|  |  |  |  |
| --- | --- | --- | --- |
| **《操作系统原理》实验报告** | | | |
| 实验名称 | Linux内核编译 | 实验序号 | 7 |
| 实验日期 | 2023/5/19 | 实验人 |  |
| 一、实验题目   * 下载、编译内核源代码[1][2] * 启动测试所编译出来的内核 * 使用Clang编译内核\* * 成功配置Linux Kernel静态分析工具[3] \* | | | |
| 二、相关原理与知识  1.Linux内核  Linux系统的内核是Linux操作系统的核心组件，它负责管理系统资源、提供对硬件设备的访问和控制、调度进程和管理内存等。它具有以下特点：  1.开源性：Linux内核是开源的，源代码可以被任何人查看、修改和分发。这种开放性促进了广泛的参与和协作，使得Linux内核能够快速发展和改进。  2.可定制性：Linux内核的可定制性非常高。用户可以根据自己的需求编译和配置内核，选择需要的功能和驱动程序，以及优化内核的性能和资源利用率。  3.可移植性：Linux内核被设计为具有良好的可移植性，可以在各种硬件架构和平台上运行，包括个人计算机、服务器、嵌入式系统和移动设备等。  Linux内核的可更换性和可定制性指的是可以替换原始内核，安装其他内核版本或自定义内核。这种可更换性是Linux的一个重要特点，它允许用户根据特定需求选择不同的内核版本或自定义内核。用户可以选择不同的内核版本，如稳定版、开发版或特定的内核分支，以满足不同的需求。例如，服务器环境可能更倾向于使用稳定版本的内核，而开发人员可能需要使用最新的开发版本来获取最新的功能和修复。此外，用户也可以根据自己的需求自定义内核。通过配置内核选项，用户可以选择启用或禁用特定的功能和模块，以及进行性能优化。这样可以使得内核更加精简，仅包含必需的功能，提高系统的性能和资源利用率。  2.编译和安装过程  对于linux内核的编译和安装过程，主要分为以下几步：  1.获得内核源代码：在linux系统中可以选择wget、git或者直接下载等方式获取内核源代码。  2.解压源代码包：使用命令 tar -xf <source\_code.tar.gz> 解压源代码包。其中 <source\_code.tar.gz> 是下载的内核源代码包的名称。  3.配置内核：运行命令 make <config\_target> 来配置内核选项。其中 <config\_target> 可以是不同的目标，如：   * defconfig：使用默认配置。 * menuconfig：以文本菜单方式进行配置。 * xconfig：以图形界面方式进行配置。   4.编译内核：使用命令 make 开始编译内核。  5.安装内核模块：安装内核模块是为了让系统能够支持各种硬件设备、文件系统和其他功能。内核模块是一种动态加载的代码，它们可以在运行时向内核添加新的功能或驱动程序。使用命令 make modules\_install 安装编译后的内核模块。  6.安装内核：使用命令 make install 安装编译后的内核。这个命令将会将编译后的内核文件复制到适当的位置，并更新引导加载程序（如GRUB）的配置文件。  7.重启系统：在启动过程中选择新的内核进行启动。 | | | |
| 三、实验过程  1.内核版本查看  在进行内核编译、安装与验证以前，首先查看麒麟linux系统中自带的linux内核版本，以便于对后续内核安装是否成功进行验证。  此处使用以下命令完成：   |  | | --- | | **uname** -mrs |   版本查看图如下：    可以看到，我所安装的麒麟linux系统的内核版本为linux 5.10.0-8。  2.内核代码下载  编译内核代码，首先需要下载内核。此处我使用wget命令，从北京交通大学的镜像网站下载5.10.1的linux版本内核。使用以下命令：   |  | | --- | | **wget** https:**//**mirror.bjtu.edu.cn**/**kernel**/**linux**/**kernel**/**v5.x**/**linux-5.10.1.tar.gz |   内核下载图如下：    3.解压内核代码  对下载完成的内核代码需要进行解压操作，此处可使用tar命令   |  | | --- | | **tar** xvf linux-5.10.1.tar.gz |   使用解压命令时，可以看到文件列表，内容较多，解压完成后如下图所示。    4.安装编译所需的软件包  在对内核进行编译以前，需要安装必要的依赖软件包，具体命令如下：   |  | | --- | | **sudo** **apt-get install** **git** fakeroot build-essential ncurses-dev xz-utils libssl-dev **bc** **flex** libelf-dev **bison** |   上述命令将安装下列编译、安装内核必要的软件库：  git：版本控制工具Git，用于获取和管理源代码。  fakeroot：用于在非特权（非超级用户）环境下执行需要特权操作的工具。  build-essential：包含了编译和构建软件所需的基本工具集，包括编译器、链接器和库文件。  ncurses-dev：用于开发基于终端的应用程序所需的开发库。  xz-utils：用于解压和压缩XZ格式的文件的实用工具。  libssl-dev：用于开发和编译使用SSL/TLS协议的应用程序所需的开发库。  bc：用于执行数学运算的计算器工具。  flex：用于生成词法分析器的工具。  libelf-dev：用于开发和编译与ELF（Executable and Linkable Format）二进制文件格式相关的应用程序所需的开发库。  bison：用于生成语法解析器的工具。  安装软件包图如下：    5.配置内核  此处我没有对内核进行独特配置，而是选择继承麒麟linux自带的配置。即使用命令：   |  | | --- | | **cp** -v **/**boot**/**config-$**(uname** -r**)** .config |   从当前机器的启动目录拷贝配置信息到源代码目录。这步操作生成的.config文件保证我们编译内核的配置采用用当前环境一致的配置。  虽然使用默认配置，但我同样打开了配置的界面，如下：   |  | | --- | | **make** menuconfig |     由于我们采用默认设置，因此不对界面中的内容进行修改。  6.编译内核  直接使用make命令对内核进行编译，会导致速度过慢，可以在后面加上-j 2的参数，加速编译速度。   |  | | --- | | **make** -j 2 |   编译内核运行图如下：    进行make指令以后便开始编译，此过程时间比较长，使用 -2 j参数以后运行了两个半小时，累计20g左右。当编译完成后，    7.安装内核模块  为内核安装内核模块，让系统能够支持各种硬件设备、文件系统和其他功能，其命令具体为：   |  | | --- | | **sudo** **make** modules\_install |   安装内核运行图如下：    当内核模块安装完成以后，    8.安装内核  当上述工作都完成以后，即可对内核进行安装，此时使用以下命令即可：   |  | | --- | | **sudo** **make** **install** |   内核安装图如下：    当内核安装完成以后，它会表示为：    9.重启并测试内核  经过上述操作，内核安装已经完成，此处对虚拟机进行重启，并对内核进行测试，验证内核安装是否成功。  首先在开机时进入GRUB，可以查看内核版本与选择：    可以发现，5.10.1即为我所用的linux安装包版本。成功开机后，我们可以使用命令查看linux内核版本，如下图：    可以发现内核版本发生了改变，变为我们安装的linux内核版本，以此完成了新内核的测试。 | | | |
| 四、问题总结  1. 在编译过程中，由于我在创建虚拟机时给虚拟机分配的硬盘内存过小，data盘仅为10g，因此编译失败。  因此我重装了虚拟机，并扩大了data盘容量至25g，最后发现编译后占用空间为20g以上，因此需要分配足够的空间。  2. 在内核源码包下载时，第一次我选择的是linux 5.15.1的版本，但是在编译过程中报错，因为选择更换了更为稳定的linux 5.10.1，并顺利编译与安装。 | | | |