The journal of Apple technology.

Volume Number: 18 (2002)
Issue Number: 10
Column Tag: Section 7

Process Control

ps, top, kill, ...

by Rich Morin

A recent foray into Internet Radio conflicted in some manner with the OSX screen saver on my desktop machine. As a result, I was unable to get the screen saver to go away. I tried all of the usual tricks: hitting mouse buttons, moving the mouse, advancing to keyboard keys (e.g., shift, return) and finally "magic key combinations" (e.g., command-option-escape). No luck.

On Mac OS 9, my next steps would have been to try a "three-finger salute" (i.e., command-control-power), followed by pushing the reset button and if need be, pulling the plug. This being a BSD-based machine, however, I had a better solution at hand.

I went to another machine on the local net, logged into my desktop machine, and killed a couple of sorely-confused processes. I then logged out and returned to my desk, flushed with victory and the knowledge that I had a useful Section 7 topic in hand.

Advance Preparation

My "remote login" trick wouldn't have worked without a bit of advance preparation. OSX, for very good reasons, turns off most "remote services", by default. I don't recommend turning services on, willy-nilly, but ftp and ssh are so handy that I wouldn't want to live without them. And, because my desktop machine is hidden behind a firewall, I consider the risk to be pretty minimal (your mileage may vary).

In any case, here's how to allow remote ftp and ssh on an OSX 10.2 machine: In the System Preferences application, select the Sharing dialog. Under the "Services" tab, turn on the checkboxes labelled "FTP Access" and "Rmote Login".

You will also need a way to "find" your machine from the other one. This can be achieved in any number of ways, including DNS (Domain Name System), /etc/hosts files, etc. Your network administrator will (or should :-) know how to make this happen on your local network. As a last resort, you can use your machine's IP (Internet Protocol) address as a name.

Logging In

Because ftp and ssh aren't OSX-specific tools, you can log in remotely from almost any operating system. This being an OSX-related column, however, I'll assume that you're running OSX. If you're not, you'll have to deal with the vagaries of installing and starting up the needed clients on the machine you're on. On OSX, just start up a Terminal window and type:

```
% ssh <me>@<mydesk>
<me>@<mydesk>s password:
```

The first time you do this from a given machine, you'll be asked whether you trust the SSH keys on your desktop machine. Say yes (:-). After entering your password, you'll get a prompt from your desktop machine:

```
<me>@<mydesk> [~] 1:
```

There are all sorts of things you can do at this point, but try not to get too excited and do something you'll regret. Think "delicate surgery", rather than "hack and slash". Thus, the first thing we want to do is find out a bit about the programs that are running over on the desktop machine. To do this, we'll use the top command, which displays system usage statistics and then lists the top processes, in terms of CPU utilization:

```
<me>@<mydesk> [~] 2: top
Processes:  65 total, 3 running, 62 sleeping... 190 threads   17:39:16
Load Avg:   2.15, 1.46, 1.35
CPU usage:  62.3% user, 37.7% sys, 0.0% idle
SharedLibs: num =  116, resident = 23.2M code, ...
MemRegions: num = 4936, resident =  159M + 13.0M private, ...
VM:         4.22G + 45.8M  5363(0) pageins, 0(0) pageouts
   PID COMMAND            %CPU      TIME         ...
   758 top                7.9%      0:38.79   ...
   703 tcsh               0.0%      0:00.34   ...
   702 sshd               0.0%      0:00.78   ...
   701 ssh                0.0%      0:00.86   ...
   692 CCacheServ         0.0%      0:06.09   ...
   610 Finder             0.0%      0:18.75   ...
   ...
```

Unlike most BSD commands, top doesn't just run and go away. Instead, it refreshes its display about once a second, letting you see if things are changing. When you're done viewing the display, type "q" (or Conrtol-C) and the program will terminate. Alternatively, just start up another Terminal window, leaving top running in the first one...

The system usage statistics are a bit arcane; look at top's man page if you want the details. For now, just note that the number of processes should normally be under 100 and that the load average (time-averaged system load) should be under five. If you see higher values on your machine, you may have a runaway application.

Because top's process list is sorted by CPU utilization, active programs tend to "bubble up". So, look near the top of the list for suspects. In some cases, killing off a sijngle process won't be enough; to restore my desktop to proper functionality, I had to kill the screen saver, the Internet Radio application, and the Finder!

I issued a separate kill com mand for each process I wanted to terminate, specifying the PID (process ID). This let me see and evaluate the effect of each action:

```
^C<me>@<mydesk> [~] 3: kill -9 610
```

Be sure to get the PID right; you don't want to kill off an innocent application, by mistake! By the way, kill can do more than just terminate processes. It can send any desired "signal" to one or more specified processes. So, for example, it can be used to tell processes to check their configuration files, etc.

If you can't figure out which command you need to kill, you may need to get a different view of the information. The ps command lists "process status" information, in a variety of formats. For our present purposes, "ps awx" is a reasonable idiom:

```
<me>@<mydesk> [~] 4: ps awx
  PID  TT  STAT     TIME COMMAND
    1  ??  SLs    0:00.03 /sbin/init
    2  ??  SL     0:02.35 /sbin/mach_init
   41  ??  Ss     0:02.24 kextd
  ...
```

To list the commands in a different order, just pipe together some commands:

```
<me>@<mydesk> [~] 5: ps awx | sort +3 -rn | more
   62  ??  Ss    56:09.79 .../WindowServer
  ...
  PID  TT  STAT     TIME COMMAND
  994 std  R+     0:00.00 more
  993 std  R+     0:00.01 sort +3 -rn
  ...
  610  ??  S      0:18.80 .../MacOS/Finder -psn_0_3538945
  ...
```

Transliterated, the pipeline above says to run the output of ps through a line-oriented sort (sorting by the fourth column, in reverse numeric order), then pipe the result through more (a text viewing command). If you know what command you're looking for, of course, you can use grep:

```
<me>@<mydesk> [~] 6: ps awx | grep Finder
 1006 std  R+     0:00.00 grep Finder
  610  ??  S      0:18.80 .../Finder -psn_0_3538945
```

With the advent of OSX 10.2, I feel more comfortable in recommending FreeBSD-related books to OSX users and programmers. There are still some differences, to be sure, but any book that covers FreeBSD 4.X commands will be a good match for the OSX 10.2 command set.

Although processes are the workhorses of any BSD system, there are few books that spend much time on them. Quite a few man pages relate to processes, however, and the DOSSIER (www.ptf.com/dossier) volume "Processes: FreeBSD" brings them together in a single volume.

Rich Morin has been using computers since 1970, Unix since 1983, and Mac-based Unix since 1986 (when he helped Apple create A/UX 1.0). When he isn't writing this column, Rich runs Prime Time Freeware (www.ptf.com), a publisher of books and CD-ROMs for the Free and Open Source software community. Feel free to write to Rich at rdm@ptf.com.