# Ubuntu内核编译

## 前言

笔者最近在进行操作系统课设，发现网上教程和自己实际实施存在出入。自己根据自己实际情况，做出调整，并记录下来，方便自己日后查看，也希望能够帮到他人。

## 前期准备

我的环境：

* Ubuntu 18.04
* Linux内核：linux-5.12.1

## **安装VMwareTools**

tar -zxvf [VMwareTools的tar.gz文件]

cd [解压后的目录]

sudo ./vmware-install.pl

## **上网问题**

[如何让虚拟机的Ubuntu上网？](https://link.zhihu.com/?target=https%3A//blog.csdn.net/czg13548930186/article/details/79789001)

## **更改镜像源**

[Ubuntu更改镜像源两种方式](https://link.zhihu.com/?target=https%3A//blog.csdn.net/laoluobo76/article/details/108302191%23%3A~%3Atext%3D%25E4%25B8%2580%25E3%2580%2581%25E5%258F%25AF%25E8%25A7%2586%25E5%258C%2596%25E6%2596%25B9%25E5%25BC%258F%25E6%259B%25B4%25E6%2594%25B9%25E9%2595%259C%25E5%2583%258F%25E6%25BA%2590%25201%2520%25E6%2589%2593%25E5%25BC%2580%25E2%2580%259C%25E8%25BD%25AF%25E4%25BB%25B6%25E5%2592%258C%25E6%259B%25B4%25E6%2596%25B0%25E2%2580%259D%25E8%25AE%25BE%25E5%25A4%2587%25202%2520%25E9%2580%2589%25E6%258B%25A9%2520%25E2%2580%259CUbuntu%25E8%25BD%25AF%25E4%25BB%25B6%2520%252F%2520%25E6%25BA%2590%25E4%25BB%25A3%25E7%25A0%2581%2C%2528%25E9%259C%2580%25E9%25AA%258C%25E8%25AF%2581%25EF%25BC%258C%25E8%25BE%2593%25E5%2585%25A5%25E5%25AF%2586%25E7%25A0%2581%2529%2520%252F%2520%25E5%2585%25B6%25E5%25AE%2583%25E7%25AB%2599%25E7%2582%25B9%2520%252F%2520%25E9%2580%2589%25E6%258B%25A9%25E6%259C%2580%25E4%25BD%25B3%25E6%259C%258D%25E5%258A%25A1%25E5%2599%25A8%2520%25E2%2580%259D%25EF%25BC%258C%25E8%2587%25AA%25E5%258A%25A8%25E9%2580%2589%25E6%258B%25A9%25E5%2590%2588%25E9%2580%2582%25E7%259A%2584%25E9%2595%259C%25E5%2583%258F%25E6%25BA%2590%25203%2520%25E9%2587%258D%25E6%2596%25B0%25E8%25BD%25BD%25E5%2585%25A5%25E5%258F%25AF%25E7%2594%25A8%25E8%25BD%25AF%25E4%25BB%25B6%25E5%2588%2597%25E8%25A1%25A8%25EF%25BC%258C%25E5%25AE%258C%25E6%2588%2590%25E6%259B%25B4%25E6%2596%25B0)

## **更新软件源**

sudo apt update

## **更新软件**

sudo apt upgrade

## **编译内核所要依赖**

sudo apt install libncurses5-dev openssl libssl-dev build-essential pkg-config libc6-dev bison flex libelf-dev zlibc minizip libidn11-dev libidn11 qttools5-dev liblz4-tool

## **配置ccache**

提高后续编译的速度

sudo apt install ccache

sudo gedit ~/.bashrc

在末尾回车，添加如下语句，注意ubuntu要改成你的用户名。

export USE\_CCACHE=1

export CCACHE\_DIR="/home/ubuntu/.ccache"

export CC="ccache gcc"

export CXX="ccache g++"

export PATH="$PATH:/usr/lib/ccache"

按Ctrl+S或点击“保存”按钮保存，然后使其生效。

source ~/.bashrc

注:50G为需求，自行修改为自己的需求

ccache -M 50G

### **额外**

我也不知道为啥我的ccache不生效，通过查找官方文档，再次配置了以下内容才可以

which ccache # 找到ccache的位置

cd /usr/bin # 我的对应ccache路径

sudo cp ccache /usr/local/bin/

# 建立软链接

sudo ln -s ccache /usr/local/bin/gcc

sudo ln -s ccache /usr/local/bin/g++

sudo ln -s ccache /usr/local/bin/cc

sudo ln -s ccache /usr/local/bin/c++

ccache -s # 查看当前的一个状况

## 解压内核压缩包

**进入根目录下的图形界面**

sudo nautilus /

**压缩包移动到usr/src下并解压**

sudo tar -xavf linux-5.12.1.tar.xz

**进入解压后的包内**

cd linux-5.12.1

**净化内核源码**

sudo make mrproper

## 修改配置文件

根据需求更改配置

sudo make menuconfig

## **可选1**

笔者在后续编译过程中出现错误信息：  
make[1]: \*\*\* 没有规则可制作目标“debian/canonical-certs.pem”，由“certs/x509\_certificate\_list” 需求。 停止。  
下述修改就是针对这种情况

sudo nautilus /

Ctrl + H：显示隐藏文件

找到/usr/src/linux-5.12.1目录再修改.config文件【可以用Ctrl+F快捷键搜索定位到具体的行】

删除选定内容【标红部分】

## **可选2**

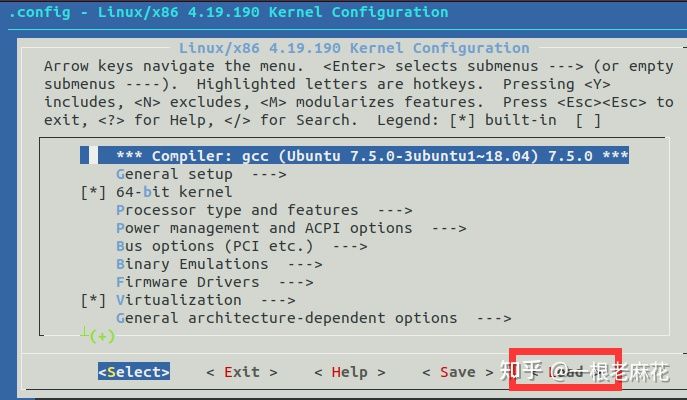
因为课上老师说最好用相近的内核版本，且用原有的配置文件，故我搜索相关资料再重新配置了一遍。【具体原因，这里不细表】

sudo su # 进入root模式

cp /boot/config-`uname -r` ./.config # 把原有的.config文件复制过去

make menuconfig # 进一步配置，先加载后配置

# load -> (.config) OK-> 修改 -> SAVE -> (.config) OK -> EXIT



**【备注：这里配置后，我又出现了一些问题，具体原因暂未查明】**

## 编译主体

sudo make bzImage -j8

## **编译模块**

sudo make modules -j8

## **安装模块**

sudo make modules\_install -j8

## **安装**

sudo make install -j8

## 扫尾

sudo mkinitramfs -o /boot/initrd.img-5.12.1

sudo cp arch/x86/boot/bzImage /boot/vmlinuz-5.12.1

sudo cp System.map /boot/System.map-5.12.1

**更新grub**

sudo update-grub2

**重启验证**

sudo reboot

## 参考文献：

[make1: 没有规则可制作目标“debian/canonical-certs.pem”，由“certs/x509\_certificate\_list” 需求。 停止。](https://link.zhihu.com/?target=https%3A//blog.csdn.net/Chenciyuan_nj/article/details/115099040)

[Ubuntu20.04编译内核教程\_Gay夜的ubuntu的博客-CSDN博客](https://link.zhihu.com/?target=https%3A//blog.csdn.net/qq_39819990/article/details/106605430)

[Ubuntu 18.04环境下编译Linux内核以及添加系统调用](https://link.zhihu.com/?target=https%3A//blog.csdn.net/qq_41037945/article/details/91352418)

# 安装所需的软件

执行make menuconfig需要前置安装libncurses-dev flex bison

安装过程中提示缺少#include <openssl/opensslv.h>，使用apt-file search后得知需要安装libssl-dev

缺少#include <gelf.h>，需要安装libelf-dev