

文件编号：HN/WI-GC-YF-066	Linux 应用程序设计指南	版本：V1.0
制定部门： 研发中心		页码：1 / 39
		生效日期：2024 年 2 月 5 日

文档名称：

Linux 应用程序设计指南

摘要：

本指南介绍 Linux 应用程序设计相关资料，帮助开发人员快速定位 Linux 应用程序设计过程中的学习资源。

当前版本	V1.0	文件状态	<input type="checkbox"/> 草稿； <input checked="" type="checkbox"/> 正式	取代版本	
完成时间	2024-2-5		被取代文档完成时间		
作者	刘新颖		审批	孙世奎	
批准	王传峰		文档编号	HN/WI-GC-YF-066	
版本历史：					
版本状态	作者	参与者	完成日期	备注	
V1.0	刘新颖	岳泉	2024-2-5		

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 2 / 39
		生效日期：2023 年 12 月 25 日

目录

- 1、 引言 6
 - 1.1 编写目的 6
 - 1.2 适用范围 6
- 2、 系统 API 6
 - 2.1 常用的 API 接口 6
 - 2.1.1 官方资源 6
 - 2.1.2 书籍与文档 6
 - 2.1.3 示例与源码 7
 - 2.1.4 在线社区与论坛 7
 - 2.1.5 教程与视频 7
- 3、 文件 I/O 编程 7
 - 3.1 标准 I/O 开发 7
 - 3.1.1 官方资源 8
 - 3.1.2 书籍与文档 8
 - 3.1.3 示例与源码 8
 - 3.1.4 在线社区与论坛 8
 - 3.2 文本流和数据流 9
 - 3.2.1 官方资源 9
 - 3.2.2 书籍与文档 9
 - 3.2.3 示例与源码 9
 - 3.2.4 在线社区与论坛 9
 - 3.3 文件和目录 10
 - 3.3.1 官方资源 10
 - 3.3.2 书籍与文档 10
 - 3.3.3 示例与源码 10
 - 3.3.4 在线社区与论坛 11
- 4、 网络编程 11
 - 4.1 TCP/IP 协议概述 11
 - 4.1.1 官方资源 11
 - 4.1.2 书籍与文档 11
 - 4.1.3 示例与源码 12
 - 4.1.4 在线社区与论坛 12
 - 4.1.5 教程与视频 12
 - 4.2 网络基础编程 12
 - 4.2.1 官方资源 12
 - 4.2.2 书籍与文档 13
 - 4.2.3 示例与源码 13
 - 4.2.4 教程与视频 13
 - 4.3 网络高级编程 13
 - 4.3.1 官方资源 14
 - 4.3.2 书籍与文档 14
 - 4.3.3 示例与源码 14
 - 4.3.4 在线社区和论坛 14
- 5、 进程间通信 14
 - 5.1 Linux 下进程间通信概述 15
 - 5.1.1 官方资源 15
 - 5.1.2 示例与源码 15

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 3 / 39
		生效日期：2023 年 12 月 25 日

5.1.3 在线社区和论坛	15
5.2 管道通信	15
5.2.1 官方资源	15
5.2.2 示例与源码	16
5.2.3 在线社区和论坛	16
5.3 信号通信	16
5.3.1 官方资源	16
5.3.2 示例与源码	16
5.3.3 在线社区和论坛	16
5.4 共享内存	17
5.4.1 官方资源	17
5.4.2 示例与源码	17
5.4.3 在线社区和论坛	17
5.5 消息队列	17
5.5.1 官方资源	18
5.5.2 示例与源码	18
5.5.3 在线社区和论坛	18
6、多线程编程	18
6.1 Linux 下线程概述	18
6.1.1 官方资源	18
6.1.2 书籍与文档	19
6.1.3 示例与源码	19
6.1.4 在线社区和论坛	19
6.2 Linux 线程实现	19
6.2.1 官方资源	19
6.2.2 书籍与文档	20
6.2.3 示例与源码	20
6.2.4 在线社区与论坛	20
6.3 线程的同步与互斥	20
6.3.1 官方资源	21
6.3.2 书籍与文档	21
6.3.3 示例与源码	21
6.3.4 在线社区和论坛	21
6.4 可重入与线程安全	22
6.4.1 官方资源	22
6.4.2 书籍与文档	22
6.4.3 示例与源码	22
6.4.4 在线社区和论坛	22
7、Qt 图形编程	23
7.1 Qt GUI 开发入门	23
7.1.1 官方资源	23
7.1.2 书籍与文档	23
7.1.3 示例与源码	23
7.1.4 在线社区与论坛	24
7.1.5 教程与视频	24
7.2 窗口部件	24
7.2.1 官方资源	24
7.2.2 书籍与文档	24
7.2.3 示例与源码	24
7.2.4 在线社区与论坛	25

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 4 / 39
		生效日期：2023 年 12 月 25 日

7.2.5 教程与视频	25
7.3 布局管理	25
7.3.1 官方资源	25
7.3.2 书籍与文档	25
7.3.3 示例与源码	26
7.3.4 教程与视频	26
7.4 主窗口框架	26
7.4.1 官方资源	26
7.4.2 书籍与文档	26
7.4.3 示例与源码	27
7.4.4 教程与视频	27
8、设备驱动开发	27
8.1 设备驱动概述	27
8.1.1 书籍与文档	28
8.1.2 示例与源码	28
8.1.3 在线社区和论坛	28
8.1.4 教程与视频	29
8.2 字符设备驱动编写	29
8.2.1 官方资源	29
8.2.2 书籍与文档	29
8.2.3 示例与源码	30
8.2.4 在线社区和论坛	30
8.3 块设备驱动编写	30
8.3.1 官方资源	30
8.3.2 书籍与文档	30
8.3.3 示例与源码	31
8.3.4 在线社区和论坛	31
8.4 中断编程	31
8.4.1 官方资源	31
8.4.2 书籍与文档	31
8.4.3 示例与源码	32
8.4.4 在线社区和论坛	32
8.5 键盘驱动实现	32
8.5.1 官方资源	32
8.5.2 书籍与文档	33
8.5.3 示例与源码	33
8.5.4 在线社区和论坛	33
9、内存管理	33
9.1 内存管理的基本原理和方法	33
9.1.1 官方资源	33
9.1.2 书籍与文档	34
9.1.3 在线社区和论坛	34
10、性能优化	34
10.1 官方资源	35
10.2 书籍与文档	35
10.3 示例与源码	35
10.4 在线社区与论坛	35
11、开源第三方库	36
11.1 常用的第三方库	36
11.1.1 Boost C++ Libraries	36

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 5 / 39
		生效日期：2023 年 12 月 25 日

11.1.2 Qt.....	36
11.1.3 GTK.....	36
11.1.4 OpenCV.....	36
11.1.5 FFmpeg.....	36
11.1.6 SQLite.....	37
11.1.7 CMake.....	37
11.1.8 gRPC.....	37
11.1.9 ZeroMQ.....	37
11.1.10 Eigen.....	37
11.1.11 libmodbus.....	37
11.1.12 Libevent.....	38

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 6 / 39
		生效日期：2023 年 12 月 25 日

1、引言

1.1 编写目的

本指南旨在为开发人员提供全面的知识，使开发人员能够更好地利用 Linux 平台进行应用程序设计和开发。有助于提高开发人员在 Linux 环境下的技能水平，从而更好地满足用户需求和系统要求。

1.2 适用范围

适用于软件开发人员或其他对 Linux 应用程序设计感兴趣的人群。为初学者提供入门指南，且为有经验的开发人员提供深入的技术指导。

2、系统 API

2.1 常用的 API 接口

在 Linux 应用程序设计中，使用系统 API 是非常重要的。以下是一些关于 Linux 系统 API 的相关资料：

2.1.1 官方资源

(1) ``man`` 命令：在终端中使用 ``man`` 命令可以查看 Linux 系统 API 的官方手册。例如，``man open`` 可以查看 ``open`` 系统调用的手册。

(2) [Linux API](https://en.wikipedia.org/wiki/Linux_kernel_interfaces)：维基百科上对 Linux API 的介绍，包括系统调用、库函数等。

2.1.2 书籍与文档

(1) 《Advanced Programming in the UNIX Environment》（译版《UNIX 环境高级编程》）by W. Richard Stevens：这是一本经典的 UNIX 编程教材，涵盖了 UNIX 系统编程的方方面面，包括系统调用、文件 I/O、进程控制等。

(2) 《Linux System Programming》（译版《Linux 系统编程》）by Robert Love：这本书专注于 Linux 系统编程，提供了详细的解释和示例，适合想要深入了解 Linux 系统 API 的读者。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 7 / 39
		生效日期：2023 年 12 月 25 日

（3）《Understanding the Linux Kernel》（译版《深入理解 LINUX 内核》）by Daniel P. Bovet and Marco Cesati: 虽然这本书更侧重于 Linux 内核，但它提供了对 Linux 系统 API 的深入理解，特别是对于想要了解系统调用和内核机制的人。

（4）《Linux Programming by Example》by Kurt Wall: 这本书通过实例介绍了如何在 Linux 环境中进行编程，包括文件操作、进程控制等。

（5）[Linux man Pages Online](https://man7.org/linux/man-pages/): 在线浏览 Linux 的 man 手册，包括系统调用、库函数等。

2.1.3 示例与源码

浏览 GitHub 上的开源项目，特别是那些与系统编程相关的项目，可以学习到许多实际应用的例子。如：<https://github.com/guodongxiaren/LinuxAPI/wiki>。

2.1.4 在线社区与论坛

在 Linux 相关的在线社区，如 Stack Overflow (<http://stackoverflow.p2hp.com/>)、Linux Questions (<https://www.linux.org/forums/>) 等，我们可以提问并学习其他人的经验。

2.1.5 教程与视频

[Linux System Programming Tutorial](https://www.tutorialspoint.com/ebook/linux-system-programming-techniques/index.asp): 一个在线教程，介绍了 Linux 系统编程的基础知识。

这些资源能够帮助我们更好地理解和使用 Linux 系统 API 进行应用程序设计。建议你根据自己的需要选择适合的学习资源，并通过实践来加深对 Linux 系统编程的理解。

3、文件 I/O 编程

3.1 标准 I/O 开发

在 Linux 应用程序设计中，标准 I/O (Standard I/O) 是进行输入和输出操作的一种常见方式。以下是一些关于 Linux 标准 I/O 开发的相关资料：

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 8 / 39
		生效日期：2023 年 12 月 25 日

3.1.1 官方资源

- (1) ``man`` 命令：使用 ``man`` 命令可以查看 Linux 系统上标准 I/O 相关函数的官方手册。例如，``man printf`` 可以查看 ``printf`` 函数的手册。
- (2) [在线手册](https://man7.org/linux/man-pages/): 在线浏览 Linux 的 man 手册，包括标准 I/O 函数。

3.1.2 书籍与文档

- (1) 《C Programming Absolute Beginner's Guide》by Perry and Miller: 这本书适合初学者，其中有关于标准 I/O 的简单而易懂的介绍。
- (2) 《C Programming for the Absolute Beginner, Second Edition》by Vine: 同样是面向初学者的一本书，涵盖了 C 语言的基础和标准 I/O 的使用。
- (3) 《The C Programming Language》（译版《C 程序设计语言》）by Kernighan and Ritchie:通常被称为 K&R，这本书是 C 语言的经典之作，其中包含了对标准 I/O 的介绍。
- (4) 《Advanced Programming in the UNIX Environment》（译版《UNIX 环境高级编程》）by W. Richard Stevens: 这本书提供了对 UNIX 环境下高级编程的全面介绍，其中包括对文件 I/O 和标准 I/O 的深入讲解。
- (5) 《Linux Programming by Example》by Kurt Wall: 这本书通过实例介绍了如何在 Linux 环境中进行编程，包括文件 I/O 和标准 I/O。

3.1.3 示例与源码

浏览 GitHub 等开源代码平台，查找使用标准 I/O 的实际项目，学习实际应用的例子。

3.1.4 在线社区与论坛

- 在 Stack Overflow (<http://stackoverflow.p2hp.com/>) 等在线社区，你可以提问有关标准 I/O 的问题，并学习其他人的经验。
- 以上资源能够帮助我们更好地理解和使用 Linux 中的标准 I/O 进行应用程序设计。建议你根据自己的需求选择适合的学习资源，并通过实践来加深对标准 I/O 的理解。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 9 / 39
		生效日期：2023 年 12 月 25 日

3.2 文本流和数据流

在 Linux 应用程序设计中，理解文本流和数据流是重要的一部分。以下是一些关于 Linux 中文本流和数据流的相关资料：

3.2.1 官方资源

(1) ``man`` 命令：通过使用 ``man`` 命令，你可以查看 Linux 系统上有关文本流和数据流的函数的官方手册。例如，``man fgets`` 可以查看 ``fgets`` 函数的手册。

(2) [在线手册](https://man7.org/linux/man-pages/)：在线浏览 Linux 的 man 手册，包括文本流和数据流相关函数。

3.2.2 书籍与文档

(1) 《C Programming Absolute Beginner's Guide》by Perry and Miller：这本书适合初学者，其中有关于文本流和数据流的简单而易懂的介绍。

(2) 《C Programming for the Absolute Beginner, Second Edition》by Vine：面向初学者，涵盖了 C 语言的基础和文本流、数据流的使用。

(3) 《The C Programming Language》（译版《C 程序设计语言》）by Kernighan and Ritchie：通常被称为 K&R，这本书是 C 语言的经典之作，其中包含了对文本流和数据流的介绍。

(4) 《Advanced Programming in the UNIX Environment》（译版《UNIX 环境高级编程》）by W. Richard Stevens：这本书提供了对 UNIX 环境下高级编程的全面介绍，包括了文本流和数据流的深入讲解。

(5) 《Linux Programming by Example》by Kurt Wall：这本书通过实例介绍了如何在 Linux 环境中进行编程，包括了文本流和数据流的应用。

3.2.3 示例与源码

浏览 GitHub 等开源代码平台，查找使用文本流和数据流的实际项目，学习实际应用的例子。

3.2.4 在线社区与论坛

在 Stack Overflow (<http://stackoverflow.p2hp.com/>) 等在线社区，你可以提问有关文本流和数据流的

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 10 / 39
		生效日期：2023 年 12 月 25 日

问题，并学习其他人的经验。

以上资源能够帮助我们更好地理解和使用 Linux 中的文本流和数据流进行应用程序设计。建议你根据自己的需求选择适合的学习资源，并通过实践来加深对文本流和数据流的理解。

3.3 文件和目录

在 Linux 应用程序设计中，文件和目录操作是基础而且重要的一部分。以下是一些关于 Linux 文件和目录操作的相关资料：

3.3.1 官方资源

- （1）`man` 命令：使用 `man` 命令可以查看 Linux 系统上文件和目录相关函数的官方手册。例如，`man open` 可以查看 `open` 系统调用的手册。
- （2）[在线手册](https://man7.org/linux/man-pages/)：在线浏览 Linux 的 man 手册，包括文件和目录操作相关函数。

3.3.2 书籍与文档

- （1）《Linux Command Line and Shell Scripting Bible》by Richard Blum：这本书介绍了 Linux 命令行和脚本编程，包括文件和目录的基本操作。
- （2）《The Linux Programming Interface》（译版《Linux/UNIX 系统编程手册》）by Michael Kerrisk：这本书是关于 Linux 系统编程的权威之作，提供了深入的文件和目录操作的解释和示例。
- （3）《Advanced Programming in the UNIX Environment》（译版《UNIX 环境高级编程》）by W. Richard Stevens：这本书涵盖了 UNIX 环境下高级编程的方方面面，包括文件和目录的操作。
- （4）《Linux Pocket Guide》（译版《Linux 口袋书》） by Daniel J. Barrett：这本小册子提供了关于 Linux 命令行和文件系统操作的简明指南。
- （5）《Linux Programming by Example》by Kurt Wall：这本书通过实例介绍了如何在 Linux 环境中进行编程，包括了文件和目录操作的应用。

3.3.3 示例与源码

浏览 GitHub 等开源代码平台，查找使用文件和目录操作的实际项目，学习实际应用的例子。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 11 / 39
		生效日期：2023 年 12 月 25 日

3.3.4 在线社区与论坛

在 Stack Overflow (<http://stackoverflow.p2hp.com/>) 等在线社区，你可以提问有关文件和目录操作的问题，并学习其他人的经验。

以上资源能够帮助我们更好地理解和使用 Linux 中的文件和目录进行应用程序设计。建议你根据自己的需求选择适合的学习资源，并通过实践来加深对文件和目录操作的理解。

4、网络编程

网络编程共用书籍与文档：

(1) 《UNIX Network Programming, Volume 2: Interprocess Communications》（译版《UNIX 网络编程：第 2 版. 第 2 卷，进程间通信(中文版)》）by W. Richard Stevens：这本书是 Stevens 的系列著作的第二卷，深入研究了进程间通信（IPC）和其他高级网络编程主题。该书介绍了各种高级网络编程概念，如多路复用、信号处理等。

4.1 TCP/IP 协议概述

在 Linux 应用程序设计中，理解 TCP/IP 协议是至关重要的。以下是一些关于 Linux TCP/IP 协议概述的相关资料：

4.1.1 官方资源

(1) Linux 内核源代码：你可以通过阅读 Linux 内核源代码来深入理解 TCP/IP 协议的实现。内核源代码通常位于 `/usr/src/linux` 或 `/lib/modules/$(uname -r)/build`。

(2) [在线手册](<https://man7.org/linux/man-pages/>)：在线浏览 Linux 的 man 手册，包括与网络编程相关的函数。

4.1.2 书籍与文档

(1) 《TCP/IP 详解 卷 1：协议》by W. Richard Stevens：这是一本经典的 TCP/IP 协议相关书籍，详细介绍了 TCP/IP 协议族的各个方面。

(2) 《Unix Network Programming, Volume 1: The Sockets Networking API》（译版《UNIX 网

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 12 / 39
		生效日期：2023 年 12 月 25 日

络编程 卷 1:套接字联网 API》) by W. Richard Stevens: 作者同样是 W. Richard Stevens, 这本书专注于 Unix 平台的网络编程, 对 TCP/IP 协议的实现进行了深入解析。

(3) 《TCP/IP Illustrated, Volume 1: The Protocols》(译版《TCP/IP 详解 卷 1: 协议(原书第 2 版)》) by Kevin R. Fall and W. Richard Stevens: 这本书结合了图示和深入的解释, 对 TCP/IP 协议进行了全面的说明。

4.1.3 示例与源码

浏览 GitHub 等开源代码平台, 查找使用 TCP/IP 协议的实际项目, 学习实际应用的例子。

4.1.4 在线社区与论坛

在 Stack Overflow (<http://stackoverflow.p2hp.com/>) 等在线社区, 你可以提问有关 TCP/IP 协议的问题, 并学习其他人的经验。

4.1.5 教程与视频

[Linux Network Programming](http://www.linuxhowtos.org/C_C++/socket.htm): 一个在线教程, 介绍了在 Linux 上进行网络编程的基础知识。

以上资源能够帮助我们更好地理解和使用 Linux 中的 TCP/IP 协议进行应用程序设计。建议你根据自己的需求选择适合的学习资源, 并通过实践来加深对 TCP/IP 协议的理解。

4.2 网络基础编程

在 Linux 应用程序设计中, 网络基础编程是一个重要的主题。以下是一些关于 Linux 网络基础编程的相关资料:

4.2.1 官方资源

[在线手册](<https://man7.org/linux/man-pages/>): 在线浏览 Linux 的 man 手册, 包括与网络编程相关的函数。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 13 / 39
		生效日期：2023 年 12 月 25 日

4.2.2 书籍与文档

- (1) 《Unix Network Programming, Volume 1: The Sockets Networking API》（译版《UNIX 网络编程 卷 1:套接字联网 API》）by W. Richard Stevens：这是一本经典的著作，详细介绍了 Unix 平台上网络编程的基础知识，包括套接字编程。
- (2) 《TCP/IP 详解 卷 1：协议》by W. Richard Stevens：这本书深入探讨了 TCP/IP 协议，对网络编程的基本概念进行了详细的解释。
- (3) 《Linux Socket Programming by Example》by Warren Gay：该书通过示例演示了在 Linux 上进行套接字编程的实际应用。
- (4) 《Beej's Guide to Network Programming》by Brian "Beej" Jorgensen：这是一份开源的网络编程指南，通过简单的例子介绍了网络编程的基本概念。
- (5) 《Advanced Programming in the UNIX Environment》（译版《UNIX 环境高级编程》）by W. Richard Stevens：这本书不仅涵盖了网络编程，还包括了 UNIX 环境下高级编程的方方面面。

4.2.3 示例与源码

在 GitHub 等开源代码平台上搜索网络编程的示例代码，学习实际应用的例子。

4.2.4 教程与视频

[Linux Network Programming](http://www.linuxhowtos.org/C_C++/socket.htm)：一个在线教程，介绍了在 Linux 上进行网络编程的基础知识。

以上资源能够帮助我们更好地理解和应用 Linux 网络基础编程。根据你的需求，选择适合的学习资源，并通过实际编程来深入学习网络编程的技能。

4.3 网络高级编程

在 Linux 应用程序设计中，网络高级编程涉及更深层次的主题，包括多线程网络编程、异步网络编程、安全性等方面。以下是一些关于 Linux 网络高级编程的相关资料：

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 14 / 39
		生效日期：2023 年 12 月 25 日

4.3.1 官方资源

[在线手册](https://man7.org/linux/man-pages/): 在线浏览 Linux 的 man 手册，深入了解与高级网络编程相关的函数。

4.3.2 书籍与文档

- (1) 《Linux System Programming: Talking Directly to the Kernel and C Library》by Robert Love: 本书深入介绍了 Linux 系统编程，包括进程控制、文件系统、网络编程等高级主题。
- (2) 《The Linux Programming Interface》（译版《Linux/UNIX 系统编程手册》）by Michael Kerrisk: 本书对 Linux 系统编程进行了广泛而深入的介绍，包括高级网络编程。
- (3) 《Beej's Guide to Network Programming Using Internet Sockets》by Brian "Beej" Hall: 这份开源的网络编程指南包括了一些高级主题，如多线程、select() 函数等。

4.3.3 示例与源码

浏览 GitHub 等开源代码平台，查找使用高级网络编程概念的实际项目。

4.3.4 在线社区和论坛

在 Stack Overflow (<http://stackoverflow.p2hp.com/>)、Reddit (<https://www.redditinc.com/>) 等在线社区上参与讨论，了解其他开发者的经验和建议。

以上资源能够帮助我们深入理解和应用 Linux 中的高级网络编程概念。根据你的需求，选择适合的学习资源，并通过实际项目和编程练习来提高技能。

5、进程间通信

进程间通信共用书籍与文档：

- (1) 《Unix Network Programming, Volume 2: Interprocess Communications》（译版《UNIX 网络编程：第 2 版. 第 2 卷， 进程间通信(中文版) 》）by W. Richard Stevens: 本书详细介绍了 Unix 环境下的 IPC，包括管道、消息队列、信号量等。
- (2) 《Advanced Programming in the UNIX Environment》by W. Richard Stevens: 该书覆盖了进

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 15 / 39
		生效日期：2023 年 12 月 25 日

程间通信、文件和目录操作等方方面面，对 Unix 环境下高级编程有全面的介绍。

(3) 《The Linux Programming Interface》（译版《Linux/UNIX 系统编程手册》）by Michael Kerrisk：本书深入介绍了 Linux 系统编程，包括信号处理的各个方面，如信号处理函数、信号集等。

5.1 Linux 下进程间通信概述

在 Linux 应用程序设计中，进程间通信（IPC）是一个重要的主题，涉及到多种机制，如管道、消息队列、信号量、共享内存等。以下是一些关于 Linux 下 IPC 概述的相关资料：

5.1.1 官方资源

[在线手册](https://man7.org/linux/man-pages/): 在线浏览 Linux 的 man 手册, 包括与 IPC 相关的函数。

5.1.2 示例与源码

浏览 GitHub 等开源代码平台，查找使用 IPC 机制的实际项目。

5.1.3 在线社区和论坛

在 Stack Overflow (<http://stackoverflow.com/>)、Reddit (<https://www.reddit.com/>) 等在线社区上参与讨论，了解其他开发者的经验和建议。

以上资源能够帮助我们更好地理解和应用 Linux 下的 IPC 概念。根据你的需求，选择适合的学习资源，并通过实际项目和编程练习来提高技能。

5.2 管道通信

管道通信是 Linux 应用程序设计中常用的进程间通信（IPC）机制之一。以下是一些关于 Linux 下管道通信的相关资料：

5.2.1 官方资源

[在线手册](https://man7.org/linux/man-pages/): 在线浏览 Linux 的 man 手册, 查找关于管道的函数和系统调用的信息。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 16 / 39
		生效日期：2023 年 12 月 25 日

5.2.2 示例与源码

在 GitHub 等开源代码平台上搜索使用管道通信的实际项目和示例代码。

5.2.3 在线社区和论坛

在 Stack Overflow (<http://stackoverflow.p2hp.com/>)、Reddit (<https://www.redditinc.com/>) 等在线社区上参与讨论，了解其他开发者在管道通信方面的经验和建议。

以上资源能够帮助我们更深入地理解和应用 Linux 下的管道通信。通过学习示例代码和实际项目，你可以提高在这一领域的技能。

5.3 信号通信

在 Linux 应用程序设计中，信号是一种用于进程间通信的机制，用于通知进程发生了某个事件。以下是一些关于 Linux 下信号通信的相关资料：

5.3.1 官方资源

[在线手册](<https://man7.org/linux/man-pages/>): 在线浏览 Linux 的 man 手册，查找与信号相关的系统调用和函数。

5.3.2 示例与源码

(1) 《Linux Signal Handling Example》(https://www.tutorialspoint.com/cplusplus/cpp_signal_handling.htm): 在 Tutorialspoint 上有一个关于 Linux 信号处理的简单示例，有助于理解信号处理的基本概念。

(2) 在 GitHub 等开源代码平台上搜索使用信号通信的实际项目和示例代码。

5.3.3 在线社区和论坛

在 Stack Overflow (<http://stackoverflow.p2hp.com/>)、Reddit (<https://www.redditinc.com/>) 等在线社区上参与讨论，了解其他开发者在信号通信方面的经验和建议。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 17 / 39
		生效日期：2023 年 12 月 25 日

以上资源能够帮助我们更好地理解和应用 Linux 下的信号通信机制。通过学习示例代码和实际项目，你可以提高在这一领域的技能。

5.4 共享内存

在 Linux 应用程序设计中，共享内存是一种高效的进程间通信（IPC）机制，允许多个进程共享同一块内存空间。以下是一些关于 Linux 下共享内存的相关资料：

5.4.1 官方资源

[在线手册](<https://man7.org/linux/man-pages/>): 在线浏览 Linux 的 man 手册，查找与共享内存相关的系统调用和函数。

5.4.2 示例与源码

(1) 《Linux Shared Memory Tutorial》(https://www.tutorialspoint.com/inter_process_communication/inter_process_communication_shared_memory.htm): 在 Tutorialspoint 上有一个关于 Linux 共享内存的简单教程，有助于理解共享内存的基本概念。

(2) 在 GitHub 等开源代码平台上搜索使用共享内存的实际项目和示例代码。

5.4.3 在线社区和论坛

在 Stack Overflow (<http://stackoverflow.p2hp.com/>)、Reddit (<https://www.redditinc.com/>) 等在线社区上参与讨论，了解其他开发者在共享内存方面的经验和建议。

以上资源能够帮助我们更好地理解和应用 Linux 下的共享内存机制。通过学习示例代码和实际项目，你可以提高在这一领域的技能。

5.5 消息队列

消息队列是 Linux 应用程序设计中常用的进程间通信（IPC）机制之一，用于在不同进程之间传递消息。以下是一些关于 Linux 下消息队列的相关资料：

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 18 / 39
		生效日期：2023 年 12 月 25 日

5.5.1 官方资源

[在线手册](https://man7.org/linux/man-pages/): 在线浏览 Linux 的 man 手册，查找与消息队列相关的系统调用和函数。

5.5.2 示例与源码

(1) 《Linux Message Queue Example》(https://www.tutorialspoint.com/inter_process_communication/inter_process_communication_message_queues.htm): 在 Tutorialspoint 上有一个关于 Linux 消息队列的简单教程，有助于理解消息队列的基本概念。

(2) 在 GitHub 等开源代码平台上搜索使用消息队列的实际项目和示例代码。

5.5.3 在线社区和论坛

在 Stack Overflow (http://stackoverflow.p2hp.com/)、Reddit (https://www.redditinc.com/) 等在线社区上参与讨论，了解其他开发者在消息队列方面的经验和建议。

以上资源能够帮助我们更深入地理解和应用 Linux 下的消息队列机制。通过学习示例代码和实际项目，你可以提高在这一领域的技能。

6、多线程编程

6.1 Linux 下线程概述

在 Linux 应用程序设计中，线程是一种轻量级的并发执行单位，能够共享同一进程的资源。以下是一些关于 Linux 下线程概述的相关资料：

6.1.1 官方资源

[在线手册](https://man7.org/linux/man-pages/): 在线浏览 Linux 的 man 手册，查找与线程相关的系统调用和函数。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 19 / 39
		生效日期：2023 年 12 月 25 日

6.1.2 书籍与文档

- (1) 《Advanced Programming in the UNIX Environment》（译版《UNIX 环境高级编程》）by W. Richard Stevens: 该书详细介绍了 Unix 环境下高级编程，包括线程的概念、创建、同步等方面。
- (2) 《Linux System Programming》（译版《Linux 系统编程》）by Robert Love: 本书涵盖了 Linux 系统编程的多个方面，包括线程的概述和使用。
- (3) 《The Linux Programming Interface》（译版《Linux/UNIX 系统编程手册》）by Michael Kerrisk: 本书深入研究了 Linux 系统编程，包括线程的实现和使用。
- (4) 《Understanding the Linux Kernel》（译版《深入理解 LINUX 内核》）by Daniel P. Bovet and Marco Cesati: 该书深入剖析了 Linux 内核，包括关于进程和线程管理的章节。

6.1.3 示例与源码

- (1) 《POSIX Threads Programming》：POSIX 线程是 Linux 下常用的线程库，可以参考 POSIX Threads 的相关文档和教程。
- (2) 在 GitHub 等开源代码平台上搜索使用线程的实际项目和示例代码。

6.1.4 在线社区和论坛

在 Stack Overflow (<http://stackoverflow.com/>)、Reddit (<https://www.reddit.com/>) 等在线社区上参与讨论，了解其他开发者在 Linux 线程编程方面的经验和建议。

以上资源能够帮助我们更好地理解和应用 Linux 下的线程编程。通过学习示例代码和实际项目，你可以提高在这一领域的技能。

6.2 Linux 线程实现

Linux 线程实现的底层细节涉及到内核级别的内容，主要是在 Linux 内核中的线程管理。以下是一些关于 Linux 线程实现的相关资料：

6.2.1 官方资源

[在线手册](<https://man7.org/linux/man-pages/>): 在线浏览 Linux 的 man 手册，查找与线程实现相关

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 20 / 39
		生效日期：2023 年 12 月 25 日

的系统调用和函数。

6.2.2 书籍与文档

- (1) 《Understanding the Linux Kernel》（译版《深入理解 LINUX 内核》）by Daniel P. Bovet and Marco Cesati: 该书深入剖析了 Linux 内核，包括关于进程和线程管理的章节，对 Linux 线程实现有较为详细的介绍。
- (2) 《Linux Kernel Development》（译版《Linux 内核设计与实现》）by Robert Love: 本书介绍了 Linux 内核的开发，包括进程和线程的管理。对于理解 Linux 线程实现有帮助。
- (3) 《Linux Kernel in a Nutshell》（译版《Linux 内核技术手册》）by Greg Kroah-Hartman: 该书简要概述了 Linux 内核的基本概念，涵盖了一些关于线程实现的基础知识。
- (4) 《Linux Kernel Documentation》:Linux 内核文档包含在源代码中，可以在 `Documentation/` 目录下找到。特别是在 `Documentation/scheduler/` 和 `Documentation/core-api/` 目录下，有一些关于线程实现的文档。

6.2.3 示例与源码

[Linux 内核源代码]可以直接查看 Linux 内核的源代码，特别是关于进程和线程管理的部分。Linux 内核源代码可以从 [kernel.org](https://www.kernel.org/) 获取。

6.2.4 在线社区与论坛

Linux 内核的开发讨论发生在内核邮件列表。你可以在 [LKML (Linux Kernel Mailing List)](https://lkml.org/) 上查找关于线程实现的讨论。

请注意，深入了解 Linux 线程实现需要对内核和系统编程有较高的理解水平。如果你是初学者，建议首先学习 Linux 下的用户级线程库（如 POSIX Threads）的使用，然后再逐渐深入了解内核级别的线程实现。

6.3 线程的同步与互斥

线程的同步和互斥是多线程编程中非常重要的概念，用于协调多个线程之间的执行。以下是一些关于 Linux 下线程同步与互斥的相关资料：

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 21 / 39
		生效日期：2023 年 12 月 25 日

6.3.1 官方资源

[在线手册](https://man7.org/linux/man-pages/): 在线浏览 Linux 的 man 手册，查找与线程同步和互斥相关的系统调用和函数。

6.3.2 书籍与文档

(1) 《Advanced Programming in the UNIX Environment》（译版《UNIX 环境高级编程》）by W. Richard Stevens: 该书详细介绍了 Unix 环境下高级编程，包括线程同步和互斥的概念，以及相关的系统调用。

(2) 《Linux System Programming》（译版《Linux 系统编程》）by Robert Love: 本书涵盖了 Linux 系统编程的多个方面，包括线程同步和互斥的使用。

(3) 《The Linux Programming Interface》（译版《Linux/UNIX 系统编程手册》）by Michael Kerrisk: 本书深入研究了 Linux 系统编程，包括线程同步和互斥的实现和使用。

(4) 《C++ Concurrency in Action》（译版《C++并发编程实战》）by Anthony Williams: 本书介绍了 C++11 中引入的多线程和并发编程的内容，其中包括了一些关于线程同步和互斥的实践。

6.3.3 示例与源码

(1) POSIX Threads Programming: POSIX 线程是 Linux 下常用的线程库，可以参考 POSIX Threads 的相关文档和教程，了解如何使用互斥锁和条件变量等。

(2) 在 GitHub 等开源代码平台上搜索使用线程同步和互斥的实际项目和示例代码。

6.3.4 在线社区和论坛

在 Stack Overflow (<http://stackoverflow.p2hp.com/>)、Reddit (<https://www.redditinc.com/>) 等在线社区上参与讨论，了解其他开发者在线程同步和互斥方面的经验和建议。

以上资源能够帮助我们更好地理解和应用 Linux 下线程同步与互斥的机制。通过学习示例代码和实际项目，你可以提高在这一领域的技能。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 22 / 39
		生效日期：2023 年 12 月 25 日

6.4 可重入与线程安全

在 Linux 应用程序设计中，可重入性（reentrancy）和线程安全性（thread safety）是编写多线程应用程序时需要特别关注的概念。以下是一些关于 Linux 下可重入性和线程安全性的相关资料：

6.4.1 官方资源

[在线手册](https://man7.org/linux/man-pages/): 在线浏览 Linux 的 man 手册，查找与可重入性和线程安全性相关的系统调用和函数。

6.4.2 书籍与文档

- (1) 《Advanced Programming in the UNIX Environment》（译版《UNIX 环境高级编程》）by W. Richard Stevens: 该书详细介绍了 Unix 环境下高级编程，包括可重入性和线程安全性的概念。它涵盖了许多系统调用和库函数的可重入性和线程安全性特性。
- (2) 《The Linux Programming Interface》（译版《Linux/UNIX 系统编程手册》）by Michael Kerrisk: 本书深入研究了 Linux 系统编程，包括可重入性和线程安全性的实现和使用。
- (3) 《C Programming FAQs: Frequently Asked Questions》by Steve Summit: 本书中有一些关于可重入性和线程安全性的常见问题的解答，可以帮助理解这两个概念。
- (4) 《C Interfaces and Implementations: Techniques for Creating Reusable Software》（译版《C 语言接口与实现》）by David R. Hanson: 本书介绍了 C 语言中实现可重入性和线程安全性的技术。

6.4.3 示例与源码

- (1) 《POSIX Threads Programming》: POSIX 线程是 Linux 下常用的线程库，了解 POSIX 线程的文档和规范可以帮助你编写可重入和线程安全的代码。
- (2) 在 GitHub 等开源代码平台上搜索使用可重入和线程安全编程实践的项目和示例代码。

6.4.4 在线社区和论坛

在 Stack Overflow (<http://stackoverflow.com/>)、Reddit (<https://www.reddit.com/>) 等在线社区上参与讨论，了解其他开发者在可重入性和线程安全性方面的经验和建议。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 23 / 39
		生效日期：2023 年 12 月 25 日

通过深入学习这些资料，我们可以更好地理解和应用 Linux 下的可重入性和线程安全性原则。

7、Qt 图形编程

7.1 Qt GUI 开发入门

Qt 是一个跨平台的 C++图形用户界面开发框架，用于开发桌面、移动、嵌入式和实时操作系统上的应用程序。以下是一些入门 Qt GUI 开发的相关资料：

7.1.1 官方资源

- (1) [Qt 官方文档](https://doc.qt.io/): Qt 的官方文档是学习和使用 Qt 的最佳资源。包含了详细的教程、示例和 API 文档。
- (2) [Qt Creator 集成开发环境]Qt Creator 是 Qt 官方提供的集成开发环境，使用它可以方便地进行 Qt 应用程序的开发。

7.1.2 书籍与文档

- (1) 《C++ GUI Programming with Qt 4》（译版《C++ GUI Qt4 编程》）by Jasmin Blanchette and Mark Summerfield: 这是一本经典的 Qt GUI 编程书籍，适合想要深入学习 Qt 的开发者的。
- (2) 《Qt5 C++ GUI Programming Cookbook》by Lee Zhi Eng: 提供了许多实用的 Qt 编程技巧和示例，适合希望通过实际项目经验学习的人。
- (3) 《An Introduction to Design Patterns in C++ with Qt》（译版《C++ Qt 设计模式》，第二版）by Alan Ezust and Paul Ezust: 介绍 Qt 基础知识与核心类、面向对象概念、UML 和设计模式等。该书普遍适用，不需要任何 C/C++编程经验。
- (4) 《Advanced Qt Programming: Creating Great Software with C++ and Qt4》(译版《Qt 高级编程》): 讲解多媒体、Model/View 架构、多线程等技术。适合想要深入学习 Qt 的开发者的。

7.1.3 示例与源码

Qt 安装目录下有丰富的示例项目，通过查看这些示例可以学到很多实际应用的技巧。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 24 / 39
		生效日期：2023 年 12 月 25 日

7.1.4 在线社区与论坛

- (1) [Qt Blog](https://blog.qt.io/): Qt 官方博客，提供有关 Qt 最新开发和技术的信息。
- (2) [Qt Centre Forum](https://www.qtcentre.org/): Qt Centre 是一个 Qt 开发者社区，你可以在这里找到许多问题的答案。

7.1.5 教程与视频

[Qt 教程 - C++ GUI 开发](https://www.bogotobogo.com/Qt/Qt5_TutorialHelloWorld.php): 一个简单的 Qt 教程，适合初学者入门。

请根据自己的水平和需求选择适合的学习资源。Qt 的学习过程中，实践是非常重要的，因此尝试编写一些简单的 Qt 应用程序以加深理解。

7.2 窗口部件

Qt 提供了丰富的窗口部件（Widget）来构建用户界面。以下是一些关于 Qt 窗口部件的相关资料：

7.2.1 官方资源

- (1) [Qt Widgets 模块](https://doc.qt.io/qt-5/qwidget.html): Qt 官方文档中关于 Widgets 模块的详细介绍，包括各种窗口部件的说明和示例。
- (2) [Qt Creator 集成开发环境]Qt Creator 是 Qt 官方提供的集成开发环境，它集成了代码编辑、调试和可视化设计等功能，方便开发者进行 Qt 应用程序的开发。

7.2.2 书籍与文档

(1) 《C++ GUI Programming with Qt 4》（译版《C++ GUI Qt4 编程》）by Jasmin Blanchette and Mark Summerfield: 这本经典书籍不仅涵盖了 Qt 的基础知识，还深入介绍了 Qt Widgets 的使用。

7.2.3 示例与源码

Qt 安装目录下有许多示例项目，涉及到不同类型的窗口部件和功能。通过查看这些示例，可以更好

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 25 / 39
		生效日期：2023 年 12 月 25 日

地理解如何使用 Qt 窗口部件。

7.2.4 在线社区与论坛

(1) [Qt 中文社区](http://www.qtcn.org/bbs/index.php): 这是一个 Qt 中文社区，你可以在这里找到一些建议和问题的解答，也可以参与讨论。

(2) [Qt 开源社区](https://www.qter.org/portal.php): 这是一个 Qt 开源社区，你可以在这里找到一些建议和问题的解答，也可以参与讨论，同时可以找到一些教程。

7.2.5 教程与视频

[Qt Designer 教程](https://doc.qt.io/qt-5/qtdesigner-manual.html): Qt Designer 是一个可视化的界面设计工具，允许你通过拖拽和放置的方式设计界面。

以上资源能够帮助我们更好地了解和使用 Qt 窗口部件。记得实际动手编写一些简单的应用程序，以加深对窗口部件的理解。

7.3 布局管理

Qt 中的布局管理是一种用于自动管理窗口部件位置和大小的机制，以确保在不同平台和不同屏幕尺寸下，应用程序的用户界面能够良好地适应。以下是关于 Qt 布局管理的一些相关资料：

7.3.1 官方资源

(1) [Qt Layouts](https://doc.qt.io/qt-5/qlayout.html): Qt 官方文档中关于布局管理的详细说明，包括各种不同类型的布局管理器和如何使用它们的示例。

(2) [Qt Creator 集成开发环境]Qt Creator 集成了 Qt Designer，你可以在其中使用可视化工具来创建和编辑布局。

7.3.2 书籍与文档

《C++ GUI Programming with Qt 4》（译版《C++ GUI Qt4 编程》）by Jasmin Blanchette and Mark Summerfield: 这本书不仅覆盖了 Qt 的基础知识，还提供了关于布局管理的深入教程和示例。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 26 / 39
		生效日期：2023 年 12 月 25 日

7.3.3 示例与源码

Qt 安装目录下有很多示例项目，其中包括使用不同布局管理器的示例。通过查看这些示例，可以更好地理解如何设计灵活的用户界面。

7.3.4 教程与视频

- (1) [Qt 布局管理教程](https://www.bogotobogo.com/Qt/Qt5_Layouts.php): 一个简单的 Qt 布局管理教程，适合初学者入门。
- (2) [Qt 布局管理器介绍](https://www.devbean.net/2012/09/qt-study-road-2-layout/): 针对 Qt 布局管理学习者的教程。
- (3) [Qt Designer 教程](https://doc.qt.io/qt-5/qtdesigner-manual.html): Qt Designer 是一个可视化的界面设计工具，你可以使用它来创建和调整布局。

以上资源能够帮助我们更好地了解和使用 Qt 布局管理器。实践是学习的关键，所以尽量动手编写一些包含不同布局的应用程序以加深理解。

7.4 主窗口框架

Qt 的主窗口框架通常是通过继承 `QMainWindow` 类来创建的，这个框架包括了菜单栏、工具栏、状态栏等基本组件。以下是一些关于 Qt 主窗口框架的相关资料：

7.4.1 官方资源

- (1) [Qt QMainWindow Class](https://doc.qt.io/qt-5/qmainwindow.html): Qt 官方文档中关于 `QMainWindow` 类的详细说明，包括构建主窗口框架的基本步骤和示例。
- (2) [Qt Creator 集成开发环境]Qt Creator 集成了 Qt Designer，你可以在其中使用可视化工具来创建和编辑主窗口框架。

7.4.2 书籍与文档

《C++ GUI Programming with Qt 4》（译版《C++ GUI Qt4 编程》）by Jasmin Blanchette and Mark Summerfield: 这本经典书籍包括了关于 `QMainWindow` 的深入教程和示例。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 27 / 39
		生效日期：2023 年 12 月 25 日

7.4.3 示例与源码

Qt 安装目录下有很多示例项目，其中包括了使用 `QMainWindow` 的示例。通过查看这些示例，可以更好地理解如何设计和使用主窗口框架。

7.4.4 教程与视频

- (1) [Qt 主窗口框架教程](https://www.bogotobogo.com/Qt/Qt5_MainWindowTriggerAction.php): 一个简单的 Qt 主窗口框架教程，适合初学者入门。
- (2) [Qt Creator 教程](https://doc.qt.io/qtcreator/creator-tutorial.html): Qt Creator 是一个集成开发环境，它提供了创建和编辑 `QMainWindow` 的可视化工具。
- (3) [Qt Designer 教程](https://doc.qt.io/qt-5/qtdesigner-manual.html): Qt Designer 是一个可视化的界面设计工具，你可以使用它来创建和调整主窗口框架。

以上资源能够帮助我们更好地了解和使用 Qt 的主窗口框架。实践是学习的关键，因此尽量动手编写一些包含主窗口框架的应用程序以加深理解。

8、设备驱动开发

设备驱动开发共用书籍与文档：

- (1) 《Linux Device Drivers》（译版《Linux 设备驱动程序》）by Jonathan Corbet, Alessandro Rubini, and Greg Kroah-Hartman: 该书是关于 Linux 设备驱动开发的经典之作，提供了深入的介绍和实践经验。第三版包括了对最新内核版本的支持。
- (2) Linux Kernel Documentation:Linux 内核文档中有关于设备驱动的详细文档，包括各种设备类型的驱动开发指南。这些文档通常位于 `Documentation/` 目录下。

8.1 设备驱动概述

在 Linux 应用程序设计中，设备驱动是一个关键的领域，特别是在嵌入式系统和嵌入式 Linux 开发中。以下是一些关于 Linux 设备驱动概述的相关资料：

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 28 / 39
		生效日期：2023 年 12 月 25 日

8.1.1 书籍与文档

- （1）《Linux Kernel Development》（译版《Linux 内核设计与实现》）by Robert Love：本书不仅介绍了 Linux 内核的基本概念，还涵盖了设备驱动的开发和调试。
- （2）《Linux System Programming》（译版《Linux 系统编程》）by Robert Love：该书涵盖了 Linux 系统编程的多个方面，包括设备驱动的一些基础知识。
- （3）《Linux 设备驱动开发详解》作者宋宝华：这是一本介绍 Linux 设备驱动开发理论、框架与实例的书，《Linux 设备驱动开发详解（第 2 版）》基于 LDD6410 开发板，以 Linux2.6 版本内核为蓝本，详细介绍 Linux 设备驱动的架构和框架中各个复杂数据架构和函数的关系，并讲解了 Linux 驱动开发的大量实例，使读者能够独立开发各类 Linux 设备驱动。
- （4）《嵌入式 Linux 驱动开发教程》作者姜先刚、刘洪涛：本书结合大量实例，在基于 ARM Cortex-A9 四核处理器 Exynos4412 的硬件教学平台和 PC 上，全面详细地讲解了 Linux 设备驱动开发。在需要重点关注的地方还加入了大量的内核源码分析，使读者能够快速并深刻理解 Linux 设备驱动的开发。

8.1.2 示例与源码

- （1）Linux Device Drivers, Third Edition (Free Electrons): Free Electrons 公司维护了一份关于 Linux 设备驱动开发的在线教程，可以免费访问。这份教程包含了许多实例和案例。
- （2）Linux Driver Project: [Linux Driver Project](https://www.linuxdriverproject.org/) 是一个旨在帮助开发者贡献设备驱动到 Linux 内核的项目。该网站包含一些建议的项目和开发资源。
- （3）[GitHub 仓库]https://e.coding.net/weidongshan/01_all_series_quickstart.git: 韦东山驱动开发资料。
- （4）[内核源代码]了解 Linux 设备驱动开发的最好方式之一是直接查看内核源代码。可以从 [kernel.org](https://www.kernel.org/) 获取最新的内核源代码。

8.1.3 在线社区和论坛

在 Stack Overflow (<http://stackoverflow.p2hp.com/>)、Reddit (<https://www.redditinc.com/>)、以及 Linux 内核邮件列表（发送教程: <https://zhuanlan.zhihu.com/p/476648206>）上参与讨论，向其他设备驱动开发者请教问题和分享经验。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 29 / 39
		生效日期：2023 年 12 月 25 日

8.1.4 教程与视频

- (1) <https://pan.baidu.com/s/1JU95JHG-v7MKvkvXsMNhdw> 提取码：n3ju：阿尔法 Linux 开发板 Linux 驱动开发资料；
- (2) <https://www.bilibili.com/video/BV14f4y1Q7ti>：嵌入式 Linux 驱动开发基础知识。

以上资源能够帮助我们深入理解 Linux 设备驱动的概念和实践。请注意，设备驱动开发通常需要对硬件和内核具有深入的理解，因此可能需要一些时间和实践来掌握这一领域。

8.2 字符设备驱动编写

编写 Linux 字符设备驱动是 Linux 内核模块开发的一部分，它涉及到字符设备的注册、文件操作、缓冲区管理等方面。以下是一些关于 Linux 字符设备驱动编写的相关资料：

8.2.1 官方资源

[在线工具](<https://elixir.bootlin.com/linux/latest/source>): 在线查看 Linux 内核源代码的工具，可用于浏览字符设备相关的代码和数据结构。

8.2.2 书籍与文档

- (1) 《Linux System Programming》（译版《Linux 系统编程》）by Robert Love：该书涵盖了 Linux 系统编程的多个方面，其中包括字符设备驱动的一些基础知识。
- (2) [Linux Kernel Module Programming Guide](<http://tldp.org/LDP/lkmpg/2.6/html/index.html>): Linux Documentation Project 维护的在线指南，包括了关于字符设备驱动开发的一些内容。
- (3) 《嵌入式 Linux 驱动开发教程》作者姜先刚、刘洪涛：本书结合大量实例，在基于 ARM Cortex-A9 四核处理器 Exynos4412 的硬件教学平台和 PC 上，全面详细地讲解了 Linux 设备驱动开发。在需要重点关注的地方还加入了大量的内核源码分析，使读者能够快速并深刻理解 Linux 设备驱动的开发。
- (4) 《Linux 设备驱动开发详解》作者宋宝华：这是一本介绍 Linux 设备驱动开发理论、框架与实例的书，《Linux 设备驱动开发详解（第 2 版）》基于 LDD6410 开发板，以 Linux2.6 版本内核为蓝本，详细介绍 Linux 设备驱动的架构和框架中各个复杂数据架构和函数的关系，并讲解了 Linux 驱动开发的大量实例，使读者能够独立开发各类 Linux 设备驱动。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 30 / 39
		生效日期：2023 年 12 月 25 日

8.2.3 示例与源码

- (1) Linux Device Drivers, Third Edition (Free Electrons): Free Electrons 公司维护了一份关于 Linux 字符设备驱动开发的在线教程，包含实例和案例。
- (2) [内核源代码]查看 Linux 内核源代码是学习字符设备驱动编写的重要途径。可以从 [kernel.org](https://www.kernel.org/) 获取最新的内核源代码。

8.2.4 在线社区和论坛

在 Stack Overflow (<http://stackoverflow.p2hp.com/>)、Reddit (<https://www.redditinc.com/>)、以及 Linux 内核邮件列表（发送教程：<https://zhuanlan.zhihu.com/p/476648206>）上参与讨论，向其他设备驱动开发者请教问题和分享经验。

通过深入学习这些资料，我们可以掌握字符设备驱动的编写原理和实践方法。请注意，在进行字符设备驱动开发时，了解 Linux 内核的基本概念和 API 是非常重要的。

8.3 块设备驱动编写

编写 Linux 块设备驱动是一项复杂的任务，因为它涉及到底层磁盘和文件系统的交互。以下是一些关于 Linux 块设备驱动编写的相关资料：

8.3.1 官方资源

[在线工具](<https://elixir.bootlin.com/linux/latest/source>): 在线查看 Linux 内核源代码的工具，可用于浏览块设备相关的代码和数据结构。

8.3.2 书籍与文档

- (1) 《Linux System Programming》（译版《Linux 系统编程》）by Robert Love: 该书涵盖了 Linux 系统编程的多个方面，其中包括块设备驱动的一些基础知识。
- (2) Linux Block I/O Layer:[Linux Block I/O Layer](<https://lwn.net/Articles/28345/>): LWN.net 上的文章，深入解释了 Linux 块 I/O 层的工作原理，对于理解块设备驱动很有帮助。
- (3) 《Linux 设备驱动开发详解》作者宋宝华: 这是一本介绍 Linux 设备驱动开发理论、框架与实

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 31 / 39
		生效日期：2023 年 12 月 25 日

例的书，《Linux 设备驱动开发详解（第 2 版）》基于 LDD6410 开发板，以 Linux2.6 版本内核为蓝本，详细介绍 Linux 设备驱动的架构和框架中各个复杂数据架构和函数的关系，并讲解了 Linux 驱动开发的大量实例，使读者能够独立开发各类 Linux 设备驱动。

8.3.3 示例与源码

- (1) Linux Device Drivers, Third Edition (Free Electrons): Free Electrons 公司维护了一份关于 Linux 块设备驱动开发的在线教程，包含实例和案例。
- (2) [内核源代码]查看 Linux 内核源代码是学习块设备驱动编写的重要途径。可以从 [kernel.org](https://www.kernel.org/) 获取最新的内核源代码。

8.3.4 在线社区和论坛

在 Stack Overflow (<http://stackoverflow.p2hp.com/>)、Reddit (<https://www.redditinc.com/>)、以及 Linux 内核邮件列表（发送教程：<https://zhuanlan.zhihu.com/p/476648206>）上参与讨论，向其他设备驱动开发者请教问题和分享经验。

注意，块设备驱动开发通常需要对磁盘和文件系统的工作原理有一定的了解。深入学习这些资料，结合实际的编程实践，将帮助我们更好地理解和掌握 Linux 块设备驱动的开发。

8.4 中断编程

在 Linux 应用程序设计中，中断编程是一个关键的主题，尤其是在与硬件交互和驱动开发中。以下是一些关于 Linux 中断编程的相关资料：

8.4.1 官方资源

[在线工具](<https://elixir.bootlin.com/linux/latest/source>): 在线查看 Linux 内核源代码的工具，可用于浏览与中断相关的代码和数据结构。

8.4.2 书籍与文档

- (1) 《Linux System Programming》（译版《Linux 系统编程》）by Robert Love: 该书涵盖了

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 32 / 39
		生效日期：2023 年 12 月 25 日

Linux 系统编程的多个方面，其中包括中断编程的一些基础知识。

(2) Linux IRQ Handling:[Linux IRQ Handling](https://www.kernel.org/doc/html/v5.10/core-api/irq/index.html): Linux 内核文档中关于 IRQ 处理的部分，包括注册中断处理程序等内容。

(3) 《嵌入式 Linux 驱动开发教程》作者姜先刚、刘洪涛：本书结合大量实例，在基于 ARM Cortex-A9 四核处理器 Exynos4412 的硬件教学平台和 PC 上，全面详细地讲解了 Linux 设备驱动开发。在需要重点关注的地方还加入了大量的内核源码分析，使读者能够快速并深刻理解 Linux 设备驱动的开发。

8.4.3 示例与源码

(1) Linux Device Drivers, Third Edition (Free Electrons): Free Electrons 公司维护了一份关于 Linux 中断编程的在线教程，包含实例和案例。

(2) [内核源代码]查看 Linux 内核源代码是学习中断编程的重要途径。你可以从 [kernel.org](https://www.kernel.org/) 获取最新的内核源代码。

8.4.4 在线社区和论坛

在 Stack Overflow (<http://stackoverflow.p2hp.com/>)、Reddit (<https://www.redditinc.com/>)、以及 Linux 内核邮件列表（发送教程：<https://zhuanlan.zhihu.com/p/476648206>）上参与讨论，向其他内核开发者请教问题和分享经验。

深入学习这些资料，特别是理解中断处理的机制和实现，将有助于我们在 Linux 系统上进行更高级的硬件交互和设备驱动开发。

8.5 键盘驱动实现

在 Linux 应用程序设计中，键盘驱动的实现通常属于设备驱动开发的范畴。以下是一些关于 Linux 键盘驱动实现的相关资料：

8.5.1 官方资源

(1) `evtest` 是一个在 Linux 下测试输入设备的工具，可以用于调试和测试键盘驱动。你可以在终端中运行 `evtest` 并选择键盘设备来查看输入事件。

(2) [在线工具](https://elixir.bootlin.com/linux/latest/source)：在线查看 Linux 内核源代码的工具，

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 33 / 39
		生效日期：2023 年 12 月 25 日

可用于浏览与输入设备和键盘相关的代码和数据结构。

8.5.2 书籍与文档

Linux Input Subsystem:[Linux Input Subsystem](https://www.kernel.org/doc/html/latest/input/input.html): Linux 内核文档中关于输入子系统的文档，包括了键盘和其他输入设备的处理。

8.5.3 示例与源码

[内核源代码]查看 Linux 内核源代码是学习键盘驱动实现的重要途径。你可以从 [kernel.org](https://www.kernel.org/) 获取最新的内核源代码。

8.5.4 在线社区和论坛

在 Stack Overflow (<http://stackoverflow.p2hp.com/>)、Reddit (<https://www.redditinc.com/>)、以及 Linux 内核邮件列表（发送教程：<https://zhuanlan.zhihu.com/p/476648206>）上参与讨论，向其他设备驱动开发者请教问题和分享经验。

学习这些资料将有助于我们理解 Linux 系统中键盘驱动的实现原理和开发流程。

9、内存管理

9.1 内存管理的基本原理和方法

内存管理是操作系统和应用程序设计中一个重要的主题。以下是一些关于 Linux 应用程序设计中内存管理的基本原理和方法的相关资料：

9.1.1 官方资源

[在线工具](https://elixir.bootlin.com/linux/latest/source): 在线查看 Linux 内核源代码的工具，可用于浏览与内存管理相关的代码和数据结构。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 34 / 39
		生效日期：2023 年 12 月 25 日

9.1.2 书籍与文档

- (1) 《Understanding the Linux Kernel》（译版《深入理解 LINUX 内核》）by Daniel P. Bovet and Marco Cesati: 该书深入介绍了 Linux 内核的各个方面，包括内存管理。它对虚拟内存、物理内存、页表等进行了详细的讲解。
- (2) Linux Kernel Documentation:Linux 内核文档中有关于内存管理的详细文档，包括内存分配、虚拟内存、页表等方面的内容。这些文档通常位于 `Documentation/vm/` 和 `Documentation/mm/` 目录下。
- (3) 《Understanding the Linux Virtual Memory Manager》(译版《深入理解 Linux 虚拟内存管理》) by Mel Gorman: 该书专注于 Linux 虚拟内存管理器的理解，提供了对内核中关键概念的详细解释。
- (4) Linux Memory Management Documentation:[Linux Memory Management Documentation](<https://www.kernel.org/doc/html/latest/admin-guide/mm/index.html>): Linux 内核文档中关于内存管理的在线文档，涵盖了多个方面，包括内存区域、页表、物理内存等。
- (5) 《Linux System Programming》（译版《Linux 系统编程》）by Robert Love: 该书涵盖了 Linux 系统编程的多个方面，包括了一些关于内存管理的基础知识。
- (6) Linux Performance by Brendan Gregg: Brendan Gregg 的网站包含了丰富的关于 Linux 性能和调试的资源，其中有关于内存的文章对理解和调优内存管理非常有帮助。

9.1.3 在线社区和论坛

在 Stack Overflow (<http://stackoverflow.p2hp.com/>)、Reddit (<https://www.redditinc.com/>)、以及 Linux 内核邮件列表上参与讨论，向其他开发者请教问题和分享经验。

通过深入学习这些资料，将更好地理解 Linux 系统中内存管理的基本原理和方法，能够更有效地进行应用程序设计和调优。

10、性能优化

以下是 Linux 应用程序性能优化的相关资料：

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 35 / 39
		生效日期：2023 年 12 月 25 日

10.1 官方资源

(1) **Linux Documentation:** Linux 内核文档包含了很多关于性能调优和内核参数的信息。可以在 `/usr/src/linux/Documentation` 目录下找到这些文档，或者在 [Linux Kernel Archives](https://www.kernel.org/doc/) 上在线查看。

(2) **GNU Compiler Documentation:** 如果涉及到编译器优化，可以查阅 GCC 的官方文档，了解不同编译器标志的含义和使用方法。

(3) **Perf Examples:** Perf 工具提供了许多示例用法，可以通过查看 `man perf` 或在终端中运行 `perf --help` 来获取更多信息。

10.2 书籍与文档

(1) 《Optimized C++: Proven Techniques for Heightened Performance》（译版《C++性能优化指南》，杨文轩译） by Kurt Guntheroth: 本书介绍了 C++代码优化的意义和原则、性能分析方法和工具等，详细描述了如何优化字符串、动态内存分配、数据结构和并发等内容。

(2) 《Linux 高性能服务器编程》作者游双: 本书介绍了高级 I/O 函数、高性能服务器程序框架、I/O 复用等内容，讲解了高性能 I/O 框架库 Libevent 的应用、程序调试和测试工具、多线程/多进程编程技术和系统状态监测工具的应用。

(3) 《Linux 性能优化大师》作者赵永刚: 本书介绍了 Linux 系统监测工具和性能测试工具的应用，讲解如何定位性能瓶颈和调整操作系统的内容。

10.3 示例与源码

Linux Kernel Source Code: 阅读 Linux 内核源代码是学习系统级性能优化的好方法。可以从 [Linux Kernel Archives](https://www.kernel.org/) 下载最新的内核源代码。

10.4 在线社区与论坛

在 Stack Overflow (<http://stackoverflow.com/>)、Reddit (<https://www.reddit.com/>)、以及 LinuxQuestions(<https://www.linuxquestions.org/>)上参与关于 Linux 性能和优化的讨论。

性能优化是一个广泛的领域，最好的学习方法是实际操作。通过分析示例、阅读源码、参与社区讨论，我们将逐渐积累经验并提高对性能调优的理解。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 36 / 39
		生效日期：2023 年 12 月 25 日

11、开源第三方库

11.1 常用的第三方库

在 Linux 应用程序设计中，使用第三方库是相当常见的，它们提供了丰富的功能和工具，以便加速开发。以下是一些常用的 Linux 下的第三方库，以及学习它们的相关资料：

11.1.1 Boost C++ Libraries

Boost 是一个广泛使用的 C++ 库集合，提供了许多工具和数据结构的实现，包括线程、文件系统、智能指针等。它是一个跨平台的库，为 C++ 开发者提供了丰富的工具。可以从 [Boost 官方网站](<http://www.boost.org/>) 获取文档和教程。

11.1.2 Qt

Qt 是一个跨平台的 C++ 库，用于创建图形用户界面和应用程序。它提供了丰富的功能，包括 GUI、数据库访问、网络通信等。官方文档和教程可以在 [Qt 官方网站](<https://www.qt.io/>) 上找到。

11.1.3 GTK

GTK (GIMP Toolkit) 是一个用于创建图形用户界面的库，特别是在 Linux 系统上。它被许多 Linux 桌面环境使用，如 GNOME。学习资源可以在 [GTK 官方网站](<https://www.gtk.org/>) 上找到。

11.1.4 OpenCV

OpenCV 是一个开源计算机视觉库，提供了许多图像处理和计算机视觉算法的实现。它支持各种编程语言，包括 C++ 和 Python。你可以在 [OpenCV 官方网站](<https://opencv.org/>) 获取文档和教程。

11.1.5 FFmpeg

FFmpeg 是一个用于处理多媒体数据的开源工具库。它可以用于音频和视频处理。学习资源可以在 [FFmpeg 官方网站](<https://ffmpeg.org/>) 找到。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 37 / 39
		生效日期：2023 年 12 月 25 日

11.1.6 SQLite

SQLite 是一个嵌入式关系型数据库管理系统，适用于小型到中型的数据库应用。它是一个轻量级、零配置的数据库，非常适合嵌入到应用程序中。它的文档和教程可以在 [SQLite 官方网站](https://www.sqlite.org/) 上找到。

11.1.7 CMake

CMake 是一个用于管理跨平台构建过程的工具，它的语法简洁，并且支持多种编程语言。你可以在 [CMake 官方网站](https://cmake.org/) 上找到学习资源。

11.1.8 gRPC

gRPC 是一个开源的高性能 RPC（远程过程调用）框架，支持多种编程语言。学习资源可以在 [gRPC 官方网站](https://grpc.io/) 上找到。

11.1.9 ZeroMQ

ZeroMQ 是一个消息传递库，用于构建分布式和并发应用程序。它提供了多种消息传递模式，如请求-应答、发布-订阅等。可以在 [ZeroMQ 官方网站](https://zeromq.org/) 上找到学习资源。

11.1.10 Eigen

Eigen 是一个用于线性代数运算的 C++ 模板库。它提供了矩阵和向量的运算，适用于科学计算和图形学等领域。它的文档可以在 [Eigen 官方网站](https://eigen.tuxfamily.org/) 上找到。

11.1.11 libmodbus

libmodbus 是一个用于支持 Modbus 通信协议的 C 语言库。Modbus 是一种用于工业自动化领域的通信协议，常用于设备之间的数据交换。以下是学习 libmodbus 的相关资料：

11.1.11.1 官方网站

[libmodbus 官方网站](https://libmodbus.org/)：可以在官方网站上找到库的最新版本、文档、示例和其他资源。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 38 / 39
		生效日期：2023 年 12 月 25 日

11.1.11.2 文档

`libmodbus` 包含详细的文档，介绍了库的 API、使用方法以及示例代码。可以在官方网站上找到在线文档。

11.1.11.3 GitHub 仓库

[libmodbus GitHub 仓库](https://github.com/stephane/libmodbus): 可以在 GitHub 上获取源代码、报告问题、查看贡献等。

11.1.11.4 Example 目录

`libmodbus` 的源代码包中通常包含一个名为 "example" 或 "examples" 的目录，其中包含一些使用库的示例代码。

11.1.11.5 使用指南

`libmodbus` 的使用指南和教程可以在官方文档中找到，以帮助你更好地理解如何在应用程序中集成和使用该库。

11.1.12 Libevent

libevent 是一个事件通知库，用于在网络编程中进行异步 I/O 操作。以下是学习 libevent 的相关资料：

11.1.12.1 官方网站

[libevent 官方网站](https://libevent.org/): 可以在官方网站上找到库的最新版本、文档、示例和其他资源。

11.1.12.2 文档

`libevent` 包含详细的文档，介绍了库的 API、使用方法以及示例代码。可以在官方网站上找到在线文档。

文件编号：HN/WI-GC-YF-061	C 和 C++安全编程指南	版本：V1.0
制定部门：研发中心		页码： 39 / 39
		生效日期：2023 年 12 月 25 日

11.1.12.3 GitHub 仓库

[libevent GitHub 仓库](https://github.com/libevent/libevent): 可以在 GitHub 上获取源代码、报告问题、查看贡献等。

11.1.12.4 Example 目录

`libevent` 的源代码包中通常包含一个名为 "example" 或 "examples" 的目录, 其中包含一些使用库的示例代码。

11.1.12.5 使用指南

`libevent` 的使用指南和教程可以在官方文档中找到, 以帮助我们更好地理解如何在应用程序中集成和使用该库。