有一些处理器设计得特殊导致不能安装某些类型的操作系统（操作系统背锅），比如因为寄存器个数或者类型，能够运行的操作系统的性能估计因为通用而不能发挥最大

Gentoo是一个基于Linux的自由操作系统，它能为几乎任何应用程序或需求自动地作出优化和定制。追求极限的配置、性能，以及顶尖的用户和开发者社区，都是Gentoo体验的标志特点。 Gentoo的哲学是自由和选择。得益于一种称为Portage的技术，Gentoo能成为理想的安全服务器、开发工作站、专业桌面、游戏系统、嵌入式解决方案或者别的东西——你想让它成为什么，它就可以成为什么。

linux渗透测试平台：Kali、Parrot、Black Box、Blackbuntu、Fedaro Security Spin......

google for Linux PenetrationTesting Distributions

Canonical 公司引导 Ubuntu的开发，RedHat是一家上市公司，如果没有公司的话真的，这个庞大的社区进步会越来越缓慢

deepin的桌面环境在国外也很受到欢迎，在国内的话除了桌面环境这个原因还有商城的方面。

biolinux 就是 ubuntu。 配套安装了一系列生物信息学的软件而已。

GNOME/GTK和 KDE/Qt 是Linux历史上最大的决裂，因为两个版本都各行其道，两边一开始又不相互兼容，反正企业在里面起到了绝对的作用，之后为了Linux更好的生态环境，出现了桌面环境库，比如只要安装了GTK就能使用gnome程序。

能够友好安装QQ的，除了上次见到有开发者自己做好了一个轻松安装的包以外，最特殊的就是AppImage格式的WineQQ，就是deepin、arch系列。从deepin15.6开始已经不使用crossover，将应用全部移植到deepin-wine。

在安装系统的时候可以不设置交换分区，但是你要保证你的内存是足够的，不然要卡。

Linux Mint 和 Ubuntu 是两个非常流行的 Linux 发行版的桌面操作系统

Xmanager是一款小巧、便捷的浏览远端X窗口系统的工具，可以在本地PC上同时运行Unix/Linux和Windows图形应用程序。Xmanager 是全新标准的跨平台集成解决方案。它是一个一站式解决方案，这个软件包含有以下一些产品：Xmanager 3D(OpenGL)，Xshell，Xftp和Xlpd。Xftp是一个用于MS Windows平台的强大的FTP和SFTP文件传输程序。Xlpd是一个用于MS Windows平台的LPD(行式打印机虚拟后台程序)应用程序。安装了Xlpd后，带有打印机的本地PC就成为了一个打印服务器，来自不同远程系统的打印任务都能在网络环境中得到请求和处理。(共享软件)。Xstart是一个窗口化的登陆界面，可进行远程登陆。

X11也叫做X Window系统，X Window系统 (X11或X)是一种 位图 显示的 视窗系统 。图形用户界面 的标准工具包和协议，并可用于几乎所有已有的现代操作系统。所有的窗口管理器和桌面环境都是运行在x window之上的。KDE是和一个叫KWM的窗口管理器一起的。 GNOME则没和任何窗口管理器做在一起，你可以使用任何你想用的窗口管理器，虽然有一些是特地为GNOME写的(Enlightenment就是一 个)。但是它们都需要X来运行。

STOP的进程不能被kill

ps -aux |grep vim

root 2720 0.0 0.4 45472 4752 pts/0 T 23:49 0:00 vim PE\_scan.php PE\_scan.php

make install 比make快多了。

Termux 一款Android终端模拟器。分屏工具tmux。

gnome-tweak-tool可以改主题，记得主题需要移动到/usr/share/themes。Unity下面是Unity-tweal-tool。

文件类型：

普通文件：“-”

目录文件：“d”，类似windows的文件夹

设备文件：所有设备均为文件，如打印机。块文件“b”，字符文件“c”

链接文件：“l”

组：

/etc/group

组账户信息。

/etc/gshadow

安全组账户信息。

/etc/login.defs

Shadow 密码套件配置。

/etc/fstab文本文件包含了你的电脑上的存储设备及其文件系统的信息。它是决定一个硬盘（分区）被怎样使用或者说整合到整个系统中的唯一文件。

/etc/hosts 设置ipv4、6的回环地址

/etc/shadow

/etc/passwd

/etc/hostname 查看主机名

/etc/issue 分支版本与版本号

/etc/resolv.conf #没准可以查看到内部的DNS服务器

/etc/profile这个文件是每个用户登录时都会运行的环境变量设置

/etc/bash.bashrc bash的配置文件

/etc/inetd.conf 内部配置处理网络请求的服务进程(从前的finger就是在这里面)

/etc/rc\*.d 和/etc/init.d 启动脚本

/etc/smb.conf Samba的配置文件

/etc/openldap/ldap.conf ldap配置文件

流stderr,stdin,stdout都是在/dev下面

/dev/tcp/x.x.x.x

/dev/urandom cat之后会不停产生随机数据

dd if=/dev/urandom of=random.dat bs=1M count=512

/etc/syslog.conf(老版本)-》/etc/rsyslog.conf 记录大部分各级别各功能日志的位置。

/var/log/wtmp wtmp日志文件是二进制格式的，给who命令读取。联想到apk编译以后AndroidMainfest.xml也变成二进制的了。

inetd，也叫作“超级服务器”，就是监视一些网络请求的守护进程，其根据网络请求来调用相应的服务进程来处理连接请求。inetd.conf则是inetd的配置文件

/dev/net/tun 文件节点只是内核提供给用户的接口而已（字符设备作为用户空间和内核空间交换数据的接口）。[tun/tap](https://www.baidu.com/s?wd=tun/tap&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao" \t "/home/vega/Documents\\x/_blank) 驱动程序实现了[虚拟网卡](https://www.baidu.com/s?wd=%E8%99%9A%E6%8B%9F%E7%BD%91%E5%8D%A1&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao" \t "/home/vega/Documents\\x/_blank)的功能，tun表示虚拟的是点对点设备，tap表示虚拟的是以太网设备。详细介绍：https://segmentfault.com/a/1190000009249039 。具体作用就是按照当前的路由表，将部分包通过虚拟网卡移交给用户层的某一个程序处理后再发送（数据包拦截），发送则是将包处理以后通过虚拟网卡放入协议栈，这种方式放入协议栈具体的发包交给了系统处理，如果使用socket自己发包则需要对不同协议处理。协议栈从哪个网卡发出去是看路由的，iptables里面相关规则有PREROUTE、POSTROUTE用于在route前与route后过滤。真正通信的时候还是要借助协议栈里面的socket实现。openvpn基于此驱动实现，拦截了所有的包然后加密/重新打包再重新发送。

tun0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST> mtu 1400

inet 10.0.0.2 netmask 255.255.255.255 destination 10.0.0.1

inet6 fe80::a8c8:c925:1de9:6b03 prefixlen 64 scopeid 0x20<link>

unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00 txqueuelen 500 (UNSPEC)

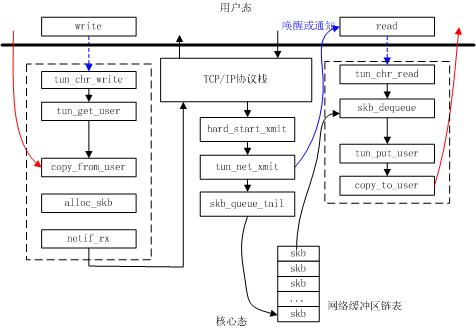
RX packets 264 bytes 103448 (101.0 KiB)

RX errors 0 dropped 0 overruns 0 frame 0

TX packets 322 bytes 58217 (56.8 KiB)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

如果tun0就是创建的tun虚拟网卡，inet为10.0.0.2，dest为10.0.0.1。只要看到发送到10.0.0.1，都会选择走这个网卡，并设置原地址为10.0.0.2 。



Linux 将设备当做文件进行处理,下面展示了如何打开设备文件 /dev/input/event5 并读取文件内容。文件 event5 表示一种输入设备，其可能是鼠标或键盘等。查看文件 /proc/bus/input/devices 可知 event5 对应设备的类型。设备文件 /dev/input/event5 使用 read() 以字符流的方式被读取。结构体 input\_event 被定义在内核头文件 linux/input.h 中。

##### 清单 2. 打开并读取设备文件

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | int fd;  struct input\_event ie;  fd = open("/dev/input/event5", O\_RDONLY);  read(fd, &ie, sizeof(struct input\_event));  printf("type = %d  code = %d  value = %d\n",              ie.type, ie.code, ie.value);  close(fd); |

在 Linux 中，元数据中的 inode 号 才是区分文件的唯一标识。

硬链接创建相同inode，但是不同文件名的节点。只能引用同一个文件系统的文件。

软链接与硬链接不同，若文件用户数据块中存放的内容是另一文件的路径名的指向，则该文件就是软连接。软链接就是一个普通文件，只是数据块内容有点特殊。软链接有着自己的 inode 号以及用户数据块。

我从gedit打开的文本中复制了一些东西，然后通过C-C直接在终端结束了gedit，结果发现剪切板里面什么字也没了。

mkdir新建目录的默认权限：

drwxr-xr-x 2 battery battery 4096 4月 15 14:00 test

battery@battery-pc:~/Desktop/test$ ls -l #在新建目录中touch建立一个文件的权限

-rw-r--r-- 1 battery battery 0 4月 15 14:01 test

同时明白了目录中的权限不会被文件中的权限继承

-rw-r--r-- 也是/etc/passwd的默认权限

想录像一段放在网页上，本来是想把录制出来的mp4转为gif，先用Kazam来录制视频得到mp4格式，而之后转化的方式是

ffmpeg -t 3.6 -ss 00:00:01 -i Screencast\_2016-01-13-17-01-38.mp4 small-clip.gif或者用ffmpeg从视频中截图成jpg然后在用工具合并成gif。然后出了问题，可能是我录制的视频有点长？反正把jpg合并成为gif的程序直接自己挂掉了

最后mp4录像没有转gif，而是用HTML5 video的标签，结合格式工厂转换得到的ogg和webm完成了

硬件-》内核-》系统调用-》用户层

内核头文件的作用主要有两个：1）定义内核组件间的接口2）定义内核与用户空间的接口

Glibc与内核是整个系统的根基，内核头文件的话kali就没有。不过编译glibc需要内核头文件。头文件似乎也就编译的时候用到，如果是deb或者apt安装的话没有必要。

linux的行为有些时候看上去很奇特，root@kali:~/test# rm -r ../test。删除了以后用户路径还是在test文件里面，不过已经什么都不能操作了。

名字以.开头的文件会被隐藏，Unix类系统。

/bin/sh (已经被 /bin/bash 所取代)

/bin/bash (就是 Linux 预设的shell)

/bin/ksh (Kornshell 由 AT&T Bell lab. 发展出来的，兼容于 bash)

/bin/tcsh (整合C Shell ，提供更多的功能)

/bin/csh (已经被 /bin/tcsh 所取代)

/bin/zsh (基于 ksh ）

编码：统一码一般用UTF-8, 简体中文用GBK, 繁体中文用big5

linux 一般使用Plymouth作为开机关机动画渲染软件（开机动画是指引导之后，登录账号之前的部分）。

deepin系统中/etc/plymouth 配置了关机动画

主题、开机动画 可以从gnome-look.org上下载。

https://www.gnome-look.org/p/1185488/

https://www.gnome-look.org/p/1173365/

https://www.gnome-look.org/p/1143453/ 这篇文章中的安装命令：

sudo update-alternatives --install /usr/share/plymouth/themes/default.plymouth default.plymouth /usr/share/plymouth/themes/dandelion/dandelion.plymouth 100 会引起配置文件/usr/share/plymouth/themes/default.plymouth的改变，会将主题“安装”到配置文件中，到时候sudo update-alternatives --config default.plymouth进行选择

http://blog.csdn.net/zphj742/article/details/21178501 安装命令

Ubuntu可以使用plymouth-manager工具，deepin安装失败，因为路径不同，一个在/lib一个在/usr

dell xps 需要在bios中将SupportAissist处于OFF。

Grub2的图形化前端管理工具:Grub Customizer

deepin开关机的音乐在/usr/share/sounds/deepin/stereo，只要通过mv命令将指定的文件重命名就不会有声音了。

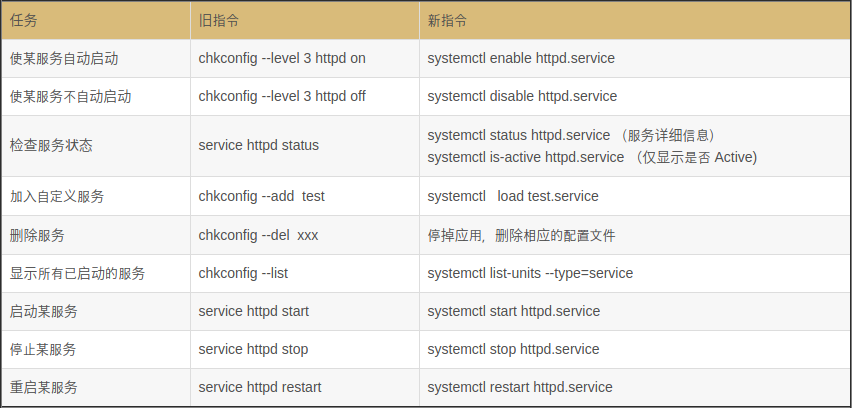
deepin已经有了自己版本的内核,如果安装了Ubuntu等内核版本中不是很匹配的内核就会导致开机开在启动动画(也正是加载内核的地方). deepin的内核多了几个patch。

PATH环境变量可以在/etc/profile中export，每一次开shell都会执行这个文件

开机启动：initd 被systemd取代

可以看到/etc/rc0.d中README说同目录脚本源自/etc/init.d

可以发现没有/etc/rc.local，如果要用就要配合systemctl去开启



开机自执行有这么几种形式：

1. 创建一个服务，然后systemctl enable一下，比如说nginx，apache这种服务器。本质上还是使用update-rc.d实现。
2. 添加执行命令到rc.local，比如你想要在开机的时候默认关闭蓝牙省电：rfkill block bluetooth
3. 添加desktop文件到~/.config/autostart或者/etc/xdg/autostart/。deepin在launcher中右键的“add to startup”会添加到~/.config/autostart，所以效果会只限于你这个用户，其他用户登录没有效果。如果要弄成全局的，就要放到/etc/xdg/autostart/，如果你安装了blueman，它的开机自启动就在这里：sudo mv /etc/xdg/autostart/blueman.desktop /etc/xdg/autostart/blueman.desktop.back 即可失效

注意：似乎如果有设计UI内容的东西，只有3会成功，比如开机自动启动conky

deepin在launcher中设置开机启动会创建xxx.desktop到~/.config/autostart/:

[Desktop Entry]

Name=Shadowsocks-Qt5

GenericName=Shadowsocks-Qt5

Comment=Shadowsocks GUI client

Exec=ss-qt5

Icon=shadowsocks-qt5

Terminal=false

Type=Application

Categories=Network;

StartupNotify=true

X-Deepin-CreatedBy=com.deepin.SessionManager

X-Deepin-AppID=shadowsocks-qt5

Hidden=false

还有一个目录：/etc/xdg/autostart/

[Desktop Entry]

Type=Application

Name=Conky

Comment=Start conky script

Exec=conky -d

chmod 4755与chmod 755 的区别在于开头多了一位，这个4表示其他用户执行文件时，具有与所有者相当的权限。可以解决/usr/bin/dumpcap的问题，而chgrp不能完全解决这个问题。代表文件所有者的权限，同组用户的权限，其他用户的权限。

将一个账号完整地提升到root权限：

**需求**：不想要再一个个打sudo，就像kali一样，但是不能直接使用root账号登录，因为root的家目录下没有我目前账号的配置文件。

**不完整**：如果将一个账号增加到root用户组（gid=0），还是会缺失不少权限。比如说执行/usr/bin/dumpcap。而将一个账号添加到root用户组，可以使用usermod命令，这会更改/etc/group，或者直接对/etc/passwd进行gid更改。

/etc/passwd中内容格式：用户名：密码：用户ID：主要组ID：COMMENT：主目录：登录shell

**完整**：如果想要完整的root权限就还需要对/etc/passwd中的uid更改改为0，不少程序都直接对uid进行了判断。但是像deepin这样使用lightdm登录管理器，就会导致开机的时候不能正常登录了。

sudo chgrp xxx xxx 执行之后会对权限重新设置，如果之前设置过chmod 4755 xxx就需要重新执行一遍。

UI桌面下的剪切比shell中的mv命令慢，估计因为剪切是复制+删除。

动态链接库：

相关命令：ldd、ldconfig（它的作用就是将/etc/ld.so.conf列出的路径下的库文件缓存到/etc/ld.so.cache 以供使用）

LINUX动态链接库存放的目录,它包含/lib,/usr/lib以及/etc/ld.so.conf文件内所列的一系列目录.

环境变量LD\_LIBRARY\_PATH 指定了搜索路径

或者编译代码时指定该程序的动态链接库搜索路径

字体渲染，一个是灰度渲染（Windows），一个是次像素渲染（Linux）。  
灰度渲染看起来会有锯齿，而且长时间阅读容易引起眼睛疲劳。次像素渲染过的字体看起来更平滑，但是分辨率不高的显示器上会发虚。你可以试试取消掉次像素渲染配置文件的软链接（/etc/fonts/conf.d或/etc/fontconfig/conf.d)。Deepin的字体应该是采用了RGB次像素渲染加上full hinting。

conky类似于windows下的水滴桌面，主要用于对资源的UI监控，可以加上天气插件。conky的窗口透明实现并不是真正透明，而是实时读取桌面背景，结合透明度的变化作为conky窗口的背景。

Apple File System 简称叫做APFS， 中文叫苹果文件格式。是一个很先进的文件系统。主要的改变就是将分区概念升级为容器。在容器里新建的卷宗不再受到分区大小的限制。并且采用了文件快照模式, 当你复制文件的时候,默认是创建一个快照，而不是复制一个文件。 期初只会对文件的修改部分进行存储。这些改进都是基于SSD的特性进行优化的。

Launcher Menu中的几种分类按照 <https://specifications.freedesktop.org/menu-spec/latest/apa.html> FreeDesktop制定的标准来分类，所以dde-launcher中直接将分类写死在代码中，而不是提供一个配置文件允许定制。

shell 中的\*，是shell解析，而不是作为一个参数被程序特殊处理，由以下代码验证：

#include <stdio.h>

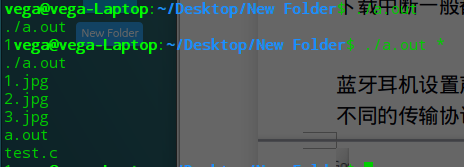
int main(int argc, char \*argv[]) {

for(int i1=0;i1<argc;++i1)

printf("%s\n",argv[i1]);

printf("1");

}

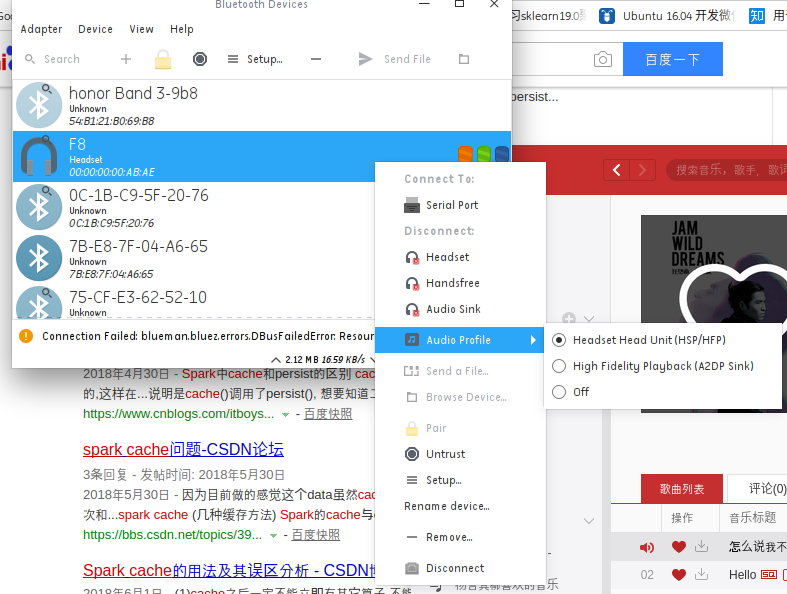


\*会直接被替换成一个个参数，然后仅仅执行一次命令

supervisor管理进程，是通过fork/exec的方式将这些被管理的进程当作supervisor的子进程来启动。

蓝牙耳机设置声音：

不同的传输协议会影响耳机能不能用



容器的文件系统都是用户态文件系统FUSE，因为容器与内核相比，这只是一个外部程序。传统文件系统完全实现在内核态，最终倒是内核维护上的麻烦，现在变成了一个内核的模块（fuse模块）+与内核通信的接口（libfuse）+用户态文件系统（fuse提供了挂载工具fusermount）。 fuse本质上（数据处理时）是处于现有文件系统之上的（具体实现是和现有文件系统处于同一个层次的），fuse不参与底层磁盘数据的存取，只负责处理对读取和写入的数据在逻辑上的操作而已。

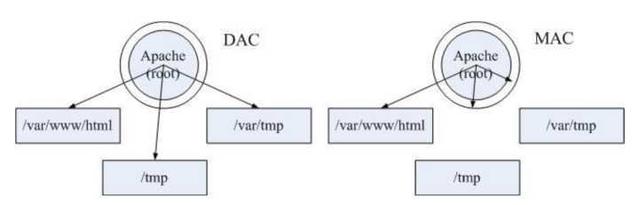
OJ平台后台使用沙箱，并且需要MAC强权限控制 来防止恶意代码。

Linux系统通过命名空间可以实现资源的隔离与共享，通过cgroup实现对资源性能上的控制。

period 进程的调度时间片

runtime 时间片内进程的实际执行时间

DAC是按照用户身份来管理权限，但是必须使用root运行的那些服务，一旦出现漏洞就是一件很爆炸的事情，所以更好的做法是MAC（Mandatory Access Control 强访问控制），除了DAC这样还需要判断进程是否对资源具有访问权限，即进程也有了权限角色。



SELinux是Linux系统的MAC，有三种工作模式，在 /etc/selinux/config 中设定，分别是：

1. enforcing：强制模式。违反 SELinux 规则的行为将被阻止并记录到日志中。

2. permissive：宽容模式。违反 SELinux 规则的行为只会记录到日志中。一般为调试用。

3. disabled：关闭 SELinux。

Apparmor类似于SELinux，也是增加了对进程的权限控制。它将先进的静态分析和基于学习的工具结合起来。

seccomp 是 secure computing 的缩写，是Linux的一种沙箱机制。在Linux系统里，大量的系统调用（systemcall）直接暴露给用户态程序。但是，并不是所有的系统调用都被需要，而且不安全的代码滥用系统调用会对系统造成安全威胁。通过seccomp，我们限制程序使用某些系统调用，这样可以减少系统的暴露面，同时是程序进入一种“安全”的状态。

旧版本Linux内核还不支持的功能，可以通过打patch，来增加（理论上所有的都能这么做，因为新版本总是基于旧版本代码）。如果能力不是很强，还是升级内核吧。

内核错误(Kernel panic)是指操作系统在监测到内部的致命错误，并无法安全处理此错误时采取的动作。

chroot 改变当前程序执行的根目录，一般用于守护进程的编写或者在linux usb系统中修复主系统，直接更改根目录从而方便使用主系统上的程序实现修复。

chroot对应C代码：

    chroot("."); //改变根目录的位置

    chdir("/"); //改变当前工作目录至根目录

C-C一个文件，粘贴在剪切板中是绝对路径。

linux的目录表示其实还有一个斜杠在后面。

Wine与 Crossover 的本质区别就是 Crossover 是商业版的Wine; Wine 是免费的，Crossover 是需要花钱买的。Crossover 能做到而 Wine不能的几个优势：一键安装Window 应用、图形安装程序、附有 Bottles（便携式虚拟Windows环境）、完美结合桌面环境、始终对受支持的应用程序进行测试。

国产出了一个xDroid <https://www.linzhuotech.com/> ，国防科大退役军人弄的，估计也是一款情怀产品。

Linux通过socket实现lock，从而防止程序被反复启动

Linux通过文件头识别文件，比如将图片后缀去掉，还是能够识别出来这是一张图片，但是去掉后缀就不能正常插入WPS文档。Linux而支持通过后缀来识别文件，从而做到一些格式的默认打卡软件。

linux的驱动程序分两个部分实现：top-half和bottom-half。top-half在运行时，不能被其他任何中断再次中断，也不能被其他进程中断，它通过对CPU内的中断屏蔽置位实现，而bottom-half则只对top-half开中断。

内核:

<https://github.com/MintCN/linux-insides-zh>

《深入理解linux内核》

RPM 和 DPKG 为最常见的两类软件**包管理工具**：

RPM 全称为 Redhat Package Manager，最早由 Red Hat 公司制定实施，随后被 GNU 开源操作系统接受并成为很多 Linux 系统 (RHEL) 的既定软件标准。

与 RPM 进行竞争的是基于 Debian 操作系统 (Ubuntu) 的 DEB 软件包管理工具 DPKG，全称为 Debian Package，功能方面与 RPM 相似。

YUM 基于 RPM，具有依赖管理功能，并具有软件升级的功能。

虚拟文件系统： cgroupfs、sysfs、proc

sysfs 被挂在/sys，是proc的改良版，设计上更加好。新设计的内核机制应该尽量使用 sysfs 机制，而将 proc 保留给纯净的“进程文件系统”。sysfs 给应用程序提供了统一访问设备的接口。https://www.ibm.com/developerworks/cn/linux/l-cn-sysfs/

如果你正在开发的设备驱动程序中需要与用户层的接口，一般可选的方法有：

注册虚拟的字符设备文件，以这个虚拟设备上的 read/write/ioctl 等接口与用户交互；但 read/write 一般只能做一件事情， ioctl 可以根据 cmd 参数做多个功能，但其缺点是很明显的： ioctl 接口无法直接在 Shell 脚本中使用，为了使用 ioctl 的功能，还必须编写配套的 C语言的虚拟设备操作程序， ioctl 的二进制数据接口也是造成大小端问题 (big endian与little endian)、32位/64位不可移植问题的根源；

注册 proc 接口，接受用户的 read/write/ioctl 操作；同样的，一个 proc 项通常使用其 read/write/ioctl 接口，它所存在的问题与上面的虚拟字符设备的的问题相似；

注册 sysfs 属性；

linux内核调度器有两种CFS、BFS，具体在操作系统.docx中介绍。