



دانشگاه صنعتی شریف
دانشکده‌ی مهندسی کامپیوتر

درس آ‌ز سیستم‌عامل

استاد حمید بیگی

گزارش آزمایش اول

علی‌رضا دیزجی، امیرحسن فتحی، مصطفی قدیمی

سؤال ۱. نصب سیستم‌عامل

با توجه به صحبت‌های انجام‌شده نیازی به آوردن اسکرین‌شات‌های این قسمت نیست.

سؤال ۲. آشنایی با دستورات پایه‌ی لینوکس

۱. دستور pwd

```
> pwd
/home/mostafa/Desktop/University/OS-Lab
```

۲. دستور cd و دستور mkdir

```
> cd /tmp
> mkdir oslab1
> cd oslab1
```

۳. دستور nano

```
> sudo nano information.txt
```

۴. دستور mv

```
> sudo mv information.txt myinformation.txt
```

۵. دستور cp

```
> sudo cp information.txt backupinfo.txt
```

۶. دستور cat

```
> cat myinformation.txt
```

۷. دستور «>» تمام محتویات قبلی را پاک کرده و محتوای جدید را به فایل اضافه می‌کند. اصطلاحاً به این کار «overwrite» می‌گویند. اما دستور «>>»، محتوای قبلی را نگه می‌دارد و محتوای جدید را به آخر محتوای قبلی اضافه می‌کند. اصطلاحاً به این کار «append» می‌گویند.

۸. ساخت فایل جدید با دستور cat

```
> cat > testfile.txt
```

۹. لیست پردازها

```
> ps aux
```

۱۰. دستور grep

```
> ps aux | grep a
```

۱۱. دستور ls

```
> cd /usr/bin
> ls
```

۱۲. نمایش حجم و نام فایل‌ها

```
> ls -sh
```

۱۳. جست‌وجو در نام فایل‌ها

```
> ls -h | grep 'id\|fs'
```

فعالیت

- کاربرد دستورات

```

mostafa@mostafa-UX303UB: ~
File Edit View Search Terminal Help
> ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1   0.0   0.1 226276   9588 ?        Ss   Jul23   1:12 /lib/systemd/sy
root         2   0.0   0.0      0      0 ?        S    Jul23   0:00 [kthreadd]
root         3   0.0   0.0      0      0 ?        I<   Jul23   0:00 [rcu_gp]
root         4   0.0   0.0      0      0 ?        I<   Jul23   0:00 [rcu_par_gp]
root         9   0.0   0.0      0      0 ?        I<   Jul23   0:00 [mm_percpu_wq]
root        10   0.0   0.0      0      0 ?        S    Jul23   0:05 [ksoftirqd/0]
root        11   0.0   0.0      0      0 ?        I    Jul23   2:54 [rcu_sched]
root        12   0.0   0.0      0      0 ?        S    Jul23   0:00 [migration/0]
root        13   0.0   0.0      0      0 ?        S    Jul23   0:00 [idle_inject/0]
root        14   0.0   0.0      0      0 ?        S    Jul23   0:00 [cpuhp/0]
root        15   0.0   0.0      0      0 ?        S    Jul23   0:00 [cpuhp/1]
root        16   0.0   0.0      0      0 ?        S    Jul23   0:00 [idle_inject/1]
root        17   0.0   0.0      0      0 ?        S    Jul23   0:00 [migration/1]
root        18   0.0   0.0      0      0 ?        S    Jul23   0:04 [ksoftirqd/1]
root        21   0.0   0.0      0      0 ?        S    Jul23   0:00 [cpuhp/2]
root        22   0.0   0.0      0      0 ?        S    Jul23   0:00 [idle_inject/2]
root        23   0.0   0.0      0      0 ?        S    Jul23   0:00 [migration/2]
root        24   0.0   0.0      0      0 ?        S    Jul23   0:03 [ksoftirqd/2]
root        27   0.0   0.0      0      0 ?        S    Jul23   0:00 [cpuhp/3]
root        28   0.0   0.0      0      0 ?        S    Jul23   0:00 [idle_inject/3]
root        29   0.0   0.0      0      0 ?        S    Jul23   0:00 [migration/3]
root        30   0.0   0.0      0      0 ?        S    Jul23   0:03 [ksoftirqd/3]

```

شکل ۱: نمایش لیست پردازنده‌ها با دستور ps

- دستور cut: برای برش یا جدا کردن بخشی از هر خط از فایل و چاپ کردن نتیجه درون خروجی استاندارد، از این دستور استفاده می‌شود.
- دستور find: از این دستور می‌توان برای پیدا کردن فایل‌ها و دایرکتوری‌ها و همچنین انجام عملیات پی‌درپی روی آن‌ها استفاده می‌شود.
- دستور head: همان‌طور که از نامش پیداست، (به صورت پیش‌فرض) ده خط اول از یک فایل را نشان می‌دهد. این تعداد خط را می‌توان تغییر داد.
- دستور tail: همان‌طور که از نامش پیداست، (به صورت پیش‌فرض) ده خط آخر از یک فایل را نمایش می‌دهد. این تعداد خط را می‌توان تغییر داد.
- دستور touch: دستور استاندارد سیستم‌عامل لینوکس برای ایجاد و تغییر timestamp فایل است. از آن برای ایجاد یک فایل بدون محتوا نیز استفاده می‌شود.
- دستور wc: از این دستور برای شمارش تعداد خط، واژه، بایت و کاراکترهای یک فایل می‌توان استفاده کرد.
- دستور kill: از این دستور برای خاتمه دادن پردازنده‌ها به صورت دستی استفاده می‌شود.

- پیدا کردن تعداد خطوط در یک فایل متنی به نام mybook.txt

```
> wc -l mybook.txt
```

- پیدا کردن تعداد فایل‌هایی که با حرف A شروع می‌شوند.

```
> find . -type f -name '[A]*' | wc -l
```

• پیدا کردن حجم فایل به نام mybook.txt

```
> find . -name mybook.txt -ls | awk 'print $1'
```

سؤال ۳. اعمال تغییرات و کامپایل مجدد هسته‌ی سیستم عامل

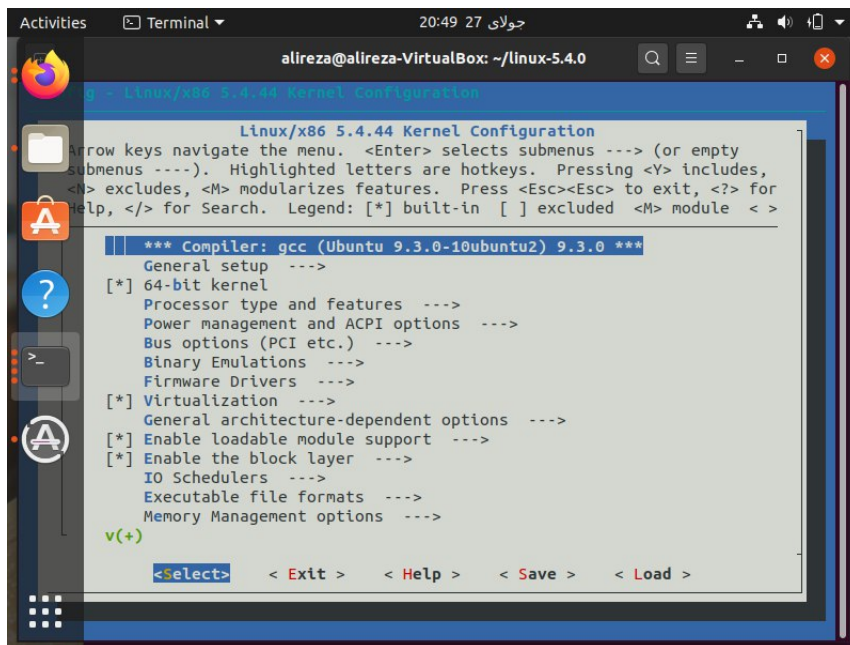
- با توجه به صحبت‌های انجام شده از طریق ایمیل و قدیمی بودن کد منبع هسته‌ی موجود در آزمایش نسخه‌ی ۴.۵ با کمک دستور زیر نصب شده است:

```
> sudo apt-get install linux-source-5.4.0
```

- پس از نصب وارد پوشه‌ی linux-source-5.4.0 می‌شویم. قبل از کامپایل کردن هسته^۱ ابتدا باید مازول‌های مورد نیاز را مشخص کنیم. برای این کار می‌توان از پیکربندی^۲ موجود در کرنل فعلی استفاده کنیم. برای این کار باید زیر استفاده کرد:

```
> cp /boot/config-$(uname -r) .config
```

- در صورت نیاز به اعمال دیگر تغییرات می‌توان از دستور make menuconfig استفاده کرد تا مازول‌های مورد نظر را فعال یا غیرفعال کنیم.



شکل ۲: استفاده از دستور make menuconfig

- حال که مازول‌های مورد استفاده تعیین شده‌اند، اولین قدم در کامپایل کردن هسته‌ی سیستم عامل، استفاده از دستور make است.

- پس از پایان دستور make با دستور make modules_install، مازول‌هایی تعیین شده، نصب می‌شوند.

- با استفاده از دستور sudo make install، هسته نصب می‌شود.

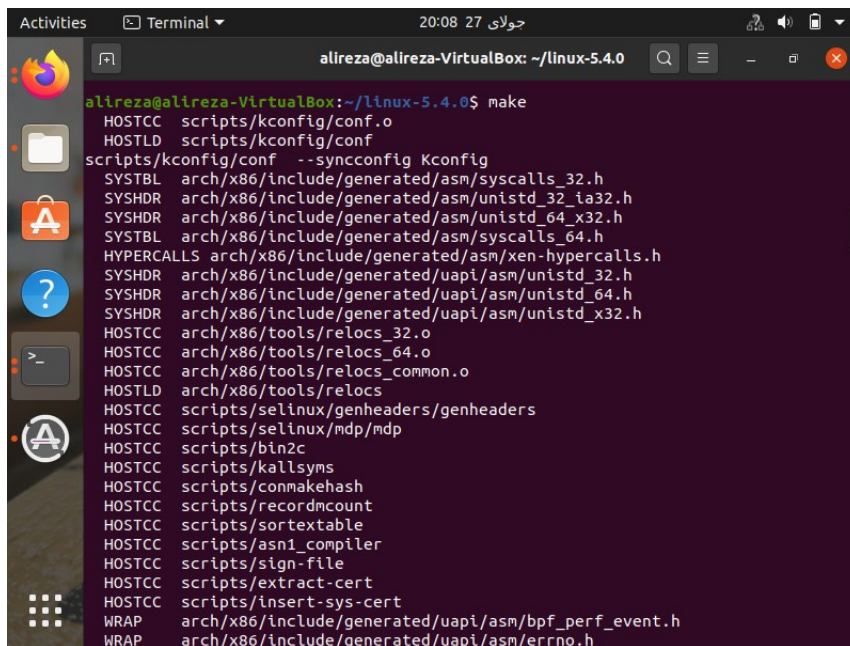
- برای فعال کردن هسته‌ی نصب شده در بوت، از فرمان زیر بهره می‌بریم:

```
> sudo update-initramfs -c -k 5.4.0
```

- برای اعمال تغییرات نیاز به به‌روزرسانی grub داریم:

```
> sudo update-grub
```

^۱kernel
^۲configuration



```
alireza@alireza-VirtualBox:~/linux-5.4.0$ make
HOSTCC scripts/kconfig/conf.o
HOSTLD scripts/kconfig/conf
scripts/kconfig/conf --syncconfig Kconfig
SYSTBL arch/x86/include/generated/asm/syscalls_32.h
SYSHDR arch/x86/include/generated/asm/unistd_32_ia32.h
SYSHDR arch/x86/include/generated/asm/unistd_64_x32.h
SYSTBL arch/x86/include/generated/asm/syscalls_64.h
HYPERCALLS arch/x86/include/generated/asm/xen-hypercalls.h
SYSHDR arch/x86/include/generated/uapi/asm/unistd_32.h
SYSHDR arch/x86/include/generated/uapi/asm/unistd_64.h
SYSHDR arch/x86/include/generated/uapi/asm/unistd_x32.h
HOSTCC arch/x86/tools/relocs_32.o
HOSTCC arch/x86/tools/relocs_64.o
HOSTCC arch/x86/tools/relocs_common.o
HOSTLD arch/x86/tools/relocs
HOSTCC scripts/selinux/genheaders/genheaders
HOSTCC scripts/selinux/mdp/mdp
HOSTCC scripts/bin2c
HOSTCC scripts/kallsyms
HOSTCC scripts/conmakehash
HOSTCC scripts/recordmcount
HOSTCC scripts/sortextable
HOSTCC scripts/asn1_compiler
HOSTCC scripts/sign-file
HOSTCC scripts/extract-cert
HOSTCC scripts/insert-sys-cert
WRAP arch/x86/include/generated/uapi/asm/bpf_perf_event.h
WRAP arch/x86/include/generated/uapi/asm/errno.h
```

شکل ۳: استفاده از دستور make

- با راه اندازی مجدد ۳ کامپیوتر می توان هسته ی نصب شده را انتخاب کرد.

restart^۳