

In the name of god
OS project
phase_1 report
Dadfar Mo'meni
965222006

attention:

1. commits may seem too close together,
the reason is there was some problem in
git branches and commits on them
so I took advantage of the extra time
to make it nice and tidy.

2. I tried to follow this step by step tutorial in Medium:

<https://medium.com/@viduniwickramarachchi/add-a-new-system-call-in-xv6-5486c2437573>

STEP 0:

1. download or clone xv6 project.

2. install qemu.

in order to add the system call to XV6 we only have to make changes
to these 5 following files:

1. syscall.h
2. syscall.c
3. sysproc.c
4. usys.S
5. user.h

STEP 1:

1. give an index to the system call in syscall.h

```
#define SYS_getProcessInfo 23
```

2. Add pointer to the system call in syscall.c

```
[SYS_getProcessInfo] sys_getProcessInfo,
```

When the system call with number 23 is called by a user program, the function pointer sys_getProcessInfo which has the index SYS_getProcessInfo or 23 will call the system call function.

STEP 2:

1. to implement the system call function, first we have to add function prototype in syscall.c

```
extern int sys_getProcessInfo(void);
```

2. to implement the system call function we can either implement it in proc.c or sysproc.c (I chose to write it in proc.c)

The basic implementation of the system call is now complete. However, there are 2 more minor steps remaining.

STEP 3:

1. In order for a user program to call the system call, an interface needs to be added. Therefore, we need to edit the usys.S file.
SYSCALL(getProcessInfo)

2. Next, the user.h file needs to be edited.

```
int getProcessInfo(struct proc_info*);
```

STEP 4:

1. in order to test the functionality of the system call, we need to add a user program which calls this system call.
get_process_info.c

2. edit the Makefile file to run the program using this tutorial:

<https://ampleux.wordpress.com/2018/02/22/how-to-add-a-user-program-to-xv6/>

STEP 5:

1. to test the system call we need to run these commands in terminal where all xv6 files are located

```
$ make clean
```

```
$make
```

```
$make qemu
```

2. to run our test program we simply run this command in qemu terminal:

```
./get_process_info
```