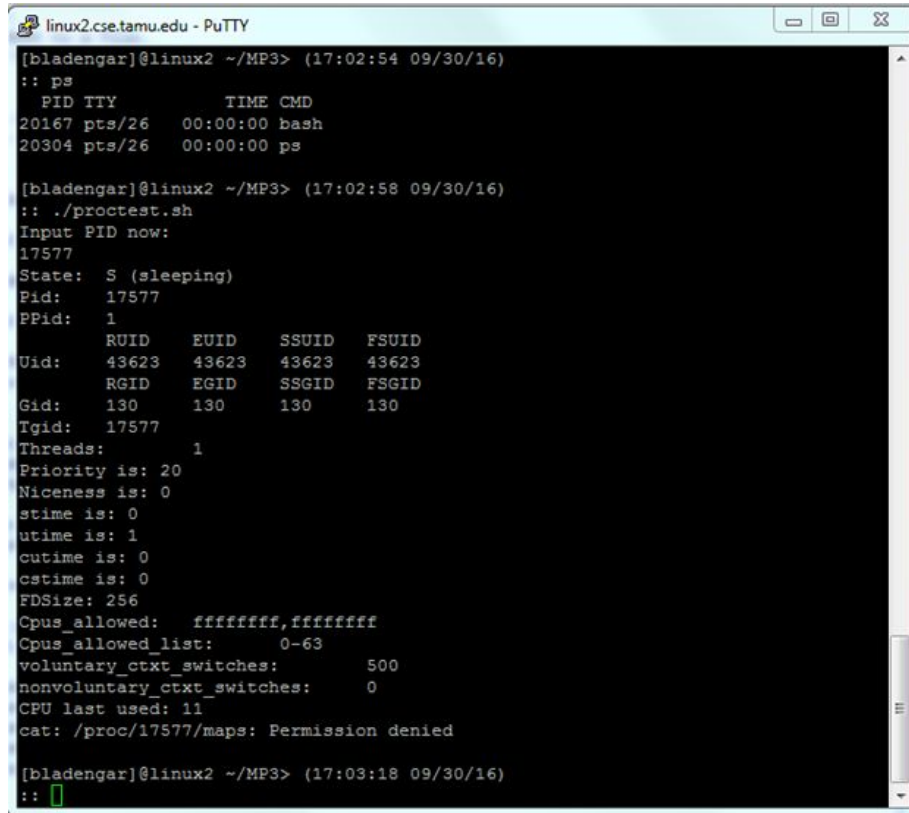


Kerry Zeng
Cesar Ortega

Machine Problem 3

The objective of this Machine problem was to create a bash file that could take in an PID from the user and display various properties of the process. Through this, we learned basic shell commands and the different properties of a process and their meanings.

For a process that was not mine, we got the following for the following information:

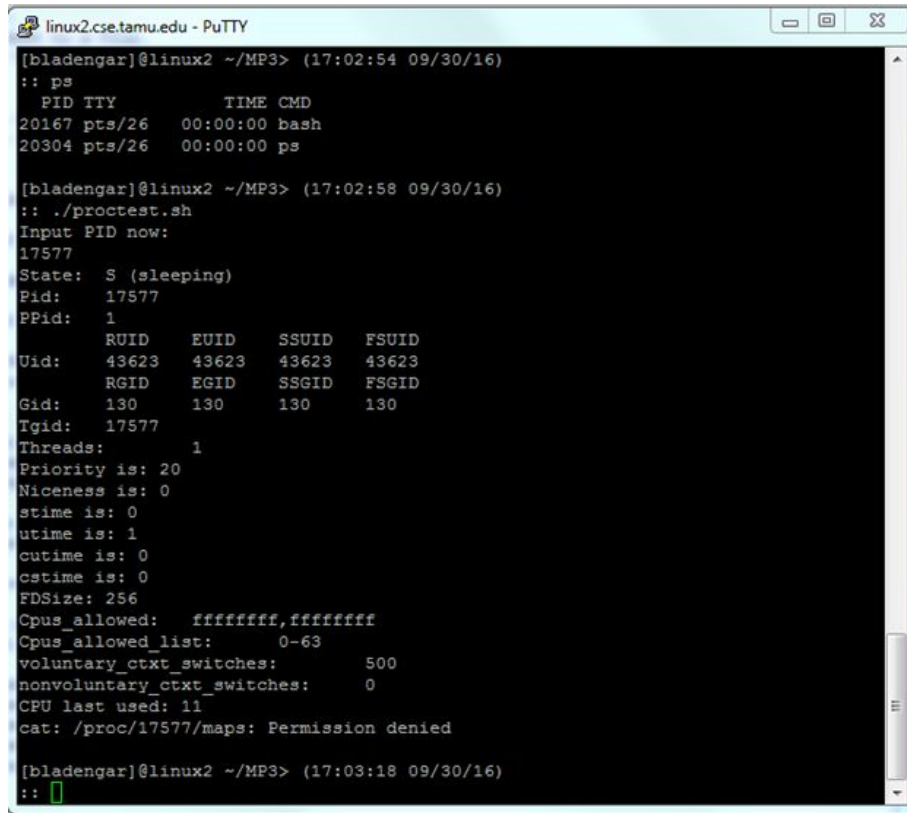


```
linux2.cse.tamu.edu - PuTTY
[bladengar]@linux2 ~/MP3> (17:02:54 09/30/16)
:: ps
  PID TTY          TIME CMD
 20167 pts/26    00:00:00 bash
 20304 pts/26    00:00:00 ps

[bladengar]@linux2 ~/MP3> (17:02:58 09/30/16)
:: ./proctest.sh
Input PID now:
17577
State:  S (sleeping)
Pid:    17577
PPid:   1
      RUID  EUID  SSUID  FSUID
Uid:    43623 43623 43623 43623
      RGID  EGID  SSGID  FSGID
Gid:    130  130   130   130
Tgid:   17577
Threads: 1
Priority is: 20
Niceness is: 0
stime is: 0
utime is: 1
cutime is: 0
cstime is: 0
FDSize: 256
Cpus_allowed:  ffffffff,ffffffff
Cpus_allowed_list:  0-63
voluntary_ctxt_switches:  500
nonvoluntary_ctxt_switches:  0
CPU last used: 11
cat: /proc/17577/maps: Permission denied

[bladengar]@linux2 ~/MP3> (17:03:18 09/30/16)
::
```

For a process that was mine, we got the following information:

A screenshot of a PuTTY terminal window titled 'linux2.cse.tamu.edu - PuTTY'. The terminal shows a user named 'bladengar' at a prompt. The first command is 'ps', which displays a table of running processes. The second command is './proctest.sh', which prompts for an 'Input PID now:'. The user enters '17577'. The terminal then displays detailed information about the process with PID 17577, including its state (S - sleeping), parent PID (1), and various IDs (UID, EUID, SSUID, FSUID, RGID, EGID, SSGID, FSGID). It also shows thread count (1), priority (20), niceness (0), and time statistics. At the bottom, it shows CPU usage and a permission denied error for the '/proc/17577/maps' file.

```
[bladengar]@linux2 ~/MP3> (17:02:54 09/30/16)
:: ps
  PID TTY          TIME CMD
 20167 pts/26    00:00:00 bash
 20304 pts/26    00:00:00 ps

[bladengar]@linux2 ~/MP3> (17:02:58 09/30/16)
:: ./proctest.sh
Input PID now:
17577
State:  S (sleeping)
Pid:    17577
PPid:   1
      RUID   EUID   SSUID   FSUID
Uid:    43623 43623 43623 43623
      RGID   EGID   SSGID   FSGID
Gid:    130   130   130   130
Tgid:   17577
Threads: 1
Priority is: 20
Niceness is: 0
stime is: 0
utime is: 1
cstime is: 0
ostime is: 0
FDSize: 256
Cpus_allowed:  ffffffff,ffffffff
Cpus_allowed_list:  0-63
voluntary_ctxt_switches: 500
nonvoluntary_ctxt_switches: 0
CPU last used: 11
cat: /proc/17577/maps: Permission denied

[bladengar]@linux2 ~/MP3> (17:03:18 09/30/16)
::
```

The map information was not printed in the terminal, but can be found in the memMaps.txt file included.

In our programs that we used to test this, the real and effective IDs were the same. This will be the case for most of the processes, but there is a difference between them. The real user ID is basically the person who owns the process. The effective user ID is the person who's currently using it. The two can be different for example, the root(or admin) may need to take control of the process to do certain things. Other users can also take control of the process, but you need to give them access.

The proc "folder" is mostly read-only to prevent malicious/ignorant attempts to modify the running processes. There are however some select files that can be written to by the kernel to change some parameters.