

# 1. 流程

1. 下载内核和补丁  
rt补丁下载  
载 <https://www.kernel.org/pub/linux/kernel/projects/rt/>  
稳定版补丁下载  
[https://wiki.linuxfoundation.org/realtime/preempt\\_rt\\_versions](https://wiki.linuxfoundation.org/realtime/preempt_rt_versions)  
内核源码下载  
<https://mirrors.edge.kernel.org/pub/linux/kernel/>

## Note

对于某些内核版本找不到的场合，寻找相近的内核源码就可以，但是要注意到RT补丁与所下载的内核源码版本号需要严格对应。下载好后，将补丁包与解压后的内核源码文件夹放在同一级目录下。

## 2. 安转依赖

SHELL

```
sudo apt-get install libncurses5-dev libssl-dev  
build-essential openssl zlibc libelf-dev minizip  
libidn11-dev libidn11 bison flex dwarves  
libncurses-dev zstd -y
```

3. 开始安转
  - a. 解压内核源码  
`tar -zxvf linux-xxxxxx.tar.gz`
  - b. 解压补丁  
`gunzip patch-xxxx.patch.gz`

- c. 打补丁  
cd linux-xxx  
patch -p1 < ../patch-xxxx.patch
- d. 安装编译内核需要的模块  
sudo apt-get install libncurses-dev  
sudo apt-get install libssl-dev
- e. 配置内存
  - i. cp -v /boot/config-\$(uname -r) .config
  - ii. 这一步是根据当前系统内核的配置生成配置文件，我们一路enter即可  
make localmodconfig
  - iii. 进入一个图形界面，有一些地方需要进行设置  
make menuconfig
    - 1. 将内核设置为全抢占式的  
General setup ->Preemption Model->Fully Preemptible Kernel(RT)
    - 2. 关闭内存溢出检测(找不到就忽略此步)  
Kernel hacking —> Memory Debugging —> []Check for stack overflows
    - 3. 关闭Optimize very unlikely/likely branches(找不到就忽略此步)  
General setup —> Optimize very unlikely/likely branches 子选项Stack Protector buffer overflow detection设置为None，因为编译器不支持更强的堆栈保护
    - 4. Device Drivers ->[] staging drivers(如果默认开启，按N键取消)(可选择跳过)
- f. 修改当前目录下配置文件./.config
  - i. CONFIG\_MODULE\_SIG\_KEY="certs/signing\_key.pem"
  - ii. CONFIG\_SYSTEM\_TRUSTED\_KEYS=""
  - iii. CONFIG\_SYSTEM\_BLACKLIST\_HASH\_LIST=""

iv. (CONFIG\_SYSTEM\_REVOCATION\_KEYS=""

### Example

一个ubuntu20.04可用的配置文件  
[example](#)

g. 开始编译

i. (sudo make -j核心数

ii. (sudo make modules

iii. (sudo make bzImage

iv. (sudo make modules\_install -j核心数

v. (sudo make install

vi. (sudo update-grub

vii. (sudo reboot

h. (在引导界面选择刚搞定的内核即可

## 2. 参考博客

## 3. 脚本

SHELL

```
#!/bin/bash
```

```
sudo apt-get install libncurses5-dev libssl-dev  
build-essential openssl zlibc libelf-dev minizip  
libidn11-dev libidn11 bison flex dwarves -y
```

```
sudo wget
```

```
https://cdn.kernel.org/pub/linux/kernel/projects/rt/  
5.15/patch-5.15.49-rt47.patch.gz
```

```
sudo wget
```

```
https://mirror.tuna.tsinghua.edu.cn/kernel/v5.x/linu  
x-5.15.49.tar.gz
```

```
sudo tar -zxvf linux-5.15.49.tar.gz
```

```
sudo gunzip patch-5.15.49-rt47.patch.gz
```

```
cd linux-5.15.49
```

```
sudo patch -p1 < ../patch-5.15.49-rt47.patch
```

```
sudo wget https://github.com/xiao-xiao-jiang/linu-  
ubuntu-kernel-RT/blob/master/.config
```

```
sudo make -j$(nproc)
```

```
sudo make modules -j$(nproc)
```

```
sudo make bzImage -j$(nproc)
```

```
sudo make modules_install -j$(nproc)
```

```
sudo make install
```

```
sudo update-grub
```