[**每天一个linux命令（51）：lsof命令**](http://www.cnblogs.com/peida/archive/2013/02/26/2932972.html)

lsof（list open files）是一个列出当前系统打开文件的工具。在linux环境下，任何事物都以文件的形式存在，通过文件不仅仅可以访问常规数据，还可以访问网络连接和硬件。所以如传输控制协议 (TCP) 和用户数据报协议 (UDP) 套接字等，系统在后台都为该应用程序分配了一个文件描述符，无论这个文件的本质如何，该文件描述符为应用程序与基础操作系统之间的交互提供了通用接口。因为应用程序打开文件的描述符列表提供了大量关于这个应用程序本身的信息，因此通过lsof工具能够查看这个列表对系统监测以及排错将是很有帮助的。

**1．命令格式：**

lsof [参数][文件]

**2．命令功能：**

用于查看你进程开打的文件，打开文件的进程，进程打开的端口(TCP、UDP)。找回/恢复删除的文件。是十分方便的系统监视工具，因为 lsof 需要访问核心内存和各种文件，所以需要root用户执行。

lsof打开的文件可以是：

1.普通文件

2.目录

3.网络文件系统的文件

4.字符或设备文件

5.(函数)共享库

6.管道，命名管道

7.符号链接

8.网络文件（例如：NFS file、网络socket，unix域名socket）

9.还有其它类型的文件，等等

**3．命令参数：**

-a 列出打开文件存在的进程

-c<进程名> 列出指定进程所打开的文件

-g  列出GID号进程详情

-d<文件号> 列出占用该文件号的进程

+d<目录>  列出目录下被打开的文件

+D<目录>  递归列出目录下被打开的文件

-n<目录>  列出使用NFS的文件

-i<条件>  列出符合条件的进程。（4、6、协议、:端口、 @ip ）

-p<进程号> 列出指定进程号所打开的文件

-u  列出UID号进程详情

-h 显示帮助信息

-v 显示版本信息

**4．使用实例：**

**实例1：无任何参数**

**命令：**

lsof

**输出：**

[root@localhost ~]# lsof

COMMAND     PID USER   FD      TYPE             DEVICE     SIZE       NODE NAME  
init          1 root  cwd       DIR                8,2     4096          2 /  
init          1 root  rtd       DIR                8,2     4096          2 /  
init          1 root  txt       REG                8,2    43496    6121706 /sbin/init  
init          1 root  mem       REG                8,2   143600    7823908 /lib64/ld-2.5.so  
init          1 root  mem       REG                8,2  1722304    7823915 /lib64/libc-2.5.so  
init          1 root  mem       REG                8,2    23360    7823919 /lib64/libdl-2.5.so  
init          1 root  mem       REG                8,2    95464    7824116 /lib64/libselinux.so.1  
init          1 root  mem       REG                8,2   247496    7823947 /lib64/libsepol.so.1  
init          1 root   10u     FIFO               0,17                1233 /dev/initctl  
migration     2 root  cwd       DIR                8,2     4096          2 /  
migration     2 root  rtd       DIR                8,2     4096          2 /  
migration     2 root  txt   unknown                                        /proc/2/exe  
ksoftirqd     3 root  cwd       DIR                8,2     4096          2 /  
ksoftirqd     3 root  rtd       DIR                8,2     4096          2 /  
ksoftirqd     3 root  txt   unknown                                        /proc/3/exe  
migration     4 root  cwd       DIR                8,2     4096          2 /  
migration     4 root  rtd       DIR                8,2     4096          2 /  
migration     4 root  txt   unknown                                        /proc/4/exe  
ksoftirqd     5 root  cwd       DIR                8,2     4096          2 /  
ksoftirqd     5 root  rtd       DIR                8,2     4096          2 /  
ksoftirqd     5 root  txt   unknown                                        /proc/5/exe  
events/0      6 root  cwd       DIR                8,2     4096          2 /  
events/0      6 root  rtd       DIR                8,2     4096          2 /  
events/0      6 root  txt   unknown                                        /proc/6/exe  
events/1      7 root  cwd       DIR                8,2     4096          2 /

**说明：**

**lsof输出各列信息的意义如下：**

COMMAND：进程的名称

PID：进程标识符

PPID：父进程标识符（需要指定-R参数）

USER：进程所有者

PGID：进程所属组

FD：文件描述符，应用程序通过文件描述符识别该文件。如cwd、txt等

（1）cwd：表示current work dirctory，即：应用程序的当前工作目录，这是该应用程序启动的目录，除非它本身对这个目录进行更改

（2）txt ：该类型的文件是程序代码，如应用程序二进制文件本身或共享库，如上列表中显示的 /sbin/init 程序

（3）lnn：library references (AIX);

（4）er：FD information error (see NAME column);

（5）jld：jail directory (FreeBSD);

（6）ltx：shared library text (code and data);

（7）mxx ：hex memory-mapped type number xx.

（8）m86：DOS Merge mapped file;

（9）mem：memory-mapped file;

（10）mmap：memory-mapped device;

（11）pd：parent directory;

（12）rtd：root directory;

（13）tr：kernel trace file (OpenBSD);

（14）v86  VP/ix mapped file;

（15）0：表示标准输出

（16）1：表示标准输入

（17）2：表示标准错误

一般在标准输出、标准错误、标准输入后还跟着文件状态模式：r、w、u等

（1）u：表示该文件被打开并处于读取/写入模式

（2）r：表示该文件被打开并处于只读模式

（3）w：表示该文件被打开并处于

（4）空格：表示该文件的状态模式为unknow，且没有锁定

（5）-：表示该文件的状态模式为unknow，且被锁定

同时在文件状态模式后面，还跟着相关的锁

（1）N：for a Solaris NFS lock of unknown type;

（2）r：for read lock on part of the file;

（3）R：for a read lock on the entire file;

（4）w：for a write lock on part of the file;（文件的部分写锁）

（5）W：for a write lock on the entire file;（整个文件的写锁）

（6）u：for a read and write lock of any length;

（7）U：for a lock of unknown type;

（8）x：for an SCO OpenServer Xenix lock on part      of the file;

（9）X：for an SCO OpenServer Xenix lock on the      entire file;

（10）space：if there is no lock.

TYPE：文件类型，如DIR、REG等，常见的文件类型

（1）DIR：表示目录

（2）CHR：表示字符类型

（3）BLK：块设备类型

（4）UNIX： UNIX 域套接字

（5）FIFO：先进先出 (FIFO) 队列

（6）IPv4：网际协议 (IP) 套接字

DEVICE：指定磁盘的名称

SIZE：文件的大小

NODE：索引节点（文件在磁盘上的标识）

NAME：打开文件的确切名称

**实例2：**查看谁正在使用某个文件，也就是说查找某个文件相关的进程

**命令：**

lsof /bin/bash

**输出：**

[root@localhost ~]# lsof /bin/bash  
COMMAND   PID USER  FD   TYPE DEVICE   SIZE    NODE NAME  
bash    24159 root txt    REG    8,2 801528 5368780 /bin/bash  
bash    24909 root txt    REG    8,2 801528 5368780 /bin/bash  
bash    24941 root txt    REG    8,2 801528 5368780 /bin/bash  
[root@localhost ~]#

**说明：**

**实例3：递归查看某个目录的文件信息**

**命令：**

lsof test/test3

**输出：**

[root@localhost ~]# cd /opt/soft/  
[root@localhost soft]# lsof test/test3  
COMMAND   PID USER   FD   TYPE DEVICE SIZE    NODE NAME  
bash    24941 root  cwd    DIR    8,2 4096 2258872 test/test3  
vi      24976 root  cwd    DIR    8,2 4096 2258872 test/test3  
[root@localhost soft]#

**说明：**

使用了+D，对应目录下的所有子目录和文件都会被列出

**实例4：不使用+D选项，遍历查看某个目录的所有文件信息的方法**

**命令：**

 lsof |grep 'test/test3'

**输出：**

[root@localhost soft]# lsof |grep 'test/test3'

bash      24941 root  cwd       DIR                8,2     4096    2258872 /opt/soft/test/test3  
vi        24976 root  cwd       DIR                8,2     4096    2258872 /opt/soft/test/test3  
vi        24976 root    4u      REG                8,2    12288    2258882 /opt/soft/test/test3/.log2013.log.swp  
[root@localhost soft]#

**说明：**

**实例5：列出某个用户打开的文件信息**

**命令**：

lsof -u username

**说明:**

-u 选项，u其实是user的缩写

**实例6：列出某个程序进程所打开的文件信息**

**命令：**

lsof -c mysql

**说明:**

 -c 选项将会列出所有以mysql这个进程开头的程序的文件，其实你也可以写成 lsof | grep mysql, 但是第一种方法明显比第二种方法要少打几个字符了

**实例7：列出多个进程多个打开的文件信息**

**命令：**

lsof -c mysql -c apache

**实例8：列出某个用户以及某个进程所打开的文件信息**

**命令：**

lsof  -u test -c mysql

**说明：**

用户与进程可相关，也可以不相关

**实例9：列出除了某个用户外的被打开的文件信息**

**命令：**

lsof -u ^root

**说明：**

^这个符号在用户名之前，将会把是root用户打开的进程不让显示

**实例10：通过某个进程号显示该进行打开的文件**

**命令：**

lsof -p 1

**实例11：列出多个进程号对应的文件信息**

**命令：**

lsof -p 1,2,3

**实例12：列出除了某个进程号，其他进程号所打开的文件信息**

**命令：**

lsof -p ^1

**实例13：列出所有的网络连接**

**命令：**

lsof -i

**实例14：列出所有tcp 网络连接信息**

**命令：**

lsof -i tcp

**实例15：列出所有udp网络连接信息**

**命令：**

lsof -i udp

**实例16：列出谁在使用某个端口**

**命令：**

lsof -i :3306

**实例17：列出谁在使用某个特定的udp端口**

**命令：**

lsof -i udp:55

**或者：特定的tcp端口**

**命令：**

lsof -i tcp:80

**实例18：列出某个用户的所有活跃的网络端口**

**命令：**

lsof -a -u test -i

**实例19：列出所有网络文件系统**

**命令：**

lsof -N

**实例20：域名socket文件**

**命令：**

lsof -u

**实例21：某个用户组所打开的文件信息**

**命令：**

lsof -g 5555

**实例22：根据文件描述列出对应的文件信息**

**命令：**

lsof -d description(like 2)

例如：lsof  -d  txt

例如：lsof  -d  1

例如：lsof  -d  2

**说明：**

0表示标准输入，1表示标准输出，2表示标准错误，从而可知：所以大多数应用程序所打开的文件的 FD 都是从 3 开始

**实例23：根据文件描述范围列出文件信息**

**命令：**

lsof -d 2-3

**实例24：列出COMMAND列中包含字符串" sshd"，且文件描符的类型为txt的文件信息**

**命令：**

lsof -c sshd -a -d txt

**输出：**

[root@localhost soft]# lsof -c sshd -a -d txt  
COMMAND   PID USER  FD   TYPE DEVICE   SIZE    NODE NAME  
sshd     2756 root txt    REG    8,2 409488 1027867 /usr/sbin/sshd  
sshd    24155 root txt    REG    8,2 409488 1027867 /usr/sbin/sshd  
sshd    24905 root txt    REG    8,2 409488 1027867 /usr/sbin/sshd  
sshd    24937 root txt    REG    8,2 409488 1027867 /usr/sbin/sshd  
[root@localhost soft]#   
[root@localhost soft]#

**实例25：列出被进程号为1234的进程所打开的所有IPV4 network files**

**命令**：

lsof -i 4 -a -p 1234

**实例26：列出目前连接主机peida.linux上端口为：20，21，22，25，53，80相关的所有文件信息，且每隔3秒不断的执行lsof指令**

**命令：**

lsof -i @peida.linux:20,21,22,25,53,80  -r  3