Linux其他常见命令的使用

## wc命令

Linux系统中的wc(Word Count)命令的功能为统计指定文件中的字节数、字数、行数，并将统计结果显示输出。

***命令格式***：wc [选项] 文件

***命令功能***：

统计指定文件中的字节数、字数、行数，并将统计结果显示输出。该命令统计指定文件中的字节数、字数、行数。如果没有给出文件名，则从标准输入读取。wc同时也给出所指定文件的总统计数。

命令参数：

* -c 统计字节数。
* -l 统计行数。
* -m 统计字符数。这个标志不能与 -c 标志一起使用。
* -w 统计字数。一个字被定义为由空白、跳格或换行字符分隔的字符串。
* --version 显示版本信息

## wget命令

wget命令用来从指定的URL下载文件。wget非常稳定，它在带宽很窄的情况下和不稳定网络中有很强的适应性，如果是由于网络的原因下载失败，wget会不断的尝试，直到整个文件下载完毕。如果是服务器打断下载过程，它会再次联到服务器上从停止的地方继续下载。这对从那些限定了链接时间的服务器上下载大文件非常有用。  
wget (选项) (参数)

## fuser命令

fuser 可以显示出当前哪个程序在使用磁盘上的某个文件、挂载点、甚至网络端口，并给出程序进程的详细信息。fuser显示使用指定文件或者文件系统的进程ID。默认情况下每个文件名后面跟一个字母表示访问类型。访问类型如下：

* c 代表当前目录
* e 可执行的被运行
* f 打开的文件。默认不显示。
* F 打开的文件，用于写操作。默认不显示。
* r 根目录。
* m 映射文件或者共享库。

当指定的文件没有被访问，或者出现错误的时候，fuser会返回非零。为了查看使用tcp和udp套接字的进程，需要-n选项并指定名称空间。默认IpV4和IpV6都会显示。套接字可以是本地的或者是远程的端口，和远程的地址。所有的域是可选的，但是其前面的','必须存在。如下：  [lcl\_port][,[rmt\_host][,[rmt\_port]]]

对于ip地址和port，名称和数字表示都可以使用。

fuser只把PID输出到标准输出，其他的都输出到标准错误输出。

常用选项：

* -a 显示所有命令行中指定的文件，默认情况下被访问的文件才会被显示。
* -c 和-m一样，用于POSIX兼容。
* -k 杀掉访问文件的进程。如果没有指定-signal就会发送SIGKILL信号。
* -i 杀掉进程之前询问用户，如果没有-k这个选项会被忽略。
* -l 列出所有已知的信号名称。
* -m name 指定一个挂载文件系统上的文件或者被挂载的块设备（名称name）。这样所有访问这个文件或者文件系统的进程都会被列出来。如果指定的是一个目录会自动转换成"name/",并使用所有挂载在那个目录下面的文件系统。
* -n space 指定一个不同的命名空间(space).这里支持不同的空间文件(文件名，此处默认)、tcp(本地tcp端口)、udp(本地udp端口)。对于端口，可以指定端口号或者名称，如果不会引起歧义那么可以使用简单表示的形式，例如：name/space (即形如:80/tcp之类的表示)。
* -s 静默模式，这时候-u,-v会被忽略。-a不能和-s一起使用。
* -signal 使用指定的信号，而不是用SIGKILL来杀掉进程。可以通过名称或者号码来表示信号(例如-HUP,-1),这个选项要和-k一起使用，否则会被忽略。
* -u 在每个PID后面添加进程拥有者的用户名称。
* -v 详细模式。输出似ps命令的输出，包含PID,USER,COMMAND等许多域,如果是内核访问的那么PID为kernel.
* -V 输出版本号。
* -4 使用IPV4套接字,不能和-6一起应用，只在-n的tcp和udp的命名存在时不被忽略。
* -6 使用IPV6套接字,不能和-4一起应用，只在-n的tcp和udp的命名存在时不被忽略。
* -重置所有的选项，把信号设置为SIGKILL.

举例：

|  |
| --- |
| --显示使用某个文件的进程信息  #这个命令在umount的时候很有用，可以找到还有哪些用到这个设备了。  fuser -um /dev/xvdb |
| --杀掉所有以任何形式访问文件系统 /home的进程  fuser -km /home |
| --查看那些程序使用tcp的80端口  fuser -n tcp 80  或$fuser -v -n tcp 80  或$fuser -v 80/tcp |

## xargs命令

### 简介

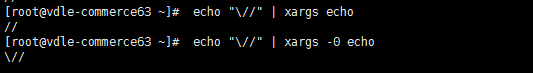
很多命令不支持|管道来传递参数，而日常工作中有有这个必要，所以就有了xargs命令。xargs 可以读入 stdin 的资料，并且以空白字元或断行字元作为分辨，将 stdin 的资料分隔成为 arguments 。 因为是以空白字元作为分隔，所以，如果有一些档名或者是其他意义的名词内含有空白字元的时候， xargs 可能就会误判。

find /sbin -perm +700 |ls -l #命令是错误的

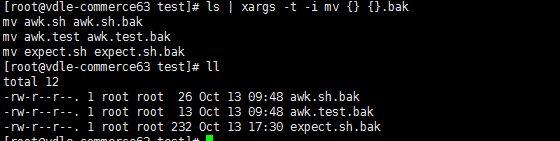
find /sbin -perm +700 |xargs ls -l #正确的

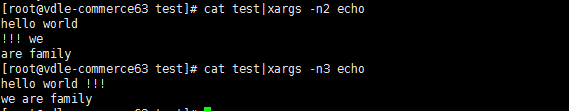
### 选项解释

* **-0：**当sdtin含有特殊字元时候，将其当成一般字符，想/'空格等
* **-a file：**从文件中读入作为sdtin
* **-e flag** ：注意有的时候可能会是-E，flag必须是一个以空格分隔的标志，当xargs分析到含有flag这个标志的时候就停止。
* **-p ：**当每次执行一个argument的时候询问一次用户。
* **-n num：**后面加次数，表示命令在执行的时候一次用的argument的个数，默认是用所有的。
* **-t ：**表示先打印命令，然后再执行。
* **-i** -I：这得看linux支持了，将xargs的每项名称，一般是一行一行赋值给{}，可以用{}代替。
* **-r**no-run-if-empty： 当xargs的输入为空的时候则停止xargs，不用再去执行了。
* **-s num** ：命令行的最好字符数，指的是xargs后面那个命令的最大命令行字符数。
* -**L** /l num ：Use at most max-lines nonblank input lines per command line.-s是含有空格的。
* **-d delim**分隔符：默认的xargs分隔符是回车，argument的分隔符是空格，这里修改的是xargs的分隔符
* **-x exit：**主要是配合-s使用。
* -**P :**修改最大的进程数，默认是1，为0时候为as many as it can









### 应用

通过这个命令你可以有效地处理多个文件，节省你的时间和精力。在这篇教程中，你可以学到如何一次性对多个文件执行命令或脚本操作，再也不用担心像单独处理无数个日志或数据文件那样吓人的任务了。xargs命令有两个要点。

在使用find命令的-exec选项处理匹配到的文件时， find命令将所有匹配到的文件一起传递给exec执行。但有些系统对能够传递给exec的命令长度有限制，这样在find命令运行几分钟之后，就会出现溢出错误。错误信息通常是“参数列太长”或“参数列溢出”。这就是xargs命令的用处所在，特别是与find命令一起使用。 find命令把匹配到的文件传递给xargs命令，而xargs命令每次只获取一部分文件而不是全部，不像-exec选项那样。这样它可以先处理最先获取的一部分文件，然后是下一批，并如此继续下去。

* 你必须列出目标文件。
* 你必须指定对每个文件需要执行的命令或脚本。

find . -name 'hell\*' | sort | xargs wc –l

## nc命令

nc命令是netcat命令的简称，都是用来设置路由器。

nc/netcat(选项)(参数)

选项：

* -g<网关>：设置路由器跃程通信网关，最多设置8个；
* -G<指向器数目>：设置来源路由指向器，其数值为4的倍数；
* -h：在线帮助；
* -i<延迟秒数>：设置时间间隔，以便传送信息及扫描通信端口；
* -l：使用监听模式，监控传入的资料；
* -n：直接使用ip地址，而不通过域名服务器；
* -o<输出文件>：指定文件名称，把往来传输的数据以16进制字码倾倒成该文件保存；
* -p<通信端口>：设置本地主机使用的通信端口；
* -r：指定源端口和目的端口都进行随机的选择；
* -s<来源位址>：设置本地主机送出数据包的IP地址；
* -u：使用UDP传输协议；
* -v：显示指令执行过程；
* -w<超时秒数>：设置等待连线的时间；
* -z：使用0输入/输出模式，只在扫描通信端口时使用。

参数：

* 主机：指定主机的IP地址或主机名称；
* 端口号：可以是单个整数或者是一个范围。

### 远程拷贝文件

|  |
| --- |
| 接收文件端监听端口：nc -lp 1234 > install.log  发送端向指定服务器端口发送数据：nc -w 1 115.29.241.37 1234 < install.log |
| tar -cvzv /…/file | nc –q 1 server\_ip 1234  nc -l -p 1234 | tar -zxvf |

### 克隆硬盘或分区

操作与上面的拷贝是雷同的，只需要由dd获得硬盘或分区的数据，然后传输即可。克隆硬盘或分区的操作，不应在已经mount的的系统上进行。

|  |
| --- |
| dd if=/dev/sda | nc 192.168.228.222 1234  nc -l -p 1234 | dd of=/dev/sda |

### 端口扫描

|  |
| --- |
| nc -v -w 1 172.16.129.201 -z 1-1000 |

### 动态输入命令

交互敲入命令：echo "flush\_all" | nc 10.161.171.195 11211

## curl命令

curl命令是一个利用URL规则在命令行下工作的文件传输工具。它支持文件的上传和下载，所以是综合传输工具，但按传统，习惯称curl为下载工具。作为一款强力工具，curl支持包括HTTP、HTTPS、ftp等众多协议，还支持POST、cookies、认证、从指定偏移处下载部分文件、用户代理字符串、限速、文件大小、进度条等特征。

使用语法：curl 选项 参数

选项：

* -A/--user-agent ：设置用户代理发送给服务器
* -b/--cookie cookie：字符串或文件读取位置
* -c/--cookie-jar ：操作结束后把cookie写入到这个文件中
* -d/--data ：HTTP POST方式传送数据
* -D/--dump-header：把header信息写入到该文件中
* -G/--get：以get的方式来发送数据
* -H/--header ：自定义头信息传递给服务器
* -I/--head ：只显示请求头信息
* -o/--output：把输出写到该文件中，--progress显示进度条
* -s/--silent：静默模式。不输出任何东西
* -u/--user ：设置服务器的用户和密码

### 文件下载

|  |
| --- |
| #抓取网页  curl http://www.helpedu.com -o helpeduHead.txt --progress |

### linux curl处理cookies

|  |
| --- |
| #cookies保存到/tmp/cookies文件  curl -c /tmp/cookies http://www.baidu.com  #发送cookies  curl -b "key1=val1;key2=val2;" http://www.baidu.com #发送cookies文本  curl -b /tmp/cookies http://www.baidu.com #从文件中读取cookies |

### linux curl发送数据

|  |
| --- |
| #linux curl get方式提交数据  curl -G -d "name=value&name2=value2" <http://www.baidu.com>  #linux curl post方式提交数据  curl -d "name=value&name2=value2" http://www.baidu.com #post数据  curl -d a=b&c=d&txt@/tmp/txt http://www.baidu.com #post文件 |

### linux curl http header处理

|  |
| --- |
| #设置http请求头信息  curl -A "Mozilla/5.0 Firefox/21.0" http://www.baidu.com #设置http请求头User-Agent  curl -e "http://pachong.org/" http://www.baidu.com #设置http请求头Referer  curl -H "Connection:keep-alive \n User-Agent: Mozilla/5.0" <http://www.aiezu.com>  curl -I http://www.aiezu.com #仅仅返回header  curl -D /tmp/header http://www.aiezu.com #将 header保存到/tmp/header文件 |

### linux curl认证

|  |
| --- |
| #用户名密码认证  curl -u aiezu:password http://www.aiezu.com  #采用证书认证  curl -E mycert.pem https://www.baidu.com |

### 其他

|  |
| --- |
| #以“#”号输出进度条  curl -# http://www.baidu.com  #保存http响应到/tmp/aiezu  curl -o /tmp/aiezu http://www.baidu.com |