<https://www.cnblogs.com/lakeone/p/3789117.html>

#include <stdio.h>

#include <pthread.h>

#include <sys/types.h>

#include <sys/syscall.h>

struct message

{

int i;

int j;

};

void \*hello(struct message \*str)

{

printf("child, the tid=%lu, pid=%d\n",pthread\_self(),syscall(SYS\_gettid));

printf("the arg.i is %d, arg.j is %d\n",str->i,str->j);

printf("child, getpid()=%d\n",getpid());

while(1);

}

int main(int argc, char \*argv[])

{

struct message test;

pthread\_t thread\_id;

test.i=10;

test.j=20;

pthread\_create(&thread\_id,NULL,hello,&test);

printf("parent, the tid=%lu, pid=%d\n",pthread\_self(),syscall(SYS\_gettid));

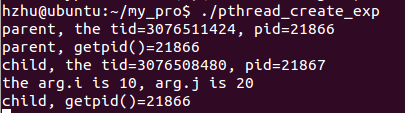
printf("parent, getpid()=%d\n",getpid());

pthread\_join(thread\_id,NULL);

return 0;

}

[复制代码](javascript:void(0);)



getpid()得到的是进程的pid，在内核中，每个线程都有自己的PID，要得到线程的PID,必须用syscall(SYS\_gettid);

pthread\_self函数获取的是线程ID，线程ID在某进程中是唯一的，在不同的进程中创建的线程可能出现ID值相同的情况。

[复制代码](javascript:void(0);)

#include <stdio.h>

#include <pthread.h>

#include <stdlib.h>

#include <unistd.h>

#include <sys/syscall.h>

void \*thread\_one()

{

printf("thread\_one:int %d main process, the tid=%lu,pid=%ld\n",getpid(),pthread\_self(),syscall(SYS\_gettid));

}

void \*thread\_two()

{

printf("thread two:int %d main process, the tid=%lu,pid=%ld\n",getpid(),pthread\_self(),syscall(SYS\_gettid));

}

int main(int argc, char \*argv[])

{

pid\_t pid;

pthread\_t tid\_one,tid\_two;

if((pid=fork())==-1)

{

perror("fork");

exit(EXIT\_FAILURE);

}

else if(pid==0)

{

pthread\_create(&tid\_one,NULL,(void \*)thread\_one,NULL);

pthread\_join(tid\_one,NULL);

}

else

{

pthread\_create(&tid\_two,NULL,(void \*)thread\_two,NULL);

pthread\_join(tid\_two,NULL);

}

wait(NULL);

return 0;

}

[复制代码](javascript:void(0);)

https://images0.cnblogs.com/i/578474/201406/171020531457661.png