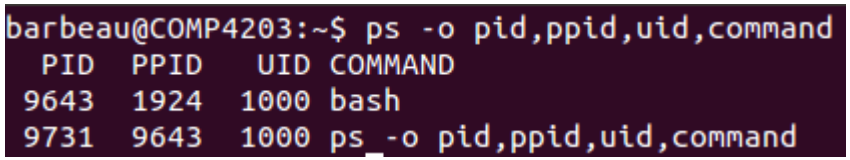
PID, PPID, UID & GID

Looking at Processes

```
barbeau@COMP4203:~/Documents/COMP3000/ALP-listings/chapter-3$ more print-pid.c
#include <stdio.h>
#include <unistd.h>

int main ()
{
  printf ("Process ID is %d\n", (int) getpid ());
  printf ("Parent Process ID is %d\n", (int) getppid ());
  printf ("User ID is %d\n", (int) getuid ());
  return 0;
}
barbeau@COMP4203:~/Documents/COMP3000/ALP-listings/chapter-3$ ./print-pid
Process ID is 9610
Parent Process ID is 9357
User ID is 1000
```

Process ID (PID), Parent Process ID (PPID) & User ID (UID)


barbeau@COMP4203:~$ ps -o pid,ppid,uid,command
PID PPID UID COMMAND
9643 1924 1000 bash
9731 9643 1000 ps -o pid,ppid,uid,command

`ps` command

`kill` command

Sends a `SIGTERM` signal.

```
barbeau@COMP4203:~/Documents/COMP3000/ALP-listings/chapter-3$ more system.c
#include <stdlib.h>

int main ()
{
  int return_value;
  return_value = system ("ls -l /");
  return return_value;
}

barbeau@COMP4203:~/Documents/COMP3000/ALP-listings/chapter-3$ ./system
total 92
drwxr-xr-x   2 root root  4096 Jan  9 19:46 bin
drwxr-xr-x   3 root root  4096 Jan  9 19:47 boot
drwxr-xr-x   2 root root  4096 Jan  9 19:39 cdrom
drwxr-xr-x  15 root root  4120 Jan 13 14:58 dev
drwxr-xr-x 127 root root 12288 Jan 13 14:58 etc
drwxr-xr-x   3 root root  4096 Jan  9 19:40 home
lrwxrwxrwx   1 root root    36 Jan  9 19:44 initrd.img -> boot/initrd.img-3.2.0-
29-generic-pae
drwxr-xr-x  21 root root  4096 Jan 12 17:43 lib
drwxr-xr-x   2 root root  4096 Jan 12 17:44 lib64
drwx------   2 root root 16384 Jan  9 19:30 lost+found
```

Creating process using *system*

Create a new shell process and hands the command to that shell.

```
barbeau@COMP4203:~/Documents/COMP3000/ALP-listings/chapter-3$ more system.c
#include <stdlib.h>

int main ()
{
  int return_value;
  return_value = system ("ps -o pid,ppid,command");
  return return_value;
}

barbeau@COMP4203:~/Documents/COMP3000/ALP-listings/chapter-3$ ./system
  PID  PPID COMMAND
 9357  1924 bash
 9978  9357 ./system
 9979  9978 sh -c ps -o pid,ppid,command
 9980  9979 ps -o pid,ppid,command
```

Inefficiency of *system*

```
root@COMP4203:/home/barbeau# more fork.c
#include <stdio.h>
#include <sys/types.h>
#include <unistd.h>

int main ()
{
  pid_t child_pid;

  child_pid = fork ();
  if (child_pid != 0) {
    printf ("this the parent with pid %d\n", (int) getpid ());
    printf ("child's pid is %d\n", (int) child_pid);
  }
  else
    printf ("this is the child with pid %d\n", (int) getpid ());

  return 0;
}
root@COMP4203:/home/barbeau# ./fork
this the parent with pid 10001
child's pid is 10002
root@COMP4203:/home/barbeau# this is the child with pid 10002
```

Calling *fork*

```c
#include <stdio.h>
#include <unistd.h>

int main(void) {
   int x;

   x = 0;
   fork();
   x = 1;
   printf("I am process %ld and my x is %d\n", (long)getpid(), x);
   return 0;
}
```

```
clude <stdio.h>
clude <unistd.h>
clude <sys/types.h>
clude <sys/wait.h>

 main (void) {
pid_t childpid;

childpid = fork();
if (childpid == -1) {
   perror("Failed to fork");
   return 1;
}
if (childpid == 0)
   fprintf(stderr, "I am child %ld\n", (long)getpid());
else if (wait(NULL) != childpid)
   fprintf(stderr, "Wait interrupted!\n");
else
   fprintf(stderr, "I am parent %ld with child %ld\n", (long)getpid
        (long)childpid);
return 0;

beau@COMP4203:~/Documents/COMP3000$ ./parentwaitpid
m child 10144
m parent 10143 with child 10144
```

fork and *wait*

```
for (i = 1; i < n; i++)
    if (childpid = fork())
        break;
```

```
for (i = 1; i < 4; i++)
{
    childpid = fork();
    if (childpid != 0)  /* i.e., in parent */
        break;
}
```

```
beau@COMP4203:~$ more simplechain.c
clude <stdio.h>
clude <stdlib.h>
clude <unistd.h>

 main (int argc, char *argv[]) {
pid_t childpid = 0;
int i;

for (i = 1; i < 4; i++)
   if (childpid = fork())
      break;

fprintf(stderr, "i:%d  PID:%ld  PPID:%ld  child ID:%ld\n"
        i, (long)getpid(), (long)getppid(), (long)childpi

beau@COMP4203:~$ ./simplechain
  PID:2537  PPID:1941  child ID:2538
beau@COMP4203:~$ i:2  PID:2538  PPID:1  child ID:2539
  PID:2539  PPID:1  child ID:2540
  PID:2540  PPID:1  child ID:0
```

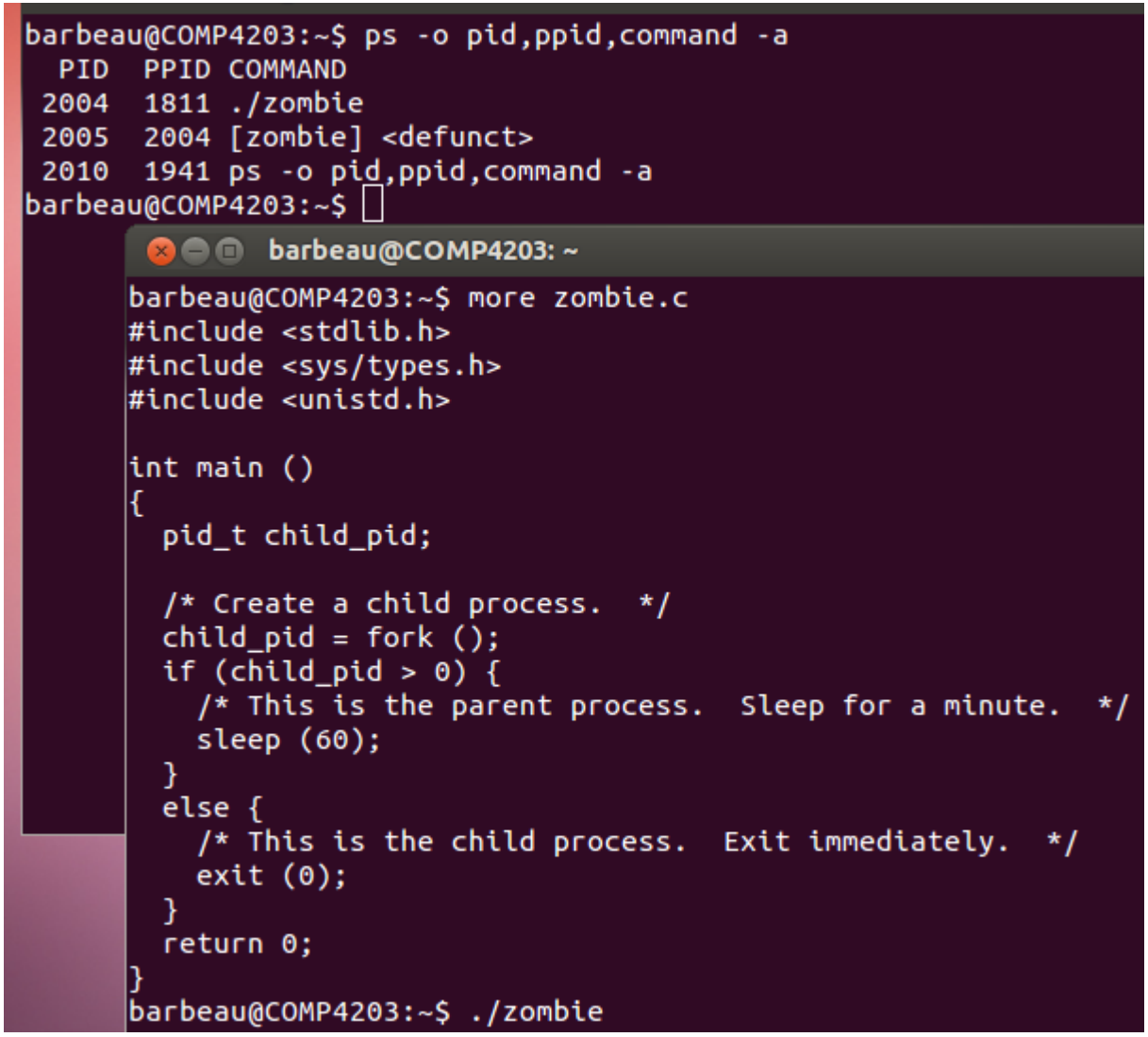
```
for (i = 1; i < n; i++)
    if ((childpid = fork()) <= 0)
        break;
```

```
for (i = 1; i < n; i++)
    if ((childpid = fork()) == -1)
        break;
```

```
barbeau@COMP4203:~/Documents/COMP3000$ more execls.c
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/wait.h>

int  main(void) {
   pid_t childpid;

   childpid = fork();
   if (childpid == -1)  {
       perror("Failed to fork");
       return 1;
   }
   if (childpid == 0) {                          /* child code */
       execl("/bin/ls", "ls", "-l", NULL);
       perror("Child failed to exec ls");
       return 1;
   }
   if (childpid != wait(NULL)) {                 /* parent code */
       perror("Parent failed to wait due to signal or error");
       return 1;
   }
   return 0;
}
barbeau@COMP4203:~/Documents/COMP3000$ ./execls
total 200
drwxr-x--- 15 barbeau barbeau  4096 Jun 26  2001 ALP-listings
-rw-------  1 barbeau barbeau 47675 Sep  9  2009 ALP-listings.tar.gz
```

Creating process with *execl*

```
nt  main(void) {
  pid_t childpid;

  childpid = fork();
  if (childpid == -1)  {
      perror("Failed to fork");
      return 1;
  }
  if (childpid == 0) {                          /* child code *
      execl("/bin/ps", "ps", NULL);
      perror("Child failed to exec ls");
      return 1;
  }
  if (childpid != wait(NULL)) {                 /* parent code *
      perror("Parent failed to wait due to signal or error");
      return 1;
  }
  return 0;

arbeau@COMP4203:~/Documents/COMP3000$ ./execls
 PID TTY          TIME CMD
9357 pts/2    00:00:02 bash
0491 pts/2    00:00:00 execls
0492 pts/2    00:00:00 ps
```



***Zombie* process** (terminated child not cleaned up by parent, using *wait*)

```
barbeau@COMP4203:~$ ps -o pid,ppid,command -a
  PID  PPID COMMAND
 2017  1811 ./zombie
 2018  2017 [zombie] <defunct>
 2019  1941 ps -o pid,ppid,command -a
barbeau@COMP4203:~$ kill 2017; ps -o pid,ppid,command -a
  PID  PPID COMMAND
 2020  1941 ps -o pid,ppid,command -a
```

Killing a parent process with zombies

```
barbeau@COMP4203:~$ more myzombie.c
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <unistd.h>

int main ()
{
  /* Create a child process.  */
  if (fork () > 0) {
    /* This is the parent process.  Exit immediately.  */
    printf ("Process ID is %d\n", (int) getpid ());
    exit (0);
  }
  else {
    /* This is the child process.  Sleep for a minute.  */
    printf ("Child ID is %d\n", (int) getpid ());
    sleep (60);
  }
  return 0;
}
barbeau@COMP4203:~$ ./myzombie; ps -o pid,ppid,command
Process ID is 2442
Child ID is 2443
  PID  PPID COMMAND
 1811  1801 bash
 2443     1 ./myzombie
 2444  1811 ps -o pid,ppid,command
```

Orphan process

```
groupadd developers
grep developers /etc/group
developers:x:1002:

id barbeau
uid=1000(barbeau) gid=1000(barbeau)
groups=1000(barbeau),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),109(lpadmin),
124(sambashare)

usermod -a -G developers barbeau

id barbeau
uid=1000(barbeau) gid=1000(barbeau)
groups=1000(barbeau),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),109(lpadmin),
124(sambashare),1002(developers)
```

```
ls -l
-rw-r----- 1 barbeau barbeau  721 Jan 14 21:01 zombie.c

chown :developers zombie.c

ls -l
-rw-r----- 1 barbeau developers  721 Jan 14 21:01 zombie.c

chmod g+w zombie.c

ls -l
-rw-rw---- 1 barbeau developers  721 Jan 14 21:01 zombie.c
```