实验1 编译Linux内核

班级： 07112102 学号： 1120210964 姓名： 王英泰

一、实验目的

学习重新编译Linux内核，理解、掌握Linux内核和发行版本的区别，提高对操作系统的认识和理解。

二、实验内容

1. 配置实验环境，在虚拟机上安装Ubuntu操作系统
2. 下载内核源码，安装编译内核所需要的工具
3. 配置编译选项，编译Linux内核
4. 分析实验结果

三、实验步骤

实验环境准备

* 下载VMware Workstation Pro安装包和Ubuntu镜像，并将Ubuntu安装在虚拟机上。
* 图形用户界面, 网站

  描述已自动生成VMware Workstation Pro17下载地址为：<https://www.vmware.com/go/getworkstation-win>

图 1 VMware Workstation Pro下载页面

* 图形用户界面, 应用程序

  描述已自动生成Ubuntu操作系统的最新版Ubuntu 23.04下载地址为：<https://releases.ubuntu.com/23.04/ubuntu-23.04-desktop-amd64.iso>

图 2 Ubuntu 23.04的下载页面

* 电脑萤幕画面

  描述已自动生成在VMware Workstation Pro中安装Ubuntu操作系统，注意分配的硬盘空间要大一些，这里选择大小为60GB。

图 3 Ubuntu在虚拟机中安装成功后的显示

下载内核源码并解压

* 表格

  描述已自动生成到Linux内核官网下载Linux内核最新稳定版6.5.5：<https://cdn.kernel.org/pub/linux/kernel/v6.x/linux-6.5.5.tar.xz>

图 4 Linux内核下载页面

* 将其复制到虚拟机中，输入如下命令进行解压：

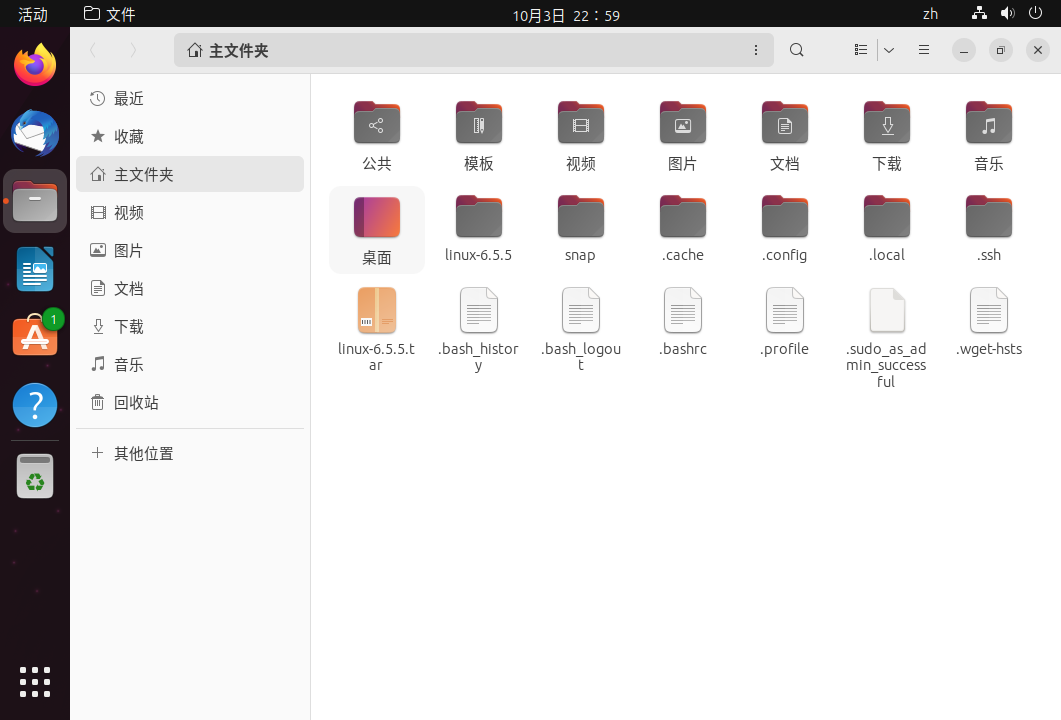
tar -xavf linux-6.5.5.tar.xz

图 5 解压后的内核源码目录

安装编译内核所需要的工具

sudo apt install update # 更新安装源

sudo apt install make

sudo apt install gcc g++

sudo apt install flex

sudo apt install bison

sudo apt install libncurses-dev

sudo apt install build-essential

sudo apt install libssl-dev

sudo apt install libc6-dev

sudo apt install libelf-dev

sudo apt install openssl

配置内核编译选项

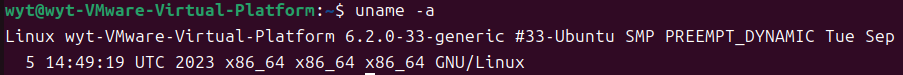
* 输入uname -a查看Linux内核版本，可以看到更换内核之前的版本为6.2.0

图 6 查看更换内核之前的内核版本

* 进入内核代码解压后的目录

cd ~/linux-6.5.5

* 清理之前的编译结果，如果之前没有编译过，则不需要这步操作

sudo make mrproper

* 复制本机的配置文件

cp /boot/config-`uname -r` ./.config

* 进入内核配置菜单，注意要放大屏幕，否则无法打开配置菜单

sudo make menuconfig

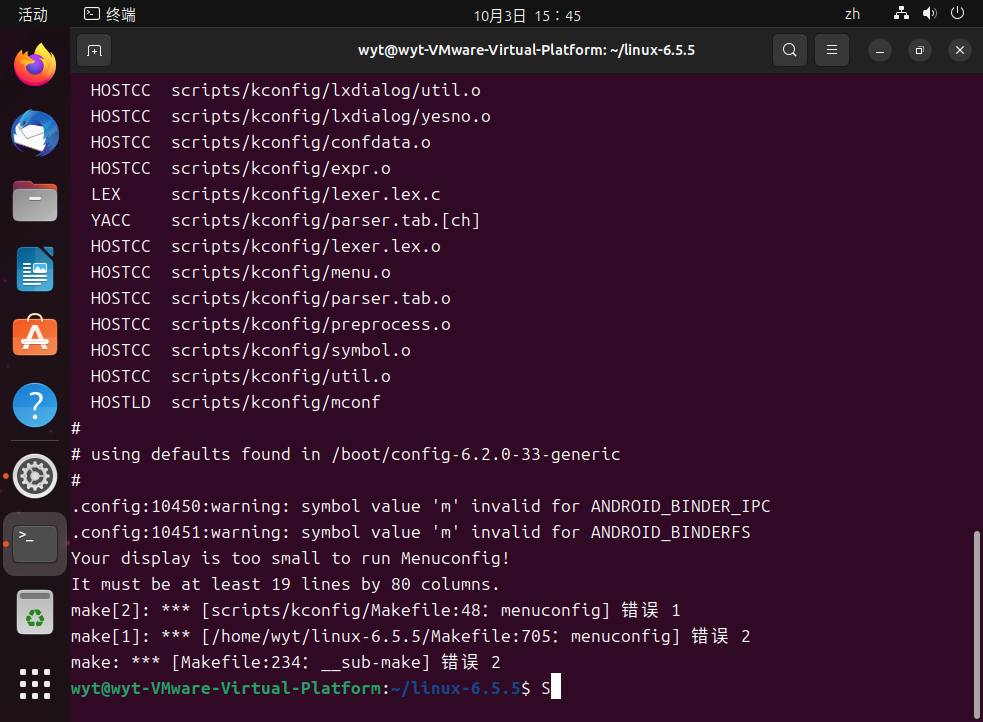
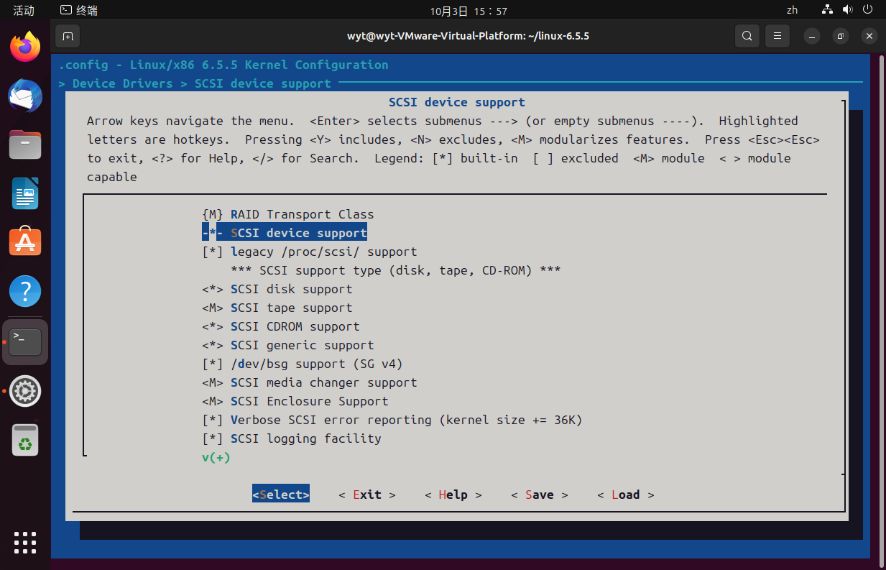


图 8 配置菜单

图 7 没有放大屏幕会报错，无法启动配置菜单

**操作方法如下：**

* **Esc退出**
* **空格改变选项状态（M是编译成运行时加入内核，\*表示编译进内核，空表示不加入内核）**

**由于编译选项过于繁杂，为了简化配置过程，可以使用原有内核配置，所以这里不需要更改配置，直接选择save，然后保存本次配置，最后选择exit退出配置界面**

编译内核

* 修改.config文件

**将CONFIG\_SYSTEM\_TRUSTED\_KEYS="debian/canonical-certs.pem"更改为CONFIG\_SYSTEM\_TRUSTED\_KEYS=""**

**将CONFIG\_SYSTEM\_REVOCATION\_KEYS="debian/canonical-revoked-certs.pem"更改为CONFIG\_SYSTEM\_REVOCATION\_KEYS=""**

* 编译代码

sudo make -j4 # -j4表示使用4个线程进行编译

以上命令等价于：

sudo make bzImage -j4

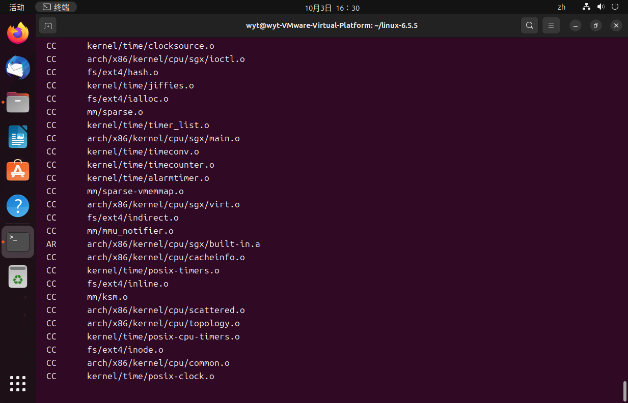
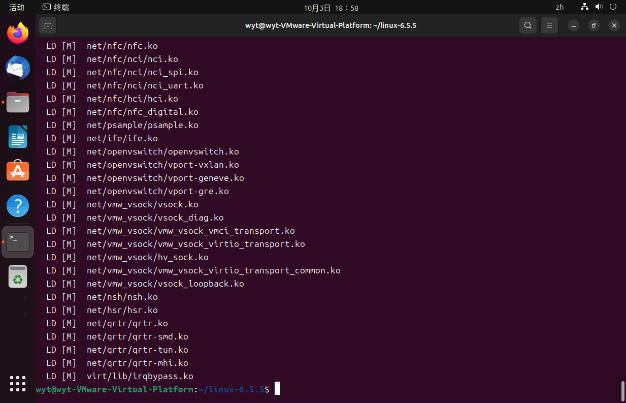
sudo make modules -j4

图 9 编译开始和结束

* 模块安装

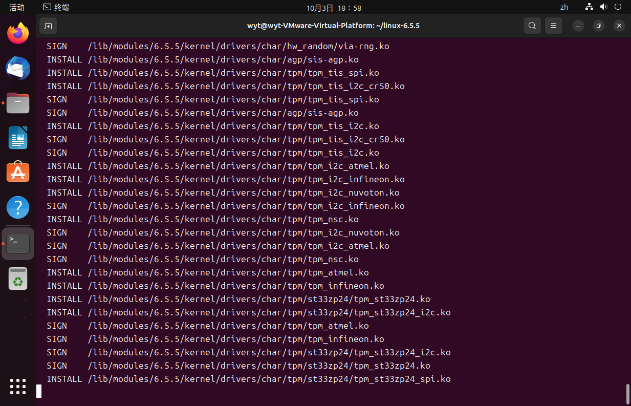
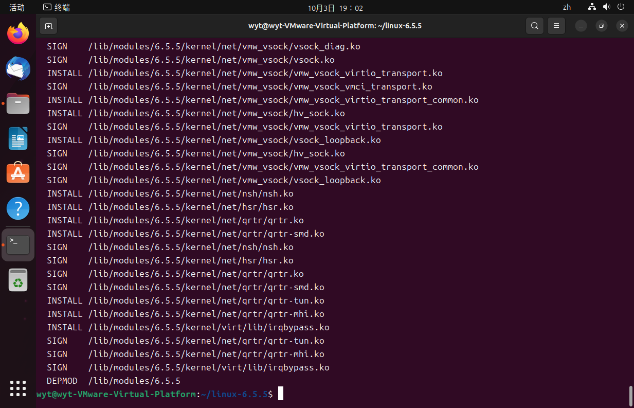
sudo make modules\_install -j4

图 10 模块安装的开始和结束

* 安装编译后的内核

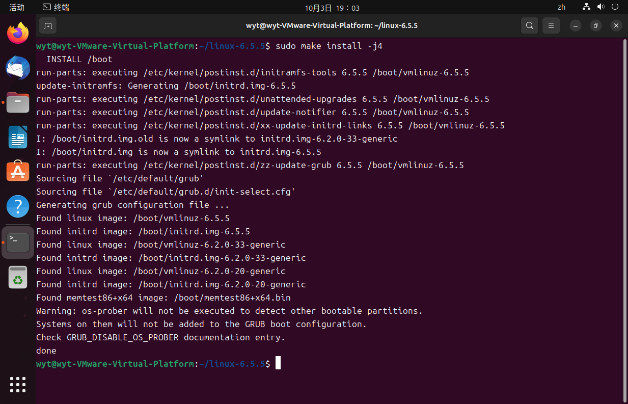
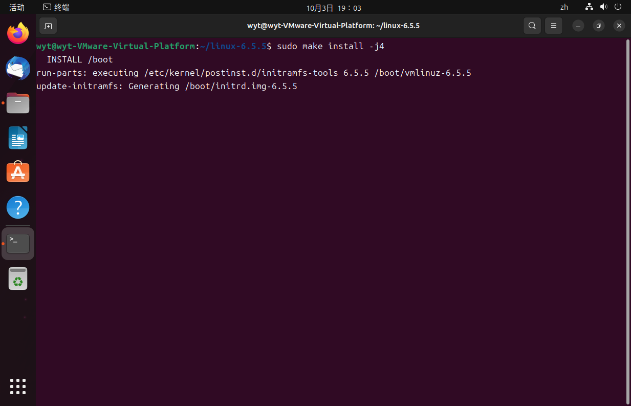
sudo make install -j4

图 11 安装内核的开始和结束

* 更新grub引导程序

sudo update-grub2

* 重新启动

sudo reboot

* 查看更新后的内核版本，确认更新内核版本成功

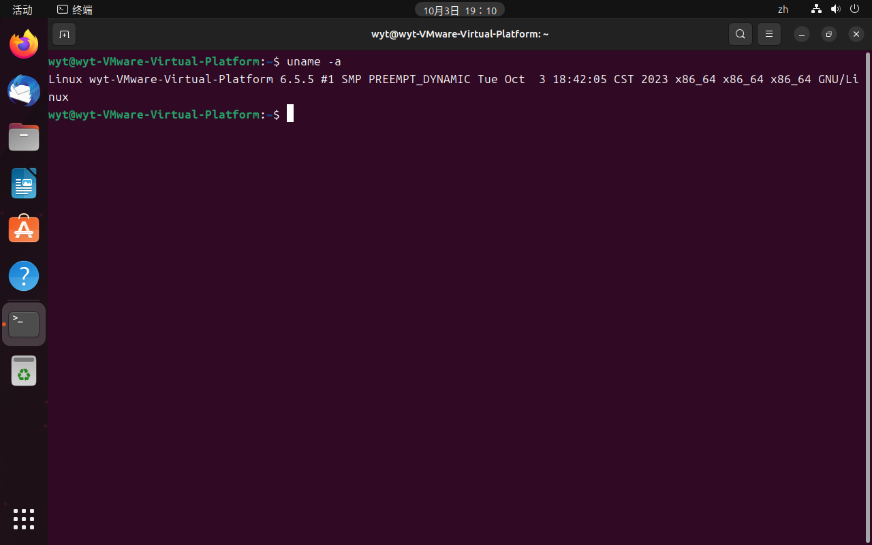
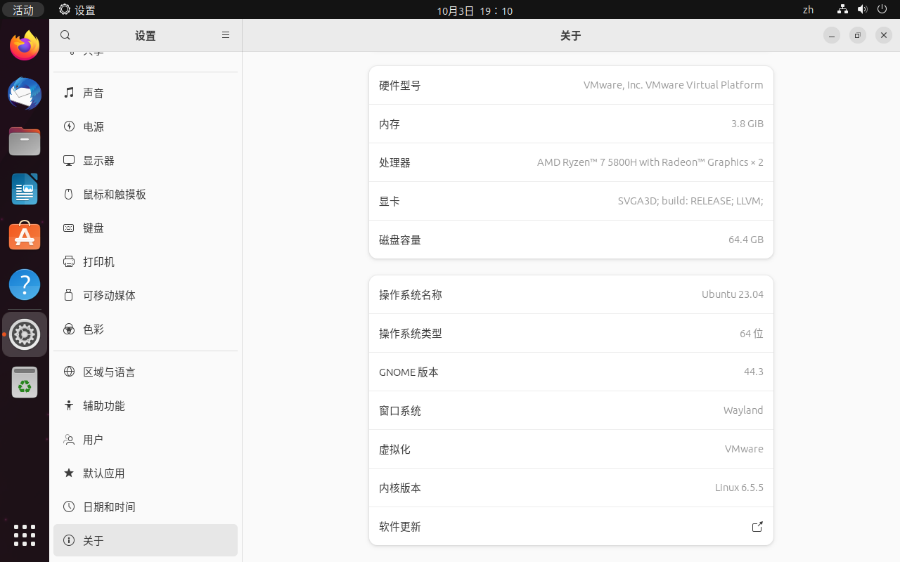
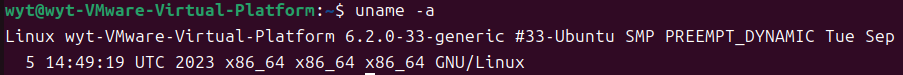
uname -a

图 12 查看更新后的内核版本号

四、实验结果及分析

* 文本

  描述已自动生成图形用户界面, 应用程序

  描述已自动生成更换内核版本前后对比

图 14 更换后的内核版本为6.5.5

图 13 更换前的内核版本为6.2.0

* 图形用户界面, 文本, 应用程序, 电子邮件

  描述已自动生成不同过程花费时间：具体见实验过程截图

图 15 不同过程花费时间

* 图形用户界面, 应用程序

  描述已自动生成图形用户界面, 应用程序

  描述已自动生成编译前后目录大小

图 17 编译后目录大小为22.8GB

图 16 编译前目录大小为1.4GB

五、实验收获与体会

通过本次实验，我掌握了Linux内核的编译过程，对操作系统内核有了大概认识。同时学习了一些实际的技能和经验，如如何配置交叉编译工具链、如何分析和处理编译过程中的问题和警告等，提高了自己的技能和能力。并且提高了自己的独立思考和解决问题的能力，在面对问题时能够更快地找到解决方案。总之，本次实验对我未来的学习和工作有一定的帮助。

Linux内核和发行版本的区别如下：

* Linux内核：这是系统的心脏，是Linux中最基层的代码。它提供了硬件抽象层、硬盘及文件系统控制以及多任务功能。内核是Linux的基本组成部分，安装完后，用户界面和软件都没有。
* Linux发行版本：这是在Linux内核的基础上，加入了用户界面和各种软件的支持后形成的版本。例如，Ubuntu、小红帽等都是Linux的发行版本。这些发行版本在内核的基础上，开发了不同的应用程序，组成了一个完整的操作系统。Linux发行版通常包含了桌面环境、办公包、媒体播放器、数据库等应用软件。这些操作系统通常使用Linux、以及来自GNU计划的软件和基于X Window的图形界面组成。此外，Linux发行版可以分为商业发行版和社区发行版。