被拒绝。

6. 权能体系结构的优缺点：

权能体系结构（Capability-based architecture）的优点包括精细的访问控制、清晰的权限管理和较高的安全性。系统可以精确地为每个程序或用户分配权能，规定它们能够执行的操作。这种体系结构还可以有效地避免权限的非法传递。缺点可能包括实现复杂性和性能开销。在实际系统中，权能的管理和传递可能很复杂，并且需要额外的机制来防止权能的滥用。

习题8

1.基于一般操作系统开发安全操作系统一般有3种方法，即虚拟机法、改进/增强法、仿真法。请比较这3种开发方法之间的优缺点及它们各自所适用的场合。

2.目前人们大多是通过对Linux内核进行安全性增强的方式来开发所需的各种安全机制。试说明在这个过程中如何保障安全机制的完备性?

3.对照国标GB17859—1999各安全等级的要求，请具体给出《安全标记保护级》和《结构化保护级》安全操作系统在安全保证中对TCB设计与实现的具体要求。

4.结合TCB的设计与实现，请举例说明安全操作系统开发过程中的安全机制的设计与实现。

5.在安全操作系统的开发过程中，如何协调系统的安全性、兼容性及效率三方面关系?

1. 开发安全操作系统的三种方法的优缺点及适用场合：

- 虚拟机法:

优点: 安全性较高，因为可以在虚拟机中隔离不同的操作环境，使得系统和应用的安全性得到增强; 可以同时运行多个操作系统实例，方便测试和部署。

缺点: 性能开销较大，需要额外的资源来维护虚拟化环境; 对硬件的支持要求高。

适用场合: 需要高度隔离的环境，如云计算平台和多用户操作环境。

- 改进/增强法:

优点: 可以针对特定的安全需求进行定制; 通常与现有系统兼容性好，易于部署。

缺点: 可能会引入新的安全漏洞; 对现有系统的理解要求高，错误可能导致系统不稳定。

适用场合: 对现有操作系统进行增强以满足特定的安全需求，如企业服务器和专用设备。

- 仿真法:

优点: 在不影响原有系统的情况下，可以在用户态模拟操作系统功能; 方便开发和测试新的安全特性。

缺点: 性能开销大，可能不适合性能敏感的应用; 仿真层可能成为新的安全漏洞点。

适用场合: 研究和教学，以及在开发早期阶段测试新的安全机制。

2. 保障安全机制完备性的方法：

在对Linux内核进行安全性增强时，保障安全机制的完备性通常涉及以下步骤：

- 形式化验证：使用形式化方法对增强的安全机制进行建模和验证，以确保它们符合预定的安全策略。

- 代码审查：对修改后的内核代码进行严格的审查，以确保没有引入新的安全漏洞。

- 安全测试：通过广泛的测试来检测安全机制的有效性和潜在的漏洞，包括渗透测试和模糊测试。

- 安全评估：引入第三方安全评估，对安全增强的内核进行独立的安全评估。

3. 《安全标记保护级》和《结构化保护级》的TCB设计与实现要求：

安全标记保护级（B级）要求TCB能够实施标记政策，确保信息流控制和访问控制符合安全标记规则。实现要求包括：

- 强制访问控制（MAC）机制，以确保只有适当的安全级别的用户才能访问相应的数据。

- 可信路径，用于安全地进行用户认证。

- 审计机制，记录所有对敏感对象的访问尝试。

结构化保护级（C级）要求TCB提供基础的安全保证，实现要求包括：

- 自主访问控制（DAC）机制，让用户能够控制对其拥有的对象的访问。

- 用户身份认证机制，确保用户必须通过认证才能访问系统资源。

- 审计功能，记录用户的活动，以便在出现安全问题时进行追踪。

4. TCB的设计与实现的安全机制设计与实现示例：

以SELinux为例，它是在Linux内核中实现的一个TCB，提供了基于策略的强制访问控制。SELinux的设计包括：

- 安全策略：定义了不同类型的对象（如文件、进程）和它们之间可能的交互。

- 安全上下文：为系统中的每个对象和主体（如用户、进程）分配一个安全标签。

- 安全决策点：在系统中的关键点（如文件打开、网络连接）进行安全检查，以确保操作符合安全策略。

实现方面，SELinux通过Linux安全模块（LSM）框架集成到内核中，提供了必要的钩子（hooks）来实施安全策略。

5. 协调安全操作系统的安全性、兼容性及效率：

在安全操作系统的开发过程中，需要平衡安全性、兼容性和效率三者之间的关系。以下是一些策略：

- 设计时考虑性能优化，避免不必要的性能开销。

- 使用模块化和可配置的安全机制，允许在不同的安全需求和性能要求之间调整。

- 保持对现有应用和工具的兼容性，尽可能使用标准的接口和协议。

- 定期进行安全审计和性能评估，确保在提高安全性的同时，不会对系统性能造成严重影响。

- 通过教育和培训，提高用户和管理员对于安全操作系统使用的认识，以便更好地利用其安全特性而不影响效率。

习题10

1.简述基本概念：可信计算、可信构建块(TBB)、可信边界、信任链扩展。

2.概括TPM和TCM的三个主要功能。

3.阐述TCM与TPM的主要区别。

4.简述平台完整性证实过程。一个完整的远程证实，你认为主要分为哪几部分?

5.TCG所定义的信任链是如何逐步扩展的?TCG定义的信任链已扩展到哪一步骤?从TCG所定义信任链的终点开始，信任链扩展应该到应用层才可以充分保证系统的可信属性，那么你认为这一步骤该如何完成?

6.保存在PCR中的Hash运算值具有先后性，这个属性在系统的引导过程中确实非常必要。但如果在启动不同的应用程序时，这样的先后顺序是否有必要?如果有必要，请说明理由；如果没有必要，也请说明理由。

7.简述基于可信硬件(TPM/TCM)的可信操作系统的核心机制及其发展趋势。

1. 基本概念：

- 可信计算：指通过硬件和软件的结合，实现对计算平台的安全性和信任性的度量、验证和报告。

- 可信构建块(TBB)：是构成可信计算环境的基本组件，通常包括硬件、固件、软件等，它们可以被验证并被信任以执行特定的安全功能。

- 可信边界：是指在可信计算环境中，可信部分与不可信部分之间的分界线，确保可信部分内的操作不受外部不可信因素的影响。

- 信任链扩展：是指从根信任出发，通过一系列的验证步骤逐步扩展信任到系统的其他部分。每一步都依赖于前一步的验证结果。

2. TPM和TCM的三个主要功能：

- 身份认证：提供设备的唯一身份，用于证明设备的真实性。

- 完整性度量：通过度量计算机启动时的关键组件（如BIOS、引导加载程序、操作系统）的完整性，并将度量结果存储在平台配置寄存器（PCR）中。

- 密钥管理和加密服务：生成和管理密钥，提供加密、解密、数字签名和验证等服务。

3. TCM与TPM的主要区别：

- TCM（Trusted Cryptography Module）是中国自己的可信计算标准，而TPM（Trusted Platform Module）是国际标准，由TCG（Trusted Computing Group）制定。

- TCM通常对中国国内的密码算法有更好的支持，而TPM则遵循国际标准，使用国际通用的密码算法。

- TCM和TPM在细节上可能存在一些不同，比如在隐私保护、密钥生成和管理方面可能有所区别。

4. 平台完整性证实过程：

平台完整性证实过程通常包括度量、存储、报告三个阶段。一个完整的远程证实主要分为以下几部分：

- 本地度量：在系统启动或运行时，度量软件和硬件的完整性。

- 本地存储：将度量值安全地存储在本地的TPM/TCM模块中。

- 证实报告：将存储的度量值通过安全通信传递给远程的验证方进行验证。

- 远程验证：验证方对接收到的度量值进行验证，以确保系统的完整性。

5. TCG所定义的信任链如何逐步扩展，以及目前扩展到的步骤：

TCG定义的信任链从根信任（通常是TPM）开始，然后度量并扩展到BIOS、引导加载程序、操作系统、驱动程序等。目前，信任链扩展到操作系统层。信任链的扩展到应用层可以通过操作系统提供的API或服务完成，例如，使用TPM提供的功能来度量和验证应用程序的完整性。

6. PCR中Hash运算值的先后性在系统引导和应用程序启动中的必要性：

- 在系统引导过程中，先后性非常重要，因为引导过程中各个组件的加载顺序决定了系统的初始完整性状态。

- 在启动不同的应用程序时，先后性可能不是必要的，因为应用程序之间通常相互独立，它们的完整性状态不依赖于加载顺序。然而，在某些情况下，如果应用程序间有依赖关系或者应用程序的加载顺序对系统安全态有影响，保持先后性就变得重要。

7. 基于可信硬件(TPM/TCM)的可信操作系统的核心机制及其发展趋势：

核心机制：

- 完整性度量和存储：系统启动和运行过程中的关键组件会被度量，并将度量值存储在TPM/TCM中。

- 安全启动：确保系统从信任的根（如TPM/TCM）开始，按照信任链顺序启动。

- 密钥和证书管理：TPM/TCM负责生成和保护密钥，为系统提供加密和认证服务。

发展趋势：

- 向更高层次扩展：从硬件和操作系统层面向应用层扩展，提供更全面的安全保护。

- 云计算和物联网的集成：随着云计算和物联网的兴起，可信计算也在向这些领域扩展，以确保数据和设备的安全。

- 隐私保护：在提供安全的同时，越来越多地关注用户隐私的保护，开发更加智能和用户友好的隐私保护技术。

习题11

1.简述基本概念：虚拟机、虚拟机监控器(VMM)、虚拟机迁移、虚拟可信平台。

2.简述虚拟化技术的分类，以及x86架构虚拟化的实现技术有哪些?

3.虚拟化平台的核心技术有哪些?它们的安全性如何?

4.Xen虚拟化平台的虚拟机监控器安全体系结构如何?

5.虚拟机安全监控技术主要有哪些?

6.请举例说明虚拟化平台的隐蔽通道，并论述它们可能给云计算环境带来的风险。

7.简述虚拟可信平台的主要组件和架构，画出简图。

8.构造虚拟机的迁移流程，思考所构造的迁移流程在虚拟可信平台中是否适用?对比虚拟可信平台中的虚拟机迁移和没有可信属性的虚拟化平台中的虚拟机迁移，有何不同?请思考虚拟可信平台中虚拟机迁移应该考虑的安全问题和解决方案。

9.有一种说法“虚拟化技术是云计算的催化剂”,调查当前工业界的虚拟化技术进展情况，简述虚拟化技术和云计算的关系。你如何看待目前业界的研究热点——云计算?云计算会是昙花一现还是会持续发展?如果能够持续发展，是否能够成为计算机领域发展的一个主流

方向?为什么?

1. 基本概念：

- 虚拟机（VM）：是指通过软件模拟的、与底层物理硬件相对独立的计算环境，它可以像物理机一样安装操作系统和应用程序。

- 虚拟机监控器（VMM）：也称为虚拟化管理程序或虚拟机管理器，是一种中间软件层，用于创建、运行和管理虚拟机，同时提供对物理资源的抽象和分配。

- 虚拟机迁移：是将一个运行中的虚拟机或容器从一个物理主机迁移到另一个物理主机的过程，这个过程对最终用户透明，可以实现负载均衡和维护无停机操作。

- 虚拟可信平台：是一种整合了可信计算技术的虚拟化环境，可以提供虚拟机的信任状态度量和验证，确保虚拟环境的可信性和安全性。

2. 虚拟化技术的分类及x86架构虚拟化实现技术：

- 全虚拟化：提供完整硬件抽象，不需要修改客户操作系统，使用二进制翻译技术来处理特权指令。

- 准虚拟化：需要修改客户操作系统以使用一套特定的超调用接口，提高效率。

- 硬件辅助虚拟化：使用处理器的虚拟化支持功能，如Intel VT和AMD-V，来提供高效的全虚拟化支持。

- 操作系统级虚拟化：提供隔离的用户空间实例，通常称为容器或轻量级虚拟化。

3. 虚拟化平台的核心技术及其安全性：

- CPU虚拟化：通过时间片分配确保虚拟机的处理器资源分配。

- 内存虚拟化：通常使用页表和影子页表技术来隔离和管理虚拟机的内存。

- I/O虚拟化：包括直接设备分配和使用中间层软件来抽象设备的访问。

- 网络虚拟化：通过虚拟交换机和网络接口来管理虚拟机的网络流量。

虚拟化平台的安全性面临虚拟机逃逸、共享资源攻击和隔离失败等风险，需要通过安全设计和管理策略来缓解。

4. Xen虚拟化平台的虚拟机监控器安全体系结构：

Xen采用微内核设计，将虚拟机监控器保持在最小功能，主要管理CPU调度、内存分配和设备I/O，同时利用Xen安全模块（XSM）来控制虚拟机的访问权限，增强安全性。

5. 虚拟机安全监控技术主要有：

- 虚拟机内省（Introspection）：监控虚拟机内部状态，检测异常行为。

- 安全审计：记录和分析虚拟机的活动日志，以检测潜在的安全事件。

- 隔离机制：确保虚拟机之间的逻辑隔离，防止资源泄露和交叉攻击。

- 虚拟网络安全：使用防火墙、入侵检测系统（IDS）和入侵防御系统（IPS）来保护虚拟网络环境。

6. 虚拟化平台的隐蔽通道举例及风险：

隐蔽通道可能通过共享缓存、内存页表、网络带宽等资源在不同虚拟机之间传递信息。在云计算环境中，隐蔽通道可能被用于数据泄露、侧信道攻击等，给云服务的安全性和隐私保护带来风险。

7. 虚拟可信平台的主要组件和架构：

虚拟可信平台主要包括可信平台模块（TPM），用于存储度量值和密钥，以及虚拟化管理程序（VMM）来创建和管理虚拟机。还有度量和认证机制，用于验证虚拟机的信任状态。基本架构可以表现为一个三层模型，底层是物理硬件，中层是VMM，顶层是运行在VMM之上的虚拟机。

8. 虚拟机迁移流程及虚拟可信平台中的安全问题和解决方案：

虚拟机迁移流程通常包括准备迁移、内存和状态同步、切换控制和恢复运行等步骤。在虚拟可信平台中，迁移流程还需包括信任度量和验证，以确保目标平台的可信性。与非可信平台相比，需额外关注迁移过程中的度量数据和密钥材料的安全性，解决方案可能包括加密传输、完整性验证和访问控制等。

9. 虚拟化技术与云计算的关系及未来发展：

虚拟化技术是云计算的基础，它允许云服务提供商高效地共享和管理硬件资源，为用户提供弹性的服务。虚拟化技术的发展推动了云计算的普及。云计算目前已是计算领域的一个重要趋势，因为它提供了成本效益、可扩展性和灵活性。未来，随着技术的不断进步和安全性的提高，云计算有望成为计算领域的主流方向。