A Unix Utility You Should Know About: lsof

Article Number: 548 | Rating: Unrated | Last Updated: Tue, Apr 16, 2019 5:49 PM

A Unix Utility You Should Know About: lsof

This is the third post in the article series about Unix and Linux utilities that you should know about. In this post I will take you through the useful lsof tool. If netcat was called the Swiss Army Knife of Network Connections, then I'd call lsof the Swiss Army Knife of Unix debugging.

Lsof follows Unix philosophy closely. It does just one task and it does it perfectly -- it lists information about files opened by processes. An open file may be a regular file, a directory, a NFS file, a block special file, a character special file, a shared library, a regular pipe, a named pipe, a symbolic link, a socket stream, an Internet socket, a UNIX domain socket, and many others. Since almost everything in Unix is a file, you can imagine how incredibly useful lsof is!

See the first post on pipe viewer for the introduction to this article series. If you are interested in articles like this one, I suggest that you subscribe to my rss feed to receive my future posts automatically!

How to use lsof?

In this article I will try to present lsof based on as many use cases as I can think of. Let's start with the simplest (that you probably already know) and proceed to more complicated ones.

List all open files.

lsof
Running lsof without any arguments lists all open files by all processes.

Find who's using a file.

lsof /path/to/file
With an argument of a path to a file, lsof lists all the processes, which are using the file in some way.

You may also specify several files, which lists all the processes, which are using all the files:

lsof /path/to/file1 /path/to/file2
Find all open files in a directory recursively.

lsof +D /usr/lib
With the +D argument lsof finds all files in the specified directory and all the subdirectories.

Note that it's slower than the usual version with grep:

lsof | grep '/usr/lib'
It's slower because +D first finds all the files and only then does the output.

List all open files by a user.

lsof -u pkrumins
The -u option (think user) limits output of files opened only by user pkrumins.

You can use comma separated list of values to list files open by several users:

lsof -u rms,root
This will list all the files that are open by users rms and root.

Another way to do the same is by using the -u option twice:

lsof -u rms -u root
Find all open files by program's name.

lsof -c apache

The -c option selects the listing of files for processes whose name begins with apache.

So instead of writing:

lsof | grep foo
You can now write the shorter version:

lsof -c foo
In fact, you can specify just the beginning part of the process name you're looking for:

lsof -c apa
This will list all the open files by a processes whose starts with apa.

You can also specify several -c options to output open files by several processes:

lsof -c apache -c python
This will list all open files by apache and python.

List all open files by a user OR process.

lsof -u pkrumins -c apache
Lsof options can be combined. The default is to OR between options. It means it will combine outputs of -u pkrumins and -c apache producing a listing of all open files by pkrumins and all open files by apache.

List all open files by a user AND process.

lsof -a -u pkrumins -c bash
Notice the -a option. It combines the options with AND. The output listing is files opened by bash, which is run under pkrumins user.

List all open files by all users EXCEPT root.

lsof -u ^root
Notice the ^ character before root username. It negates the match and causes lsof print all open files by all users who are not root.

List all open files by the process with PID.

lsof -p 1
The -p option (think PID) filters out open files by program's id.

Remember that you can select multiple PIDs by either comma separating the list or using multiple -p arguments:

lsof -p 450,980,333
This selects processes with PIDs 450, 980 and 333.

List all open files by all the processes EXCEPT process with PID.

lsof -p ^1
Here the negation operator ^ is used again. It inverts the list and does not include process with PID 1.

List all network connections.

lsof -i
Lsof with -i option lists all processes with open Internet sockets (TCP and UDP).

List all TCP network connections.

lsof -i tcp
The -i argument can take several options, one of them is tcp. The tcp option forces lsof to list only processes with TCP sockets.

List all UDP network connections.

lsof -i udp
The udp option causes lsof to list processes with UDP sockets.

Find who's using a port.

lsof -i :25
The :25 option to -i makes lsof find processes using TCP or UDP port 25.

You may also use service port name (found in /etc/services) rather than port number:

lsof -i :smtp

Find who's using a specific UDP port.

lsof -i udp:53
Similarly, to find who's using a TCP port, use:

lsof -i tcp:80
Find all network activity by user.

lsof -a -u hacker -i
Here the -a option combines -u and -i to produce listing of network file usage by user hacker.

List all NFS (Network File System) files.

lsof -N
This option is easy to remember because -N is NFS.

List all Unix domain socket files.

lsof -U
This option is also easy to remember because -U is Unix.

List all files for processes with a specific group id.

lsof -g 1234
Process groups are used to logically group processes. This example finds all files opened by processes with PGID 1234.

List all files associated with specific file descriptors.

lsof -d 2
This lists all files that have been opened as file descriptor 2.

You may also specify ranges of file descriptors:

lsof -d 0-2
This would list all files with file descriptors 0, 1 and 2.

There are also many special values, such as mem, that lists memory-mapped files:

lsof -d mem
Or txt for programs loaded in memory and executing:

lsof -d txt
Output PIDs of processes using some resource.

lsof -t -i
The -t option outputs only PIDs of processes. Used together with -i it outputs PIDs of all processes with network connections. It's easy to kill all processes that use network:

kill -9 `lsof -t -i`
Repeat listing files.

lsof -r 1
The -r option makes lsof repeatedly list files until interrupted. Argument 1 means repeat the listing every 1 second. This option is best combined with a narrower query such as monitoring user network file activity:

lsof -r 1 -u john -i -a

Posted - Tue, Apr 16, 2019 5:49 PM. This article has been viewed 1914 times.

Online URL: http://kb.ictbanking.net/article.php?id=548