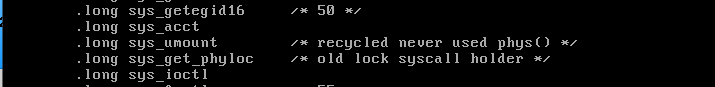
오퍼레이팅시스템 14주차 과제

1280626 성시열

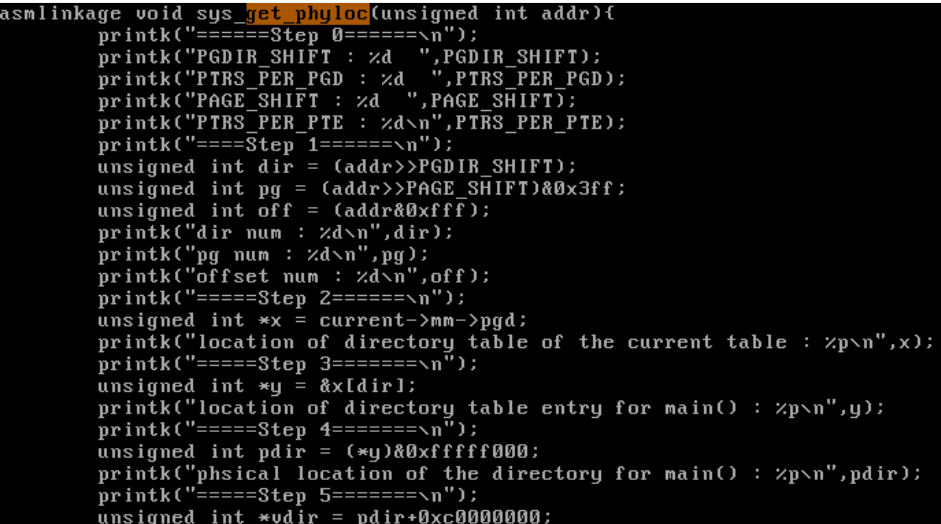
hw 7) Make a system call, sys\_get\_phyloc(), which will display the physical address of main().

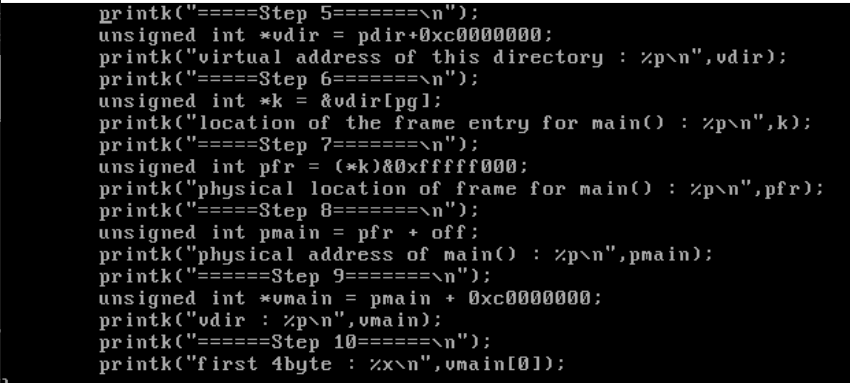
1) Write a simple program that prints the address of main().



cd linux-2.6.25.10/arch/x86/kernel로 이동 후 syscall\_table\_32.S를 수정한다.

53번에 sys\_get\_phyloc을 추가한다.



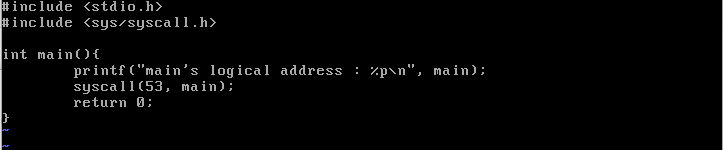


Step에 따라 다음과 같이 함수를 작성한다.

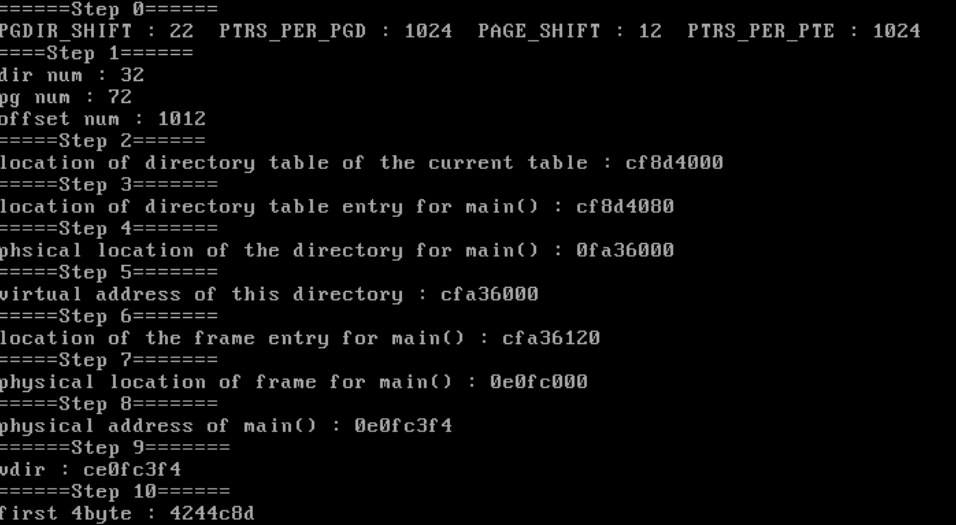
2) Call sys\_get\_phyloc(main) in this program which passes the address of main.

3) sys\_get\_phyloc(addr) is a system call that performs following steps in order:

Ex14.c 선언

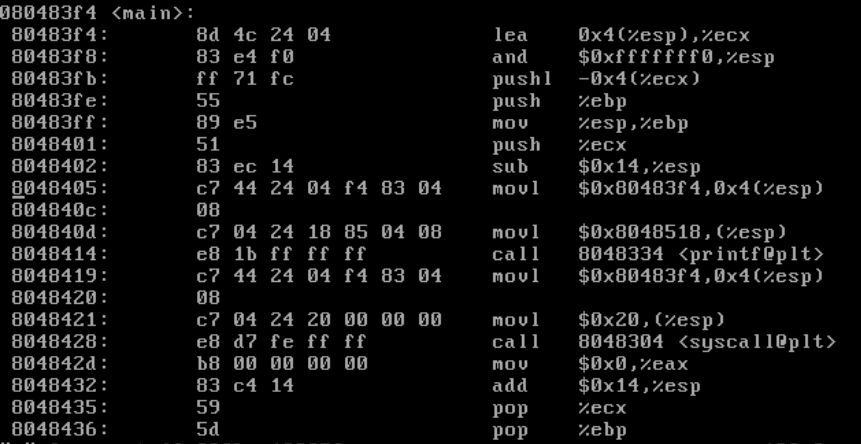


Ex14.c 실행 결과



위와 같이 각각의 해당하는 주소값들을 출력할 수 있었다.

결국 main의 physical 한 주소를 찾아내어 main의 첫바이트를 가져올 수 있었다.



Main의 첫 4바이트가 Step 10에서 찾은 최종 main의 첫 byte와 일치하는 것을 볼 수 있었다.

step0: print the value of PGDIR\_SHIFT, PTRS\_PER\_PGD, PAGE\_SHIFT, PTRS\_PER\_PTE

PGDIR\_SHIFT=22: number of shifting to extract directory number from a logical address. Logical address 0x080484a4 = (dir 20h, page 48h, offset 4a4h)

pgd\_index=20h, pte\_index=48h

PAGE\_SHIFT=12: number of shifting to extract page number from a logical address.

PTRS\_PER\_PGD=1024: number of directory entries in a directory table

PTRS\_PER\_PTE=1024: number of frame pointer entries in a directory

step1: extract directory number (dir), page number(pg), and offset(off) from addr, and display them.

step2: print the location of directory table of the current process: x

step3: print the location of directory table entry for main(): y

y= &x[dir];

step4: print the physical location of the directory (partial page table) for main():pdir

pdir= \*y & 0xfffff000; // the physical address should be at frame boundary

step5: print the virtual address of this directory: vdir

vdir = pdir + 0xc0000000; // physical to virtual mapping for kernel address

// read about kernel address space in Section 7.4.

step6: print the location of the frame entry for main(): k

k = &vdir[pg];

step7: print the physical location of frame for main(): pfr

pfr = \*k & 0xfffff000; // the physical address should be at frame boundary

step8: print the physical address of main(): pmain

step9: print the virtual address for the physical address of main(): vmain

step10: display the first 4 bytes in vmain and compare them with the first 4 bytes of main in the original executable file(use "objdump -d program-name" to see the first 4 bytes of main in the original program). If they are same, you have the correct physical address of main.