

```

// Allocate two pages at the next page boundary.
// Make the first inaccessible. Use the second as the user stack.
sz = PGROUNDUP(sz);
if((sz = allocuvm(pgdir, sz, sz + 2*PGSIZE) == 0))
    goto bad;
clearpteu(pgdir, (char*)(sz-PGSIZE));

//sp = sz;
if((sp = allocuvm(pgdir, STKBASE - PGSIZE, STKBASE)) == 0)
    goto bad;

```

Changed the start of the stack and also

```

// Fetch the int at addr from the current process.
int
fetchint(uint addr, int *ip)
{
    struct proc *curproc = myproc();

    // if(addr >= curproc->sz || addr+4 > curproc->sz)
    // return -1;
    *ip = *(int*)(addr);
    return 0;
}

// Fetch the nul-terminated string at addr from the current process.
// Doesn't actually copy the string - just sets *pp to point at it.
// Returns length of string, not including nul.
int
fetchstr(uint addr, char **pp)
{
    char *s, *ep;
    struct proc *curproc = myproc();

    // if(addr >= curproc->sz)
    // return -1;
    *pp = (char*)addr;
    ep = (char*)curproc->sz;
    for(s = *pp; s < ep; s++){
        if(*s == 0)
            return s - *pp;
    }
}

```

Changing checks since it used to compare to our old sz

```

pde_t*
copyuvm(pde_t *pgdir, uint sz)
{
    struct proc *cur = myproc();
    pde_t *d;
    pte_t *pte;
    uint pa, i, flags;
    char *mem;

    if((d = setupkvm()) == 0)
        return 0;
    for(i = 0; i < sz; i += PGSIZE){
        if((pte = walkpgdir(pgdir, (void *) i, 0)) == 0)
            panic("copyuvm: pte should exist");
        if(!(*pte & PTE_P))
            panic("copyuvm: page not present");
        pa = PTE_ADDR(*pte);
        flags = PTE_FLAGS(*pte);
        if((mem = kalloc()) == 0)
            goto bad;
        memmove(mem, (char*)P2V(pa), PGSIZE);
        if(mappages(d, (void*)i, PGSIZE, V2P(mem), flags) < 0){
            kfree(mem);
            goto bad;
        }
    }
}

for(i = (STKBASE - p->numpages * PGSIZE + 4); i < STKBASE; i += PGSIZE){
    if((pte = walkpgdir(pgdir, (void *) i, 0)) == 0)
        panic("copyuvm: pte should exist");
    if(!(*pte & PTE_P))
        panic("copyuvm: page not present");
    pa = PTE_ADDR(*pte);
    flags = PTE_FLAGS(*pte);
    if((mem = kalloc()) == 0)
        goto bad;
    memmove(mem, (char*)P2V(pa), PGSIZE);
    if(mappages(d, (void*)i, PGSIZE, V2P(mem), flags) < 0){
        kfree(mem);
        goto bad;
    }
}

```

Changed copyuvm to now go over the new stack

```

        break;
    case T_PGFLT:
        if((rcr2() < STKBASE)){
            allocuvm(myproc()->pgdir, STKBASE- ((myproc()->numpages+1)*PGSIZE), STKBASE-
            ((myproc()->numpages)*PGSIZE));
            myproc()->numpages++;
        }
    break;

```

Made a new page fault test