



EDDI

Electronic Design
Development Institute

에디로봇아카데미

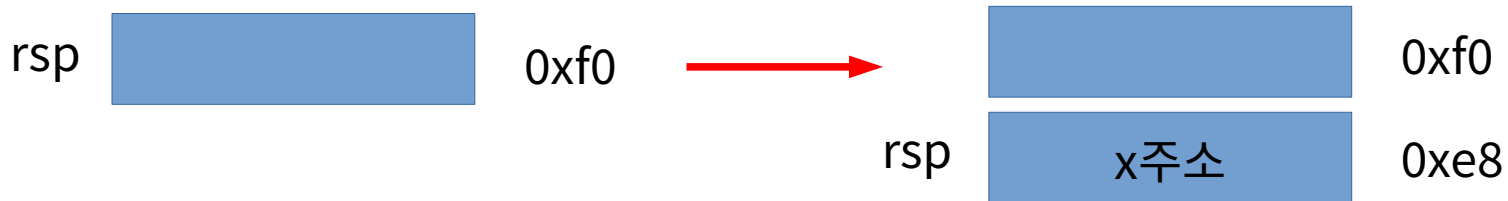
임베디드 마스터 Lv2 과정

제 1기

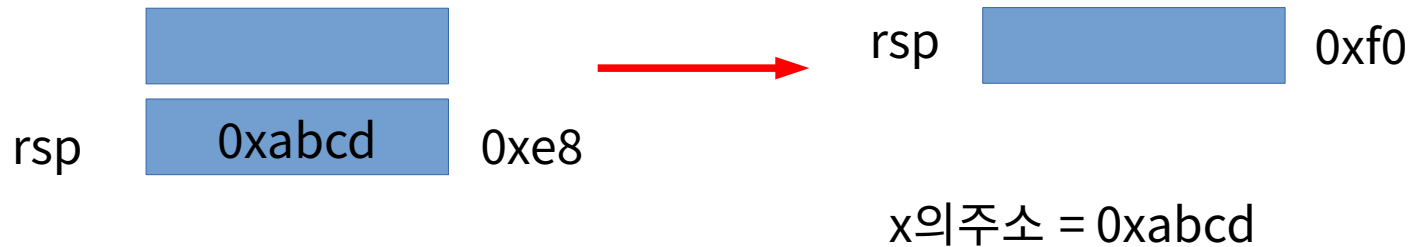
2021. 10. 08

이충재

Push %x = rsp 주소를 8바이트 만큼 내리고
x의 주소값을 넣는다.



Pop %x = rsp에 있는 값을 x의 주소값으로 하여라.
그리고 rsp 주소를 8바이트 만큼 올려라.



`mov %a %b`: a의 주소를 b의 주소로 옮겨라

`mov $c %d`: c값을 d주소로 옮겨라

`mov $c ±x(%e)` : x의주소 보다 x만큼 위 또는 아래에 c값을 넣어라.

`retq` : rsp에 있는 주소값을 rip의 주소로 하여라
rip는 다음에 실행할 명령어가 있는 주소를 가진다.
Pop rip와 같은 동작을 한다.

Leaveq : rsp를 rbp가 있는 주소로 옮기고
rsp에 있는값을 rbp 주소로 옮긴다.

mov rbp rsp + Pop rbp와 같은 동작을 한다.

gcc -g: 컴파일시 디버깅 메타 정보를 기록한다.

gcc -O0: 어떠한 최적화 기법도 적용되지 않는다.

gcc -O2: 최대한의 보수성을 지키며 최적화를 수행

gcc -O3: 문제 발생 여부 상관없이 최고의 효율로 컴파일
문제가 발생할 가능성이 있다.

b main: 중단점을 main으로 설정한다.

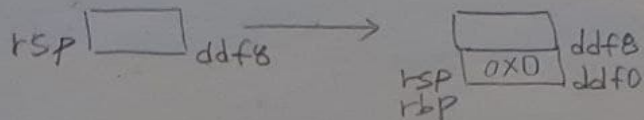
disas: 컴파일한 어셈블리어를 보여준다.

Si: 어셈블리어 기준으로 한 줄 실행

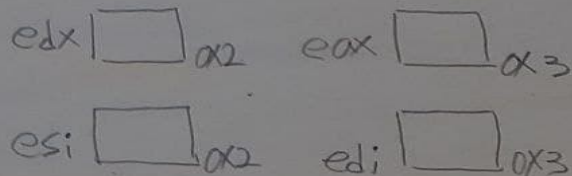
ni: c코드 기준으로 한 줄 실행

숙제

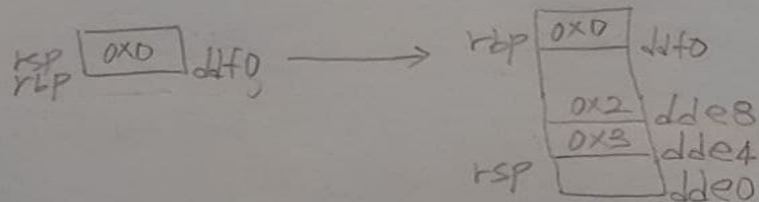
```
push %rbp
mov %rsp, %rbp
```



```
mov -0x8(%rbp), %edx
mov -0xc(%rbp), %eax
mov %edx, %esi
mov %eax, %edi
```

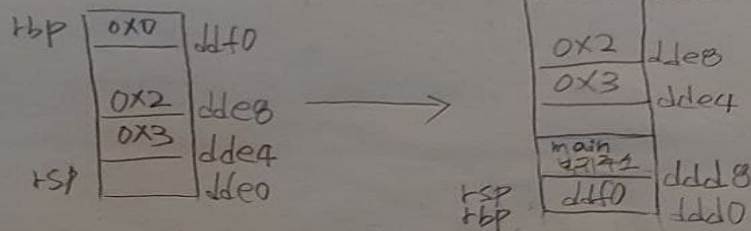


```
sub $0x10, %rsp
movl $0x3, -0xc(%rbp)
movl $0x2, -0x8(%rbp)
```

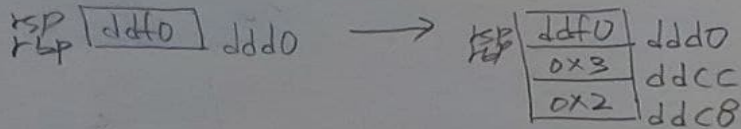


```
callq 0x55555555/401 <mult>
```

```
push %rbp
mov %rsp, %rbp
```

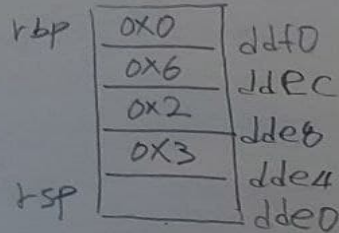



```
mov %edi, -0x4(%rbp)
mov %esi, -0x8(%rbp)
mov -0x4(%rbp), %eax
```

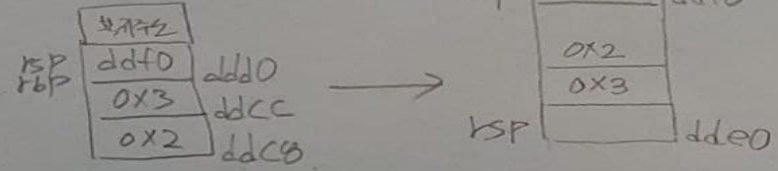


eax [] 0x3

```
mov %eax, -0x4(%rbp)
```



```
imul -0x8(%rbp), %eax
pop %rbp
retq
```



eax [] 0x3

eax [] 0x6