**工具dd**

**dd 命令：convert and copy a file**

**用法：**  
dd if=/PATH/FROM/SRC of=/PATH/TO/DEST bs=# count=#  
if=file 从所命名文件读取而不是从标准输入  
of=file 写到所命名的文件而不是到标准输出  
ibs=size 一次读size个byte  
obs=size 一次写size个byte  
bs=size block size, 指定块大小（既是是ibs也是obs)  
cbs=size 一次转化size个byte  
skip=blocks 从开头忽略blocks个ibs大小的块  
seek=blocks 从开头忽略blocks个obs大小的块  
count=n 复制n个bs  
conv=conversion[,conversion...] 用指定的参数转换文件  
转换参数:  
ascii 转换 EBCDIC 为 ASCII  
ebcdic 转换 ASCII 为 EBCDIC  
lcase 把大写字符转换为小写字符  
ucase 把小写字符转换为大写字符  
nocreat 不创建输出文件  
noerror 出错时不停止  
notrunc 不截短输出文件  
sync 把每个输入块填充到ibs个字节，不足部分用空(NUL)字符补齐  
fdatasync 写完成前，物理写入输出文件

**备份MBR**

dd if=/dev/sda of=/tmp/mbr.bak bs=512 count=1  
dd if=/dev/sda of=mbr.txt bs=64 count=1 skip=446

**破坏MBR中的bootloader**

dd if=/dev/zero of=/dev/sda bs=64 count=1 seek=446

**有一个大与2K的二进制文件fileA。现在想从第64个字节位置开始读取，需要读取的大小是128Byts。又有fileB, 想把上面读取到的128Bytes写到第32个字节开始的位置，替换128Bytes，**实现如下

dd if=fileA of=fileB bs=1 count=128 skip=63 seek=31 conv=notrunc

**备份：**  
**将本地的/dev/sdx整盘备份到/dev/sdy**

dd if=/dev/sdx of=/dev/sdy

**将/dev/sdx全盘数据备份到指定路径的image文件**

dd if=/dev/sdx of=/path/to/image

**备份/dev/sdx全盘数据，并利用gzip压缩，保存到指定路径**

dd if=/dev/sdx | gzip >/path/to/image.gz

**恢复：**  
**将备份文件恢复到指定盘**

dd if=/path/to/image of=/dev/sdx

**将压缩的备份文件恢复到指定盘**

gzip -dc /path/to/image.gz | dd of=/dev/sdx

**拷贝内存资料到硬盘**  
**将内存里的数据拷贝到root目录下的mem.bin文件**

dd if=/dev/mem of=/root/mem.bin bs=1024

**从光盘拷贝iso镜像  
拷贝光盘数据到root文件夹下，并保存为cd.iso文件**

dd if=/dev/cdrom of=/root/cd.iso

**销毁磁盘数据  
利用随机的数据填充硬盘，在某些必要的场合可以用来销毁数据，执行此操作以后，/dev/sda1将无法挂载，创建和拷贝操作无法执行**

dd if=/dev/urandom of=/dev/sda1

**得到最恰当的block size  
通过比较dd指令输出中命令的执行时间，即可确定系统最佳的block size大小**

dd if=/dev/zero of=/root/1Gb.file bs=1024 count=1000000  
dd if=/dev/zero of=/root/1Gb.file bs=2048 count=500000  
dd if=/dev/zero of=/root/1Gb.file bs=4096 count=250000