0x01

常用工具简介：

文件类型	压缩工具	解压工具
.gz	gzip	gunzip
.bz2	bzip2	bunzip2
.zip	zip	unzip

- **tar**：用来创建备份和归档
  - 常用选项：
    - -c：创建一个新归档。
    - -x：从归档中抽取文件。即解压缩。
    - -j：压缩/解压bz2格式tar文件。
    - -z：压缩/解压gz格式tar文件。
    - -f：当与-c选项一起使用时，创建的tar文件使用该选项指定的文件名；当与-x选项一起使用时，则解除该选项指定文件的归档。
    - -t：显示包括在tar文件中的文件列表。
    - -v：显示文件的归档进度。
  - 创建一个归档文件：tar -cvf [归档文件名称] [被归档文件路径1(含文件名)] [被归档文件路径2]
    - 例：

```
d16ug-a1l@mxd:~/Linux/0x00$ ls
0x00.md  1.txt  2.txt
d16ug-a1l@mxd:~/Linux/0x00$ tar -cvf test1_2.tar ./1.txt ./2.txt
./1.txt
./2.txt
d16ug-a1l@mxd:~/Linux/0x00$ ls
0x00.md  1.txt  2.txt  test1_2.tar
d16ug-a1l@mxd:~/Linux/0x00$
```

  - 列出归档文件的内容：tar -tvf [归档文件名称]
    - 例：

```
d16ug-a1l@mxd:~/Linux/0x00$ tar -tvf test1_2.tar
-rw-rw-r-- d16ug-a1l/d16ug-a1l 6 2020-04-29 21:45 ./1.txt
-rw-rw-r-- d16ug-a1l/d16ug-a1l 6 2020-04-29 21:45 ./2.txt
```

  - 解压一个归档文件：tar -xvf [归档文件名称]
    - 例：

```
d16ug-a1l@mxd:~/Linux/0x00$ tar -xvf test1_2.tar
./1.txt
./2.txt
```

  - 创建一个bz2格式的归档文件：tar -cjvf [归档文件名称] [被归档文件路径1(含文件名)] [被归档文件路径2]
    - 例：

```
d16ug-a1l@mxd:~/Linux/0x00$ tar -cjvf test1_2.tar.bz2 1.txt 2.txt
1.txt
2.txt
d16ug-a1l@mxd:~/Linux/0x00$ ls
0x00.md  1.txt  2.txt  test1_2.tar  test1_2.tar.bz2
```

  - 解压一个bz2格式的归档文件：tar -xjvf [归档文件名称]

    - 创建一个gzip格式的归档文件：tar -czvf [归档文件名称] [被归档文件路径1(含文件名)] [被归档文件路径2]
    - 解压一个gzip格式的归档文件：tar -xzvf [归档文件名称]
- **expand**：将输入制表符转换为空格
    - -t：指定一个制表符转化为几个空格
- **unexpand**：将输入空格转换为制表符
- **grep**：在指定文件中搜索关键字符串
    - 常用选项：
        - -b：在输出的每一行前显示包含匹配字符串的行在文件中的字节偏移量。
        - -c：只显示匹配行的数量。
        - -i：比较时不区分大小写。
        - -h：在查找多个文件时，指示grep不要将文件名加入到输出之前。
        - -l：显示首次匹配串所在的文件名并用换行符将其隔开。当在某文件中多次出现匹配串时，不重复显示此文件名。
        - -n：在输出前加上匹配串所在行的行号（文件首行行号为1）。
        - -v：只显示不包含匹配串的行。· - x：整行显示严格匹配的行
- **find**：查找文件的位置
    - 以名称和文件属性为条件查找：
        - -name字符串：查找文件名匹配所给字符串的所有文件，字符串内可用通配符*、？及[ ]。
        - -lname字符串：查找文件名匹配所给字符串的所有符号连接文件，字符串内可用通配符*、？及[ ]。
        - *-gid n：查找ID号为n的用户组的所有文件。*
        - -uid n：查找ID号为n的用户的所有文件。
        - -group字符串：查找用户组名为所给字符串的所有文件。
        - -user字符串：查找用户名为所给字符串的所有文件。
        - -empty：查找大小为0的目录或文件。
        - -path字符串：查找路径名匹配所给字符串的所有文件，字符串内可用通配符*、？及[ ]。
        - -perm权限：查找具有指定权限的文件和目录，权限的表示如711、644。
        - -size n[bckw]：查找指定文件大小的文件，n后面的字符表示单位，默认为b，代表512字节的块。
        - -type x：找类型为x的文件，x为b（块设备文件）、c(字符设备文件)、d（目录文件）、p（命名管道（FIFO））、f（普通文件）、l（符号连接文件）或s（socket文件）。
    - 以时间为条件查找：
        - -amin n：查找n分钟以前被访问过的所有文件。
        - -atime n：查找n天以前被访问过的所有文件。
        - -cmin n：查找n分钟以前文件状态被修改过的所有文件。
        - -ctime n：查找n天以前文件状态被修改过的所有文件。
        - -mmin n：查找n分钟以前文件内容被修改过的所有文件。
        - -mtime n：查找n天以前文件内容被修改过的所有文件。
    - 可执行的操作：
        - -exec命令名称{ }：对符合条件的文件执行所给的Linux命令，而不询问用户是否需要执行该命令。{}表示命令的参数即为所找到的文件；命令的末尾必须以“ \; ”结束。
            - 例：查找当前目录下的所有文件，并对其分别执行ls -al

```
d16ug-a1l@mxd:~/Linux/0x00$ find . -type f -exec ls -al {} \;
-rw-rw-r-- 1 d16ug-a1l d16ug-a1l 10240 4月  29 21:57 ./test1_2.tar
-rw-rw-r-- 1 d16ug-a1l d16ug-a1l 6 4月  29 21:45 ./1.txt
-rw-rw-r-- 1 d16ug-a1l d16ug-a1l 156 4月  29 22:05 ./test1_2.tar.bz2
-rw-rw-r-- 1 d16ug-a1l d16ug-a1l 0 4月  28 15:38 ./0x00.md
-rw-rw-r-- 1 d16ug-a1l d16ug-a1l 22 4月  29 23:08 ./1.c
-rw-rw-r-- 1 d16ug-a1l d16ug-a1l 6 4月  29 21:45 ./2.txt
```

    - -ok命令名称{ }：对符合条件的文件执行所给的Linux命令，与exec不同的是，它会询问用户是否需要执行该命令

- **AWK：**强大的文本（特别是表格）处理程序。命令格式为（{}为命令的内容）：awk '{在文件中查找的内容 查到后执行的动作}' [被查找的文件]。AWK将每个输入行信息分为记录和字段，记录是单行的输入，记录的分隔符是换行，每条记录包含若干字段；默认的字段分隔符是空格或制表符。当AWK读取输入内容时，整条记录被分配给变量$0。各字段以字段分隔符分开，被分配给变量$1、$2、$3，依次增加序号。
    - 例：打印启动日志中每行的第2、4、5、6个字段

```
d16ug-a1l@mxd:/var/log$ sudo awk '{print $2,$4,$5,$6}' boot.log | more
Tue 28 14:16:39 CST
clean, files, 2422225/26082560 blocks
OK Finished Set console
OK Finished Commit a
OK Finished Create Volatile
Network Resolution...
Network Synchronization...
Update about System Boot/Shutdown...
OK Finished Update UTMP
OK Started Network Time
OK Reached target System
OK Reached target System
```

- **sort：**对文件的每行内容进行排序。根据从输入行抽取的一个或多个关键字进行比较来完成。排序关键字定义了用来排序的字符序列。默认情况下以整行为关键字，按ASCII字符顺序进行排序。
    - 改变默认设置的选项：
        - -m：若给定文件已排序，则合并文件。
        - -c：检查给定文件是否已排好序，如果没有排序，则打印出错信息，并以状态值1退出。
        - -u：对排序后认为相同的行只保留其中一行。
        - -o：输出文件将排序输出写到输出文件中而不是标准输出，如果输出文件是输入文件之一，则sort先将该文件的内容写入一个临时文件，然后再排序，写输出结果。
    - 改变缺省排序规则的选项：
        - -d：按字典顺序排序，比较时仅字母、数字、空格和制表符有意义。
        - -f：将小写字母与大写字母同等对待。
        - -I：忽略非打印字符。
        - -M：作为月份比较，如“JAN” < “FEB” < 1/4 < “DEC”。
        - -r：按逆序输出排序结果。
        - -b：在每行中寻找排序关键字时忽略前导的空白（空格和制表符）。
- **nl：**为输入的文件每一行添加一个行号
    - 例：

```
d16ug-a1l@mxd:~/Linux/0x00$ nl awk.c
     1  #include <stdio.h>

     2  int main(int argc, char* argv[])
     3  {
     4          printf("hello.world1!\n");
     5          printf("hello.world2!\n");
     6  }
```

键盘组合键命令：

- ^C：< ctrl+c > 中断程序。

- ^\：< ctrl+\ > 退出程序。
- ^S：< ctrl+S > 结束程序。
- ^Z：< ctrl+Z > 挂起程序。