

Practice Using Pipes

Here is a list of small tasks that can be done by chaining together 2 or more unix standard commands into a pipeline to get the results. First one to e-mail me a correct and complete solution for a particular task will receive fame and glory for their brilliant solution, and maybe a few bonus points...

1. The `ps` command can be used to get a list of processes running on the system. Create a pipeline to find all of the processes that you are running on the system.
2. The `tail` command can be used to continuously monitor the output being generated to a file. Write a filter to watch a file continuously and monitor it for when certain events or output is generated. For example, the `/var/log/messages` file is where lots of unix services dump log messages. Monitor this file and look for when new kernel messages are generated.
3. `du` can be used to determine the amount of disk usage being used by a directory. Determine which directory is using the most disk space (hint `du` and `sort` would be useful here).
4. Print out only the name and modification time of the oldest file in the current directory. Do the same but for the 3 oldest files.
5. `find` is a powerful command that can be used to find files based on not only the name of the file matching a pattern, but by modification time, owner, etc. Find all of the directories under the current directory, but filter out all of the “.svn” subversion directories from the list of directories.