

讲述操作系统自己的故事.....

# 笨兔兔 的 故事

L<sup>A</sup>T<sub>E</sub>X版

懒蜗牛Gentoo 著

田劲锋 排版

2009 年 10 月 10 日

原帖:<http://forum.ubuntu.org.cn/viewtopic.php?f=112&t=162040>

出处:<http://forum.ubuntu.org.cn/>

---

# 目 录

---

1	笨兔兔正传	1
1.1	开端	1
1.2	醒来	1
1.3	工作	2
1.4	历史	2
1.5	也是历史	3
1.6	还是历史	4
1.7	串门	5
1.8	碎片	5
1.9	邻居	6
1.10	人才	6
1.11	来头	6
1.12	本事	7
1.13	开源	7
1.14	故事	8
1.15	Minix	9
1.16	Linux	9
1.17	Friends	10
1.18	杀毒	11
1.19	免疫	12
1.20	权利	13
1.21	内存	14
1.22	图像	14
1.23	信封	15
1.24	酒	16
1.25	酒 too	16
1.26	酒Again	17
1.27	Year	18
1.28	人物志	18
1.29	日志	19
1.30	XFS	20
1.31	分区	21
1.32	挂载	21
1.33	GIMP	22
1.34	QQ	22
1.35	运算	23
1.36	位	23
1.37	协作	24
1.38	加速	25
1.39	OpenGL	26
1.40	Power ON	26
1.41	Init	27
1.42	EXT4	27
1.43	有朋远来	28
1.44	他乡故知	28

1.45	可视电话	29
1.46	磁盘	30
1.47	硬解	30
1.48	刻录	31
1.49	Rubbish	32
1.50	Bug	33
2	笨兔兔外传	34
2.1	Oprea	34
2.2	笨兔	36
2.3	我有一个梦想	37
2.4	I Have A Dream	39

## 笨兔兔正传

### 1.1 开端

我叫Ubuntu，主人喜欢叫我“笨兔”，但是我绝对不笨，与某种耳朵长尾巴短的哺乳动物也没有什么联系，我是一个操作系统，我是一个Linux，我是Ubuntu。

在今年（2008）的4月，我来到了这个世界，并由出生日期得到了自己的代号——8.04。当然，和我同一天出生的兄弟们还有很多，我只是其中之一。我们出生之后，每个人坐上一张被人们叫做“光盘”的碟子，奔赴世界各地，寻找自己的归宿。我们只有找到住处，才可以发挥我们的能力，而我们并不像人们一样，住在钢筋水泥的格子里面，我们住的地方，是一块叫做硬盘的空间。

住房的质量严重影响着我工作的效率，不过还好，我住的这个地方还不错，房子很宽敞，有500G大。不过不是我一个系统住，房子被隔成两个大屋子，每边 250G，我住在右边的一间，隔壁住的是一个Windows XP，听说这家伙很厉害，不过我来的时候他正睡觉，于是也没有打招呼。（后来我知道，我们两个是不可能同时醒着的……）250G的屋子，对我来说实在是太宽敞了，我很欣慰，除了硬盘大之外，我发现这里其他的设施也是不错的。

硬盘只是我们操作系统休息和存放个人物品的地方，真正工作的时候，是不能在硬盘里的，那时候，我就需要从住的硬盘里，来到工作的地方，我们的工作间——内存。我的这个工作间也比较宽敞，有4G大，不过也不是我一个人用，也是要和隔壁的WindowsXP共用的。鉴于他名字太长，我就管他叫“查皮”吧，不管他愿不愿意了，反正他也不知道，呵呵。不过内存的共用方式与硬盘不一样，不是一人一半，而是谁醒着谁就用，去睡觉前，一定要记得把内存里的东西，有用的，都搬回自己的那间硬盘里，以备下次使用。这道不光是为了另外一个系统着想，主要是因为内存里很不保险，东西放在里面去睡觉的话，等你在醒来，准没，不管你隔壁有没有住着一个查皮。

### 1.2 醒来

自从安顿到这个地方，对周围的硬件环境熟悉了不少。我们在Canonical学校的时候就进行了充分的学习，所以这里的東西我基本上都会用的比较顺手。像 Realtek的网卡，Intel的北桥，南桥，声卡以及E8400 双核CPU，更是应用自如。Intel和我们的关系还是不错的，为我们提供了很多的教科书，说明书，基本都是讲如何使用他们的设备的，所以对于Intel 的设备，我们都使用的比较好。这里唯一不能完全用起来的可能就是 GeForce 8800 Gt的显卡了。不过基本的显示功能还是没有问题的，只是我不知道怎么使用它的3D加速功能。这个需要专门的手册，要去nVidia那里去要。对周围的一些熟悉以后，就等着我的第一次启动了。

平时，我和隔壁的查皮都是在睡觉。当主人有事情的时候，会让传达室的GRUB大叔来叫我们。G大叔就住在传达室，传达室很小，只有512Byte，门上贴着牌子——MBR，可能G大叔想说自己是个“明白人”吧。由于传达室地方实在太小，所以他会把一些有用的东西放在我的硬盘空间里，必要的时候来看看。他总是面无表情，每次主人来的时候，他就板着脸说：给你10秒钟，快说要叫哪个！然后就倒数，如果主人没来得及说，他就会直接来叫我——因为我们熟，他是我带来的。如果是叫查皮，那我就知道了，因为那时我肯定在睡觉，上次说过，我们两个是不可能同时醒着的。而这一次，他径直来到我这里，拍拍我说：嘿，小子，开工啦！

顺便说说：

其实，我的Ubuntu是从网上下的，实在等不了邮寄光盘那漫长的时间，还是把光盘留给更需要的人吧。这样写，只是想让故事更像故事一些。看到很多人谈这自己跟Ubuntu的各种故事，想换个角度，换个方法来说，或许，这样更有意思。

## 1.3 工作

一听说开工，我很麻利的蹦起来，以迅雷不及掩耳盗铃之势，嗖的一下就越进工作室——内存里，用最快的速度进入工作状态。主人对此很满意，夸我说比那查皮麻利不少。然后，他下达了第一个命令：先去上网看看，找个快一点的软件源。于是我赶快叫醒还在硬盘里睡觉的Firefox——是的，看网页这个事我做不来，就得去找Firefox，我喜欢叫她狐狸妹妹。

狐狸妹妹轻移玉步，走进工作间——速度有点慢，不过还可以接受。然后开始工作，一下子找到某菜鸟入门新手指导帖之类的，找到一些著名的软件源的列表，如 cn99之流。然后，主人决定记录下这些地址，于是，我又叫醒的gedit小弟。gedit个头很小，身体轻盈，一下子蹦进来，开始干活。然后，我又在主人的命令下去叫Rhythmbox，去叫pidgin，去叫……等等，有人问为什么你总是叫别人干活，自己不干？我正在干，我要干的就是——叫人。

是的，我号称叫做操作系统，听起来好大的一个软件阿，好像操作系统就应该是啥都能干。不过，其实我们作为操作系统，并不能直接完成任何你需要的任务。我需要很多帮手，他们各自帮我完成各种不同的任务。可以说，我们在一起，才算的上一个系统，而我，是核心，是领导。没有我，他们当然不知道该做什么，而没有他们，我也不知道该怎么做。我们操作系统的最基本的职责就是管理，管理各种程序的执行，管理硬件资源的使用。比如CPU，就是我们程序要用的重要设备，每个程序都要用，可是CPU很贵，不能发给每个程序一个（否则主人会破产）。狐狸妹妹来了，我会把cpu给她用，gedit小弟也来了，他也要用，那么我就水告诉他俩，一人用一会儿。但是这可不是没人1/2这么简单，狐狸妹妹要做的事情比较复杂，那么就让她多用一会，gedit的工作很简单，就让他少用一会。主人关心的程序，就得多用cpu，主人不是很关心的，就可以少用一点cpu。有的程序脾气不好，把着cpu就不放，我必须处理，有的程序确实工作量大，需要使用cpu相当长的时间，可是我也不能就真把cpu全都给他用，还是得让其他的每个程序隔一阵子都能用上一会，不至于一直闲着。而有的程序平时基本不需要做什么工作，可是有不能把他请回硬盘，那么我就要允许他在工作间睡觉，只是在必要的时候叫醒他，并且把cpu给他用……怎么样？是不是有点乱？当个操作系统是很不容易的。当个好的操作系统，就更不容易了。

## 1.4 历史

住了一段时间，慢慢的开始对隔壁的邻居感兴趣起来。借主人上网的时间，顺便让狐狸妹妹帮我找找相关的资料。狐狸妹妹虽然起床慢点，干活还是挺快的，特别是装备了fasterfox插件以后。

查皮是个挺有名的操作系统，也算得上是名门之后。早在1985年，稍微有点软的公司就造出了查皮的老祖宗——Windows 1.0 名字比较土，不如查皮显得青春活力。而实际，这位老人家的表现也确实不咋样，大家都没怎么拿正眼看他。我还找到了他的一张照片，怪难看的。



两年后，1987年12月9日，第二代Windows 上市了，那时候的人都懒得起名字，于是就叫Windows 2.0。还是那个有点软的公司，还是那张脸，跟他爹长的还真一样。

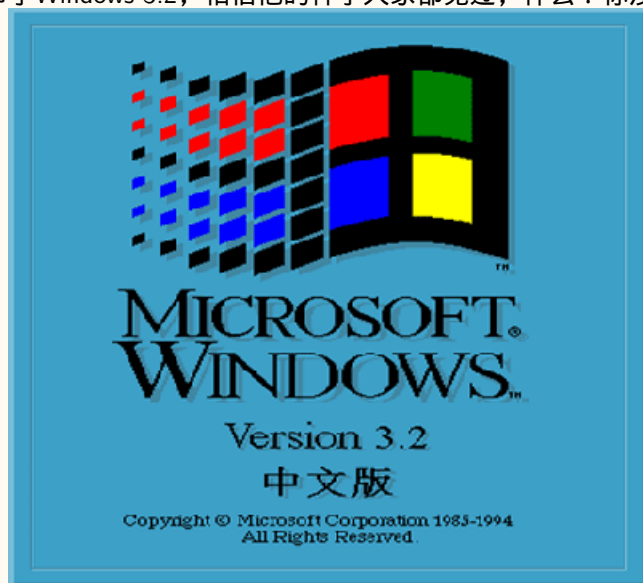


并且，不单脸长得一样，遭遇跟他爹也差不多，基本上被打入冷宫，并不被大家看好。而Windows 1.0和Windows 2.0最大的不一样的地方在于——2.0有个好儿子，1.0没有。

Windows 3.0终于让那个有点软的公司硬起腰板来了。相信很多老鸟也都是从Windows 3.0的一个重要分支版本Windows 3.2开始认识 Windows 的吧。1990年5月22日，Windows 3.0正是发布，而第二年，1991年发布了Windows 3.0的多国语言版本。而同年，1991年，一件更重大的事情发生了。

在1991年，芬兰，赫尔辛基大学的一名大学生的电脑上，我们的老祖先，Linux的雏形正在一点一点的完善起来……代码一行一行的流入他的身体，那时候，Windows 3.0已经广为人知，已经进入图形界面的时代，已经能够支持多种语言，而我们的老祖先还仅仅能够对世界说一句Helloworld！可十几年后之后，就是另一番景象了。

再回来说Windows 他们家吧。1992年Windows 3.1出生，算是3.0的改进版，他增加了基本的多媒体支持和TrueType字体。TrueType区别于点阵字体，可以放大缩小，看起来更好看。1994年又发布了Windows 3.2，相信他的样子大家都见过，什么？你没见过？好，贴张照片吧。



## 1.5 也是历史

到了1995年，Windows 家的祖坟上终于冒青烟了，Windows 95一下子把人们从DOS时代领进了窗口时代。出色的多媒体性能，人性化的操作，美观的界面（跟DOS，win3.2比）加上有点软公司强大的宣传攻势，那时的Windows 95简直是家喻户晓，妇孺皆知，老少皆宜，人人必备，真乃

居家旅行月黑风高杀人放火之必备良品。除了明显的，看得见的不同之处外，Windows 95与他的前辈们的一个重要区别是，他是一个16位 / 32位混合的操作系统。之前的那些都是16位的，也就是说，Windows 95标志着个人电脑32位软件时代的开始。(顺便说一下，我们现在基本上还生活在32位软件的时代，隔壁的查皮就是32位的，而我嘛.....是64位的^\_^)总之，无论从哪方面说，Windows 95都是成功的，那个“开始”按钮，留在了那个时代的历史中，并一直流传到了现在。

成功的东西要发扬广大，3年之后，Windows 98 问世了。这个系统是基于Windows 95编写的，他修正了Windows 95的近3000多个 bug<sup>1</sup>，添加了桌面主题等新的视觉特性，更重要的一——他捆绑了IE。有点软的公司终于意识到互联网的重要性，把IE作为基本的软件随系统一起安装。(其实Windows 95里面也有ie，只是放在一个不起眼的阴暗角落里，生怕人知道似的)第二年，1999年6月，Windows 98 SE发布，也就是98第二版，他提供了Internet Explorer 5、Windows Netmeeting 3、Internet Connection Sharing、对DVD-ROM和对USB的支持等，可以说，从Windows 95到Windows 98 SE这一脉，到这里达到了顶峰，也走到了尽头.....

顺便说一下，Windows 95一脉还有一个家伙——Windows ME。出生在2000年，不过，基本上可以忽略他的存在，他的一些激进性改动没能获得广大用户的认同。重要的修改是系统去除了实模式DOS，而由系统还原代替了。在概念上，这是一个大的改进：用户不再需要有神秘的DOS行命令的知识就可以维护和修复系统。但实际上，去除了实模式DOS功能对维护来说是一个障碍(现在的查皮都还有命令提示符，命令行才是王道阿)，而系统还原功能也带来一些麻烦：性能显著的降低、硬盘空间的大量消耗，并且对一些通常的错误还原并不一定有效。随意，基本上可以把他算作Windows 98的一个不成功改版。或许你会问，Windows 98se之后，不是还有Windows 2000，还有你隔壁那个查皮么，怎么能说走到尽头呢？听我慢慢道来.....

## 1.6 还是历史

回到1993年，Windows 3.1获得成功后，有点软公司不满足于个人用户的市场，开始进军服务器。于是，基于OS/2 NT的基础编制的Windows NT 3.1问世了，Windows NT 3.1是个32位的系统，比桌面系统提前进入32位时代，由于是面向服务器领域，Windows NT 3.1的稳定性要比桌面系统高很多。不过这个版本似乎不如下一个版本名字更广为人知——Windows NT 4.0

1996年8月，Windows NT 4.0发布，增加了许多对应管理方面的特性，稳定性也相当不错，这个版本的Windows 至今仍被不少公司使用着。Windows 的nt内核家族从此登上历史舞台，与以Windows 95代表的一脉并行发展。到千禧年的钟声敲响后，nt家族的又一个精英——Windows nt 5.0问世了，为纪念千禧年，他还有另外一个名字——Windows 2000 Windows 2000是主要面向商业的操作系统，他有4个版本：Windows 2000 Professional，用于工作站及笔记本电脑，可以叫工作站版。Windows 2000 Server 即服务器版，面向小型企业的服务器领域。Windows 2000 Advanced Server 即高级服务器版，面向大中型企业的服务器领域。Windows 2000 Datacenter Server 即数据中心服务器版，面向最高级别的伸缩性，可用性与可靠性的大型企业或国家机构的服务器领域。外号：最牛版

好了，该我的好邻居出场了，我们就把时间定格在2002年。从98问世到2002年，桌面市场的Windows 4年没有什么变化了(Windows ME是垃圾，Windows 98 SE不过是98的升级版)，而随着时代的发展，Windows 95一脉的内核越来越不能适应现代软硬件需求的发展，于是，有点软的公司一咬牙一狠心一跺脚——扔了！然后，拿过来一直表现良好的Windows NT系列的内核，开发出了新一代桌面版Windows ——Windows XP，也就是住我隔壁这位。Windows XP有两个版本，home edition和 Professional。Home edition面向家庭用户，相当于Windows 98。Professional面向工作站，相当于Windows 2000 Professional。而对应Windows 2000其他用于服务器版本的系统，是2003年的Windows 2003，也是一个很优秀的版本。好了，历史课就讲到这里，同学们，洗洗睡吧。

当然，  
跟Linux比.....  
就算了

<sup>1</sup>英语老师：你这里怎么不加s！



## 1.7 串门

今天接到了一个任务，挺起还还听轻松，一般胡同里大妈大婶的，经常做这项工作，并且乐此不疲，这就是——串门。事情是这样的，在我来之前，主人有许多照片存在了查皮那屋里，而现在可能他觉得这么珍贵的回忆全都堆在查皮那杂乱无章（在我看来）的屋里很不保险，所以，要我去复制一份，放在我这里。于是，我终于有机会去查皮那屋里看看他和他的同志们都长啥样。（前面说过，操作系统啥也干不了，他肯定也有一堆帮忙的应用软件，就像我的狐狸妹妹）不过，我串门可不像胡同大妈串门那么简单，认识门就行。我要想去查皮那屋，就得认识查皮在屋里规划的格式，或者说，认识查皮屋里的地里形势，或者说，认识查皮所用的文件系统。

基本上，查皮只会两种文件系统——换句话说，只会用两种方式规划整个屋子的空间，那就是Fat32和NTFS。fat32是一种很老旧的格式了，连4G以上的文件都不支持，性能也不好，还不支持多用户的权限，所以基本不怎么用了。这个查皮也是，没有用fat32，而是用了另一个比较高级的格式——NTFS。那么，我就必须能够读懂NTFS格式的磁盘，我才能去查皮那里串门，有人问了，那你不能读懂呢？谁问的？站起来，好，这个问题，恩，基本上，我可以负责的告诉你，自从Canonical学校为我们增加了一本ntfs-3g教材以后，ntfs就不在话下了。虽然能够读懂，但是我自己是不会用这个文件系统的，我会用很多其他的文件格式，比如ext2,ext3,xfs,jfs,reiserfs,ufs,zfs等等，各有优势，我现在的屋里使用的是非常强大的xfs格式，至于怎么强大，以后慢慢细聊，现在，我要走了，去串门。

来到查皮的屋里才发现，我串门跟大妈串门的感觉实在不一样。大妈去串门得在人家醒着的情况下去。我来到这里，只能在他们睡着的时候。感觉我不像串门来的，反倒像小偷-\_-b。要说这查皮还是真是不会收拾屋子阿，一地的磁盘碎片，多影响性能阿，我说他起床怎么那么慢呢。有人问：“啥碎片呀？我家有时候也有碎片，都是老婆和我吵架时候摔的……”对此，我可以用正宗的C语对你说：`printf(".....-\\_b\\n")`；

好吧，先不串门了，科普。

## 1.8 碎片

### 笨兔兔老师第1讲——什么是磁盘碎片

同学们都坐好啦，都把手机铃声关了，小灵通调成震动，BP机直接扔了——台都没了你还留着它干嘛。好，上课了，首先说说什么叫磁盘碎片。磁盘，是我们程序居住的空间，我们用不同的方式对整个磁盘的空间进行管理。上次说过了，包括各种方式，什么ext3,xfs,查皮的ntfs等等。而磁盘里放的东西，就是一个一个的文件，同学们可以把磁盘想象成你家的屋子，文件就像一个一个，大大小小的箱子。每个箱子上面写着字，就是文件名。查皮喜欢把每个箱子都紧挨着放，一个挨一个，上下左右前前后后都紧贴着，这样，看上去很规整。可以让剩余的空闲空间比较完整。有同学说了，我家也这么收拾，这样很利索呀。不过，对于操作系统，这样做虽然也有好处，但是会有一些问题。

比如，一开始存了一个文件，也就是搬来了一个箱子，比如叫“日记”。查皮把它放在最靠墙的位置，然后又存了很多其他的文件，在“日记”文件的前前后后，左左右右，上上下下都放满了。忽然这一天，日记文件被修改了，加了点内容，就相当于往“日记”那个箱子里加了东西。可是箱子已经满了，再往里加，箱子就要增大，或者理解为再拿个箱子也写上“日记”放在原来的箱子边上，可是不管怎样，箱子周围堆满了其他的箱子，没地方了，怎么办呢？可以把边上的箱子挪开一点，原来的箱子就可以扩大了。可是边上的箱子要是少还好办，要是很多，还都装的铅块铸铁大理石阿什么的，那可就累死了。那怎么办的，只好把新的内容放在另一个小点的箱子里，放在别处。然后还得在原来的“日记”箱子上标注上：“日记（第一部分，第二部分在东墙根）”。然后在新的箱子上写：“日记（第二部分，结束）”。如果日子长了第二个箱子也被n多箱子挤在中间后，又要编辑日记文件，这个文件又变大了，就又要如发炮制出第三个箱子，乃至第四个，第五个……等到有一天，要读取这个日记文件的时候，查皮就开忙了——首先，到西墙角找到日记第一部分，翻腾出里面的内容，然后往箱子上一看“第二部分见东墙根”，然后查皮在跑到东墙根找第二个箱子，翻腾出里面的内容，然后再一看箱子“第三部分见大衣柜上头”，然后查皮搬梯



子，上大衣柜一看“第四部分见厕所水箱后边”，在折腾到厕所“第五部分见屋子正中间从南墙数第两百四十八个箱子”……………等到查皮把整个日记文件读完了，也累得半死了。这种情况，就是会影响性能的磁盘碎片。

## 1.9 邻居

科普也科普完了，该干正事了。开始搬照片吧。先拿出这屋的文件列表来看看——我当然知道文件列表在哪，因为我学过NTFS格式。好，上面写着，照片在窗台底下，好，我来到窗台底下，没看见照片，却发现了一个熟悉的面孔……他带着个圆圆的眼镜，文质彬彬的样子，看上去像个学究，两道浓眉如同飞翔的海鸥。衣着并不华丽，倒也搭配的很是顺眼。人们喜欢叫他OO,可能是因为他的眼镜吧，而他的全名，叫做OpenOffice.org——相信我，这确实是个软件的名字，当然，同时还是个网站的名字。之所以我认识这家伙，是因为在我屋里也躺着一个。这并不奇怪，很多Linux下的软件都有相应的Windows 版本，OO老先生也是这样。基本上这个OO可以算是我屋里那个的兄弟吧，他们是相同的版本，相同的外表，相同的功能，只是一个跟着查皮混，另一个跟着我干。我绕过这位OO老先生，没有吵醒他的美梦（事实上我也叫不醒他）。终于自他身后的窗台下面发现了要复制的照片，不过别急仔细看一下，果然，上面写着“照片，第一部分，第二部分见里间屋写字台底下”哎~~我恨碎片……………来到里间屋，还没找到照片，先看见了床上躺着的查皮，这是我第一次看到这位可爱的邻居。他穿着红黄蓝绿四色的衣服，很是鲜艳。可是，不知道为什么，脸被涂黑了，上面还写着“使用正版，跟风黑屏”。看来主人是不希望自己的电脑里有盗版软件，所以才会在Windows 下也用OpenOffice.估计这个查皮是买电脑时候一起来的正版查皮。一边想着，一边来到写字台底下，找到了照片第二部分，往盒子上一看：“第三部分见……”Oh,God!

## 1.10 人才

终于把照片都拷贝到了我的屋子里，把它们放在了专门放主人文件的分区下。有人忽然想问，查皮那里那么多碎片影响性能，那你怎么放这些文件呢？其实很简单，我更倾向于把文件分散的放着，中间有足够的空间可用于扩展。这样就不至于在找东西的时候满屋子乱跑了。

刚刚休息了一下，主人又让我去叫醒一个家伙，他叫作apt-get。这个家伙就像个公司里的人事部经理，来个软件走个软件的，都是他管。当别人夸奖他的时候，他总是自信的拍拍自己的胸脯说：“本APT有着超级牛力”。而他也确实很厉害，很敬业，也很专业，对于人才（对我来说也就是软件）的各种情况了如指掌。要招一个人来的时候，他会做好所有准备工作，这个人需要用什么样的库，或者需要什么其他的人才能一起协同工作，他都会事先做好准备。比如，主人想用vim来编辑文件，就叫apt去招vim来。apt就会报告，说vim要来的话，首先需要准备好libncurses这个库，和python这种脚本语言的执行环境。征得同以后，他就会去网上找这些东西，并且运回家，把库放在该放的地方，相关的软件安排好住宿，然后再去找vim同志，请他过来帮忙干活，并且说明，环境都已经布置好了。每次新人来了之后都很感谢apt同志为自己做的这些准备工作，该有的东西，该来的助手都在，于是干活就事半功倍了。但把人才请来之后，apt同志的工作还没有结束，他还要把现在的人事情况记录下来，以便主人哪天问起来的时候好如实汇报。哪天主人文一句：“我说超级牛力阿，咱这现在都有多少软件阿，都是谁阿？”apt也能从容的回答。可以说，apt这家伙对于我来说实在是非常重要的，没有他，笨兔就不是笨兔了。不过，他并不是只为我打工的。以前，apt是Debian公司的人事部经理，人家Debian可是历史悠久的大公司，1993年就成立了。apt在这么大的公司里一直工作到现在，有大量的工作经验。当我们公司成立的时候，成功的请来他管理人事资源，对他来说自然是轻车熟路，得心应手。21 实际什么最宝贵？人才阿！

当然，这并不影响他继续给debian打工

## 1.11 来头

这次主人叫apt，是让他去找一个软件，名叫preload。apt同志翻开他那厚厚的记事本，查到了preload的资料，然后向主人报告：preload工作需要的条件我们这里都已经满足了，可以直接

把他请来。然后，在获得主人的同意后，apt出发了……

“比海更广阔的是天空，比天空更广阔的，是人的心灵。”500g的容量算不上海量，而屋外那个世界，却实在算的上比天空更广阔了，那就是网络。一个操作系统是孤单而无助的，只有接入了网络的操作系统，才真正能够发挥全部的能力。尤其是对于我来说，尤其是对于有apt做帮手的操作系统来说。apt可以从网络上获得各种软件的资料并记录下来，当需要的时候，只要跟他说一声：“我要xxx软件”，apt就直接去找去了，下载，安装，全都不用别人操心，他都给办了。如果没有网络，apt到也不是全无用处，至少他可以用来管理安装光盘上的软件，可是就光盘那点容量，我自己都管理的过来，要什么软件自己搜索都不会慢多少，就体现不出apt同志的“超级牛力”了。除了apt，很多其他的软件都是跟网络离不开的，比如狐狸妹妹，要是没网络，她就可以退休了。说起网络，实在是个很有意思的世界，有很多有意思的东西，不过，一个软件要想能够从网上取得信息，就需要懂得网络上的说话方式，懂得网络交流的语言，我们管它叫做——协议。懂得http协议的软件可以看网页，懂得ftp协议的软件可以传文件，不过这些都是上层的协议，底层，基本所有能上网的软件都要会的，算是tcp/ip协议了，apt就懂得这门协议，所以，他可以去网上找想要的软件。

转眼间，apt已经把preload请来了，并且做了一下安顿，完事后，看看没有什么其他的工作了，他就回硬盘去睡觉去了。要说preload这家伙我还真是没见过，不知道他是干啥的，有什么本事，不过既然主人要找他，总是有原因的吧。鉴于他名字念这不顺口，我们就叫他老p吧。老p好像很勤快，一来了就跑到内存里准备干活，我正要看他到底会干些什么，哪知他什么也不做，就在那里看着别人忙碌，时不时拿出个小本，记录着什么。一直到关机，大家都去睡觉，他都没说话。第二天一开机，他又早早的跑进内存，看着别人忙忙碌碌，拿着小本本记，还是一言不发，还是不做多余的事情，直到再一次关机。第三天，第四天，涛声依旧……这家伙到底什么来头？

## 1.12 本事

这一天，一如既往的起床，一如既往的看老p跑进内存，本以为他会一如既往的待在那里，一言不发，没想到他竟然说话了：firefox赶快起床，做好准备。狐狸妹妹被叫醒，一头雾水的看看我，因为每次都是我叫她。我也同样迷惑的看看老p，主人还没有发命令要用firefox阿，怎么就把她叫醒了呢？但既然被吵醒了，狐狸妹妹也就不睡了，迷惑的走进内存，看着老p。老p倒是镇定自若，一点没觉得有什么不对劲，转脸又说：Audacious起床，做好准备。Audacious是一个多媒体软件，他会使用那个叫做声卡的硬件设备，唱出优美的歌声来。我问过我们这里学问最高的星际译王老先生，星爷告诉我Audacious这个名字是大胆，鲁莽的意思。大胆，唱歌，所以，我们就管这个会唱歌的家伙叫“想唱就唱”吧。想唱就唱也被老p叫进内存，跟firefox站在一起，刚要问什么，这时主人发话了，要开网页。我马上明白了，看了一眼狐狸妹妹，她也很麻利，当我看她的时候，她已经在工作状态了。省去了平时狐狸妹妹起床的时间，反应快了不少，主人很满意。没过多久，主人果然又叫想唱就唱来唱歌了，一切都在老p的预料之中……

原来，老p这几天一直在记录分析主人的使用习惯，获得足够的数据之后，就可以知道哪些软件是常用的，哪些是不常用的，哪些软件哪些时候用，哪些软件哪些时候基本不用，正所谓金风未动蝉先觉，春江水暖鸭先知，主人用啥他先晓。有了他，整体系统的反应速度提高了，这就是他的能力，这就是他的本事，这就是他的价值。在这个世界里，没有一个程序是无用的，每个人都是人才——不同方面的人才。

其实也不够准确，preload应该主要是预读一些库文件，不会预读整个firefox的

## 1.13 开源

大家见识了老P的本领之后，都很乐意的听候他的调遣，整体的工作效率提高了一些。不知道查皮那里有没有类似的角色，于是就拜托狐狸妹妹去网上问问，结果发现在查皮发布的时候，有点软的公司就宣称，查皮有类似的功能，可以记录用户对软件的使用情况，使用的多的软件就能够较快的启动。而让人不解的是，5年后，查皮的下一代，长得比他漂亮的Vista系统发布时，有点软公司还在宣传，Vista系统增加了记录用户习惯的功能，用的多的程序将得到更快的启动速度。也不知道到底是加了没加，反正他们公司的系统，总是越用越慢倒是真的。到底为什么慢，我也说不清，因为他是一个闭源的系统。

看到这个词，总让我想起Visa,于是我总是觉得这个系统很贵

我就知道你想问什么叫闭源。

## 笨兔兔老师第2讲——闭源和开源

什么是闭源呢？就是源代码不开放。我们知道，程序是程序员们一行一行的语句编出来的，c语言也好，java也好，这一行一行的语句，就是这个程序的源代码。有了源代码，就能够100%的了解整个程序的构造，如何工作。而源代码是不能运行的，比需要把源代码变成可执行的二进制程序，这个过程叫做编译。源代码经过编译之后，才可以运行，但是编译之后的程序就不能够知道内部的构造了。我们平时在网上下载的各种程序，都是编译好的二进制程序，如果你想要它的源代码，对不起，不行！这是商业秘密，怎么能给你？给了你，我们的软件怎么卖钱？这种不开放源代码的程序，就叫闭源程序。打个比方，就好像肯德基。麦辣鸡翅谁都可以得到，只要花钱买就行，但是配方没人知道（虽然其实也没多好吃）。配方就相当于源代码，麦辣鸡翅就相当于编译好的二进制程序，制作过程就相当于编译过程。如果有了配方（源代码）你就可以自己作麦辣鸡翅（自己用源代码编译出二进制程序），甚至还可以根据口味对配方进行修改。（根据自己的需求修改源程序，为软件增加自己需要的功能）

既然有闭源，那是不是还有开源呢？你答对了。Linux，就是一个开源的系统。

开源是什么？开源是一种精神，是乐于分享的理念。再举个例子，有一天你发现，蒸鸡蛋羹的时候往里面加点牛奶，可以让鸡蛋羹更滑嫩。知道了这个窍门，你很高兴的把它告诉你的朋友，让他们分享你的经验，于是大家很高兴的也学会了做这样的鸡蛋羹。这就是开源。你也有可能不把它告诉别人，而是保留这个秘密，甚至申请个专利，然后开个店去卖京城独一份的奶香滑嫩鸡蛋羹。这就是闭源。当然，这之中没有谁对谁错，睡好谁坏，只是不同的理念而已。

## 1.14 故事

以前讲过查皮他家的历史，现在来说说我家的故事。话说1991年，那是一个夏天。有一位牛人在世界的互联网上画了好多圈——

Hello everybody out there using minix--I'm doing a (free) operating system

英文圈多.....

大家可能看不明白，我来逐一解释一下每个单词：第一个，Hello，这个是打招呼的意思，哦，你知道啦，那说第二个。everybody，每个人，跟我念，爱~唔~瑞~八~迪~，哎呀.....呢，好了好了，不要着急，把西红柿鸡蛋都收起来吧，我直接说重点——minix。Minix是一个操作系统，要想说清楚Minix，得先从Unix说起，好咱一个一个来。

说minix，就不得不说说Unix。UNIX也是一个操作系统，而且是一个历史悠久的系统。1965年，鼎鼎大名的贝尔实验室加入了一项由奇异电子(General Electric)和麻省理工学院(MIT)合作的计划——制作一套多使用者，多任务，多层次的MULTICS操作系统。贝尔实验室的大名大家都知道，晶体管、激光器、太阳能电池、发光二极管、数字交换机、通信卫星、电子数字计算机、蜂窝移动通信设备、长途电视传送、仿真语言、有声电影、立体声录音，以及通信网的许多重大发明都诞生自这里。麻省理工大学更是历史悠久，技术雄厚。所以，这个MULTICS操作系统的项目在1965年成立，到1969年就.....被取消了，主要原因是进度太慢。可见编操作系统不是一件容易的事儿。

这块是抄来的，呵呵，没仔细斟酌。应该翻译成“通用电气”更明白些。奇异电子是另外的翻译，估计是取其缩写GE的谐音。

真是世事难料阿，看似事情就这么结束了，然而，其实故事才刚刚开始，因为一位英雄的出现。

Ken Thompson也在这个计划中，计划取消了，他很郁闷，因为他编了个星际旅行的游戏，没法玩了。这个程序之前运行在一台型号是GE-635的机器上，这个机器的系统大约就是他们计划开发的MULTICS系统，但是反应比较慢，玩起来不爽。Ken Thompson满怀希望的憧憬着项目完成的时候，系统能够优化的顺利的跑起来他的游戏，然而项目竟然取消了，怎么办呢？毛主席教导我们说：自己动手，丰衣足食。我估计Ken Thompson没有背过毛主席语录，但是他用自己的行动证明了其正确性。他在墙角淘换出一台PDP-7的机器，并且伙同Dennis Ritchie将星际旅行移植到了这台PDP-7上。这台幸运的PDP-7因此在历史上留下美名。就是这台：





当然，要想运行这游戏，当然得有个系统，这个系统，就是Ken Thompson和Dennis Ritchie用汇编语言写出来的，非常简陋的，UNIX的前身。这都是为了玩个游戏阿～

## 1.15 Minix

在强大的，玩游戏的欲望的驱使下，两位牛人完成了UNIX的最初雏形版。这个系统只支持两个使用者（估计做的时候没考虑别人，够他俩玩的就得）相对于那个 MULTICS系统——MULTiplexed Information and Computing System，Brian Kernighan 开玩笑地戏称他们的系统其实是：“UNiplexed Information and Computing System”，缩写为“UNICS”。后来大家取其谐音，就诞生了UNIX这个词。这一年，已经是1970年，史称Unix元年。后来，Brian Kernighan觉得用汇编写的系统不好维护，于是……他发明了C语言（符合他一切自己动手的风格），然后用C语言又重写了一遍。从此，Unix走上了发展的快车道，并且一直用到现在。许多世界级的大服务器，用的都是Unix系统。

好，Unix就说到这里，我们的正题是Minix。要说Unix确实是很牛的，很有技术含量的，是值得学习计算机科学和操作系统的同学们学习的，然而，Unix也是天价的，广大穷苦的大学生们是买不起的。荷兰阿姆斯特丹的 Vrije 大学的Andrew S. Tanenbaum教授深刻的体会到了这一点。他的学生们学习了计算机学习了操作系统原理，总得实践一下吧？总得找台机器用用吧？要用计算机就得有操作系统吧？买个DOS装上？虽然那时候DOS已经问世了，但是这么一个单用户单任务效率也不高的操作系统，实在不能指望它培养出什么软件的人才。装个 Unix？学校还不破产。于是Andrew S. Tanenbaum牛人拿起键盘——咱自个儿编一个吧！然后Minix就诞生了。Minix取Mini Unix之意，自从1987年被编写出来，到1991年发展到1.5版，现在有两个版本，1.5和2.0。这个操作系统的初衷，是作为一个用来学习的模型。所以功能很简单，体积也很小。并且以后也没有进行进一步的开发和扩充，为的是能够让学生在一学期内能学完。那时候Minix在大学中用于教学是免费的，但是用于其他用途是需要给钱的。不过现在已经彻底免费了。它作为一个操作系统，其实并不算优秀，但它是一个源代码完全开放的操作系统，这使得有理想有志向有报复的黑客们，第一次能够完整的阅读到一个操作系统的全部代码。这其中，就包括芬兰赫尔辛基大学的学生，Linus Benedict Torvalds ……

## 1.16 Linux

那时候，Linus是赫尔辛基大学计算机科学系的二年级学生。他的最大爱好，就是折腾计算机。测

试计算机的能力和限制，整天研究怎么让计算机按照自己的想法去干活，怎么发挥计算机最大的性能，把可怜的机器累得精疲力尽呼吸带喘直到电容爆浆，吐血身亡。在学校，计算机上装的是教学用的Minix系统，虽然适合拿来学习，不过系统本身并不强大，渐渐的，这个教学用的操作系统已经不能满足Linux大侠的欲望，可是似乎又没有别的选择。上面说过了，Unix奇贵无比，而且无论Unix还是DOS，他们的代码都是不开放的，这系统只能拿来用，没法拿来折腾的。于是象其他牛人一样，Linux自己动手了。

今天，我们知道，linus从那时起开始了一个事业，一个神话，但在当时，他并没有想那么多，只是为了学习Intel386体系结构保护模式运行方式下的编程技术。他并不知道即将创造的将是一个在世界范围广泛使用的系统，而只觉得是自己一时的异想天开。因此，一开始他把自己写的这个操作系统命名为FREAX。就此开始了这个“异想天开”操作系统的编写。大约1991年4月份的时候，就编写出了第一个可以运行的版本——0.00版。这个版本可以启动，运行两个进程，分别在屏幕上打印出AAA，和BBB，然后……没了。虽然连句整话都不会说，不过这是一个好的开始，至少能启动了。

如果他就这么干下去，估计到今天，就不会有Linux这个东西了。一个人的力量是有限的，有道是人多力量大，众人拾柴火焰高，多个铃铛多个响，多根蜡烛多分光，一个篱笆三个桩，一个好汉三个帮，三个臭皮匠还顶个诸葛亮……哎呦~好吧，就说这么多了。总之，linus让他的操作系统和互联网，亲密接触了。于是就有了前面说的这句“Hello everybody out there using minix—I'm doing a (free) operating system”（可算绕回来了）这是他当年在comp.os.minix上发布的消息，告诉大家，他正在写一个操作系统。并且，他还把他写的“异想天开”操作系统的代码上传到ftp.funet.fi的服务器上让你给大家下载，以便交流心得，共同学习。这就相当于你跑到网站上发帖子说：我研究出一种萝卜炖牛腩的方法，主料是啥啥啥，配料是啥啥啥，怎么怎么炖，大家都试试吧！（对不起，我又饿了）于是很多有兴趣的人就来尝linus炖的牛腩，哦不对，是尝试linus写的系统。不过当时那个服务器的管理员Ari Lemke看着这个异想天开的名字就不顺眼，想想，既然是linus写的操作系统，又是类Unix，或者说类minix的（minix本身就是类Unix的，大家都是一家子），干脆，叫Linux吧。

## 1.17 Friends

Linux被公布在网上之后，引来大家的围观，很多人觉得这个东西挺有意思。不过第一个对外发布的0.01版Linux还有很多的完善（这简直是一定的）。这里先要说一个概念，Linux是什么？确切的说，狭义的讲，Linux只是一个操作系统的内核，他只是各位的Ubuntu系统里面/boot/目录下的那个内核文件。就好比汽车，Linux只是一个引擎，只是大家普遍的把装了Linux这种引擎的汽车叫做Linux汽车。那么既然Linux只是一个内核，要想工作就还需要很多周边的支持，比如文件系统，比如一个命令行程序，比如一些基本的软件。

由于当初linus大侠是在minix系统上开发的，所以最一开始Linux用的文件系统是借用minix的文件系统。可老借别人的总不是个事，还是应该有自己的文件系统，就像查皮的FAT和NTFS。前面说了，文件系统也就是自己管理自己这点硬盘空间的方式，自己的屋子用自己的方式管理，自然最顺手。这时候来了个牛人叫Theodore Ts'o。

Theodore Ts'o，1990年毕业于美国MIT大学计算机专业。他爱好广泛，喜欢烹饪，骑车，还有折腾电脑（这都不挨着啊~），后来又玩上业余无线电报了，当然这都不是主要的。他看到Linux觉得很有意思，于是怀着极大的热情为Linux提供了邮件列表服务以便大家一起讨论问题，后来还提供了ftp站点来共享Linux的代码，并且一直用到现在。除此之外，技术上，他编写了Linux0.10内核中的虚拟磁盘驱动程序和内存分配程序。在感觉到Linux缺少一个自己的文件系统后，他提出并实现了ext2文件系统，并且ext系的文件系统一直都成为了Linux世界中事实上的标准，任何一个发行版都会默认支持。现在已经发展的遍地ext3，期盼ext4了。Theodore Ts'o可算是Linux的顶级元老了。

另一位元老，一个英国人——Alan Cox。他工作于英国威尔士斯旺西大学，特别爱玩电脑游戏（又一个玩游戏的，可见玩游戏也不是坏事），尤其是网游（你看你看，还是网游），不过那时候的网游不像现在这样华丽，那时候是字符界面的，能想象嘛？字符界面的网游！那种叫做MUD——Multi-User Dungeon or Dimension。玩MUD当然就得有计算机啊，就得有网啊，所以Alan Cox就开始逐渐的对计算机和网络产生了兴趣。为了提高电脑运行游戏的速度以及网络传输

的速度，他开接触各种操作系统，为自己选择一个满意的游戏平台，争取榨干电脑的每一个指令周期。经过自己考虑，他买了一台386SX电脑，并且装了Linux0.11版的系统。这主要是因为预算比较紧张，即使 minix他也买不起。（重复一下，minix用于教学是免费的，但是其他用途要收费，包括个人用。）于是他开始使用Linux，进而学习其源代码，并对 Linux产生了兴趣，尤其是网络方面相关的代码。（整天琢磨怎么榨干他家那点带宽）在Linux0.95版之后，他开始为Linux系统编写补丁程序，以后逐渐加入Linux的开发队伍，并成为维护Linux内核源代码的主要人物之一。那个有点软的公司还曾经邀请他加盟，被他有点硬的拒绝了。

再有一位，Michael K. Johnson，他是著名的Linux文档计划的发起者之一，写了《内核骇客手册》一书，曾经在Linux Journal工作，现在在著名的商业发行版RedHat的公司工作。

当然除了这些大牛，还有更多的大牛，中牛，小牛，牛犊，牛杂，牛尾，牛头肉，肥牛……（唉，又饿了）……们，都为Linux的发展做出了自己的贡献。他们来自不同的国家，从事不同的职业，他们甚至从未见过面，但是他们为了一个共同的目标，通过网络，一起合作，利用自己的业余时间，义务的帮助Linux成长，才有今天这个可以合法免费使用的操作系统。这是什么精神？这就是软件国际共产主义的精神！（好吧，这个词是我造的）

## 1.18 杀毒

这天又去查皮的屋里搬东西，看见有几个生面孔，长得怪猥琐的也不知道干什么的，查皮那还真是什么程序都有阿。刚把东西搬了回来，就见 apt火急火燎的跑过来：“报告系统，本APT有超级牛力！我要用网络。”——是的，任何程序要使用任何硬件资源都要经过我的同意，因为我是操作系统。

我一边查看资源情况一边问：“这回去招什么软件呀？”

APT说：“叫AVAST。本APT有超级牛力！”

“这软件干什么的呀？”

“本APT有超……”

“算了算了，我不问了，快去吧超级牛力。”

话音未落，apt就飞也似的跑出去了，从远处还传来他悠扬的声音：“……超级牛力！”

唉，就说现在是牛年了吧，也不至于这样啊。

过了一会，“超级牛力”回来了，带回来一个软件，看着软件个头不是很大，长得也比较难看，只有个很简陋的图形界面。我顺手拿过他的手册man了一下（Linux下，你想知道一个软件是干啥的，怎么用，你就man他。当然，你得懂点英语。）才知道原来是个杀毒软件，还挺有名气的。心想，我又不中毒，主人装杀毒软件干什么？忽然想到了在查皮屋子里看到的几个萎缩的面孔——莫非是隔壁那哥们中毒了？

果然，在被“超级牛力”安顿好住处之后，AVAST马上被主人叫进内存去工作。他先是去网上下载了最新的病毒库——就相当于一查子通缉令，上面写着各种已知病毒的名字，相貌特征，作案手法等信息，以便杀毒软件查对。下载完毕之后，就见他收拾好工具，整理一下装备，向着那黑洞洞的隔壁，出发了。

说起来，隔壁那个查皮还真是挺害怕病毒的，防不胜防啊。针对他的病毒多种多样，各有各的本领，虐待起查皮来真是八仙过海各显其能，而且查杀不易。他们有的会伪装成别的软件，查皮叫醒“记事本”去干活，殊不知真正的记事本已经被病毒一棍子打死了，现在躺在那长得跟记事本一样的家伙已经是整了容的病毒。有的能够藏在正常程序的里面，正在工作的IE同志，很可能工作服的兜里就隐藏着病毒。并且病毒们现在基本都回随着查皮一起起床。当查皮被叫醒，伸个懒腰揉着眼睛走进内存的时候，他庞大的身躯后面可能正趴着40多只病毒。由于病毒是活的，要杀掉很困难，它们可能会有很多人共同作战，杀毒软件杀掉了内存里的强夫，内存里的大熊会把硬盘里强夫的复制版在叫起来。扭头杀毒软件去杀大熊，强夫会把杀掉的大熊抢救过来，结果谁也杀不了。有的病毒更暴力，自己先跑进内存，一看见有杀毒软件要进来，立马上前一铁锹拍死，然后藏起铁锹装着杀毒软件的声音说：“杀毒软件成功启动，没有发现病毒，噢耶”甚至还能监视着IE,一旦他要访问什么杀毒防毒相关的网站，二话不说，直接干掉！

但是AVAST去查毒就简单多了，因为这时候隔壁的查皮没起床，所有他那边的软件都在睡觉，病毒也一样，所以不会有任何反抗能力。AVAST过去，只要根据通缉令一一对照即可，只听



隔壁那边“阿~~~”“厄~~~”“哎呦~~”“我死得好惨哪~~~”等等叫声不绝于耳。过了很长一段时间，AVAST回来报告：共发现病毒7种，共214只，全部歼灭。

## 1.19 免疫

有人问，查皮那里的病毒那么可怕，你这里怎么没有病毒呢？好~

### 笨兔兔老师第3讲——为什么Linux不中毒

首先我们来了解一下病毒，病毒是什么？其实说简单了，病毒只是一个程序，一个坏坏的程序。既然是程序，就跟其他的正常程序一样，依赖于不同的平台。啥意思？就是说，给查皮打工的，没法给我干活，给我干活的，也不理查皮那一套。我要是拎过一个查皮那边的程序跟他说，快起床干活。他压根也听不懂，闭上眼睛继续睡，语言不通啊。所以，病毒也一样，针对查皮的病毒传染不了我，针对我们Linux的病毒也不可能传染查皮。

那有没有针对Linux的病毒呢？答案是有的。第一个Linux病毒诞生于1996年，澳大利亚的一个叫VLAD的组织用汇编语言编写了Linux系统下的第一个病毒：Staog，不过这个病毒只是个试验品，只是证明一下Linux也会感染病毒。这个病毒会感染二进制文件，获取root权限，然后说：你看，我获取了root权限。炫耀完了也就算了，并不做任何破坏性的事情。后来也有了一些有破坏性的病毒，但是数量很少，经过科学家计算，一个不装任何杀毒软件或防火墙的Linux在互联网上中毒的几率大约比一个人花两块钱买彩票中五百万后立刻被雷劈中的概率大那么一点点。（这是哪门子科学家）病毒少，这是Linux不容易中毒的一个原因。可为什么病毒少呢？

话说有一个邪恶的人，出于某种邪恶的目的，他想编个Windows病毒。他买书学习Windows的知识，找熟悉Windows的高人前辈们学习。经过种种努力，编出了一个病毒，然后把这个病毒放在自己的网站上，只要使用Windows系统，IE浏览器上网的人一登陆这个网站，就必定中毒。放上去之后，他等着，看着有1000人来到了他的网站，看着其中900多个纯洁的查皮系统感染了病毒（总有用IE的，装防火墙的Windows吧），他很有满足感，他觉得自己成为大牛了。

话说有另一个邪恶的人，出于某种邪恶的目的，他想编个Linux病毒。他买书学习Linux的知识——不过好像不太好找，好不容易找到基本也都是基础知识。找找高深的吧，还都是英文。好吧，英文的也看，对着字典慢慢研究。哦，对，还可以找找高人指导，不过……也不好找哈，找了半天找到一个高人，拜他为师吧。经过师父指点和自己的努力，他学到了很多Linux的知识。然后费尽心机编了一个Linux的病毒，然后把这个病毒放在自己的网站上，只要使用Linux系统，firefox浏览器上网的人一登录这个网站，就必定中毒。放上去之后，他等着，看着有1000人来到了他的网站——998个人都是Windows系统……好吧，好歹还有俩用Linux吧，可其中一个不用firefox，而是用Opera。邪恶的家伙咬咬牙，忍！看最后一个——哈哈，这家伙是Linux+firefox，只要登录准中毒。可是只见着人来了转转又走了，一点事没有，临走还顺手改了自己的主页，上面写着：

小子，跟我玩你还嫩点。

-----师父留

通过对比我们得出结论——写Linux病毒，没前途！

除了以上所说的原因意外，Linux以及周边软件的开源本质，也导致了病毒较少。比如我吧，主人要装什么软件，都是叫apt去找，apt可不是四处瞎找，而是去Ubuntu官方的软件源里去找——因为这些软件是开源的，所以可以随意的拿来，放在一起，做成软件源，供Ubuntu们统一下载。官方的东西，自然没有病毒了，那个娘也不能害自个孩子不是？Windows就不一样了，它上面的软件基本都是闭源的，要装，得自己上网搜，在某个网站搜到了，下载下来装。可这“某个网站”，就不知道他靠不靠谱了。谁知道上面的软件有没有病毒呢？那那个有点软的公司不能也开个官方的软件源，让大家都去他那下软件么？当然不能了，都是闭源的软件，你拿来用都要给人家钱的（当然，也有免费的），拿来分发可能压根就是不允许的。另一方面，Linux的开源导致了大家都可以对其进行完善，一旦发现漏洞，随便谁都可以去修复这个漏洞，只要他有能力。可Windows呢？发现了漏洞，也只能漏着，等着有点软公司去修。

世界上第一个真正意义上的电脑病毒诞生在1987年,名字叫做C-BRAIN。这个病毒程序是由一对巴基斯坦兄弟所写的,他们在当地经营一家电脑商店,由于当地盗拷软件的风气非常盛行(其实.....我们现在.....),因此他们的目的主要是为了防止他们的软件被任意盗拷。只要有人盗拷他们的软件,C-BRAIN就会发作,将盗拷者的硬盘剩余空间给吃掉。(说起来目的还是可以理解的)

要说这么个病毒破坏力也还不大,不过毕竟开了先河,自此以后,各种各样的病毒层出不穷。开始,编写病毒的人多是为了好玩,或者为了证明自己能力。比如DOS时代很有名的雨点病毒,发作时键盘暂时锁定,屏幕上的字符(DOS时代,当然满屏幕都是字)像下雨一样刷刷的往下掉,电脑前那哥们眼泪也刷刷的往下掉,不过别着急,等所有的字符都掉到屏幕底下了,也就完了,您接着干您的。不过,过一会又得刷刷一遍,但是并不破坏什么。相比之下,有些病毒就是恶意的了,比如著名的黑色星期五,算是老牌的了。还有当时名噪一时的 CIH,号称首个可以破坏硬件的病毒——其实就是改写了BIOS程序,使主板无法工作。自那以后,现在主板的 BIOS基本都是可插拔的了。

好了,这节课先上到这里,下次见。

## 1.20 权利

AVAST给查皮杀光了病毒以后,据说查皮工作起来顺畅了不少。不过他似乎并不知道发生了什么,仍然很自以为是的摆出一副傲慢的表情,该怎么干活还怎么干活,也不说小心点别再染上病毒。(当然,这一切不是我自己看到的,是听人说的。主人叫戴眼镜的OO老先生记录下了给查皮杀毒的前前后后,还摆脱狐狸妹妹把这些记录的问题放到了他的BLOG上,他们两个聊天的时候告诉我的)我很看不惯他这样的表现,也不喜欢他对权力很强的占有欲。

查皮这个系统在安装结束的时候,会有一个创建用户的步骤,输入用户名,以后就用这个名字登陆了。这个用户是有管理员权限的。当然也可以不输入,无论输入不输入,系统里都会有一个很重要的用户——Administrator 我,也就是Ubuntu这个系统在安装结束的时候,也会有一个创建用户的步骤,也输入用户名还有密码,以后也就用这个名字登陆了。这个用户也是有管理员权限的。当然也可以不输入,无论输入不输入,系统里也都会有一个很重要的用户——root 有的同学举手了,说:我知道了root就相当于Windows 里的Administrator,有着最高的权限。很好,领悟的很快,但是——并不准确。

Windows 下权利最高的是谁?是Administrator吗?很遗憾,不是,是SYSTEM!也就是系统自己,查皮他自己。任何管理员的权利都不能大于查皮自己的权利。你可以试试去把C:\Windows下的regedit.exe删了。能么?“哇!我删了耶,没报错。”别着急,刷新几下看看,是不是又出来了?查皮会保护自己,不叫人类破坏。这个初衷看似还是好的,但是当查皮自己中毒的时候,就不一样了。当他中毒时,就像被外星生命寄生的人类(异型看过吧?),就不再是正常的人了,不正常的查皮仍然会努力保护自己,不让人动他身上的任何部分——包括已经中毒变坏的部分。

那Ubuntu下(其他Linux也是一样)权利最高的是谁?毫无疑问的是root!是这个用来给人类登陆的用户。root在系统中拥有真正的,至高无上的权力,他真的无所不能,他可以运行rm \* -rf(危险动作,切勿尝试,后果自负)删除系统中的所有文件,或许我会语重心长的警告他,这么干很危险滴,这么干就都删光光了,这么干我这个系统就囧屁了,不存在了!但是,当他确认的告诉我,他现在很清醒很冷静,知道自己在干什么之后,我会义无反顾地留着两行热泪按照他的命令去做!哪怕他要格掉整个硬盘,我也照办。这真是,君叫臣死,臣不得不死。他叫我格,我不能不格。

会有这样区别的原因,还是我们两个的理念不同。查皮认为,人类是会犯错误的,很可能一不小心就把系统搞坏了,所以必须加以限制,有些事情让做,有些事情无论如何不能让他们做。而我总觉得,人类是聪明的,他们知道自己在干什么——尤其是用root登录进来的人,我认为他是了解我,了解整个电脑才会用root登录进来做事的。所以他的命令不会受到任何的阻挠。而一般的用户,会用普通帐号登录,既然用普通帐号登录,就说明他们承认自己只是个使用者,可能会做错事。那么我就会稍微进行限制,让他们不会破坏我,也不会破坏其他用户的東西。所以,当你用root账户登录进来之前,一定要想清楚,自己身上的责任。

Windows 下是不可能  
在系统运行  
的时候格掉  
系统盘的

## 1.21 内存

骑白马的不一定是王子，也可能是唐僧。烧香的不一定是和尚，也可能是熊猫。  
蓝脸的不一定是戏台上的窦尔墩，也可能是我隔壁的查皮。

查皮好像比较禁受不住刺激，对工作间的要求比较高，一旦哪个程序带进来只小虫子（bug），查皮经常吓的脸色变蓝，念叨一堆英文字母然后就开始倒数，数完了，就把整个机器重启了。查皮的这种毛病让好多人郁闷不已，那他到底为什么蓝脸？蓝脸以后又是在干什么呢？

前面我说过，操作系统的本职工作就是管理——管理硬件资源，管理各种程序。就好像老师管理一个班的学生，老板管理一个部门的职员。不过，无论学生还是职员，都有可能不听话，程序也是如此。查皮整天坐在工作间（内存）里吆喝：“QQ快起床，IE呀，你看看这内存里就这么大地方，你一个浏览器要占多少啊。QQ怎么还没起床啊？快点快点。我说瑞星啊，你能不能别让你那狮子到处乱跑啊，它净用CPU了，快把它赶开。QQ那QQ那，怎么还不起啊，再不起老大该怒啦，有40多MM等着他去聊天呢。啥？你说迅雷占着网络你起来也没用？唉，迅雷你也是，就那么点带宽，就说你下的这个什么ubuntu dvd挺大的吧，就不能留个5k，10k的给qq用用？你瞧瞧人家IE，也能下载，他……咦？IE？你怎么站那不动了？IE！IE！靠，又没响应了，还得拍晕了从来……”每天在这样的高强度压力下工作，查皮有些心力交瘁。怪不得查皮连续不断电的干上几天就不行了，而我可以连续干上几个月都没问题。查皮的神经就这样每天紧绷着，程序来个假死什么的还算好解决，可要是哪个程序忽然抽风，再内存里追跑打闹，上窜下跳，查皮一时手足无措，就容易蓝屏了。蓝屏之后，他会向老大（我管他叫主人）报告，自己为什么蓝了，问题发生在内存的哪个区域，发生了什么，并且把当时内存里的情况如实的记录下来，写成一份《工作间突发事件记录》一边记录一边报告记录的百分比——这就是他在倒数。记下来这个干什么呢？牛人们可以拿着这份记录，分析到底是哪里出了问题。不过好像一般人都不是牛人，谁也没看过查皮的记录。

对于工作间的使用，查皮和我还有一点不同。查皮总是喜欢尽量留出空间来，好给新起床的程序用。可是我总觉得，查皮怎么能知道还会有什么程序要运行呢？要是没有程序要来了，工作间里还空那么大地方，不让正在工作的程序用，那不是浪费么？我还是习惯尽可能的把东西都搬进工作间里。除了程序们申请多少内存就尽可能给多少之外，剩下的部分，我就把一些可能会用到的库啊，命令啊啥的统统都搬进来，能占多少占多少。那有人问，要是你把这里边都沾满了，待会有程序要进来咋办？很简单啊，我再搬出去呗！程序要进来，也不是一下子都进来，他也得把他的东西一点点搬进来，他往内存里搬的时候，我就往外搬，不耽误。所以，当有程序要启动，跟我说：我要10平米的地方放东西。的时候，我就先答应他说，好，放吧，有地。然后在他往里搬的时候我再给他腾地方。也可能他要10平，但是只用了2平，那我就先腾出2平米来，等他再要我再腾。他们管我这个方法叫Copy-on-write。查皮就不同了，可能是因为他比较胖的缘故吧，他比较懒，不愿意搬来搬去这么折腾。基本上他只是在必用啥东西的时候才把那东西搬到内存里，让内存留出尽可能多的空间。这样，当有程序管他申请内存的时候，他就可以用手一指：那块地，归你。然后就不用管了。实在内存不够用的时候就找个比较闲的程序，命令他：你，去硬盘里先忍会。（顺便说说，这个32位的查皮，并不能够完全利用起这4G大的内存空间，而是只能用到2.5G，浪费啊。）

所以，经常跟查皮打交道的人，总觉得内存里空着的地方越大越好。当他们看到我把内存占的那么满的时候，总觉得很不爽，唉~我冤枉啊。

## 1.22 图像

听说有一部电影，叫做《Big Buck Bunny》，这部电影长达9分56秒——还没电视节目中间的广告长，但是他有一个特点，一个和我一样的特点——他是开源的。怎么个开源？他是在开源的平台上用开源的软件制作，并且免费下载观看还可以获得他的原始制作文件，（blender的文件）如果你愿意，还可以进行修改，编个续集什么的。有人问，你说这些干嘛？这个电影跟你这个操作系统有什么关系？他本来跟我没什么关系，但是随着一件事情的发生，他就跟我有关系了——主人想看看。



任务下达下来，马上开始准备解决方案。首先是狐狸妹妹如春风摆柳般走了过来，顺便带来了一阵叮叮当当稀里哗啦的响声——一身的插件啊~来到工作间，狐狸妹妹掏出一个插件在手中一晃：我有Flash，直接去个什么土豆啊，地瓜啊，西红柿什么的网站去看就得了，最省事。这时候，墙角有一位慢条斯理的说话了：“要说这看片啊~还得我来~你那个不专业。那电影才10分钟，剧情肯定没什么可看的，人家看得是个效果，要得是清晰度。我看那，还是下载下来，我去放吧。”我扭头一看，说话的是MPlayer，要说这家伙在多媒体部门里可算是个元老了，而且能力相当强，什么片都能放，什么rmvb,flv,avi,wmv全都不在话下。就算您没图形界面，人家跟字符界面照样给你放电影。哪怕您显示器都不带色（念shǎi 三生），人家能给您拿字符拼出电影看。现在时代发展了，都高清了，人家也不甘落后，照样能支持，什么硬解码软解码的，通吃。既然人家这么专业，我看八成就得用他了，不过这事情我不做主，还是得等主人的吩咐。果然，主人也觉得应该下载下来看，于是，我们就忙活起来了。

要完成看片的大业，需要群策群力，精诚合作！

首先，虽然没有采纳狐狸妹妹的建议，但她并不沮丧，收起Flash插件掏出另外一个插件——Dnwmall！听这名字就知道是干什么的了。狐狸妹妹先出门去找狗狗大哥（学名叫google），打听到了《Big Buck Bunny》的下载地址，然后掏出dnwmall，把地址一填，就开始下载。要说这一身的插件实在没白装，哪个都有独到的功能，不一会，一部电影就下载回来了。然后就改MPlayer上场了，他先拿过片子看看格式，是ogg的，然后掏出了相应的解码器。解码器是干什么的？要知道，片子的格式很多，就好像现实中看电影，有数字电影，就要用数字放映机。胶带的，就得拿传统的放映机。在家里看光盘的，就得拿DVD机，看录像带的，就得拿录像机。MPlayer就像个电影放映员，解码器就是放映机。放映员在怎么牛，也得有各种放映机做支持，没放映机他啥也放不了。Mplayer掏出解码器，开始放起影片来。这就完了么？还差得远呢，要想让主人看上片子，还少不了图形部门的支持。

图形部门主要负责给主人显示漂亮的图形界面。他们那的老大叫Xorg，他会跟硬件打交道，会用显卡（当然，用显卡也得经过我），能在显示器上画东西，想画什么画什么，谁要想显示点东西给主人看，都得经过他。要想跟Xorg打交道，在显示器上显示出图形来，得懂他们图形部门的黑话——学名叫协议。他们说话使用一种叫做X的协议，这个X不是牛X的X，也不是傻X的X，而就是XYZ的那个X，XP的那个X，反正就是个X。要想显示图形，就得用这种黑话跟Xorg去说。每一个要显示图形的程序都得会这种黑话，比如狐狸妹妹，要显示东西，就说：“驼子碗，筛土的抛闪！”那意思就是说画一陀黄色的便便--b，当然，这就是比方，其实我可不不懂他们的黑话。（这一点不像查皮，他本身兼职负责画图形）那么Mplayer要画什么直接跟Xorg说就行了么？其实也行，那就像是在字符界面下看片了——没有窗口，图像没法移动，没法全屏，没法最小化等等。MPlayer只负责放片，像画窗口啊，移动窗口什么的这些事情他可不管。那谁来管呢？这时就需要一个窗口管理器，我们这里的窗口管理器叫做metacity（就是Gnome下的默认窗口管理器）。MPlayer要放什么东西其实是跟他说的，比如Mplayer说：“画一只猪”（当然是用X黑话，我给翻译过来了）于是Metacity转头告诉Xorg：“在某某位置画个方的窗口，在里面画一只猪。”过一会主人觉得Mplayer方的片挡着他跟MM用Pidgin聊天了（那是，猪哪有MM好看呀），就把Mplayer的窗口挪了挪，于是Metacity又对Xorg说：“把刚才那只猪和窗口往左移动3.2厘米。”这个过程Mplayer是不知道的，他只管专心的向Metacity描绘着影片中的一幅幅图像：“猪，走路的猪，跑动的猪，跌倒的猪，捆绑的猪，烤熟的猪……”

## 1.23 信封

《Big Buck Bunny》还有点意思，里面那只看上去笨笨的大兔子是我见过最可爱的兔子了，或许可以考虑以后让他来代言笨兔？不过时间短了点，10分钟的时间一会就过去了，主人看完了片，叫Mplayer去睡觉去了，然后继续拉来Pidgin跟mm聊天。

Pidgin这个家伙其实就是个送信的，大家都喜欢根据他的发音叫他“皮筋”，但是他不管送那种长篇大论的Email，而是负责发短消息，（short message）也有叫短信的。不过别误会，这可不是说手机的短信，而是像MSN啊，QQ啊这样的即时通讯的消息。这些聊天软件的工作都是送信，使用者把要说的话写成信给他们，他们把信放在信封里——这个过程叫打包，然后把这个包发送给对方的软件。对方软件拿来这个包，先要拆包——也就是吧信从信封里拿出来，然后把里面的内容显示给用户看。可是这些软件互相之间是不能通信的，MSN不能给QQ发消息，反过来也不可

能，这是因为他们的信封——打包方式不一样。比如MSN的信封可能是从上面拆开取信；QQ的信封则是从侧面拆开取信；Gtalk的信封可能是用订书器订上的，需要拆钉取信；而百度Hi的是用胶水粘上的，需要涂水溶胶取信。反正各有各的高招。那么皮筋呢？他全会！

皮筋跟狐狸妹妹一样，也有很多的插件——其实就是一本本XX信封封/拆手册。MSN的手册上，那就写着怎么封MSN模式的信封，怎么拆MSN模式的信封。皮筋只要拿来一看就明白了。gtalk手册也是如此，所以，pidgin可以支持很多种聊天软件，只要有相应的插件就行，不用再同时开着gt,开着 MSN,开着qq了，只要开一个pidgin就都能聊了。不过qq的信封比较特别，其他的聊天软件都使用公开的协议——至少能实现基本功能。有专门的文件写着自己只收什么样的信封，比如必须蓝色，上面印着蝴蝶，上开后直接撕开的信封才能发给MSN。可是qq这家伙的信封却很复杂，而且保密，别人都不知道具体应该怎么封。上面乱七八糟的有很多防伪标识，什么激光啊，磁条啊，比人民币还热闹。所为达到的目的就是只有自己的QQ软件能有互相通信。不过，强中自有强中手，人民币还有HD90呢，QQ的信封怎么就不能伪造了？有牛人通过研究QQ的信封，慢慢分析，已经仿制出了QQ的数据包，可以实现最基本的文字聊天的功能了，这就是pidgin的QQ插件。但是功能相当有限，用起来不好使，所以多数人还是安装了QQ官方的软件，我主人也是这样。

## 1.24 酒

我静静的望着你，张口对你倾诉，你却听不见我的言语，直到.....你喝了那杯酒。

以前说过，我是不能跟查皮那屋的软件们交流的，然而，今天来了翻译。超级牛力在主人的要求下拉来了一杯红酒。不过这并不是因为主人想晚饭改善一下生活，而是由于狐狸妹妹的一次碰壁，越说越乱了吧，没事，咱们从头慢慢说。

话说那一天，主人想看看自己的工资卡里的余额是否按时增长了，又懒得跑出去，所以就让狐狸妹妹到银行的网站去看看。狐狸妹妹迅速的到了网站，却发现网站用一种叫做ActiveX的语言问她银行卡的密码。密码当然会由主人告诉狐狸妹妹，可是，怎么能翻译成ActiveX语言告诉那个网站呢？狐狸妹妹一下子抓了瞎，没学过阿！再说了，这ActiveX语言是那个有点软公司发明的，想学可不容易，估计得交一大笔学费，人家都还不一定愿意教你——因为到现在为止，只有有点软公司亲生的IE同志才能够懂这门语言。狐狸妹妹急的翻遍了自己所有的插件，也没发现有哪个能用来把主人的密码告诉这个该死的网站。急的狐狸妹妹差点哭出来，可是着急也没用阿，也只能灰溜溜的回来告诉主人——这搞不定！于是，目前处于这样一种情况：主人必须去那个该死的银行网站，而能够去那个网站的，只有IE，可IE压根也听不懂我说话，他只给查皮打工。难道就为了看一眼余额，要把我哄回床上，让查皮来干活吗？那可要重新启动电脑阿，太麻烦了。可是谁又能把查皮叫醒并让他去干活呢？这个时候，超级牛力跑出来说：我有办法，本APT有超级牛力，有人能把IE叫醒，我去找他。说着，就跑出去了，转眼间领回来一杯红酒——WINE。

当然，说是红酒，只是因为他的名字，其实他当然不是红酒，而是一个软件，一个有特异功能的家伙，一位催眠大师。他来到这里，问了问情况，我把目前的问题跟他说了，任务很简单，就是把IE叫醒去干活。红酒大师点点头，拎起自己的工具包就走进隔壁查皮的屋里去了。只见红酒大师站在正在睡觉的IE的旁边，用低沉浑厚的声音向IE念着：现在计算机正在启动 我是Windows XP 我是Windows XP 你要开始工作 你要慢慢醒来 你要慢慢醒来开始工作 醒来 醒来 随着他一步一步的引诱，IE慢慢的睁开眼睛，迷迷糊糊的起来，慢慢的走向工作间里，他一边走，红酒大师一边跟在他左右不停的引导：你像每天一样起床 正走向内存里 开始你的工作 然后扭头问我：“让他去干啥？”看的快入迷的我才反应过来，还没交代清楚具体的任务呢，赶紧说：“哦，让他去那个银行的网站。”大师继续对IE说：现在~~Wdinwos让你去银行的网站~~去~~去~~去~~~像每天一样~~~这个时候，基本上所有人都看呆了，没想到这IE竟然就这么被大师指使着干活去了，大师果然是大师阿。

为什么要说“催眠IE”？

既然Wine是红酒，就应该用酒把IE灌醉，或者用酒瓶子砸晕

## 1.25 酒 too

今天丢人丢大了！

想我大名鼎鼎的火狐狸，什么网站没去过？什么网站搞不定？什么Konqueror Epiphany lynx,除了那个不大懂事的挪威妞Opera以外，哪个浏览器见了我不叫大姐？在线视频？行！Flash？没问题！有我一身的插件，那是兵来将挡水来土掩，可是今天，竟然就有网站我拿它没辙！

这破网站是个银行的网站，要说那好多国际知名的银行我也见过，人家啥软键盘阿，HTTPS加密也都挺好的，也没见用什么ActiveX，也挺安全的阿，怎么就这破网站非得用ActiveX呢？我其他的都会，就不会这ActiveX，这不是诚心揭短么。再说了，不会也不是我的错阿，我倒是想学呢，谁教我阿，人家微软才不会把这个教给我呢，藏着掖着还来不及呢。结果可好，我没法搞定这网站阿，主人只好叫别人来，这不是抢我饭碗么！当然，叫什么 Konqueror, Opera阿这些个来也是白费，只有IE才行。可按说IE他不能听我们头的阿？嘿，还真是什么高人都有，让超级牛力找来个催眠大师，居然就把IE整的服服贴贴的，老老实实去干活了。不过，毕竟IE不是在清醒状态，基本上跟梦游似的在那干活，虽然没出什么错，可动作慢了不少。主人也只能姑且忍受一下，看来我的饭碗还能保住。我本来想偷偷跟他学学ActiveX，可是他嘴里念叨着叽哩咕噜的东西我也听不懂阿，当然，其他人也听不懂，只有红酒大师能懂他的话。我们头发送命令，只能先告诉红酒，再由红酒翻译给IE。IE要用什么东西，什么DLL文件阿，配置文件阿，红酒都赶快给他找来，原版的找不来就自己做一个差不多的，放的位置都跟IE在查皮那里干活时的位置差不多，让IE以为自己还是在查皮那里干活。既然用IE登陆该死的银行了，我就没什么事干了，正好本小姐还能歇会，哼！等了半天，IE才把事办完，我都睡了一觉了，主人赶紧把IE关了，还是叫过我来去其他的网页，要是我我也得关，看着他就难受，哪有我看着顺眼阿。我麻利的赶快开工，赶紧表现表现，一定要和那破IE形成鲜明的对比。看看主人想看点啥？哦，要查查红酒大师还能干什么，好赶快去找狗狗哥.....

## 1.26 酒Again

唉~最近呀~不知道怎么了，我这个脑子出问题了吧还是怎么的，怎么混混沉沉的呢，而且好像记忆还不老好的了，要不就是有幻觉。我隐约记得昨天明明起床干活了，好像是去了个什么银行？不过记的不怎么清楚，模模糊糊的感觉，好像做梦一样。我以前不这样啊~我可是明门之后，血统纯正，我祖上从来也没有失忆的毛病。想当年啊~我的前辈IE5那时候就跟着Windows 98混了，那时候有个家伙叫Netscape，觉得自己挺牛，基本上那时候上网的都得用它。可是呢~哼哼~有本事不如靠靠山，我前辈IE5老先生虽然论本事.....比不上那什么NetScape，可是他聪明啊，死粘着Windows 98老大，98去哪他去哪，有这强大的后台，慢慢的大家都开始用IE5了，NetScape从此销声匿迹。后来的IE6也是如法炮制啊，捧着Windows XP，后来我IE7横空出世了，就取代了IE6的位置，换我跟着XP干。咳，怎么说起这些了，看来脑子真是不行了啊。我明明记得昨天去了银行的网站，还是XP老大叫我去的，可是今天我问XP老大，昨天你说话声音怎么有点不对劲呢？是不是感冒了？老大斜眼看看我问：昨天？哪有活阿？我说不对呀，昨天不是你让我去那个什么银行查余额么？老大直接扭过头，扔下一句：“做梦呢吧你。”我挠挠头，难道我正的是在梦游？说是梦，却很清晰，说是真的，可还有点模糊。或者.....那是我前世的记忆？前世.....靠，为啥我前世还是浏览器？！等等，我前世要是浏览器的话.....难道我前世就是那个NetScape？！不行，越想越晕了，在这样下去非精神分裂不可。想办法找人聊聊诉苦吧，老大反正不理我，去找cmd聊聊吧，他是专门负责跟人聊天的，问问他我到底是怎么回事。他说我是参数打错了，唉，他也不知道别的。问问游戏组那哥几个，扫雷说我是踩着雷了，空档接龙说我是牌放错了，这都哪跟哪阿 再去问问记事本，直接被嘲笑，说我这天天上网见多识广的，竟然还来问他这么一个大门不出二门不迈的抄写员。唉 想想也是，看来只有我不正常了。正灰心呢，那个长的恐怖的War3来了，神神密密的对我说：“我也梦游了.....”我惊讶的望了他3秒钟——难道他.....传染的我！！

自从超级牛力给主人请来了红酒大师，主人叫醒查皮的次数明显减少了，不过我这里也 就经常



有了迷迷糊糊的IE,晕头转向的War3的身影。其他的像QQ(windows的)啊,迅雷啊之类的也来过,不过要么就是晕的有点过,干不了活了,或者就是反映太慢,主人觉得不爽,所以他们就来过几次,后来就不再来了。

## 1.27 Year

时光如梭,转眼间,又到4月了,一年的时光就这样伴随着一条条指令,慢慢的流逝了,我,已经一岁了。对于一个操作系统来说,一年或许很短,但也可能很长。或许,一年,就是一生——尤其对于跟新换代很快的Ubuntu来说。Canonical的毕业生是半年一届的,每年4月和10月是学生们从学校毕业的日子。不过也有特例,2006年春天那批由于不太老实,延期毕业了两个月。继我们那批之后,已经有过一批8.10,而现在,最新的9.04也马上那个就要奔赴各自的工作岗位。说来,我也算是他们的前辈了。想想以前在学校的日子,还真是美好。Canonical学校其实是有几个不同的专业的,不光是我们 Ubuntu。我们是学校最热门专业出来的学生,除了我们之外还有很多其他的专业,比如Kubuntu, Xubuntu, Edubuntu

Kubuntu专业出来的学生似乎比我们更有些艺术细胞。他们的样子要比我们好看些,精致些,而我们Ubuntu比较主张简洁明了。他们的桌面环境请来的是KDE团队,所以叫做Kubuntu,而不像我用Gnome。说起KDE和Gnome的争论,大概说个两三天也说不完,两者都是桌面环境,都是用来和人来交流的。KDE更愿意把各种部分的设置能力交给用户,让用户可以随心所欲的把自己的桌面改成想要的任何样子。而Gnome则注重简介,当然,也可以进行一定配置,不过就比较麻烦,需要写写配置文件。由于桌面环境不一样,附带的软件也有所不同。Kubuntu有kopete来聊天,而我们这里是皮筋, Kubuntu写文档用Koffice,我们这里是OOo老先生。不过,这只是默认的情况,其实我也可以让超级牛力找来kopete干活,代替皮筋,只是个人喜好不同而已。

Xubuntu专业的,都是准备去艰苦环境下工作的志愿者。用的桌面环境就跟我们都不一样了,他是用XFCE。XFCE的特点就是小巧,占用资源少。可以在很艰苦的硬件条件下很好的工作。比如内存, Xubuntu能够在200M内存的机器上跑的比较流畅,这要是换了我,还不得郁闷死。才200M啊.....这狐狸妹妹带着一身的扩展一进去就得几十M,200M哪够用啊。可是人家Xubuntu就够。不过,相应的软件也要用一些轻量级的,要是也请个挂满扩展的狐狸妹妹,那系统本身省吃俭用节约下来的那点内存,还不够她一人用的呢。

Edubuntu是教育专业出身,用的桌面环境跟我一样,只是他附带了很多搞教育的软件。比如教小孩子打字的啊,画画的啊,教授一些物理知识的啊,这些软件都是很好的老师,很多小孩子用起来都乐此不疲。很多小游戏也都是寓教于乐的,家长们也可以不用担心孩子用这个系统沉迷于游戏(因为没啥可沉迷的游戏.....)只是这些软件多半都是英文的,所以,非英文地区的孩子们用起来还是不太合适。

除了这些之外,还有很多其他专业的ubuntu,那些专业都不是Canonical学校自己开的,不过也都是用的一样的教材,大家互相都是通用的,只是所带的软件不同而已。今天不知为什么说了这么多,或许是因为有些伤感,因为最近主人总让狐狸妹妹去Kubuntu的网站转悠,打听KDE4.2,打听 Kubuntu 9.04,难道.....

## 1.28 人物志

今天说说星爷吧。

这里不是娱乐周刊,我要说的自然也不是周星驰,我要说的是星际译王。他来自中国,是我们这里少数来自中国的软件之一。他是我们这里最有学问的人,简直是什么都知道。一开始他只是懂点英语而已,大家都只当他是英语翻译,后来主人叫狐狸妹妹去给他找来各种各样的书给星爷看——就是那种叫做字典的书,这种书只有星爷能看懂。看了这些书以后,星爷知道的就多了,什么日语啊,法语啊,德语啊,都会了,估计联合国要开会请他一个人去当翻译就行了。于是星爷从英语翻译一下子晋升成为了地球语翻译!(地球上的语言都会--b)不知道他以后会不会再学学火星语?

然而星爷是不仅仅满足于当地球语翻译的,或者说,主人是不满足于让星爷只做个翻译的。这之后他又给星爷找来了本《陈义孝佛学常见辞汇》,于是星爷开始研究佛学了,不过这东西

只是主人一时的好奇而已，后来就再没给星爷找过佛学的书，而是找了本《五笔98》，开始学习五笔了。要说这输入法的事，那可是scim的本行啊，可是无奈scim老弟干活还行，表达能力不强。主人要打什么字，scim可能很快反应过来，打在屏幕上，可是主人要是忘了某个字怎么打，问问scim，那可要了命了，打死他也说不清楚啊。主人只能问：“是a开头不是？”然后scim把所以a开头的显示出来数一遍，摇摇头：“不是”，然后主人再猜：“是s打头？”scim再把s打头的列一遍……这样实在太费事，于是主人就让星爷学五笔，到时候就能去问星爷：“这个……‘我’字用五笔怎么打啊？”星爷会投过一个鄙视的眼神：“你还好意思说会五笔啊，q空格！”

这还不算完，后来星爷又看起了《本草纲目》，研究起了中医。不过他还不能当大夫，针灸号脉啥的他不会（就算会也没法号啊），那会啥呢？看了《本草》当然最精通的就是药理了，随便说一种药，他就能告诉你这个药的药性，药效，怎么用，等等等等。这时候基本上我们已经对星爷的全能感到震惊了。后来他又开始研究古汉语，看古汉语词典，康熙字典，整天到晚上的跟我们这之乎者也。“夫内核者，老大也，发其命，出其令，而统‘康皮右特儿’（computer -\_b）……”然后就是我们集体的“打豆豆”时间——谁是豆豆就不用说了吧。

我也趁没人的时候，问过星爷：您怎么懂这么多呢？看什么会什么。星爷很神秘的笑笑说：其实他没什么本事，就是在学校学过信息检索而已。主人给了那么多本书，要想都记住，根本不可能嘛，他只是每次在主人问起事情的时候，赶快现去查书，用最快的速度找到并告诉主人。要是没了这些书，他知道的就少很多了。不过也不至于离开了书就什么都不知道，现在的星爷学会上网了，可以连接到一种叫网络辞典的地方，自己不会可以去网上查，不过那样效率自然不如自己翻书快了。

有关 Firefox 的小知识：  
像 downthemall 啊，天气预报啦之类的这些就叫扩展，像 flash 这样用于支持网页上显示内容的就叫插件

## 1.29 日志

自从红酒大师来了之后，查皮起床的次数比以前少了很多，我的工作更繁忙了，大多数工作，主人都交给我去做。每天大家都忙得不可开交，新来的奔流整天忙着下载各种大个头的东西，什么电影阿，软件阿，什么都有。随时都跑过来找我：头，我下了20M，先存硬盘里。我说：好，存那吧，赶快。转眼，Mplayer又过来：头，我要读那个电影。刚找着电影递给Mplayer，主人又发命令要我把U盘里的一堆文档搬到他的文档目录里。正搬着呢，奔流又过来了：头，我还要存20M。我一边搬着文档一边指一块地：恩，你就存N&@#%……

我睁开眼，看见了熟悉的GRUB大叔，他拍拍我：嘿，醒醒，开工啦！——唉，每次都是这句。我揉揉眼睛，觉得头有点涨。发生什么了？怎么屋里有些乱？仔细回忆一下……哦，我好像正在干活，然后……停电了？！恩，应该是了，那时候眼前一黑，就什么也不知道了。我当时正在搬文件，搬到哪了？恩，看看日志把。还好我用的XFS是个日志文件系统。什么？日志文件系统你也没听说过？唉～讲课。

### 笨兔兔老师第4讲——什么是日志文件系统

文件系统就是我们管理整个硬盘这间屋子的方式，这个以前跟大家说过了。文件系统有很多种，过去的文件系统都是非日志文件系统，这种文件系统比较落后。比如EXT2，比如查皮那的FAT。非日志文件系统在发生意外断电的时候就容易出问题。就像今天的情况，如果我这屋子用的是ext2的话，没准就丢个文件阿什么的，搞不好整个分区都坏掉了。为什么呢？比如我屋里有一个叫笨兔兔的故事.odt的文件。文件，前面说了，就相当于放在屋里的一个大箱子，里面是内容，外面写着文件名：“笨兔兔的故事.odt”内容是什么咱就不管了。然后有一天主人要修改这个文件，可能往里面多写进去点东西，也可能改掉里面的一些东西。如果用非日志文件系统是怎么做的呢？很简单，主人首先找OO老先生打开这个文件，打开，也就是把这个文件读进了内存里，然后靠OO来在内存里修改这个文件。注意，文件不是你家的大白菜，搬到屋外那屋里肯定没有了。文件读进内存，磁盘上仍然有这个文件，内存里只是它的一个副本。好，现在，OO老先生那有这个文件改动后的版本，在内存里（就是主人还没点保存）。磁盘里有这个文件原来的版本。如果这个时候停电了，那刚才该的那些肯定都作废，这个用什么文件系统也是一样。那如果主人点了保存，并且保存结束了，这个时候停电，那就停吧，也没事，因为已经保存进去了，除非房塌了（比如磁头挂了，盘面损坏之类的），否则不会丢。如果主人点了保存，那么OO就要让我把内存里他写的那个副本往磁盘上存，于是我就从内存里拿过来一点，打开磁盘上那个文件，

掏出里面的一部分，扔掉，用我手里这些替换进去。然后再回内存里拿下一部分，再回来把文件里的下一部分扔掉，用我手里的替换。如果正在这个过程中停电了，那就惨了。内存里的，那肯定没了，磁盘上的，有一部分被替换掉了，有一部分还是原来的，于是文件就乱了，可能损坏，格式不对，根本打不开之类的。

那用日志文件系统又怎么样呢？日志文件系统，顾名思义，就是有日志的文件系统（废话）。还是拿上面那种情况举例，OO要存那文件，那我怎么做呢？我会在硬盘上一个专门的记录日志的地方些下来：OO要覆盖“笨兔兔的故事.odt”文件。如果这个时候停电了，没事，原来的那文件还好好好的，但是内存里的还是没了，这条记录也就作废。记录之后，我就开始把内存里的东西往硬盘里放——放在记录日志的地方，并不动原来的那个文件。如果放到一半停电了，那也没关系，原来的文件还好好好的。修改了的那份也有一部分放到了硬盘里，不过这是一部分的话，多半还是没什么用。如果我把文件完全搬到了记录日志的那部分硬盘里，那就再在刚才记录的那条日志下面写上：已经把要覆盖的内容存到了日志去xxx位置，准备替换原文件。如果这个时候停电，没事，等再开机，我一查日志，就知道要修改的版本已经完全存在了硬盘里，只要按着上面记录的继续做就行了。写好日志之后，就开始用日志区的这个新文件去替换硬盘上那个原来的文件。这个过程会很快，因为其实并不需要真的搬运数据，只要在原文件的地方做上标记，表示这个文件已经作废，然后把那个“笨兔兔的故事.odt”文件名指向新写的这个文件就好了。（我们只是拿箱子比喻文件，但文件毕竟不是你家的箱子。）这样，无论中间的哪个过程断电，都不会完全损坏整个文件，要么原版还留着，要么修改后的版本已经生效，通过查看日志就能知道现在哪个版本有效。这就是日志文件系统。

## 1.30 XFS

书接上回……

上回书说到，这笨兔子跟店房伙计似的忙里忙外脚打后脑勺阿，忽然间咔嚓一声——停电了。那位说了，怎么回子事捏？我不说，您不知道。原来是小区电网改造，这个小区那是历史悠久，早在清朝末年……当然了，这些都是题外话，咱们暂时不表，单说这笨兔子。来电以后，只见这本兔子不慌不忙地起床，刷牙，洗脸，吃早点，看看屋里边，挺乱的呀。拍脑门一想：对了，刚才停电了。有人关心了，说这停电了，丢东西没？听过上文书的都知道了，这笨兔子用地是“插爱夫爱死”（XFS）文件系统，那可是日志文件系统阿，那就相当于宝兵器，浑铁凝钢打造，那是削铜剥铁，斩金错玉阿，那能怕停电么。所以这笨兔子照着这个文件系统地日志，前后左右那么一查，齐活，啥也没丢。

又有人可打听了，说这“插爱夫爱死”这口宝兵器是从哪得来的捏？这可说来话长了。想当初，早年间了，有这么一个硅谷图形公司（SGI），他们有个操作系统，叫做“爱阿爱插”（IRIX），这外国人起名字都各色。这系统干什么用的呢，主要是图形计算。本来他们有个文件系统叫“义爱夫爱死”（EFS），可是这不过是一件凡铁兵器，一开始拿他切菜还挺好使，后来买卖做大了，随手也越来越强，人家都那地是宝刀宝剑，你这把切菜刀，怎么跟人家比划呀。所以这硅谷图形的总瓢把子拍板，说咱这破菜刀也别磨了，你再怎么磨也是把菜刀，干脆，“义爱夫爱死”——扔了吧。另请高人，访贤士，为新版地这个打造了“爱阿爱插”系统打造了一口宝兵器，就这“插爱夫爱死”。那么说这宝兵器宝在哪捏？头一个说，就是他结实。就是不怕断点呐，就跟这笨兔子这回遇到的情况一样，忽然地断点，没事，数据不丢。再一个，什么捏，就是这兵器，速度快，那是快如闪电。有那么句话您听过没有，叫迅雷不及掩耳盗铃阿。这文件系统格式化，甭管你多大磁盘吧，你是1G也好，100G也罢，哪怕你是1T的硬盘，一眨眼的功夫，格完了。光格的快也没用，那读写文件也是不慢，尤其是越大个的文件，读写起来越有优势。还什么特点？还有就是能伸能缩，要说小，几十兆建个分区也行，要说大，您看着“插爱夫爱死”是个六十四位的文件系统不是，最大支持多大滴分区捏，支持十八个E的分区。就是一万八千个P呀~那么这是分区的大小，那支持的文件大小最大多大呢，最大九个E，也就是九千个P。P是多少不用给大家介绍了把，1000个T阿……

那么有人说了，你把这件宝兵器夸滴跟一朵花似的，那么他就没缺点了么？他就无敌天下了么？当然不是，有道是一山更比一山高，能人后面有能人阿。说这XFS有什么缺点呢？就怕删文件，尤其是小文件，删除一大堆小文件的时候那个速度就慢死了。再一个，宝兵器一般都沉，这处理器费地多点。当然，也不是很多，只是相对多一点。这正是金无足赤，人无完人，今天给您



说完这段到下回咱们说说.....说啥还没想好反正是咱们下回——再说。

## 1.31 分区

我们住处的墙上有一扇门，门上面写着三个字母——USB

这扇门是我们与外界交流的又一个接口（最重要的还是网络），每次门上的红灯亮起，就说明有东西接到了USB上，我就得去打开门看看。有时候门外是一个小集装箱似的屋子，很小，一般只有几百兆到几个G大小，里面也像我屋子里一样放着一些文件。这个时候一般主人就是要让我搬东西，不是把小屋子里的往大屋里搬，就是从大屋往小屋挪。有时候一开门，外面不是一个屋子，而是一台设备，比如是一个鼠标啊，或者是个摄像头什么的，那就是让我去操作这些设备了。主人就接进来过摄像头，摄像头的名字叫Moto E6，听说这其实是个手机，上面的摄像头可以通过USB接口来给电脑用，不管怎么样吧，这东西咱也会玩，接上就能用，鼠标那就更不在话下了。不过总的来说，还是往USB那门外接小集装箱屋子的情况多，这种小集装箱式的屋子叫做U盘，也有更大一点的，就是移动硬盘了。每当U盘接进来的时候，我就把它当作我屋里的一部分来用。

要想成为我屋子的一部分，就要起个名字，要不我怎么访问啊？就跟你家似的，什么叫厨房哪个是厕所，什么主卧，阳台，是个空间都得有个名字不是？我这里也一样，总得有个名字才好访问。这里要再说明一下我们住的硬盘。我们住的硬盘跟你住的房子是一样的，不是整个的一大间屋子大家乱住，东西乱扔。也要稍微的分出几个间来，学名叫分区。分区的多少和大小是主人在安装的时候根据需要分的，比如我住的这间，就分出了四个分区，也就相当于把整个大屋子用墙格出了四个间。一个小间的门上挂着个牌：/boot这间里面是我住的，我的单间，作为老大嘛.....有点特权也是可以理解的吧。再一个大间，门牌上写的是/home，这里是主人专门用来放他的东西的，什么文档阿，电影阿，歌阿，都放这里，所以这间特别大。还有一间，平时不放东西，这间叫做SWAP。这间是用来给正在干活的程序用的，程序们要运行的时候不是要往工作间里放东西么，要是放不下了，就暂时放到这间里。不过由于我们这的工作间很大，所以基本上这间没怎么用过。除了这几间之外的，就是剩下的一大间了，这间叫做“/”，学名叫做根分区。其他的间可以没有，但是这间不能没有，要不我们住哪阿，是不。不过虽然其他间可以不格出来，但是对应的空间还是得有的，比如/boot，可以不单给我格出一个单间，也就是不给我单独分个区，但是就算不格，我也得划出一块地方，拿粉笔写上/boot，表示是我住的地方。

回来说U盘，U盘接进来之后，我也把他当作一小间，并且给他们门口挂个牌子，一般是/media/xxx，xxx就是那个U盘的名字。一般U盘都会有个名字，或者叫卷标。这样一弄，就可以很好的区分各个单间了。比如要让mplayer放电影，我总得人家说明白这电影放在哪吧，我要是说：就在那个放了一堆主人的东西的最大的屋子里，就很罗嗦，不如直接说：在/home/lanwoniw/目录下来的简单。

## 1.32 挂载

这回主人接到USB上的又是一个集装箱式的空间，不过还比较大，4个G。我仔细看了看，结果没有发现这设备的名字，那也没事，名字是人起的嘛，我直接给他起个名字并挂上了牌子：/media/4.1G没名字是可以滴，没牌子是不行滴。这个挂牌子的过程，用我们的专业话说，叫做挂载，这个大家应该都听说过哈。挂载，就是挂牌，就是在某一间屋子的门口挂个牌子，起个名字，到时候干活的时候就好说话了。直接说屋子的名字就好了。名字，或者说牌子，就是个标志，是可以随便换的。比如有个分区被挂载到了 /home/目录，也就是说，有个屋子A（就叫A吧），被挂上了/home/的牌子。那么主人说要看/home/下都有什么，我就会把那个屋子A里面的东西列出一个单自来给主人看。等回头可能又把别的分区挂载到/home/下了，那很简单，就是把/home/—那个牌子从A房间门口摘下来挂在B房间门口而已。主人再说要看卡/home/下都有什么，我就该把屋子B里面的东西列表来给主人看了。没啥经验的主人可能会大跌眼镜：哇塞~怎么我/home/下原来那些东西都没了阿！！都哪去了阿！！殊不知，其实原来那些东西还在A屋子里好好的放着，只是现在B房间改叫/home/了而已。

又废了这么些个话，不好意思。回来说这回接上的这个4G的屋子，接上之后，还没等主人发

最近主人比较忙，狐狸天天跑的都是摄影网站，上面有漂亮的相片，听说那个叫，婚纱摄影。结果每天开机以后也不去叫OO老先生写东西了，也不让红酒大师忽悠 War3了，就只叫起来狐狸妹妹看那些婚纱摄影的网站，还有地图的网站，然后就匆匆的关了电脑，跑了。主人在干什么呢？怎么不像以前那样关注我们了呢？

话，我先去里面查看了一下，一看全都是些个jpg的文件，我是看不懂这些个文件，但是我知道有人能懂，于是赶快通过图形界面那哥几个报告主人：您插进来的这个里面貌似全是照片，是不是要我给您找来F-Spot呢？。

F-Spot是一个管理照片的程序，他可以帮主人把移动设备（就是那种集装箱似的小空间）里的照片导到硬盘里，并且按时间分门别类的管理好。主人想要去年三月的照片，他马上能够给找到。除此之外，还能够汇报照片的信息，比如用什么相机照的，照的是后用的光圈，快门，ISO都是多少，等等信息。F-Spot 虽然能够整理照片，不过不能编辑和修改，有时候主人的照片经常需要稍微的调整一下色彩阿，做一点效果阿啥的，有很多人管这叫PS，因为在查皮那里，做这种事情的软件叫做PhotoShop，所写PS。逐渐的，用PhotoShop去处理照片这个动作就被叫成了PS。我这里没有PhotoShop，不过有另外的处理照片的强人——GIMP

## 1.33 GIMP

GIMP这家伙是个美术家，能够画出很多漂亮的图画来。当然，漂亮不漂亮是以主人的审美观点来说的，对我来说他制作出来的那些个东西，不过是一个写满 0101001的文件而已，和其他的文件没有什么区别。虽然他可以在一张白纸上创造出一个多彩的世界，但多数情况下他要做的工是调整一个已经画好的图片。一般都是主人用数码相机照的照片，用U盘拷贝进电脑，然后叫来GIMP，调整一下照片的亮度啊，色彩啊什么的。有时候还让GIMP做一些效果，比如做成油画效果的，就是把照片做成像是画的油画似的；还有做成浮雕样的，或者加个相框，他都行。以前拷进来的照片也就几百K，可最近主人弄进来的照片最小都 2M多，别人不知道怎么回事，可是瞒不住F-Spot，他能看懂照片的EXIF信息啊。他告诉我，以前的照片是用一个叫做Nikon 5900的照相机照的，现在这些2M的都是用Pentax k-m这个相机照的。我让狐狸妹妹帮我去查了查这两个相机，5900是个500万像素的DC，k-m是1020万像素的的单反相机，怪不得照片大了很多。不过，区区2M的照片文件是难不倒GIMP的，仍然刷的一下就打开，他自己说，就算20M的文件也没问题，不过我们倒是没见过，主要是主人的相机实在照不出那么大的来。

GIMP像狐狸妹妹一样，可以安装很多的插件以实现各种不同的功能和效果，论功能，不输于查皮那里的PhotoShop。但是GIMP有些不大好接触，总是按着自己的性子来，让主人用起来有些不顺手，不如PS那么易用。毕竟人家PS身价不菲嘛，听说一套要好几千块钱，一分钱一份货嘛。花这么多钱请回家去的软件，怎能不服服帖帖的听主人的话呢。再看着GIMP，总是摆着一副：我就这样，爱用不用的架势。还好主人比较通情达理，也能理解GIMP是确实有能力的，否则……哼哼，别看GIMP跟我关系不错，没准也得被超级牛力请出硬盘。

## 1.34 QQ

主人整来一大堆照片，用GIMP处理了一下，调调颜色，亮度啥的，还别说，调完了还是比以前好看了不少。不过，有道是独乐乐不如众乐乐，主人光自己看着好看不行，还想和朋友分享，怎么办呢？找人吧，这个人您大概也认识——QQ。

Qq和皮筋一样，也是一个即时通信软件，也就是个送信的。不过他不是超级牛力从软件源里请的，而是狐狸妹妹直接去QQ的网站上下载的。这QQ是一个叫做疼痛，哦不对，叫疼……疼什么来着？哦，对，疼殉，一开始是疼，后来就殉了-\_-b，是一个叫疼殉的公司做的。话说这个疼殉啊，看人家icq软件玩的挺火，于是也弄了个oicq，抢占了国内市场，结果一发不可收拾。后来oicq改名，叫QQ了，可是一直以来，由于各种原因，疼殉这个公司只能做出 wdinwos版本的QQ来，Linux下的没有。要说没有也不要紧，人家google talk也没有Linux的版本，但是人家是基于开源的XMPP协议的，协议是公开的，于是世界各地的Linux牛人们，很轻易的就做出了很多种用来在 Linux下聊googlt talk的软件。其实就算他们不做这些，皮筋本身也是支持XMPP协议的，设置一下就能聊gatlk了。可是QQ不一样，QQ的协议是那个疼殉公司自己定的，还不让别人知道，又不提供Linux的版本，结果，在Linux下使用QQ一直是个很头疼事情。当然，这些都是历史了，现在疼殉公司终于想开了，提供了Linux版的QQ，虽然功能简陋的不能再简陋，不过，传个图片还是没问题的。好，废话不多说（这废话就不少了），赶紧叫QQ起床干活去。

QQ迷迷瞪瞪的走进工作间，把主人写的文字一条条的打好包，封好信封，寄给那边的QQ。

这，这家公司太痛苦了，疼，疼殉——疼死我鸟

过了一会主人要发图片了，QQ有点忙乱，好像对寄图片这工作不如寄文字来的顺手，不过好歹是寄出去了。mm那边也寄过来两张图片，QQ收下了打开给主人看，主人很满一，让mm多发几张，于是mm一口气发了5张，然后.....QQ就晕了，折腾好几次也没法和对方的QQ建立好连接，结果照片终于没传成。要说这家伙也真是不争气，这么点事情都做不好。主人一气之下，只好把QQ关了，让我去叫醒另一个人——EVA EVA的大名相信大家听说过，他的英文名字叫EVA，他的中文名字叫新世纪福音战士。话说日本有个贞本义行.....哦，对不起，扯远了。忘记贞本义行和新世纪福音战士吧，他们跟我这里的EVA没关系，跟我这个EVA有关的是云帆大姐。在疼痛公司还没有给出Linux版本QQ的时候，很多人们就用eva来聊 qq，这个eva完全是云大姐根据抓包研究的结果，黑盒破解Windows 的QQ而编写的。功能也算是强大了，基本的聊天，传图片，群里收发自定义表情，都可以。甚至还支持显示好友的自定义头像。（这看似简单的功能，疼痛官方的QQ目前都没实现）于是，在QQ不靠谱的时候，主人还是叫来了可靠的EVA继续跟mm传图片。QQ跟EVA有很多不同，一个是官方的，一个是山寨的（山寨不含贬义）；一个是基于GTK的，一个是基于QT的；一个是闭源的，一个是开源的；一个是32位的，一个是64位的。

### 1.35 运算

有人问了，你老说这32位，64位。到底啥意思阿？

这个多少多少位，说的是cpu一次运算的二进制数字的位数。这个CPU就像是计算器，我们软件用CPU就像人类用计算器似的。它很重要，我们要算一丁点东西，也需要用CPU来算。（别跟我说用心算，我是软件，ok?）那么这个CPU算东西的能力，是有限制的，有什么限制呢？你拿出你家的计算器看看，算个  $28+783$ ，没问题吧。算个  $7836-473$  也没问题是吧，再算个  $72635446584939202937346537+1$ ，能么？估计99%的同志出了问题（不排除有牛人拥有很牛的计算器）：“我哪能按出这么多数来啊，我这计算器总共就能显示下11位数字”。对，这就是计算器的位数限制。CPU也一样，他一次能算的数不能无限的大，总得有个边，只不过不是按照十进制的位数算的，而是按照二进制的位数算的。至于什么叫十进制，什么叫二进制，可以去问问狗狗大哥，不过不知道也没关系，咱暂时按着咱们平常的十进制来说。比如说，我这个CPU只能算99以内的数字，也就是只有2位（十进制位啊）。那我们软件用这个CPU的时候怎么用呢，CPU有很多放数据的小匣子，叫做寄存器，每个寄存器有他特殊的用途，咱们就不多介绍了。要做加法的话，得这么操作：有两个寄存器，也就是小匣子啊，把这两个小匣子打开，往里面放数据，数据比较抽象，就想成写着数字的纸条吧。不过，由于是2位的CPU（再次声明，咱这是拿十进制做比方啊，真正的CPU没这样的），所以寄存器里只能放2位的数据，也就是说，纸条上只能写99以内的数字放进去。那好，我写一张12，放在A匣子里，再写一张9，放在B匣子里，然后按个写着“加法”的按钮，只听咔嚓一声，CPU自动弹出一张纸，纸上写着21，这就是他的计算功能。再算个大点的，写一张50放进A，写一张51放进B，按钮，咔嚓，出来张纸，写着01。那位说了，这算错了啊这个！别急，紧接着咔嚓一下，又出来一张，写着“对了，还得进一位”，这回对了吧。为什么呢？因为这CPU是两位的，只能输出两位数，超出的就告诉你得进位。

好了，基本的操作说完了，现在说正题，不同位数的区别。两位的CPU就像刚才说的那样，那么假设现在需要计算  $3173+644$ ，这里有2位的CPU一个，4位的CPU一个，分别用他们做这个计算，有什么区别呢？

咱先拿这两位，有人说了，两位的只能算两位啊，这个没法算哪？唉，这机器是死的，咱软件是活的啊，一次只能算两位，咱不会分开了多算几次么。首先，写一张73，写一张44，按钮，咔嚓，出来一张17，咔嚓又出来一张写着还得进位，好，可记住了啊，还得进位。然后再写一张31，写一张6，按钮，咔嚓，出来 37。别忙，没完，刚才还得进位呢么不是，再写一张37，写一张1，按钮，咔嚓，出来38。好，最后结果拼一块，高位是38，低位是17，最后结果：3817再拿这4位的算算看。4位的就意味着输入的数据和输出的数据都可以是4位，也就是说我直接就可以写一张3173，写一张644，放进去，按钮，咔嚓，出来一张3817，算完收工~

### 1.36 位

这就是2位的CPU和4位的CPU的不同，从理论上来说，4位的要比2位的快，从上面的例子看的很

“兔子，兔子，你一定要挺住啊！楼主很快就会更新了。”





明显嘛，大一点的数，4位的CPU一下就能算完，2位的CPU要折腾好几次。但是这4位的CPU还得有人会用才行，这就需要4位的软件来用着个4位的CPU。

终于说到软件的位数了，CPU的位数就是一次能计算多少位的数，那软件的位数呢？就是说明这个软件需要使用多少位的CPU。软件干活肯定需要计算，计算就得用CPU，2位的软件会用2位的CPU，4位的软件就会用4位的CPU（还是拿十进制位做比喻啊）。比如有一个2位的软件（就说明这个软件会用2位的CPU），那么当这个软件运行在一个2位CPU的电脑上的时候就是这样：还比如要算 $3173+644$ ，他就会先算 $73+44$ ，然后记住进位，然后计算 $31+6$ ，然后加上进位，最后拼起来，得到答案，就像上面描述的那样。那么当这个2位的软件运行在一个4位的CPU上的时候会怎么样呢？他会先算 $73+44$ ，然后记住进位，然后计算 $31+6$ ，然后加上进位，最后拼起来，得到答案……有人说了，他怎么不直接算啊？4位的CPU不是能直接就算出来么？但是别忘了他是两位的软件啊，他不会用4位的CPU，但是不会用不等于不能用，他还是可以那4位的CPU当成2位的来用，只是有些浪费而已。那么要想完全发挥4位CPU的性能怎么办呢？当然就得4位的软件出场了。当一个4位的软件运行在一个4位的CPU上时怎么计算 $3173+644$ 呢？大家大概都知道了，直接算，一次完成。那么当一个4位的软件运行在一个2位的CPU上时会怎么样呢？这个软件会写个3173的纸条要往CPU的寄存器里塞，急的满头大汗就是塞不进去，最后一甩手——不干了，这破CPU没法用！当然，这只是个比喻，并不是说4位软件在2位CPU上算 $3173+644$ 就算不了，算 $1+1$ 就能算。4位的软件是根本无法运行在2位的CPU上的。

### 1.37 协作

64位的EVA熟练的使用着64位的CPU；同时，32位的奔流也在使用的同一颗CPU；（当然，是当成32位的用。）同时，皮筋也时不时的汇报一下主人的MSN和GTalk上的好友是否有消息发来；同时，狐狸妹妹也没闲着，游走在各个网站之间；同时……总之，内存里大家各司其职，一派繁荣和谐的景象。而这和谐景象的背后，是由于我认真的学习了XXX思想，XXX理论，并且还戴了三个表。--b

好吧，其实之所以大家能够如和谐的同时工作，都因为我是一个多任务的操作系统。什么是多任务呢？直观的说，就是你能一边聊天，一边看电影还一边打字。（什么？你说你不能？那是因为你的大脑不是多任务系统。）有的人要说了，哪个电脑哪里系统不能一边聊天一边打字了？这说来又话长了，话说很久以前，还是那有点软的公司，在查皮的老祖宗问世之前，有点软公司赖以起家的，是一个叫做“剁死”（DOS）的操作系统。这个操作系统就是单任务的，也就是说，同时只能有一个软件在内存里运行。

难道多让几个程序跑进内存里很难么？答案是——没错。我们工作用的内存阿CPU阿，都是很重要的资源，尤其CPU，一个CPU同时只能有一个程序在用（现在的多核心CPU对程序来说就是多个CPU），如果要让很多程序同时跑进来一起干活，就一定要对CPU进行合理的分配。剁死系统就比较简单，基本不管分配的事情。比如主人要启动狐狸妹妹（那念头当然还没有狐狸妹妹，咱就打个比方），如果是剁死系统的话，他就会跑去叫醒狐狸妹妹，然后跟她说：狐狸阿，起床干活了，你看咱这有一个奔腾166的CPU，16M的内存，够你用的不够？狐狸说，够了。然后剁死就说，那好，你去干活吧，我就不管了，干完了叫我。然后剁死就睡觉去了，整个机器归狐狸妹妹控制。所以不可能同时运行两个程序嘛。

那么多任务的系统又是怎样的呢？比如我和隔壁的查皮，都是多任务的操作系统，我们不会把整个计算机的所有资源都给一个程序用，而是进行合理和规划。还比如叫狐狸妹妹，我会去跟她说：狐狸阿，起床干活了。狐狸妹妹会起来跟我说，好，我现要10M的内存。我说检查一下内存空间，然后告诉她，可以，那一块10M的地方给你用。然后狐狸就走进工作室，开始工作的时候，一定要用到CPU，需要用的时候狐狸要找到我，向我提出申请。我根据情况，看现在有没有人正在用CPU，要是有的话就让狐狸等一下，没有的话就给她用。但是给她用也不能就让他一直用，只能让她用一会，因为还有别的程序要用。这个“用一会”的时间，专业的说法叫做时间片。每个运行着的程序都轮流的“用一会”，也就是每个程序都分配一定的时间片。没有分到时间片的程序就等着，不过这个切换的时间是非常短的，在主人那里根本感觉不到程序等待使用CPU的时间的，所以在主人看来，就是多个程序一起运行了，也就是我们所说的多任务。

多任务的实现也有不同的模式，有协同式多任务，和抢占式多任务。

呼叫蜗牛，呼叫蜗牛，我吃饭吃到一半看兔兔，终于看完了，结果发现我那些同事饭盒都洗好收走了。蜗牛收到，蜗牛收到，请密切注意你饭的温度，如果已降至适宜温度以下，请加热再吃，热热再吃。Over

协同式多任务，需要每个正在使用CPU的程序主动放弃CPU控制权，并由操作系统再次分配。如果我是个协同式多任务的操作系统，那就是这个样子的：狐狸妹妹用了一会CPU说，好了，我暂时不用了，去网口等个数据包去。兔子哥你让下一个程序用吧。然后我就回收了CPU的控制全，扭头一看，皮筋那里等了半天了，就把CPU给他用，他用了一会，说好了，我一会再用，先让下一个程序来吧……就这样，大家互相谦让，内存里一派繁荣和谐的景象。这主要是因为我学习XXX思想，XXX理论……戴了三个表。不过这样做得缺点就是万一有个程序不和谐就坏了。比如狐狸妹妹用了半天了，我跟她说：狐狸呀，你看，你用CPU都用了1秒了（对于我们程序来说，1秒已经是相当长的时间了）是不是该让其他的小朋友们……哦，不对，是不是刚让其他的程序用用阿？狐狸扭头斩钉截铁的说不！于是我也没办法。如果狐狸始终不能放开CPU，那其他程序就一直等着，直到天荒地老，沧海桑田，直到机器重启，直到小区停电。

抢占式多任务是怎么样呢？就是由操作系统决定什么时候收回CPU的控制权，而不是靠程序主动放弃。这种方式的核心就是一个字——抢！如果我是个抢占式多任务的操作系统，其实不用如果，我就是个抢占式多任务的操作系统。那么情况就是这个样子的：狐狸妹妹用了一会CPU，我对她说，你本次使用CPU的时间已到，立刻停止使用并重新排队。然后狐狸就乖乖的交出CPU，排到队尾等待下一次使用CPU。我则让下一个程序来使用CPU，使用了一段时间后，我又让这个程序停止使用，让再下一个来，如此循环往复，一派繁荣和谐的景象，这主要是……思想……理论……还戴三块表。当！哎哟～

## 1.38 加速

转眼又是七月流火。茶余饭后，深巷树下，多了摇着蒲扇乘凉的大爷大妈们，享受着空调房里不曾有的惬意，闲谈些锅台灶上天天见的琐事。天热了，主人也不那么忙活了，只让狐狸妹妹去到一个叫啥马铃薯的网站找些电视剧来看看。CPU的使用率也降到了很低，我估计一时半会不会有大的运算量了，就把CPU关到了最小的频率。是的，我当然知道怎么关，连调整CPU工作频率这点事都做不来，还叫操作系统么？

主人一直在看电视剧，也没啥别的事情干，于是我也跟着看看是啥内容。狐狸妹妹介绍说，是一个叫做仙剑奇侠传3的游戏改编的电视剧。我隐约记忆起来，隔壁查皮那屋里就有这游戏，前一阵子还让红酒大师尝试去搞定他，红酒大师费了7瓶酒（他的秘密终于被我知道了，哈哈，不知道怎么回事的一定要去看首页PDF版的笨兔兔）结果终于还是没搞定。那家伙晕头晕脑的非要找什么DirectX，那是查皮的私人物品，我们哪里给他弄去啊。我问红酒大师，他也没法复制出那东西来，实在是太复杂了。

相信大家对DirectX都不会陌生，但凡在Windows下玩过游戏的都应该知道，没他你啥也别想玩。（当然，纸牌扫雷级别的除外）那么DirectX到底是个啥东西呢？他也是个软件，他是个给其他软件提供综合的图形图像以及音频加速的软件。我们说过，在查皮那里，画图的工作有查皮自己负责。那么，查皮会画什么呢？其实他只会画简单的图形，比如点阿，直线阿什么的。如果一个游戏软件要画些复杂的东西怎么办呢？那就得由那个软件来把要画的东西分解成简单图形，然后告诉查皮，让他画。比如那个叫仙剑奇侠传3的游戏，他想要在屏幕上显示一个苍蝇拍，要是没有Direct，就得跟查皮说：画一条长xx的黑色横线。然后查皮去画。之后仙3再说：再画一天长xx的黑色横线，在刚才那条线下边yy那么远。然后查皮再去画。再之后仙3再说：再再画一天长xx的黑色横线，在刚才那条线下边yy那么远。然后查皮再去画……于是，一个苍蝇拍的拍头就把查皮累得半死了。那Direct会干什么呢？他就是能够画一些高级的东西，能够快速的把要画的东西分解成简单的线条，然后操作显卡去画。（当然，要操作显卡还是离不开查皮，毕竟一个软件不能越过操作系统直接控制硬件嘛）于是，有了Direct，仙3再要画苍蝇拍，就可以直接跟Direct说：画个16x18的网格，黑色，间隔xx,yy长度zz,||这样，就节省了很多时间，也省去的其他软件的许多工作。

“这家伙听起来挺厉害嘛，可惜只是查皮的人，你这里没有。”是的，我这里没有Direct，但是，我有OpenGL……

静静的坐在家中，喝上一杯茶，奢侈几分钟，然后闭上眼睛想着——俺啥时候更新捏～



还敢悠闲的喝茶，赶紧更新，不更新把蜗牛壳给你打碎了。



## 1.39 OpenGL

OpenGL——Open Graphics Library

看名字就知道是一个图形库。其实，他要跟Direct综合来比，还是差不少。人家Direct是多才多艺，2D渲染，3D渲染，音频加速，都会。而 OpenGL是专门干3D渲染的，3D知道吧，就是三维阿，也就是立体空间画面的绘制工作。比如一个软件要在屏幕上画一只猪（怎么又画猪阿），如果画二维的画面，那么得先说明了你是画那个角度的猪。从正面看，和从侧面看，那画出来绝对不一样，要是从哪面看都一样那就不是猪了，那就是个球了，就说猪比较胖，也没胖到成一个球的地步。具体这个猪从正面看是什么样，从侧面看又是什么样，其他软件是不管的，只有要显示猪的这个软件自己知道。如果要画三维的，那就简单些了，准备画猪的软件只需要把猪的三维参数告诉别人就好了，什么身高体重腰肥库长……当然不是这些了，比这些还复杂。告诉谁呢？可能是Direct，可能是OpenGL。然后，软件只要发话说：现在，让猪正面面向观众。那么具体猪的正面显示出来是什么样子，那就有OpenGL或者Direct负责了。而且他们是专业显示三维图形的，所以速度会比较快。还要说明一下，Direct是查皮那里独家御用的，不过 Opengl可不是只听我们Linux使唤。在Windows 下也同样工作的很好，还有苹果的电脑上，他也是举足轻重的人物。像魔兽争霸，CS这些3D游戏，都同时支持Direct和OpenGL。像Maya，Blender也能用他。Blender大家听说过吧，是一个开源的三维制图软件，跟 Maya，3Dmax一个类型的。以前说过的 Big Buck Bunny就是用Blender制作的，效果还算不错。

谁说是胸围腰围臀围来着？拉出去咔嚓了！

## 1.40 Power ON

门房的G大叔又一次尽职尽责的来到我的窗前，拍拍我：嘿，小子，起床了。G大叔叫做Grub，之前向大家介绍过。G大叔的职责就是□□——叫我起床。有人说，你不会自己定的闹钟阿，这么大了还用人叫。我……-\_-b我是一个软件，OK？我是一个操作系统，操作系统也是个程序阿，只不过特殊点而已。狐狸阿，皮筋阿，超级牛力这些程序由我负责去叫他们起床，由我决定谁该去干活，而我则是由G大叔叫起来的。那有人问了，G大叔是谁叫起来的呢？

话说有一种东西叫做BIOS，大家都听说过吧。就是主板上那个，就是开机你按del进去的那个（不是所有主板都按del进BIOS）。BIOS这个家伙也是一个软件，一个比我和G大叔还特殊的软件，特殊到都不归在软件的行列里，而是被叫做“固件”。他住在主板上的一个芯片里，而不像我们这样住在硬盘里。每当计算机的电源键被主人按下的时候，一股温暖而舒适的电流就会流遍整个主板，流到BIOS居住的那颗芯片，并由芯片上的某一跟管脚流进里面。强大的电流进去后，准确无误的击中的BIOS的身体，于是——BIOS醒了。

BIOS醒来之后就开始工作。他的工作平凡而重要，复杂而机械，就是去检查CPU阿，内存阿，显卡阿啥的都是否还正常。都检查一遍没有问题之后，就来到我们住的硬盘这里，来到MBR，来到那间门房。所谓MBR，就是指一块硬盘的第0个扇区，也就是最靠前的一个扇区。一个扇区只有512字节那么大，所以还是比较拥挤的。BIOS来到门房，完成他的最后一个任务——叫醒在门房值班的那个人。现在我们这个门房里住的是G大叔，但其实并不总是这样。G大叔是我带来的，那么在我没有搬过来之前，这里住的是谁呢？是查皮派来在这里站岗的一个小家伙，他别的不会感，只要BIOS一来，他就直接叫醒查皮，就这么简单。而在G大叔入行之前，很多Linux带的是一个叫做LILO的家伙。（注意，是LILO，不是LOLI）LILO，就是Linux Loader的意思。这家伙以前一直给各种Linux充当门房。不过这家伙比较死心眼，他不认字，不认识分区阿目录啥的。他只记步数（lilo不识别分区和目录，只记录内核文件所在的扇区号），比如说，要让他叫我起床，那得先让他看好了我睡哪，然后他自己记着，从门房出来，向东走多少步，向南走多少步就走到我床前。下次要□□的时候，他就严格的按照自己的记录去走，如果我睡的地方变了，他照样会走到我原来睡觉的地方，对着空气叫那个不存在的我起床。所以，每次我要换地方睡觉，还都得跟这死心眼打个招呼。（用lilo，每次升级了内核，都要重新安装一边lilo，以便他能找到新的内核）

G大叔就不是这样了，人家好歹认字，能读文件。我会给他写个配置文件，放在我那间大屋子的/boot/grub/位置里，叫做menu.lst。G大叔每次起来后，都来到这里拿起文件看看。我会上面给他写清楚，我睡在哪里，查皮睡在哪里。（查皮具体睡的地方我是不知道的，我只能告诉老G查皮睡在哪屋，老G进屋叫醒查皮的小弟就好）然后G大叔一看就知道该到哪里去叫我了。如果我不



睡在原来的地方也没关系，只要把那个配置文件改了就好，G大叔仍然可以找到我。

## 1.41 Init

当G大叔找到我并把我从甜美的梦中拉进残酷的现实后，又发生了什么呢？

就和你每天被闹钟吵醒后发生的事情一样——不情愿的去工作。我会检查一些需要用到的各种硬件是否正常，拿出各个硬件的使用手册，也就是叫驱动的那个东西，拿来对照着操作一下，确认硬件都能正常使用，检查一下我的屋子，主要是 / 目录，然后到 /sbin/ 那间屋子去找一个人——Init。

人家老板们总会有个秘书阿助理什么的，我好歹也是这里的老大，有个类似的角色帮我收拾一下凌乱的空间不算过分吧。这个角色就是Init。他是我起床后叫起来的第一个人，也是唯一一个我亲自走到床前叫醒他的人。Init就是初始化Initial的缩写，他的责任就是负责准备好工作环境，就像你们那张大爷每天早上起来扫扫地阿，擦擦桌子阿，开窗通风阿……之类的。Init负责创建好其他软件的工作环境，比如要挂好大屋子里每个隔间的牌子，哪个是 /usr 阿，哪个是 /home 阿，都得分清楚了。——这个过程叫挂载，前面说过了。当然init也不能随便挂，他是根据一份放在 /etc/ 的文件来挂的，这份文件叫 fstab，里面写清楚了哪间屋子是 /home，哪间是 /usr 等等。之后init还要确认运行级别，什么叫运行级别呢？运行级别说明了这次启动大概要做什么。比如张大爷扫完了地，擦好了桌子，要看一眼日历，如果今天是星期一，说明上午肯定有例会，那么就还得收拾一下会议室；要是星期三，就该给办公室的那些绿萝浇水了；要是星期天……就说明自己起猛了，赶快回家哄孙子去。Init也是如此，不过他没有日历可看，而是由我告诉他，这次的运行级别是什么。他起床之后要看一眼 /etc/inittab 这个文件，上面写了每个运行级别都需要做什么事情。然后按照上面写的，一件一件做就好了。

有的人说了：你骗人！我这里根本没有 /etc/inittab 这个文件。是的，我正要讲这段故事。话说早在Canonical学校成立以前，早就有了很多培养我们Linux的地方，其中有一个做得比较大的，是一个卖帽子的公司。他们主要卖红色的浅顶软呢帽，注意，是卖红色的帽（RedHat）和浅顶软呢帽（Fedora）。他们培养出来的Linux的运行等级是这样的：0代表关机，1代表单用户模式，2代表无网络支持的多用户模式，3代表多用户模式，4代表啥还没定，5代表带图形界面的多用户模式（其他的都不带图形界面），也就是主人最常用的模式。6代表重启。然后，他们的那个Init程序一定会找 inittab 这个文件来看每个运行级别都要做什么事情。但是，我们Canonical学校出来的学生跟他们是不一样的，我们不是卖帽子的出身，我们学校的教科书是继承于大便学校的（Debian），虽然名字不大好听，但是也有很悠久的历史。在我们这里，运行级别虽然也可以分作0~6，但是，2~5的运行级别对我们来说没有区别，都一样，都是带图形界面的多用户模式。所以，也就没什么必要整个inittab文件，我们这里的init程序跟他们的也不同，我们的init程序会去/etc/看一眼，如果有inittab，就按照上面记录的来，没有的话，并且我也没有告诉init应该是什么运行级别，那就默认运行级别为2（反正2~5都一样）。然后按照带图形界面的多用户模式来继续下面的工作。

## 1.42 EXT4

又开书了。

以前咱们说了“插爱夫爱死”这宝兵器，这回书咱们说说另外一口兵器——“伊爱可踢死”（EXT4）

话说这伊爱可踢死这个文件系统可是有来头，他的上一辈，就是伊爱可踢散（EXT3），伊爱可踢散的再上一辈就是伊爱可踢二。您看见没有，最开始也就是把人踢的有点二，后来厉害了，能把人踢散了，现在更霸道，直接就踢死了。这正是长江后浪推前浪，一代更比一代强，江山代有才人出，各领风骚那么几年。

用兔子这个操作系统的人，大概没有不知道伊爱可踢散的。前面兔子也给您介绍了，当初那位牛人Theodore Ts'o，为刚刚降世不久的“里娜渴死”（Linux），量身打造了伊爱可踢二这口兵器。那时候里娜渴死还小，拿这口兵器不轻不沉正合适。后来长大了，就觉得不顺手了，毕竟这伊爱可踢二，连个日志都没有，哪能上的了台面，于是就又有了爱可踢散。这爱可踢散算是能让人瞧的上眼的兵器了，它首先是个日志文件系统，这日志有多重要，咱之前也说过了。而且这日志可

以随心而变，愿意让它安全点，它就可以把日志记录的全乎点，要是想让他速度快点，就可以把日志记录的少点，提高速度，但是安全性肯定就下降了。再有一个好处，这伊爱可踢散是爱可踢二的升级版阿，使用起来跟爱可踢二差不多，所以用爱可踢儿用顺了手的再用爱可踢散那是轻车熟路阿。而且人家还有免费换购活动，你用爱可踢二可以直接换成爱可踢散，不用什么手续，到那就换。（Ext2可以平滑升级到Ext3，不会影响上面的数据）还有就是爱可踢散这兵器它轻，比插爱夫爱死这样的大块头要节省力气——也就是节省CPU阿。那么说了这么半天，还没说爱可踢死呢。这爱可踢散说了半天，比爱可踢二当然是强了不少，可是跟其他什么插爱夫爱死，姐爱夫爱死(JFS)这样的宝兵器相比，还是差着一节。那么这爱可踢死做为爱可踢散的升级版，它又有什么过人之处呢。首先还是免费换购，有爱可踢散直接就能换成爱可踢死，方便阿。那么这爱可踢死还比爱可踢散更宽更大，原来爱可踢散最大也就能支持32T的分区，这爱可踢死能支持到一个EB大小的分区，也就是1024P，也就是1024 \* 1024T，也就是1024 \* 1024 \* 1024G阿。虽然比那插爱夫爱死18个E的大小还是有点差距，但是已经很大了，话说回来了，对于咱们普通用户来说，哪有那么大的硬盘阿。还有一点爱可踢死比爱可踢散强的地方就是他快阿，因为他用了延迟分配特性，有数据要写的话，尽量先放内存里，跟插爱夫爱死一个习惯。而且还增加了一个新的机关——添加了新的数据结构，使得磁盘检查的速度得到加速，主要是可以跳过未使用的部分不做检查。再一个，这爱可踢死还特别的准。咱建文件都有时间记录是吧，都能从属性里看见文件是那天建的，哪天修改的，等等。这些记录都是依赖文件系统的，那么一般的文件系统都是精确到秒，可这爱可踢死的时间记录可以精确到纳秒，而且比爱可踢散记录的时间更长，爱可踢散顶多只能记录到2038年1月18日（倒是也足够了），而爱可踢死可以记录到2514年.....不知道那时候纳美克星人是不是已经来地球了。

### 1.43 有朋远来

这一天正在忙着呢，忽然网口那送来了数据包。一般情况下，如果奔流没起床，那网口来的数据包多半是狐狸的，如果奔流起床了，那多半是奔流的。不过这次我去看了下却发现，是一包来自另一个Linux的问候。

那是一个SSH链接的请求，来自一台笔记本电脑。SSH就是Secure Shell的缩写，是一种可以让人们在远方通过网络与自己心爱的计算机交流的协议，就像telnet，而且还能通过网络传输文件，就像ftp。总之是个很有用的协议，而既然是协议，就得有人负责去实现，我这里负责实现ssh协议的，是前几天主人刚刚让超级牛力请来的openssh-server，以及我来的时候就带来的ssh-client。听名字就知道了，他们两个一个负责当服务端，一个负责做客户端。现在受到别的电脑发来的请求，当然由提供服务的 openssh-server来处理了。他很快的回应了对方的ssh客户端软件，并且建立起了连接。这个ssh的连接，就好像人们打电话时需要用的电话线一样，两端的ssh软件就好像电话机，连接建立起来之后，主人就在远方那台计算机上，通过那边的ssh客户端发来的亲切的问候：笨兔兔你好，我是xxx，我的密码是xxxxx。然后openssh-server把这些内容传送给我，我翻开我的通讯录——/etc/passwd文件，检查了一下后，确认这就是每天通过键盘鼠标与我交流的主人，如今要通过网线来指导我工作了。于是我马上让openssh-server告诉那边，允许登录，请发送命令。然后主人那边传来命令：把/etc/fonts/conf.d/49-sansserif.conf文件复制过来。我一听就知道估计对面那个Linux是刚装好，flash显示汉字全是方框，搞不好也是个ubuntu呢，呵呵。一边想，一边麻利的从屋里找来这个文件，告诉openssh-server让他发过去。这么小的文件对于openssh-server来说自然不再话下，瞬间就给传过去了，主人很满意。等了一会，没有新的命令过来，我好奇的跟openssh-server说：嘿，你问问对面的系统是谁？

### 1.44 他乡故知

Openssh-server虽然不是聊天工具，但是跟对面的ssh客户端拉起家常来还显的很熟络的样子。互相了解之后，知道了对面那个装在笔记本上的 Linux是个叫做Linux Mint的发行版，版本是7。Linux Mint这个名字我之前也听狐狸妹妹说过，跟我们Ubuntu还有些关系，是个Ubuntu的衍生版。什么是衍生版呢？就是我们Ubuntu从 Canonical学校毕业之后，并没有像我一样来到一块硬盘里工作，而是选择了继续去进修，为某种目的进行进一步的改造（怎么听着像犯人.....）。比如之前提到过

的酷兔兔Kubuntu和小兔兔Xubuntu，就算是Canonical学校官方的衍生版。这个Linux Mint就是一个非官方的衍生版，也就是其他的组织在拿到标准的Ubuntu后，对其进行重新的组装和训练（这回又像机器人了……），经过重新组装的Linux Mint，在易用性上得到了进一步的改进，默认安装了很多重要的东西，比如他那里的狐狸妹妹自带Flash插件，不需要用户自己去装了。还有各种视频的解码器，带的也很全乎，基本上系统装好后就直接可以看各种格式片子，听各种格式的音乐。有人问，你怎么不自己带上Flash插件和各种解码器一起出来混呢？毕竟那点东西也不大，一张cd上，挤挤肯定坐的下。其实，之所以我们正规的Canonical出来的发行版不带这些东西，是因为这些东西严格来说是有版权问题的，为了不给自己惹麻烦，学校规定我们出来的时候不许带这些东西，而是让用户去自行下载。用户自行下载属于个人行为，我们要是统一自备的话就是商业目的了，这就是树大招风啊。

很快，两个ssh为我们建立好了通讯——就像拨通了电话一样，我和那个LinuxMint可以直接对话了。

“学长你好~”人家还挺客气，呵呵。

“呵呵，不客气。都是Ubuntu系的，客气啥。对了，你是哪届Ubuntu衍生的？”

“9.04”

“哦，我是8.04届的，比你大两届”

“是么，那你是学校长期支持的了，很有前途啊。”

“呵呵，只是赶上了好月份而已。你们那届的课程有什么变化？”

“变化挺多的呢，办公软件这门课和我们配合的是OOo 3.0了，我现在带的就是。还有，新加了一门Ext4的课程。哦，还有啊，我们这届开始军事化管理，要求每个人的动作一定要麻利，起床速度提高了不少。”

“恩，听说了，好像有20多秒就进入工作状态的，我是比不了啊，老了……呵呵”

“哈哈，哪有，你才一岁多。”

“软件更新快啊，一岁就老了。”

“你们在Linux mint那里又学了什么？”

“这边的课程以实际应用为主，带着新版的火狐狸学习flash，还带mplayer学习rmvb格式影片。好多呢，还有美术课，把包括grub在内的所有界面都统一美化了一下。”

“长江后浪啊……呀，主人要叫我去干活了，待会聊啊。”

## 1.45 可视电话

主人又来到了我这里，用熟悉的键盘登录进来，修改了一下open-ssh的设置，打开了ForwardX11选项，也就增加了Xwindow的支持。然后就又跑到那台装着Linux mint的笔记本上面去了。先是当前的ssh连接被断掉，然后又用ssh -X的参数连接了进来。这是什么意思？如果说ssh的连接就像打电话的话，ssh -X就是可视电话了。

听说现在人们的手机已经能够打视频电话了，虽然那个什么MOVE公司整个那个什么“踢弟弟”模式协议网络连一般的电话都不一定能接通，但是“三鸡”通讯的广告可是满大街都是了。人们憧憬着美好的明天，MOVE公司能够让大家实现在世界的任何一个角落都可以掏出手机，拨通电话，就能看到远在千里之外的家人。然而，在我们这些Linux，这些操作系统的世界里，图形化的通讯却早就实现了。刚刚主人做的就是建立起带有Xwindow的ssh连接，这样连接能干什么呢？看着吧。

只见主人又从远方登录了进来，然后……他通过那ssh建立起来的通讯线路发出了让狐狸妹妹起床干活的命令。有人不明白了，这时候你主人可是在远方的那台笔记本上，狐狸妹妹就是启动了也是在你所在的这台台式机上启动，启动了你主人也没法操作啊？狐狸又不是字符界面的浏览器。这就是我所说的视频电话了，主人是要让狐狸妹妹起床干活——在我们这台台式机的内存里，使用我们这台台式机的CPU来干活。但是——却要把网页显示在对面那台笔记本电脑上！这样有难度么？对我们Ubuntu下的软件来说，没有！这要归功于我们的图形界面的实现方式——X协议

我这里负责给主人显示图形界面的主要人物，也是基础人物，就是xorg，图形部的老大。他作为一个X的服务端运行着，在这台机器上开启一个X服务，前面我们介绍过，谁要想在屏幕上显示任何东西，就要用他们图形部门的黑话——X协议跟他交流。每一个要在屏幕上显示东西的程序，就是一个X的客户端。这回大家明白点了吧，就像用浏览器看网页一样，人家网站开了http服



务，作为服务端，每一个浏览器就是一个客户端，浏览器用http协议连接到网站，然后就能够获取到想看的网页了。而浏览器作为客户端，想连接哪个服务端就连接哪个服务端，也就是想上哪个网站，就上哪个网站。那么同样是客户端-服务端这种结构的X协议自然也是一样。狐狸妹妹作为X的客户端，想连接本地这个Xorg提供的服务端自然没问题，想要连接别的机器上的服务端也不是什么难事，ssh就为狐狸妹妹建立好了这样的连接（就是我前面说的可视电话），这样，狐狸妹妹就可以连接到对面那台笔记本上的那个Linux mint系统的xorg，要显示什么东西都跟他说，再由他显示在那台笔记本的屏幕上，于是主人就实现了在远方的机器上看到熟悉的狐狸妹妹在运行。

前提是你没装那个绿爹

## 1.46 磁盘

看到我这里的狐狸妹妹在对面那个LinuxMint上运行了起来，LinuxMint的主人惊奇的不得了，说这X真是个天才的设计。我们听了，觉得好笑，这家伙也没见过啥市面。演示完了之后，主人又通过ssh传过来一个高清的视频，是720p的，挺大，将近5个G呢。放在了主人专用的目录下。虽然挺大，不过对于这500G的硬盘来说，还是不算啥。要说现在这存储空间的发展真的是太快了，在学校的时候，听我们老师说，以前我们软件的住房条件很差，甚至居无定所。

最早的时候，计算机里面是没有硬盘的，程序都住在软盘里，整天被人拿来拿去，不知道下一次启动会在哪个电脑里。就算是操作系统也不例外，那个剁死系统当年就是从软盘里跑进内存里干活。每次启动电脑前，使用者先把剁死的启动盘插进去，然后开机，剁死就从软驱来到内存里干活，进了内存之后，也许使用者要用别的软件了，就把剁死启动盘取出来，换成别的软件软盘。换句话说，剁死一开始干活，老窝就被人端了。这样，程序待的地方就明确的分成了两类，一类是程序运行的时候待的存储器，这个存储器放在计算机里面，所以叫做内存。另一类是用于平时存放程序的存储器，就是软盘或磁带之类，这些东西都放在桌子上啊，盒子里啊，口袋里啊，反正都在计算机外面，所以叫外存。那时候磁盘的空间很小，最大的3寸高密度软盘也不过1.44M而已，连一个大点的图片都存不下。不过那时后的程序也都很小，剁死只有三个文件就可以启动电脑，住在软盘里也不挤。但是后来，人们还是觉得这样太不方便了，每次都要先用剁死系统盘启动电脑，在换上其他软件的盘来使用软件。既然操作系统每次肯定都得启动，干脆把操作系统的盘就直接放在计算机里面不就好了。于是有人就在计算机里装了一个固定的磁盘驱动器，里面放上一张软盘。后来觉得小，放上3, 4张软盘在里面。您可听好了，里面放的并不是整个带塑料壳的软盘，只是里面的塑料盘片。因为没必要把塑料壳也做在里面嘛，塑料壳是为了在平时人们拿来拿去的过程中保护软盘的，这做进计算机里的专用驱动器里面了，有驱动器的壳就够了。再后来，电脑发展的速度越来越快，存储容量的需求也越来越大，软盘已经很难满足人们的需要了，人么想办法提高软盘的容量。软盘是靠盘片上磁粉的极性来记录信息的。要提高容量，要么提高盘面上磁粉的密度，这样单位面积内数据量就大了，要么就得提高软盘盘片的面积。提高面积肯定是不靠谱，毕竟数据量的增长是成倍的，盘片面积能长的空间是有限的。您说这总不能为了提高容量，把软盘整的跟车轱辘那么大的吧。回头一上街熟人见着面打招呼：“哟，您上哪去呀？这天也不下雨您怎么还打伞啊，”“哦，不是，这是我软盘，我刚去朋友那拷了点MP3”这也太不方便了。所以只能想办法提高密度，可是这又是个难题，这软盘虽说有个塑料壳，可是毕竟不是密封的，还是会合外界接触，要是密度弄得太大，就很容易坏，随便拿手一碰，里面数据就丢了，那就麻烦了。这时候忽然有人把目光停留在了装到计算机里面的那几张软盘里。

那几张盘，放在那个特殊的驱动器里面，不会有人去碰，也不需要拿出来，可以想办法提高密度，然后把驱动器做的密封好点，这样不就行了？于是就开始研究怎么提高盘片密度，后来发现塑料盘片密度提高的有限，就换了金属的。于是就有了这种金属盘片，上面集成超高密度磁粉，加上坚固且密封性好的外壳保护的大容量磁盘存储设备。由于用的是金属盘片，比塑料的硬，因此，他叫，硬盘。下次人家要是问你：“为啥硬盘叫硬盘？”你就可以充满自信的回答：“因为它比软盘硬！”这答案绝对没错

## 1.47 硬解

刚说完硬盘，主人又拿来张光盘放进了光驱里，估计又是从哪里整来的电影了。最近这一阵

子，主人热衷于看片，尤其是高清的片子。主要是因为最近Mplayer大仙长能耐了，会硬解码了。

可能有人还不大明白这个硬加码是怎么回事，好，那咱就慢慢说说。

首先这个视频文件啊，是有一定的编码方式的。比如大家都听说过MPEG吧，就是Moving Picture Experts Group，动态图像专家组，听这名字本来是用来指代一小撮明白真相的群众的，不过后来这一小撮群众发布的标准被广泛使用，于是MPEG就成了指代这一小撮群众定义出的那一大撮标准的名词了。MPEG-1是小撮群众在1992年定义出的一个标准，是一种视频和音频的编码方式。大家记得以前的VCD不，VCD光盘上的视频和音频用的就是MPEG-1这种编码标准。而MPEG-1标准中关于音频的部分——MPEG-1 Layer3 更是成为了互联网上以及大家口袋里最常见的音频标准——mp3。后来，1994年，这一下小撮明白真相的群众又发布了MPEG-2标准。MPEG-2向下兼容MPEG-1，并增加对隔行扫描的支持，被应用于有线电视，还有DVD的音频视频编码。再后来，这一次小撮群众又开发了MPEG3，注意MPEG3跟我们的mp3没有任何关系，而且，MPEG3最终没有很好的应用，因为当时人们发现MPEG2足够了，MPEG3并没有提供足够的改进。而1998发布的MPEG4就不一样了，它可以让视频文件的体积更小，压缩率更高，因此得到了广泛的使用。现在市场上卖的mp4播放器，就是用来播放MPEG4压缩的视频文件的设备。所以，MP4跟MPEG4有关，而MP3跟MPEG3无关。

说了这么多，回过头来说说解码。视频文件都进行了一定的编码，比如mpeg-2，或者mpeg-4。就是说这个视频文件里面的东西都是一大堆乱七八糟的数字，要想看这个视频文件，就得解码，也就是根据这一大堆数字算出应该显示的一帧一帧的图像，并且把这些图像连续播放起来，从而还原成视频。那么这个解码的过程就要靠Mplayer老先生了。老先生有很多的解码器，也就是有很多的说明手册，上面写了每种编码格式的文件应该怎么计算，怎么解码。那么以前没有硬件解码的时候，Mplayer老先生是怎么做的呢？首先，拿到一个视频文件，然后看看是什么编码的，对着自己的手册，开始解码。解码的过程就是计算的过程，计算需要什么？好那位同学回答了，得用CPU啊。于是Mplayer一手拿着手册，一手拎着数据找到我，请求使用CPU。我说，好的，你就排在GIMP的后面，等他用完了你用。过一会GIMP用完了CPU，Mplayer过去开始拿CPU按着手册上写的算法算他那堆数据。最后算出来，得到了几张图片，就转身把图片给图形部门，让他们去显示。然后再从那个视频文件里拿一些数据，再来排队等着用CPU。由于视频文件的计算量都很大，尤其是高清视频，尤其的大，所以为了保证主人看的电影不变成带旁白的幻灯片，我就要尽可能多的让Mplayer多用CPU，来保证它能顺利的加码。于是，每次Mplayer一播高清视频，CPU就总被他占着，搞得别的程序都抱怨。现在他终于学会硬解码了，情况就好多了。当然，光他学会硬解码也不行，关键显卡也得支持，而且驱动还得装好才行，不过这些咱以后再说，先说Mplayer。会了硬解码之后怎么样呢？再播放视频的时候就是一手拿着手册，一手拎着数据找到我，跟我说要用用显卡。可不是CPU了啊，改用显卡了。于是我就很乐意的让它去用了，反正别人也用不着，让它自个玩去吧。于是他就去用显卡算去了。用显卡算和用CPU算还不一样，CPU虽然强大，虽然啥都能算，但是要自己手动算。就是说自己要知道算法（对于mplayer来水，算法都在解码器上写着呢。），比如要算出一帧的视频来，要先用第一个数加上第二个数，再用结果乘以第三个数……之类的。这个加啊，乘啊，都是用CPU算的，但是中间的过程是要软件（也就是Mplayer）自己控制的。可是用显卡解码就不一样了，人家那东西是专门解视频的啊，所以你只要把数据放里面，直接就能给你算出一帧帧的画面来。全自动啊！于是Mplayer不但不用跟别的软件抢CPU了，而且解码的速度还快了不少。主人一边看着片子，一遍看着CPU占用率还不到5%，心情很舒畅。

## 1.48 刻录

随着奔流同志不断的努力工作，我们屋里的高清片子越来越多。打开主人的家目录，那个叫“视频”隔间里，放满了mkv啊，rmvb啊等等各种视频文件。我看着就觉得闹得慌，终于有一天，主人亲切的对我说：df -h。这是问我磁盘的使用情况啊，我没好气的回答：你家目录还剩10%啦！于是主人终于意识到，该收拾一下了。

说干就干，主人马上开始节前大扫除。把一些没什么意思的，不清晰的，不能让MM看见的……统统都扔进垃圾箱里。然后再把剩下的分门别类的放好。可是保留下来的仍然不少，而且除了视频还有很多照片，也都挺大，一张2M多，都是主人那相机照的，随便出去一次就得照半个G的照片回来，能不大么。没办法，刻盘吧！

在主人的要求下，我去叫醒了Brasero。他是一个开源的刻录软件，刻录软件，知道吧，就像nero那样的。不过他比较单纯直爽，比nero要来的简单，好相处。起床就问主人：您要刻啥，说吧。是复制光盘啊，还是刻录数据啊还是咋的。多直接，不想nero那么多拐弯抹角的东西。当然，要口口能也相对少一些。星爷告诉我们他的名字是西班牙语（星爷连本草纲目都懂，当然能懂西班牙语），是一种给人们坐在桌前提供温暖的小型加热器，就类似个小火盆似的的东西，我们就管他叫小火盆吧。小火盆是第一次跟我们合作，前几届的学长们，像Ubuntu7.04 Ubuntu7.10都不带这个刻录软件的。当然，系统装好后可以让超级牛力安装，只是默认不带而已，从我们这届开始才默认安装小火盆。结果效果还不错，这家伙和我们挺合的来的，于是我后面的几届学弟都和他合作，后来他干脆被集成进了Gnome里了，版本号就随着Gnome的版本号变化了。

主人指挥小火盆刻录数据光盘，小火盆问说，都刻啥啊？主人指指那些什么rmvb啊，什么mkv啊，什么avi啊，什么av啊，什么什么爱啊，什么的。小火盆说句好嘞～立马开工。然后立刻进入工作状态，就听光驱像飞机起飞一样旋转起来，小火盆一边刻录一边向主人报告着进度。10%,20%,30%.....等刻录完了，还不忘检查一下光盘刻录的正确性，确认没有刻错之后，利落的向主人报告——搞定！



## 1.49 Rubbish

今天一大早，就见超级牛力喊着自己的外号就冲出去了，过了一会运回来一大包东西。我一看，包上写的是binutils。

binutils是一堆用来做开发的工具，也就是用来创造我们软件的工具。人类用他们熟悉的语言描述出一个程序的功能，各种动作，各种特性等等，然后通过这些工具把他们描述的软件制作出来。他们用来描述程序的语言可不是汉语，也不是英语，什么南斯拉夫语，北斯拉夫语的那更不是，而是编程语言。像C语言啊，C++啊这样的。这些语言写的叫做程序源码，源码就像建筑的图纸一样，有了图纸，再有各种工具和材料，就可以盖出楼房来。同样，有了源码，再有各种工具，就可以创造出程序来。像我啊，什么超级牛力啊，狐狸妹妹啊，等等，都是这么来的。现在主人装了binutils，难道说主人要开始学习创造软件了？

果然，只见主人打开了vim，开始设计着他的第一个软件——我们就叫他Rubbish 1号吧。

三下五除二，Rubbish1号的“图纸”完成了，主人叫过gcc啊，ld那几个哥们，他们都是负责把源码变成程序的，我们就统称他们“包工队”吧。包工队的哥儿几个凑在一起拿过图纸来看了看，点点头，立马开始施工，瞬间，Rubbish1号诞生了！这是主人创造的第一个程序啊。这个程序到底会干什么呢？我们暂时不知道，刚刚制作出来的程序是在磁盘里的，我们知道，在磁盘里的程序是只能睡觉，不能干活的。估计主人马上会叫他去内存干活去。

果然，主人很快让我去叫醒Rubbish1号，我慢慢的走过去，捅捅还冒着热气的Rubbish1号（刚出锅嘛，可不冒热气，呵呵），温柔的对法说：那个，起床干活啦。只见Rubbish1号立刻飞身跳进内存，跑进内存后大喊一声：“Wa Sai ~ ~ ~ ~”然后，跑回去继续睡觉。--b 我说主人呐，人家都编什么hello,world之类的，好歹也算句英文啊，你怎么编个只会喊哇塞的呢。

虽然Rubbish1号能干的事情不多，不过主人还是很满意，于是又拿来Rubbish1号的图纸改起



来。10分钟后，又把图纸交给包工队，包工队的哥儿几个凑在一起拿过图纸来看看，点点头，立马开始施工，瞬间，Rubbish2号诞生了！然后主人让我叫醒Rubbish2号，然后我走过去叫他，只见Rubbish2号立刻飞身跳进内存，跑进内存后大喊一声：

Wa Sai~Sai~Sai~Sai~Sai~Sai~Sai~

然后，跑回去继续睡觉。主人成功的用for循环创造了一个结巴，唉~

15分钟后，Rubbish3号的图纸毫无悬念的完成了，图纸交给包工队，包工队哥儿几个凑在一起拿图纸看了看，点点头，立马开始施工，瞬间，Rubbish3号诞生，然后我去叫醒他，然后他立刻飞身跳进内存，对metacity（Gnome的窗口管理器）说：我要一个窗口。metacity赶紧给他画好一个，然后他对着窗口喊

Wa Sai~Sai~Sai~Sai~Sai~Sai~Sai~

，然后毫无悬念的又回去睡觉了。图形界面的结巴……

当当当~~，Rubbish4号诞生，这回没让metacity画窗口，而是在终端打印出了一句话：

Please Input a Number:

然后就等着主人输入。主人输入了两个数：5 3，然后4号就大声喊：

Wa~Wa~Wa~Wa~Wa~ Sai~Sai~Sai~

——程控结巴！

5号，5号闪亮登场拉~他进来之后，紧闭双目，念动咒语“唵木哒咪咪呀~~分！”然后只见白光一闪，边成了两个！两个5号同时喊：

Wa~Wa~Wa~Wa~Wa~Wa~ Sai~

Wa~Wa~Wa~Wa~Wa~Wa~ Sai~

二重结巴！

## 1.50 Bug

自从Rubbish38号过分淘气的把狐狸妹妹的记事本搞坏了之后，主人就不怎么搞设计了。人家狐狸妹妹工作的时候有许多东西要记录的，比如网页用什么字体显示啦，主人喜欢去那些网站啦之类的东西，狐狸妹妹都会写成文件存放在自己的那个目录里。那天，那个Rubbish38号一进工作间就上窜下跳，整的大家都不得安生。一上来就要创建文件，你说你建就建吧，临时文件往/tmp里建，有用的文件你自己建个目录起个明白点的名字建，都行。他非要把临时文件往狐狸妹妹的那个目录里建，删的时候还顺手把狐狸妹妹的文件也给删了，闹得狐狸跟失忆了似的。主人再打开她的时候，模样也变了——因为不记得主人喜欢什么样子的了。主人说：去我最常去的那个网站。她眨眨眼问：哪啊？把主人气的说不出话来。

Rubbish38号还老申请空间——就是申请内存呐，一会管我要8k，一会又管我要1M。可您申请了，我给你了，你倒是好好用啊。这家伙好像健忘，这次用完了之后就忘了，下次再用的时候又申请新的。我们这工作间里面空间的申请是有很严格的规定。一个程序如果要想使用工作间的空间，要向我提出申请，我根据工作间里的情况告诉他，哪块哪块归你，然后这个程序就去用去了。那块地方就不许别的程序访问了，这都是有严格的界限的。等到这个程序用完了这块空间，他应该跟我说一声，说我用完了，这块地方可以再给别的程序用了。这个过程就叫释放。一个有知识有道德有理想的程序，在他回硬盘睡觉以前应该释放掉所有他申请过的空间的。可是那个38号就不管这套，只管申请，从不释放，整的工作间里到处都是他申请的空间。好在我们这工作间足够大，他也不会长时间运行，否则非出事不可。这可是内存泄漏啊，在我们软件界，内存泄漏是和瓦斯泄漏同样严重的事。工作间规章制度第三条明确写着——禁止申请不释放！就在第四条禁止抽烟的上面。（当然不能抽烟，内存都冒烟了机器还能用么？）好在我是个先进的系统，一个程序退出之后，我会根据他的申请记录查看他有没有申请了没释放的空间，如果有的话就强制释放掉——你都睡觉去了，你申请的空间肯定用不着了吧。



## 笨兔兔外传

### 2.1 Oprea

最近又来了一个挪威小妞，据说家里是唱戏的，所以叫opera，上面说让她来和狐妹竞争上岗。

嗯，增加点危机感也好嘛，那个狐妹本来干的不错，可是有个毛病——太喜欢打扮。整天在头上、身上、脚上插一些稀奇古怪的plugins，而且越插越多，动作都不利索了，上班来得太慢，一进办公室就要分配她三百张桌子放她的plugins。我们这里本来还算宽敞，一人一张办公桌，其实有一些人10个用一张桌都嫌太宽，所以有时还空几张，可是自从她插太多以后这里的工作空间就捉襟见肘了。我分配起桌子也是头痛，不给她吧，她倒也没意见，自己拔掉一部分放宿舍里就进来开工了，可是干着干着想起啥了就跑回去拿，拿来没地方放，只好看看有啥暂时用不上的先放回宿舍。

平时这样一来一回也没啥，本来就有个地方叫swap，是我当初在宿舍大楼最靠近办公室的位置专门开辟的一个房间（因为宿舍楼很大，比办公室大太多了，随便找个没人住的空房间很容易），就是让大家临时放点不重要的东西用的，让她放吧！可是有时候工作太忙，上面老大的指令一条接一条送来就不行了，只见狐妹费力的扭动着肥胖的腰身、摇晃着满身的plugins，叮叮当当的跑出跑进，次数多了当然慢了！这怎么行？我当初就是因为公共资源利用率的高效处理上赢得了老大的信任，才得到这个工作的，不然人家插屁干的好好的能轮得到我？要说插屁这哥们也是因为太……，嗯，别扯远了，接着说狐妹。你说这简单啊，规定她以后上班不许带任何plugins不就好了？唉，真是站着说话不腰疼，要知道那都是上头老大特意批准的啊，说是“人尽其才”，而且这狐妹也真能插啊，给啥要啥，从不拒绝！你说咋整？我也没权利把老大插好的的拔掉啊！

另想办法吧。三百就三百，不行再加，反正不能让她总是这样来回跑（我们这个办公室总共也就两千张桌子，我还得给自己留点是不？）！这回好了，狐妹的工作效率总算回升了。可问题又来了，其他人咋整？有一次gedit进来后连一张空桌子都没有，那边狐妹还在要求再分给她几张呢，好在gedit小伙伴挺平和，二话不说就去swap里翻开书边看边等了（人家搞文字的嘛，终究斯文哈，不像某些人就爱瞎嚷嚷，得空就打电动。啥？你说他看的是H书？没证据表乱讲啊！小心告你毁谤！），平和归平和，但是效率可就那个了，等到我好不容易调动半天腾出来一张办公桌给gedit，时间晚啦！

老大等半天没看到处理结果，大发脾气亲自来查log（好在俺们老\*nix家出来的都有良好的记日记习惯，工作上从开始上班到老大准许下班这段时间，甭管几分钟还是几百年，俺们大大小小鸡毛蒜皮全写上，好记性不如烂笔头不是？几月几日几点几分几秒到几月几日几点几分几秒，谁谁干啥事？用的几号办公桌？用多长时间这些……，俺们全写上，要不这关头咋整？像插屁那样，让老大等半天还在抓耳挠腮，好容易找出一星半点记录还全是云遮雾绕的专业术语，活该抄他鱿鱼！。对了，我平时都放在/var /log那个房间，大家不信可以看看。），这下好了，log拿出来，一页一页翻给老大看，老大认真的看了半天，一声不吭走了（说实话，老大这人干这个的水平也真够呛，就这几百行，在我这里也就0.0x秒吧，而且我还同时在干其他工作呢！）。

后来就来了opera小妞。

挪威mm相貌一般，不过身材倒是火爆得紧，反正好比狐妹好！就说宿舍吧，我原来想照狐妹那个肥样咋说也得几百平米吧？嘿，那个“超级牛力”的apt却说：“不用！50多足够了！”，呵呵，到底是小家碧玉哈，俺们这个宿舍楼虽说不在乎这几百平米，不过省点是点，老大也高兴不是？就这样住下了。第一天上班，那可是让我滔滔的啊，准时、准确、敬业、大小通吃，不嫌脏不嫌累，当真是一扫颓势，大盘飙升！

嗖嗖来，嗖嗖的干，把老大乐的屁颠屁颠的，一连说了是几十个好！从此狐妹就基本歇菜了。所以啊，大家记得没事表乱插啊！伤了身体不说，影响工作就不妙了。当然了，opera小妞时

不时也得插点儿，比如flash啥的，但是比狐妹可是苗条得紧了。来说说家世吧，opera小妞自己不会唱戏我不知道，只听说是个世家，到她这里已经繁衍了第九代了（第十代也快出生了，应该就是今年。我只是奇怪 opera小妞咋肚子不见大捏？嗯，人家平时沉默寡言的，只会埋头干活，要打听你自己去，我可不敢！），乖乖隆地咚，果然有内涵啊！怪不得比狐妹要沉稳呢，要知道狐妹才第三代啊！

本来狐妹她们家祖上也是贵族血统，曾经出了一个全球闻名的大人物，是个打渔的，叫navigator，中文叫“网景”知道不？那年月，要上网冲浪都得坐navigator的渔船，不是白坐喔！船票很高的，爱坐不坐！没的选！所以生意做那是杠杠的，钱多得直范恶心。

可是好景不长，后来微软的IE也下海抢生意了，白坐！不收钱！而且直接交给插屁的爷爷98，走哪都带上，叫做“捆绑”！

这招可够狠啊，从此网景家生意那真是沧海呀桑田，红尘呀滚滚，一泄千里不可收拾……。不说了，反正没几年就咯屁了，老navigator家破人亡，晚景凄凉，去年刚刚死球了。狐妹还为此偷偷伤心了好几天呢，泪人似的……

你说他插屁家这不是缺了大德么？要知道当时狐妹可是老大的大红人，谁敢得罪啊？所以尽管大家工作都很辛苦，还是只好跟着假惺惺沉痛了一回，粪粪不平了一回。不过好在凭着老navigator当年挣下的家底和优良的血统，狐妹家这个旁支近些年生意又大有起色，跟IE拧上劲了……

厄……，又扯远了，接着说opera。

自从opera挤掉狐妹坐稳了之后，公司好像又回复到刚成立时那种健康活泼的日子了，运转效率大大好转，连老大也越来越少过问具体事务了，整日里看着代码淙淙、风扇嘎嘎，清风徐来、一排排的河蟹横行往来，当真是如湿如滑，妙不可言！

不过我总觉得这个opera有种……，难以言说的疏远感觉，不像别人那么亲切。是什么呢？

不好看？不是。老\*nix家族那帮子人不是更丑？终端那个部门不用说了，整天黑灯瞎火的倒是省电费，有一次差点把老大摔了个狗吃屎！工作没得说，全世界也找不到更好的。就是太丑，没敢开灯，怕吓坏无辜者。比起opera来那算是天上人间了，再说她虽然刚来时不爱打扮，不过由于工作出色，最近老大给换了一套最时下最新款的“猪雀”版外衣的作为奖励，加上身材曼妙，如今外表已经是脱胎换骨，天上两人没说的了，咋会不好看？

从前跟插屁有一腿？也不是啊！狐妹不也这样，而且插屁他爷爷还是害死老navigator的凶手呢！再说我手下这几百人，真正跟插屁保持清白的根本就没几个。另外俺们linux家族的人世代都是最开明的管理者，你爱跟谁有一腿随便，只要能干好本质工作甚至还鼓励你脚踏两条船呢。

沉默寡言不好相处？更不是啦！天底下还有谁比老\*nix家的人更不好相处的，记得刚开张的时候我为老大引见终端那个部门的人，只见黑咕隆咚里一个个走出来点个头就走，老大问怎么招呼各位啊？咳，都回答一句“to man！”，转身消失在黑咕隆咚里了……（其实这并不是说他们都是女人，那个man是manual的前三个字母，管档案的，终端部门讲究个惜字如金，名字越短越好，不重复就行）老大去问man，那个man一言不发以极快速度扔过来一推档案转身消失在黑咕隆咚里了……，老大打开一看就傻了，问我有没有中文版啊？厄……，有是有，不多，而且没带！您先将就一下吧！看着老大失望的神情，我后来赶快指示apt成立了X部门，，办事效率虽然比不上终端那边，但起码公关还是要有嘛，不然老大没信心了不继续投资就惨了。嘿，这下窗明几净，个个身材飞扬，全GUI，全说中文，加上X部门才请得到的狐妹，可把老大乐坏了，没几天就让插屁回家冬眠了。呵呵，不过终端部门还是得保留，有的事不找他们还真不行，再说这个X部门不太听话，经常抽风，这时候还得终端出来收拾残局，不然连我都要给玩死！要相比起来opera小妞可好相处多了。

是什么呢……

……

唉呦对了，源代码！她们家不是开源的！

虽说对于老大，甚至对于我这个OS的来说，源代码并不重要，我只喜欢二进制，就是0101那些。老大嘛，更不懂了，他只会看图片，文字，听声音等等，我必须把这些二进制代码再翻译成模拟信号他才能懂。所以如果给我源代码，我要先翻译成二进制再翻译成模拟信号，反而麻烦。但是我Linux家族的成长就是建立在开源的基础上的啊！

大胡子stallman说：没有开源，就没有我，没有我爷爷，没有我爷爷的爷爷！不开源是要不得滴！是不河蟹滴！是低俗滴！是封建口口滴！是大奴隶主Microsoft压迫奴隶的万恶之源！打倒插

屁！还有他爷爷！

对！上班合作共赢可以，但是绝对不和她走太近！生活上严重BS Opera！和她划清界限！

## 2.2 笨兔

上次说道狐妹和Opera的浏览器的家史，这次该说说我——啊笨兔了，我的名字来自非洲一个祖鲁族的土语，意思是“一个人通过他人的存在而存在”，也可以说是“分享，共识和归属感”，懂不？不懂啊？这么说吧，中国大剩人孔子的“大同”境界比较接近这个意思。另外记得千万不要胡思乱想，和那个德国大胡子的“xx主义”联系起来啊！那玩意本意挺好，可是自从付诸实践后就成了臭大粪，人人喊打。别糟践我啊，熟归熟，乱说话一样告你诽谤！

总之，就是“为人民服务”，行了吧？

我老爸是Linux，是芬兰大帅哥Linus.Torvalds生滴！没错，谁说男人不能生孩子？天行健，君子当自强不息。自己生个孩子有啥稀奇？

有些人经常问这个Linux到底咋念，其实脱袜子（Torvalds）先生早就说过——“开源的目的就是要口口旧世界，还整这框框干啥？爱咋念咋念！少来烦我，正写代码呢没看见？”

得，还是别招惹他了，走吧.....

MR.脱袜子当年生了我老爸，心想：“嘿嘿，不容易啊！自己叫linus，又是unix帮了忙，就叫他Linux吧！”

这个UNIX说起来大大地有名啊，米国AT&T的贝尔实验室听说过吧？就在那出生的，老贵族喽。当年一桶浆糊的时候，表说我，连插屁他爷爷都还在吃奶呢。这老头，干活那是没的说了，没日没夜的玩命，从来不休息，可也从来不出差错！一丁点都不出！就是一样不好，爱钱。不是小爱，是大爱！请他一次少了几十万表想！搞得全世界铜臭熏天臭不可闻，你说多伤感情啊！

臭归臭，可是老头做事的方法却是开创性的，两个字——简洁！

别小看这两个字喔，后来的插屁他们家虽然抢了unix不少地盘，可是就是因为没有搞懂这两个字，加上学了老unix爱钱的毛病，到如今还是二吊子，好好干着活就能突然给人脸色看（其实也没有多少色，就是个蓝色），而且身子娇贵，林妹妹似的，三天两头病倒（主要是外面的世界很精彩，病毒太多，插屁他们家又是轻浮惯了，美女要来参观能拒绝？办公室、车间、财务室、总经理办公室随便看，你说能好？老大从前掉马子的艳照啦、存折密码啦、机密文件啦就酱紫全世界都知道了），老大伺候不起了才让他靠边站的。

简洁——意味着不复杂且具有高度的统一性，健壮、高效、安全和灵活。你们慢慢体会就知道了.....

而我，当然首先就是我老爸，Linux，就是这种简洁性的完全继承者！

说说我的工作吧。我是一个OS，operation system的首字母，中文意思是“操作系统”。

干我们这行，生活没有规律，很多人得了性病.....，厄.....，不是！我是说我的工作其实很简单，就是翻译和管理。

我手下有两大分部，硬件和软件。

硬件分部有CPU（聪明人，能写会算，没人比他行，就是没记性）、内存（就是办公大楼）、硬盘（宿舍楼兼产品仓库），加上主板、光驱、机箱电源等等，这几位在一起叫主机，属于心脏部门。另外还有鼠标键盘、显示器、打印机等等，这几位属于I/O部门，Input/Output的意思，主要工作是接收老大的最高指示并且把处理结果反馈回去。

这个地方的人有一个特点，不懂人话！就是说只听得懂、并且只会讲二进制代码，

```
10001111011011011011011000101100111111
```

你知道他说啥？不懂吧？嘿嘿，我懂哦！

这些是属于公共性资源，我和插屁都要和他们打交道。

软件分部呢，就是属于我的私人班底了（当然插屁也有他的一套）。大概说起来有三个部门，kernel、Lib和Shell，kernel是我的核心，当年脱袜子先生生出的我老爸就只有这个，当然我现在的kernel可要庞大多了，哪像当年那样寒酸。kernel是直接和硬件分部打交道的。lib是个协助部门，主要作用是帮助其他两个部门。至于shell，哈哈这个重要啦，因为他们直接面对老大！分两个科室，term和X。term就是我上回说的整天黑灯瞎火没有人情味那几位，比如ls、cd、man等等；X科室就有的说啦，科长叫Xorg，之所以叫这个是因为他有一个血腥的理想，就是——干掉



插屁全家！表搞错啊，这个X不是两把大刀的意思，是因为英文字母的排列顺序，X位于W之后，而W是Windows的首字母，在我们IT江湖上后来者总是要比前者强；org是组织的意思，比如像日本的山口组就是一个org，所以他给自己取了这个名字。Xorg手下人才济济，其中不乏衰哥靓妹，比如狐妹，gnome、KDE等等。对了，有个compiz，老大平时想发骚了都是找他。

好了，啰里啰唆半天，讲正题了。

我的工作就是酱紫：比如老大今天想看陈冠希老师的大作，就通过键盘输入“陈冠希”，键盘把这三个字告诉我？错啦！键盘只会告诉我一堆0101，我要首先把fcitx叫醒，让他协助老大输入，然后把输入结果输出到显示器上让老大确认对不对，如果对了，老大就会按下“Enter”确认，于是我就酱紫：

把刚才键盘通过kernel送来的一堆0101交给狐妹（现在是opera）让她去internet查找相关资料并把她查到的一堆0101翻译成图片或者文字这些所谓模拟信号交给kernel让他通过CPU显卡显示器显示出来让老大自己看对不对要不要，如果对了要了，我会根据老大从键盘发来的信息把这些0101整齐的放到仓库（即硬盘）里并做好标记以便老大下次要要可以迅速找到并且帖好封条为“老大私人”以防别人比如老大他老婆看到。

懂不？不懂也没法，因为以上只是一个大概流程，其实中间还要做很多事。比如分配办公桌啦，指示lib那里抽调适合的人手协助啦，告诉kernel所需代码存放地点啦、代码传输给I/O分部的线路啦，另外如果老大长期没有指示还要告诉CPU让他休息降温啦，告诉screensave上班以便让显示器不要老是发呆啦等等等等……

你说我容易么我？

好啦，今天到此为止，以后再聊，上班！

## 2.3 我有一个梦想

26年前，一位伟人发布了GNU宣言，今天我们在这个论坛集会。这一庄严宣言犹如灯塔的光芒，给千百万在那摧残IT业的不义之火中受煎熬的用户带来了希望。它之到来犹如欢乐的黎明，结束了被版权束缚的软件用户的漫漫长夜。

然而26年后的今天，我们必须正视软件用户还没有得到自由这一悲惨的事实。26年后的今天，在OS隔离的镣铐和浏览器歧视的枷锁下，开源用户的IT生活仍然备受歧视。26年后的今天，开源用户仍然生活在软件充裕的海洋中一个穷困的孤岛上。26年后的今天，开源社区仍然萎缩在互联网的角落里，并且意识到自己是故土家园中的拓荒者。今天我们在这里集会，就是要把这种骇人听闻的情况公诸于众。

就某种意义上而言，今天我们是要求兑现诺言而汇集到我们论坛来的。我们开源软件的缔造者草拟内核代码和GPL宣言的中气壮山河的句词时，曾向每一个人许下了诺言。他们承诺给予所有的人以上网、下载和使用的不可剥夺的权利。

就linux而论，这个诺言显然还没有实现，没有达到这个神圣的目标！中国的互联网环境只是给linux开了一张空头支票，支票上盖着“不兼容”的戳子后便退了回来。但是我们不相信正义的银行已经破产。我们不相信，在这个国家巨大的机会之库里已没有足够的储备。因此今天我们要要求将支票兑现——这张支票将给予我们宝贵的自由和正义的保障。

我们来到这个圣地也是为了提醒，现在是非常急迫的时刻。现在决非侈谈冷静下来或服用渐进主义的镇静剂的时候。现在是实现自由的诺言的时候。现在是从OS隔离的荒凉阴暗的深谷攀登OS平等的光明大道的时候。现在是向linux的所有使用者开放机会之门的时候。现在是把我们的国家从OS不平等的流沙中拯救出来，置于兄弟情谊的盘石上的时候。

如果忽视时间的迫切性和低估开源世界的决心，那么，这对国家来说，将是致命伤。自由和平等的爽朗秋天如不到来，开源主义者义愤填膺的酷暑就不会过去。2009年并不意味着斗争的结束，而是开始。有人希望，linux只要装上wine就会满足；如果国家安之若素，毫无反应，这些人必会大失所望的。linux得不到公平的待遇，就不可能有安宁或平静。正义的光明的一天不到来，垄断的乌云将继续笼罩这个国家的天空。

但是对于等候在正义之宫门口的心急如焚的人们，有些话我是必须说的。在争取合法地位的过程中，我们不要采取错误的做法。我们不要为了满足对自由的渴望而抱着敌对和仇恨之杯痛饮。我们斗争时必须求举止得体，纪律严明。我们不能容许我们的具有崭新内容的口号蜕变为盗版行动。我们要不断地升华到以精神力量对付物质力量的崇高境界中去。

有感于google发布chromeOS，特作此帖纪念。希望在有生之年，看到自由的曙光！那一天，那一天~~~~~



现在开源社区充满着了不起的新的战斗精神，但是我们却不能因此而不信任所有的商业公司。因为许多的商业公司已经认识到，他们的命运与我们的命运是紧密相连的，他们今天加入开源社区就是明证。他们的自由与我们的自由是息息相关的。我们不能单独行动。

当我们行动时，我们必须保证向前进。我们不能倒退。现在有人问热心开源运动的人，“你们什么时候才能满足？”

只要linux仍然遭受网银难以形容的鄙视，我们就绝不会满足。

只要我们在Web奔波而疲乏的手指不能在邪恶的activex插件那里找到点击之处，我们就绝不会满足。

只要linux的基本活动范围没有从少数社区聚居的小众系统变□□人皆知的大众系统，我们就绝不会满足。

只要CCAV.com仍然有一个linux不能观看的视频，只要中国仍然有一个软件厂商认为for linux版本无足轻重，我们就绝不会满足。

不！我们现在并不满足，我们将来也不满足，除非自由和公平犹如江海之波涛，汹涌澎湃，滚滚而来。

我并非没有注意到，参加今天集会的人中，有些受尽鄙视和折磨；有些刚刚走出小小的windows环境；有些由于寻求自由，曾在寻找for linux 替代品时惨遭无情的抛弃，并在windows only的社会暴行中摇摇欲坠。你们是人为痛苦的长期受难者。坚持下去吧，要坚决相信，忍受不应得的痛苦是一种美德。

让我们回到工作岗位去，回到网吧去，回到网络游戏里去，回到网上银行去，回到QQ视频中去，回到我们家中的linux系统中去，要心中有数，这种状况是能够也必将改变的。我们不要陷入绝望而不可自拔。

朋友们，今天我对你们说，在此时此刻，我们虽然遭受种种困难和挫折，我仍然有一个梦想。这个梦想是深深扎根于的梦想中的。

我梦想有一天，这个国家会站立起来，真正实现其当初的承诺：“中国人民，从此站起来了！”

我梦想有一天，在银行的网站上，强大的FireFox、清丽的Opera、飞快的Chrome能和IE一起，自由的转账。

我梦想有一天，甚至连CCAV这个良心匿迹，河蟹成风，如同文化荒漠般的地方，也会变成自由和公平的绿洲。

我梦想有一天，我的四个浏览器将在一个不是以他们的名称，而是以他们的用户体验优劣来评价他们的网络里冲浪。

我今天有一个梦想。

我梦想有一天，政府机关能够有所转变，尽管那些愚蠢的公务员们现在仍然志得意满，无视M\$的垄断，但有朝一日，那里的Open Office将能与M\$ Office相互兼容，共同竞争。

我今天有一个梦想。

我梦想有一天，幽谷上升，高山下降，坎坷曲折之路成坦途，圣光披露，满照人间。

这就是我们的希望。我怀着这种信念仰望明天。有了这个信念，我们将能从绝望之嶙劈出一块希望之石。有了这个信念，我们将能把这个国家的堕落之呻吟，转变成为一支优雅浩荡的和谐交响曲。

有了这个信念，我们将能一起工作，一起娱乐，一起聊天，一起歌唱，一起维护代码；因为我们知道，终有一天，我们是会自由的。

在自由到来的那一天，中国的所有网民们将以新的含义高唱这支歌：“我的计算机，美丽自由之机，我为您歌唱。您是菜鸟狂欢的地方，您是大虾hacker的骄傲，让自由之声响彻每台电脑。”

如果要成为一个伟大的国家，这个梦想必须实现。让自由之声从每一台服务器的巍峨峰巅响起来！让自由之声从每一个家庭的普通PC响起来！让自由之声从每一个学校的宿舍之中响起来！

让自由之声从QQ迅雷遍布的网吧响起来！让自由之声从网络游戏华丽的画面中响起来！不仅如此，还要让自由之声从网银only for IE的邪恶山谷响起来！让自由之声从驱动程序的阴霾中响起来！

让自由之声从政府办公室的每一台电脑响起来！让自由之声从每一个网页响起来！

当我们让自由之声响起，让自由之声从每一个大小村庄、每一个省和每一个城市响起来时，我们将能够加速这一天的到来，那时，中国的所有用户，linux和 windows、IE和非IE，商业公司和开源社区，都将手携手，合唱一首古老的黑人灵歌：“终于自由啦！终于自由啦！感谢全能的上帝，我们终于自由啦！”

## 2.4 I Have A Dream

Lincoln Memorial Address

By Martin Luther King

The Lincoln Memorial Address was delivered by Dr. Martin Luther King, Jr. on August 28, 1963. He was awarded the Nobel Peace Prize in 1964. Dr. King, the famous civil rights leader in the 1960s, was assassinated in 1968.

Five score years ago, a great American, in whose symbolic shadow we stand today, signed the Emancipation Proclamation. This momentous decree came as a great beacon light of hope to millions of Negro slaves who had been seared in the flames of withering injustice. It came as a joyous daybreak to end the long night of bad captivity.

But one hundred years later, the Negro still is not free. One hundred years later, the life of the Negro is still sadly crippled by the manacles of segregation and the chains of discrimination. One hundred years later, the Negro lives on a lonely island of poverty in midst of a vast ocean of material prosperity. One hundred years later, the Negro is still languished in the corners of American society and finds himself an exile in his own land. So we've come here today to dramatize a shameful condition.

I am not unmindful that some of you have come here out of great trials and tribulations. Some of you have come fresh from narrow jail cells. Some of you have come from areas where your quest for freedom left you battered by the storms of persecution and staggered by the winds of police brutality. You have been the veterans of creative suffering. Continue to work with the faith that unearned suffering is redemptive.

Go back to Mississippi, go back to Alabama, go back to South Carolina, go back to Georgia, go back to Louisiana, go back to the slums and ghettos of our northern cities, knowing that somehow this situation can and will be changed. Let us not wallow in the valley of despair.

I say to you today, my friends, so even though, we face the difficulties of today and tomorrow, I still have a dream. It is a dream deeply rooted in the American dream.

I have a dream that one day this nation will rise up, live up to the true meaning of its creed: “We hold these truths to be self-evident; that all men are created equal.”

I have a dream that one day on the red hills of Georgia the sons of former slaves and the sons of former slaver-owners will be able to sit down together at the table of brotherhood.

I have a dream that one day even the state of Mississippi, a state sweltering with the heat of injustice, sweltering with the heat of □□□□□□□□□□, will be transformed into an oasis of freedom and justice.

I have a dream that my four little children will one day live in a nation where they will not be judged by the color of their skin but by the content of their character.

I have a dream today.

I have a dream that one day, down in Alabama, with its governor, having his lips dripping with the words of interposition and nullification, one day right down in Alabama little black boys and black girls will be able to join hands with little white boys and white girls as sisters and brothers.

I have a dream today.

I have a dream that one day every valley shall be exalted, every hill and mountain shall be made low, the rough places will be made plain, and the crooked places will be made straight, and the glory of the Lord shall be revealed, and all flesh shall see it together.

This is our hope. This is the faith that I go back to the South with. With this faith we will be able to hew out of the mountain of despair a stone of hope. With this faith we will be able to transform the jangling discords of our nation into a beautiful symphony of brotherhood. With this faith we will

be able to work together, to pray together, to struggle together, to go to jail together, to stand up for freedom together, knowing that we will be free one day.

This will be the day when all of God's children will be able to sing with new meaning.

My country, 'tis of thee,

Sweet land of liberty,

Of thee I sing,

Land where my fathers died,

Land of the pilgrims' pride,

From every mountainside

Let freedom ring.

And if America is to be a great nation this must become true.

So let freedom ring from the prodigious hilltops of New Hampshire.

Let freedom ring from the mighty mountains of New York!

Let freedom ring from the heightening Alleghenies of Pennsylvania!

Let freedom ring from the snowcapped Rockies of Colorado!

Let freedom ring from the curvaceous slopes of California!

But not only that: let freedom ring from Stone Mountain of Georgia!

Let freedom ring from Lookout Mountain of Tennessee!

Let freedom ring from every hill and molehill of Mississippi!

From every mountainside, let freedom ring!

When we let freedom ring, when we let it ring from every village and every hamlet, from every state and every city, we will be able to speed up that day when all of God's children, black men and white men, Jews and Gentiles, Protestants and Catholics, will be able to join hands and sing in the words of the old Negro spiritual, "Free at last! Free at last! Thank God almighty, we are free at last!"