

# Process (1A)

---

- Process Command

Copyright (c) 2012 Young W. Lim.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

Please send corrections (or suggestions) to [youngwlim@hotmail.com](mailto:youngwlim@hotmail.com).

This document was produced by using OpenOffice and Octave.

# Child Process

```
young@young-530U3C-530U4C:~$  
young@young-530U3C-530U4C:~$  
young@young-530U3C-530U4C:~$ ps  
  PID TTY          TIME CMD  
 2208 pts/1        00:00:00 bash  
 2362 pts/1        00:00:00 ps  
young@young-530U3C-530U4C:~$ xterm &  
[1] 2398  
young@young-530U3C-530U4C:~$  
young@young-530U3C-530U4C:~$  
young@young-530U3C-530U4C:~$ ps  
  PID TTY          TIME CMD  
 2208 pts/1        00:00:00 bash  
 2398 pts/1        00:00:00 xterm  
 2454 pts/1        00:00:00 ps  
young@young-530U3C-530U4C:~$ ps  
  PID TTY          TIME CMD  
 2208 pts/1        00:00:00 bash  
 2398 pts/1        00:00:00 xterm  
 2500 pts/1        00:00:00 ps  
young@young-530U3C-530U4C:~$ pstree
```

Annotations:

- background**: points to the `xterm &` command.
- Job #**: points to the `[1] 2398` output.
- child process**: points to the `2398 pts/1 00:00:00 xterm` entry in the second `ps` output.

# pstree

```
gnome-terminal—bash—pstree
                  |
                  |—xterm—bash—octave
                  |
                  |—gnome-pty-helpe
                  |
                  |—3*[{gnome-terminal}]
```

chmod

chown

chgrp

# ps -ef

ps -ef

| UID   | PID  | PPID | C | STIME | TTY   | TIME     | CMD                              |
|-------|------|------|---|-------|-------|----------|----------------------------------|
| root  | 1    | 0    | 0 | 08:22 | ?     | 00:00:00 | /sbin/init                       |
| root  | 2    | 0    | 0 | 08:22 | ?     | 00:00:00 | [kthreadd]                       |
| root  | 3    | 2    | 0 | 08:22 | ?     | 00:00:00 | [ksoftirqd/0]                    |
| root  | 6    | 2    | 0 | 08:22 | ?     | 00:00:00 | [migration/0]                    |
| root  | 7    | 2    | 0 | 08:22 | ?     | 00:00:00 | [watchdog/0]                     |
| root  | 8    | 2    | 0 | 08:22 | ?     | 00:00:00 | [migration/1]                    |
| young | 2199 | 1    | 0 | 08:28 | ?     | 00:00:07 | gnome-terminal                   |
| young | 2207 | 2199 | 0 | 08:28 | ?     | 00:00:00 | gnome-pty-helper                 |
| young | 2208 | 2199 | 0 | 08:28 | pts/1 | 00:00:00 | bash                             |
| lp    | 2361 | 870  | 0 | 08:28 | ?     | 00:00:00 | /usr/lib/cups/notifier/dbus dbus |
| young | 2398 | 2208 | 0 | 08:29 | pts/1 | 00:00:00 | xterm                            |
| young | 2400 | 2398 | 0 | 08:29 | pts/2 | 00:00:00 | bash                             |
| young | 2497 | 2400 | 0 | 08:29 | pts/2 | 00:00:00 | octave                           |

# Process Related Command

---

```
pstree -h -p -n -l
```

```
pstree -u
```

```
ps -ef
```

```
bg
```

```
fg
```

```
jobs
```

```
kill -9
```

```
kill -l
```

```
killall
```

```
fuser
```

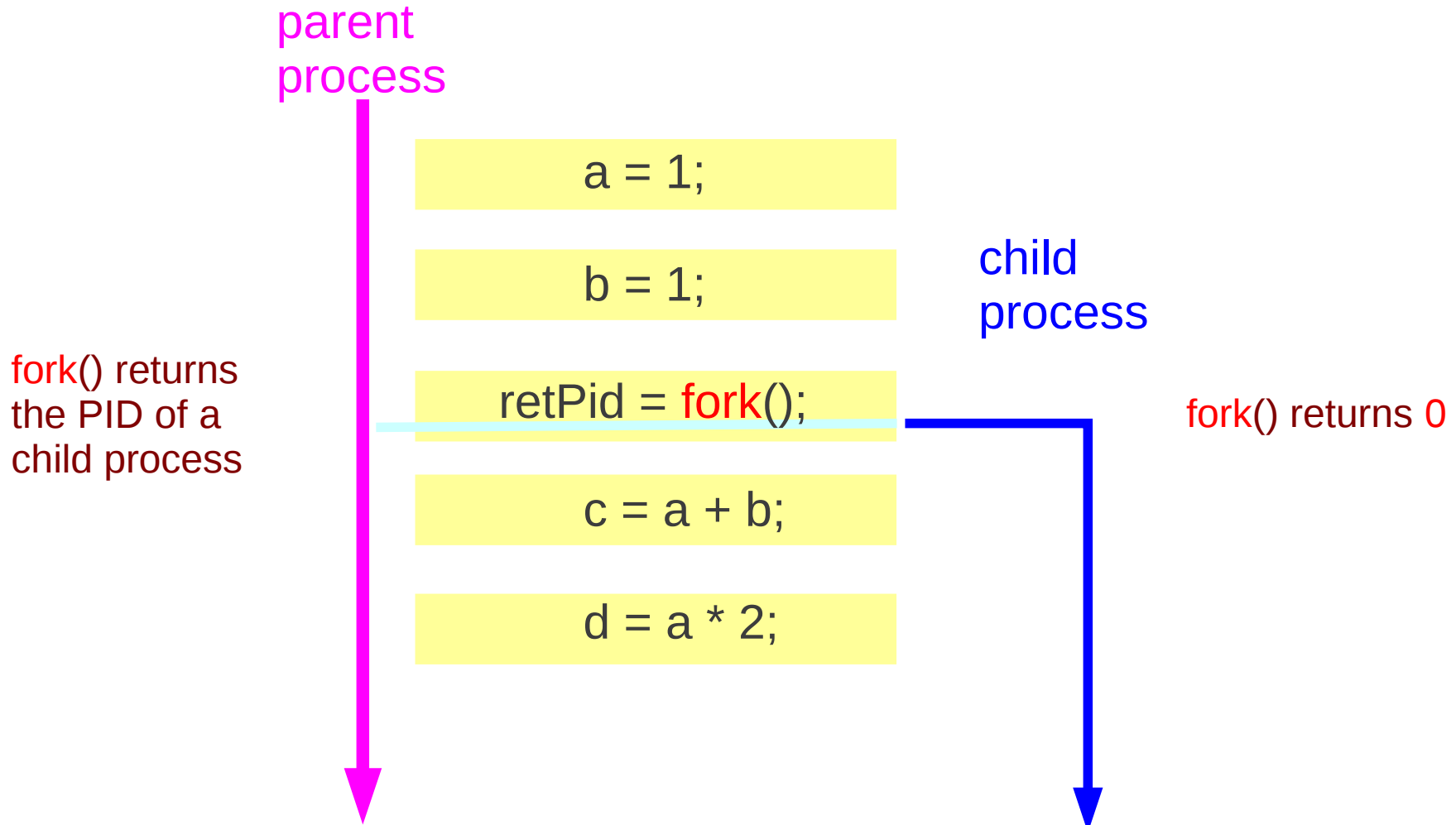
```
pidof
```

```
top
```

```
ntsysv
```

```
chkconfig
```

# fork()



# fork() example (1)

```
#include <stdio.h>
#include <unistd.h>
#include <sys/types.h>

int main(void)
{
    int x, n;
    int retPID = 9999;
    char *message;

    x = 0;
    printf("x = %d pid=%d returned PID = %d\n", x, getpid(), retPID);

    retPID = fork();

    x = 1;
    printf("x = %d pid=%d returned PID = %d\n", x, getpid(), retPID);
```



# fork() example (2)

```
if (retPID == 0) { // child process
    printf("I am a child process %d ParentPID = %d \n", getpid(), getppid());
    n = 5;
    message = "I am a child process";
}
else {
    printf("I am a parent process %d ChildPID = %d \n", getpid(), retPID);
    n = 3;
    message = "I am a parent process";

    execl("/usr/bin/firefox", "firefox" , (char *) 0);
}

for ( ; n > 0; n--) {
    puts(message);
    sleep(3);
}

return 0;
}
```

# argc, argv example

```
#include <stdio.h>
#include <unistd.h>
#include <sys/types.h>

int main(int argc, char * argv[])
{
    int n=0;

    for (n =0; n < argc; n++) {
        printf("n=%d argv[%d] = %s \n", n, n, argv[n]);
    }

    return 0;
}
```

# SetUID Example

```
young/SysP$ vi fprn.c
young/SysP$ gcc fprn.c -o fprn
young/SysP$ chmod 4755 fprn
young/SysP$ ls -l fprn
```

```
mat$ /home/young/SysP/fprn
```

```
young/SysP$ chmod 755 fprn
young/SysP$ ls -l fprn
```

```
mat$ /home/young/SysP/fprn
```

```
#include <stdio.h>

void main()
{
    FILE *fp;

    // fp = fopen("/home/young/SysP/fprn.out", "w");
    fp = fopen("fprn.out", "w");

    if (fp != NULL) {
        printf("Hello, world!\n");
        fprintf(fp, "Hello, world!\n");
        fclose(fp);
    }
    else {
        error("Cannot open /home/young/SysP/fprn.out \n");
    }
}
```

# Sticky Bit

```
young/SysP$ sudo useradd -m bob
young/SysP$ sudo useradd -m mat
young/SysP$ sudo addgroup sysp
young/SysP$ cat /etc/group
young/SysP$ id
young/SysP$ sudo usermod -g (gid of sysp) bob
young/SysP$ sudo usermod -g (gid of sysp) mat
young/SysP$ sudo cd /home/mat
young/SysP$ sudo mkdir SharedDir
young/SysP$ sudo chmod 1775 SharedDir

young/SysP$ sudo chown root:sysp SharedDir
```

```
bob$ cd /home/mat/SharedDir
bob$ vi bob.file
bob$ ls -al .
```

```
bob$ rm mat.file
```

```
mat$ cd SharedDir
mat$ vi mat.file
mat$ ls -al .
```

```
mat$ rm bob.file
```

# Reference

---

## References

- [1] <http://en.wikipedia.org/>
- [2] S.S. Park, Linux Practical Command Bible (in Korean)
- [3] S.S. Park, Linux Server Practical Administration Bible (in Korean)