1. 使用 nohup 命令的作用是？

A) 将进程转为后台运行

B) 忽略挂断信号（SIGHUP），终端关闭后进程仍运行

C) 提高进程优先级

D) 监控进程资源

2. 查找进程 ID（PID）的命令是？

A) find

B) pgrep

C) locate

D) whereis

3. 守护进程（Daemon）的特点是？

A) 始终在前台运行

B) 脱离终端并在后台运行

C) 只能由 root 用户启动

D) 必须通过 & 启动

4. 使用 exec 命令启动新进程时，会发生什么？

A) 创建子进程并保留原进程

B) 替换当前进程的地址空间

C) 将进程转为后台运行

D) 修改进程优先级

5. 如何将进程放到后台运行？

A) 按 Ctrl+C

B) 命令后加 &

C) 使用 fg 命令

D) 按 Ctrl+Z

6. 启动进程时设置优先级为最高，应使用：

A) nice -n -20 command

B) renice -n 20 command

C) priority -max command

D) nice -n 19 command

7.实时监控进程资源使用情况的命令是：

A) top

B) ps

C) lsmod

D) netstat

8. 创建守护进程（Daemon）的关键步骤不包括？

A) 调用 fork() 创建子进程

B) 调用 setsid() 创建新会话

C) 将标准输入输出重定向到 /dev/null

D) 使用 execvp() 加载新程序

9. 以下哪一项不是进程的组成部分（ ）。

A. 程序段

B. 相关数据

C. 进程控制块（PCB）

D. 文件系统

10. 进程在其生命周期内会经历多种状态，以下状态转换顺序正确的是（ ）。

A. 就绪→执行→阻塞→就绪

B. 执行→就绪→阻塞→执行

C. 阻塞→就绪→执行→阻塞

D. 就绪→阻塞→执行→就绪

11. 以下哪种进程调度算法会优先执行优先级最高的进程（ ）。

A. 先来先服务（FCFS）

B. 短进程优先（SPF）

C. 优先级调度

D. 轮转法（Round Robin）

12. 进程的地址空间是指（ ）。

A. 进程可以访问的内存区域

B. 进程的代码段

C. 进程的数据段

D. 进程的堆栈段

参考答案：请自行完成。

1. Linux 中 pthread 线程属于以下哪种类型？

A. 用户级线程

B. 内核级线程

C. 混合线程（ULT+KLT）

D. 协程

2. 用户级线程（ULT）的主要缺点是什么？

A. 切换开销大

B. 无法利用多核并行

C. 需要频繁系统调用

D. 线程数量受限

3. 以下哪种同步机制可以防止多个线程同时访问共享资源？

A. 信号量（Semaphore）

B. 条件变量（Condition Variable）

C. 互斥锁（Mutex）

D. 屏障（Barrier）

4. 以下哪种线程模型允许将多个用户级线程映射到多个内核级线程？

A. 1:1 模型

B. N:1 模型

C. M:N 模型

D. 协程模型

5. 以下哪种操作会导致线程立即终止？

A. pthread\_exit()

B. pthread\_cancel()

C. pthread\_detach()

D. pthread\_join()

6. 线程与进程的主要区别是什么？

A. 线程共享地址空间，进程独立

B. 线程切换开销更大

C. 线程不能并发执行

D. 进程不能使用多核

7.多线程编程中，以下哪种情况可能引发死锁？

A. 多个线程同时读取共享变量

B. 两个线程按相同顺序获取两把锁

C. 两个线程按相反顺序获取两把锁

D. 使用无锁数据结构

参考答案：

*B Linux的pthread线程由内核直接管理，属于内核级线程（KLT）pgrep 进程名 可快速查找进程的 PID，例如 pgrep nginx。*

*B ULT在内核视角下是单线程的，无法分配到不同CPU核心。  
C 解析：互斥锁用于保护临界区，确保同一时间仅一个线程访问共享资源。*

*C M:N 模型（混合模型）支持灵活的多对多映射，但实现复杂。*

*A pthread\_cancel 发送延迟取消请求 A pthread\_exit() 显式终止当前线程*

*A 解析：线程共享进程的地址空间和资源，进程间资源独立。*

*C 线程A持有锁1请求锁2，线程B持有锁2请求锁1，会导致循环等待*

1. Linux中，字符设备驱动的注册通常使用以下哪个函数？

A. register\_blkdev()

B. register\_chrdev()

C. netdev\_register()

D. platform\_driver\_register()

2. 卸载内核模块时，应使用以下哪个宏定义模块的卸载函数？

A. module\_init()

B. module\_exit()

C. module\_unload()

D. module\_cleanup()

3. 用户空间通过哪个虚拟文件系统访问设备驱动的配置信息？

A. /proc

B. /dev

C. /sys

D. /var

4. 驱动中实现文件操作的 read 和 write 功能时，需要填充哪个结构体？

A. struct inode

B. struct file\_operations

C. struct device

D. struct cdev

5. 驱动中从用户空间复制数据到内核空间的函数是？

A. copy\_to\_user()

B. copy\_from\_user()

C. memcpy()

D. strncpy()

6. 手动创建设备节点（如 /dev/mydev）的命令是？

A. insmod

B. mknod

C. udevadm

D. sysctl

7.块设备驱动中，表示一个磁盘设备的核心结构体是？

A. struct block\_device B. struct gendisk

C. struct bio D. struct request

参考答案:

*B， 字符设备驱动注册使用register\_chrdev()，块设备使用register\_blkdev()，网络设备使用netdev\_register()，平台驱动使用platform\_driver\_register()。*

*B module\_exit()用于定义模块卸载时调用的函数，module\_init()用于初始化函数*

*C 解析：/sys（sysfs）用于暴露设备和驱动的属性，/proc主要提供进程和内核状态信息。*

*答案：B 解析：struct file\_operations 定义了字符设备的操作函数（如 read、write、open、release 等）。*

*答案：B 解析：copy\_from\_user() 用于将用户空间数据复制到内核空间，copy\_to\_user() 用于反向操作。*

*答案：B 解析：mknod /dev/mydev c <主设备号> <次设备号> 手动创建设备节点。7.多线程编程中，以下哪种情况可能引发死锁？*

*答案：B 解析：struct gendisk 用于描述一个通用磁盘设备，包含分区、容量等信息。*

1. U-Boot的启动过程中，为什么需要分为SPL（Secondary Program Loader）和主U-Boot两阶段？

A. 为了支持多操作系统引导

B. 因硬件初始化需要分阶段完成，且主U-Boot可能过大无法直接加载

C. 为了兼容不同文件系统

D. 仅出于历史遗留设计

2. U-Boot的“重定位（Relocation）”机制的主要目的是？

A. 将自身代码从Flash复制到RAM的高地址，释放低地址空间供内核使用

B. 动态调整环境变量的存储位置

C. 优化代码执行速度

D. 支持多核CPU的负载均衡

3. U-Boot的设备树（Device Tree）处理流程中，设备树二进制文件（.dtb）何时被传递给Linux内核？

A. 编译时直接链接到内核中

B. 由U-Boot在启动时动态修改并传递

C. 由SPL阶段直接加载到指定内存地址

D. 通过环境变量bootargs传递路径

4. U-Boot在启动Linux内核时，如何从实模式切换到保护模式？

A. U-Boot自身完成保护模式切换，再跳转到内核

B. 内核接管后自行切换

C. 由SPL阶段提前切换

D. 不需要切换，直接在内核中处理

5. U-Boot在移植到新硬件平台时，必须首先实现的底层函数是？

A. 网络驱动

B. 串口输出和内存初始化代码

C. Flash操作接口

D. 中断控制器配置

6.U-Boot的环境变量存储区（ENV）如何避免在Flash擦写时损坏？

A. 使用冗余备份扇区

B. 通过ECC校验纠正错误

C. 仅在RAM中修改，不保存到Flash

D. 使用文件系统持久化

参考答案：

*B 解析：SPL负责最小化硬件初始化（如内存控制器），并加载主U-Boot到内存，解决主镜像过大无法直接运行的问题。*

*A U-Boot启动后会将自身从加载地址（如Flash）重定位到RAM的高地址，避免与内核或设备树冲突。*

*B U-Boot在启动时会解析设备树，并根据硬件配置动态修改（如修正内存大小），然后将其内存地址传递给内核。*

*B U-Boot默认在实模式下运行，内核启动后会自行切换到保护模式。*

*B。 串口调试输出和内存初始化是后续功能（如加载镜像）的基础依赖。*

*A U-Boot的ENV分区通常采用双扇区冗余设计，防止擦写过程中断电导致数据丢失。*

1. 以下哪项是 Linux 移植过程中交叉编译的主要目的？​

A. 加快编译速度​

B. 在目标平台直接编译​

C. 在宿主机上为目标平台生成可执行代码​

D. 减少编译占用空间  
2. 在 Linux 移植中，U - Boot 的主要功能不包括？  
A. 初始化硬件​

B. 加载内核镜像​

C. 文件系统管理​

D. 传递启动参数  
3. 以下哪种文件格式常用于存放 Linux 内核镜像？​

A..exe​

B..img​

C..tar.gz​

D..ko

4. 下列关于设备树（Device Tree）在 Linux 移植中的作用，说法错误的是？​

A. 描述硬件设备信息​

B. 替代了传统的硬件参数传递方式​

C. 使内核与硬件平台强耦合​

D. 方便不同硬件平台移植内核  
5. 在 Linux 移植过程中，根文件系统的作用不包括？​

A. 存放系统可执行文件​

B. 存放用户数据​

C. 提供内核启动参数​

D. 存放配置文件  
6. 交叉编译器的命名规则通常为：<目标平台>-< 编译器名称 >，例如 arm - linux - gcc，其中 “arm - linux” 表示？​

A. 宿主机平台​

B. 目标平台​

C. 编译器类型​

D. 开发环境

7. 在 Linux 移植过程中，以下哪个步骤通常在配置内核之后？​

A. 选择交叉编译工具链​

B. 初始化硬件​

C. 编译内核​

D. 制作根文件系统

参考答案：

*C 交叉编译是在宿主机（如 x86 架构的 PC）上，使用交叉编译器为目标平台（如 ARM 嵌入式设备）生成可执行代码，因为目标平台通常资源有限或架构不同，无法直接在其上编译，所以选项 C 正确。A 选项，交叉编译不一定能加快编译速度；B 选项，交叉编译不是在目标平台直接编译；D 选项，交叉编译与减少编译占用空间无关。*

*C   
B img 文件格式常用于存放完整的镜像文件，包括 Linux 内核镜像；.exe 是 Windows 可执行文件格式；.tar.gz 是压缩包格式，常用于存放源码等压缩文件；.ko 是 Linux 内核模块文件格式*

*C 设备树用于描述硬件设备信息，替代了传统的在内核中硬编码硬件参数的方式，使得内核与硬件平台解耦，方便在不同硬件平台移植内核，所以 C 选项说法错误*

*C 根文件系统用于存放系统可执行文件、用户数据、配置文件等。提供内核启动参数是引导加载程序（如 U - Boot）的功能，不是根文件系统的作用，  
B。 在交叉编译器命名中，“arm - linux” 表示目标平台，即编译后的代码将运行在基于 ARM 架构且运行 Linux 系统的设备上，而不是宿主机平台，也不是编译器类型和开发环境，  
C. 在 Linux 移植过程中，一般先选择交叉编译工具链，然后配置内核，配置完成后进行编译内核操作；初始化硬件是引导加载程序的工作；制作根文件系统和配置内核没有严格的先后顺序，但编译内核通常在配置内核之后，所以选 C。*

1. 以 QT框架的核心特性是？

A. 仅支持Windows平台

B. 使用Java语言开发

C. 跨平台C++图形用户界面库

D. 仅用于嵌入式开发

2. 在Linux下安装QT开发环境，通常需要安装哪个组件？

A. qtcreator

B. visualstudio

C. eclipse

D. xcode

3. QT的信号槽机制用于？

A. 文件读写操作

B. 对象间通信

C. 内存管理

D. 网络协议实现

4. 以下哪个命令用于编译QT项目？

A. g++ main.cpp

B. qmake && make

C. cmake .

D. ./configure

5. QT Designer的主要作用是？

A. 编写C++代码

B. 可视化设计UI界面

C. 调试程序

D. 管理版本控制

6.在QT中，哪个类用于处理主窗口？

A. QDialog

B. QMainWindow

C. QWidget

D. QLabel  
7. 要将按钮点击事件连接到槽函数，应使用？

A. connect(button, &QPushButton::clicked(), this, &MyClass::mySlot)

B. connect(button, &QPushButton::clicked, this, &MyClass::mySlot)

C. button->onClick(this, SLOT(mySlot))

D. button.link(mySlot)

8. QT项目配置文件的后缀是？

A. .cpp

B. .h

C. .pro

D. .ui

9. 在Linux QT程序中，如何实现多线程？

A. 使用QProcess

B. 继承QThread类或使用QThreadPool

C. 只能使用C++标准库

D. QT不支持多线程

10. 在Linux下部署QT程序时，通常需要？

A. 直接复制可执行文件即可

B. 使用ldd查找依赖库，或打包发布工具（如linuxdeployqt）

C. 必须静态编译

D. 需要安装Visual Studio

参考答案：C A B B B B B C B B

1 可信计算的核心目标是保证系统的（）

A. 高性能

B. 可验证行为

C. 低成本

D. 兼容性

2. TPM（可信平台模块）的主要功能不包括（）

A. 安全密钥存储

B. 远程证明

C. 磁盘加密

D. 网络流量过滤

3. 可信计算中“远程证明”（Remote Attestation）的作用是（）

A. 验证本地硬件完整性

B. 向远程方证明平台状态可信

C. 加密通信数据

D. 提升计算速度

4. 静态可信度量根（S-RTM）的度量过程始于（）

A. 操作系统启动后

B. BIOS/UEFI固件

C. 应用程序运行时

D. 网络连接建立时

5. 下列哪种攻击可通过可信计算缓解？

A. 中间人攻击

B. 硬件损坏

C. 恶意软件篡改系统

D. 电源故障

6. 下列哪项技术不属于可信计算范畴？

A. 可信启动

B. 内存隔离

C. 区块链共识

D. 完整性度量

7. 可信计算中“信任链”（Chain of Trust）的传递顺序通常是（）

A. 应用→OS→固件→硬件

B. 硬件→固件→OS→应用

C. 网络→硬件→OS

D. 用户→应用→硬件

8. 下列哪种场景不需要可信计算支持？

A. 云服务器完整性验证

B. 智能合约执行

C. 离线文档编辑

D. 防止固件级Rootkit

参考答案：B D B B C C B C

1 可信根（Root of Trust, RoT）的核心功能是？

A. 提供系统最高性能

B. 作为系统安全性的起点和信任基础

C. 管理用户数据加密

D. 优化网络传输速度

2.以下哪项是可信根（RoT）的典型特征？

A. 可被软件动态修改

B. 依赖外部输入验证

C. 具备硬件级保护且不可篡改

D. 仅用于存储日志

3. 在TCG（可信计算组）规范中，可信链的建立依赖于？

A. 随机数生成器

B. 逐级哈希度量和验证

C. 防火墙规则

D. 用户密码

4. 静态可信根（Static Root of Trust）与动态可信根（Dynamic Root of Trust）的主要区别是？

A. 静态可信根仅在启动时验证，动态可信根支持运行时验证

B. 静态可信根需要联网，动态可信根离线工作

C. 动态可信根仅用于存储加密密钥

D. 静态可信根性能更高

5. 以下哪种硬件通常作为可信根的物理载体？

A. 普通U盘

B. 可信平台模块（TPM）

C. 机械硬盘

D. 显示器

6. 在云环境中，可信链的主要作用是？

A. 提高虚拟机计算速度

B. 确保虚拟机镜像从启动到运行的完整性

C. 降低网络延迟

D. 自动扩容存储

7. 移动设备（如手机）的可信根通常存储在？

A. 应用缓存中

B. 基带处理器或专用安全芯片（如TEE）

C. SD卡

D. 云服务器

8. 信任链断裂的典型表现是？

A. 系统启动速度变快

B. 某个环节的哈希验证失败

C. 网络连接增强

D. 用户界面美化

9. 攻击者篡改Bootloader后，对可信链的影响是？

A. 信任链从Bootloader开始断裂，后续组件无法被验证

B. 仅影响操作系统性能

C. 网络连接加密增强

D. 无任何影响

参考答案：B C B A B B B B A

1. 安全杂凑（Secure Hash）的核心作用是什么？

A. 加密数据以保护机密性

B. 生成数据的唯一指纹以验证完整性

C. 压缩数据以减少存储空间

D. 生成随机密钥

2.HMAC 与普通哈希的主要区别是什么？

A. HMAC 生成更短的摘要

B. HMAC 仅用于密码存储

C. HMAC 依赖密钥来增强身份认证

D. HMAC 使用更快的算法

3. 密钥派生函数（KDF）的主要目的是？

A. 生成随机数

B. 从密码或弱熵源生成强密钥

C. 加密大文件

D. 验证数字签名

4. Nonce 在安全协议中的作用是？

A. 防止重放攻击

B. 提高加密速度

C. 压缩数据

D. 生成公钥

5. 应用程序静态可信验证通常验证什么？

A. 运行时的内存状态

B. 用户输入合法性

C. 网络延迟

D. 代码的哈希值与基准值是否一致

6. 白名单机制在进程保护中的作用是？

A. 允许所有进程运行

B. 仅允许预先授权的进程运行

C. 随机阻止进程

D. 记录进程日志

7. 以下哪种场景必须使用HMAC而非普通哈希？

A. 文件完整性校验

B. 验证API请求来源合法性

C. 生成Git提交ID

D. 压缩数据库

9. 进程保护操作中，防止代码注入的最佳实践是？

A. 使用地址空间布局随机化（ASLR）

B. 禁用所有网络连接

C. 允许所有动态链接库加载

D. 关闭日志记录

9. 白名单机制与黑名单机制的主要区别是？

A. 白名单默认拒绝所有，仅允许明确授权的项

B. 白名单性能更优

C. 黑名单无法动态更新

D. 黑名单依赖哈希值

10. 在公钥认证中，CA（证书颁发机构）的作用是？

A. 生成对称密钥

B. 验证并签名公钥的真实性

C. 加密用户数据

D. 提供哈希计算服务

11. 抗碰撞性（Collision Resistance）是哈希函数的哪类安全属性？

A. 无法找到两个不同输入生成相同哈希值

B. 无法从哈希值反推原始数据

C. 加密速度快

D. 支持密钥派生

参考答案：

B 安全杂凑（如SHA-256）通过固定长度摘要验证数据完整性，不涉及加密或压缩。

C HMAC 结合密钥和哈希函数，确保数据来源真实性和完整性。

B KDF（如PBKDF2、HNonce（一次性随机数）确保每次通信的独特性，防止攻击者重复利用旧消息。KDF）将弱输入（如密码）转换为加密安全的密钥。

A. Nonce（一次性随机数）确保每次通信的独特性，防止攻击者重复利用旧消息。

D 静态验证在程序启动前检查代码/配置文件的哈希值是否匹配可信基准。

B 白名单机制仅允许已知安全的进程执行，阻止未授权的程序。

B。 HMAC通过密钥验证消息来源，适用于身份认证场景。

A ASLR通过随机化内存地址布局，增加攻击者预测注入代码位置的难度。

A。 白名单采用“默认拒绝”策略，仅放行已知安全项；黑名单则“默认允许”，仅阻止已知恶意项。

B CA通过数字证书绑定公钥与身份，并用自己的私钥签名以确保证书可信。

A。 抗碰撞性指攻击者难以找到两个不同的输入使哈希值相同。

1 TCM的认证功能主要分为哪两类？

A. 加密与解密

B. 身份认证与完整性认证

C. 网络传输与本地存储

D. 数据压缩与解压缩

2. 完整性认证什么机制确保组件未被篡改？

A. 对称加密算法

B. PCR寄存器与哈希值度量链

C. 网络防火墙

D. 数据备份机制

3. TCM的EK证书由谁签发？

A. 用户自行生成

B. 操作系统内核

C. 厂商根密钥

D. 远程服务器

4.TCM的密钥管理体系特点是什么？

A. 根密钥始终驻留TCM内部

B. 私钥可导出到外部存储

C. 使用国际通用算法（如RSA）

D. 密钥生命周期由操作系统管理

5. 生成密钥时，TCM必须满足的前提条件是什么？

A. 网络连接正常

B. 硬件温度低于阈值

C. 用户输入生物特征

D. PCR值符合预设策略

6. 若BIOS被篡改，TCM如何检测？

A. 通过SM4加密验证

B. 重启系统自动修复

C. 比对PCR值与白名单哈希值

D. 调用网络防火墙

7. PCR值的最终结果取决于什么？

A. 最后一次写入的值

B. 所有扩展操作的顺序和内容

C. 系统时间戳

D. 随机数生成器

8. TCM的NVRAM通常用于存储什么？

A. 身份证书与策略授权信息

B. 临时计算数据

C. 操作系统日志

D. 用户个人文件

9. 属性证书在远程证明中的作用是？

A. 提高签名速度

B. 减少隐私泄露风险

C. 替代PCR值

D. 加密通信链路

参考答案:B B C A D C B A B