Mostly, we find `-o` along with `-e` as `-eo`, but when filters are applied `-o` should act alone, as mentioned in this section.

TTY filter for ps

The `ps` output can be selected by specifying the TTY to which the process is attached. Use the `-t` option to specify the TTY list, as follows:

```
$ ps -t TTY1, TTY2 ..
```

For example:

```
$ ps -t pts/0,pts/1
  PID TTY          TIME CMD
 1238 pts/0    00:00:00 bash
 1835 pts/1    00:00:00 bash
 1864 pts/0    00:00:00 ps
```

Information about process threads

Usually, information about process threads are hidden in the `ps` output. We can show information about threads in the `ps` output by adding the `-L` option. Then, it will show two columns NLWP and NLP. NLWP is the thread count for a process and NLP is the thread ID for each entry in `ps`. For example:

```
$ ps -eLf
```

Or

```
$ ps -eLf --sort -nlwp | head
UID        PID  PPID   LWP  C NLWP STIME TTY          TIME CMD
root       647     1   647  0   64 14:39 ?        00:00:00 /usr/sbin/
console-kit-daemon --no-daemon
root       647     1   654  0   64 14:39 ?        00:00:00 /usr/sbin/
console-kit-daemon --no-daemon
root       647     1   656  0   64 14:39 ?        00:00:00 /usr/sbin/
console-kit-daemon --no-daemon
root       647     1   657  0   64 14:39 ?        00:00:00 /usr/sbin/
console-kit-daemon --no-daemon
root       647     1   658  0   64 14:39 ?        00:00:00 /usr/sbin/
console-kit-daemon --no-daemon
root       647     1   659  0   64 14:39 ?        00:00:00 /usr/sbin/
console-kit-daemon --no-daemon
```

```
root       647     1   660  0   64 14:39 ?        00:00:00 /usr/sbin/
console-kit-daemon --no-daemon
root       647     1   662  0   64 14:39 ?        00:00:00 /usr/sbin/
console-kit-daemon --no-daemon
root       647     1   663  0   64 14:39 ?        00:00:00 /usr/sbin/
console-kit-daemon --no-daemon
```

This command lists 10 processes with a maximum number of threads.

Specifying output width and columns to be displayed

We can specify the columns to be displayed in the `ps` output using the user-defined output format specifier `-o`. Another way to specify the output format is with standard options. Practice them according to your usage style. Try these options:

- `-f` `ps -ef`
- `u` `ps -e u`
- `ps` `ps -e w` (`w` stands for wide output)

Showing environment variables for a process

Understanding which environment variables a process is dependent on is a very useful bit of information we might need. The way a process behaves might be heavily dependent on the environmental variables set. We can debug and make use of environment data for fixing several problems related to the running of processes.

In order to list environment variables along with `ps` entries, use:

```
$ ps -eo cmd e
```

For example:

```
$ ps -eo pid,cmd  e | tail -n 3
 1162 hald-addon-acpi: listening on acpid socket /var/run/acpid.socket
 1172 sshd: slynux [priv]
 1237 sshd: slynux@pts/0
 1238 -bash USER=slynux LOGNAME=slynux HOME=/home/slynux PATH=/usr/
local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games
MAIL=/var/mail/slynux SHELL=/bin/bash SSH_CLIENT=10.211.55.2 49277 22
SSH_CONNECTION=10.211.55.2 49277 10.211.55.4 22 SSH_TTY=/dev/pts/0
TERM=xterm-color LANG=en_IN XDG_SESSION_COOKIE=d1e96f5cc8a7a3bc3a0a73e44c
95121a-1286499339.592429-1573657095
```