

운영체제 과제2

김지원

201501480

정보통신공학부

목차

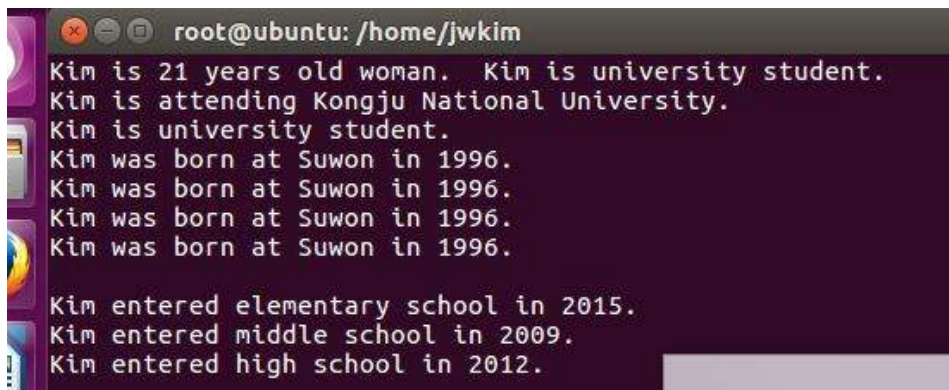
1. 과제 목적
2. 수행 내역 & 수행 결과
3. 수행 방법
4. 요구등급과 근거
5. 작업 일지

1. 과제 목적

리눅스 명령어를 공부하고 vi 사용법을 익힌후 vi 편집 실습 및 텔넷 서버, ftp 서버 및 웹서버의 구축 과정과 계정관리를 통해 공부한 것을 확인.

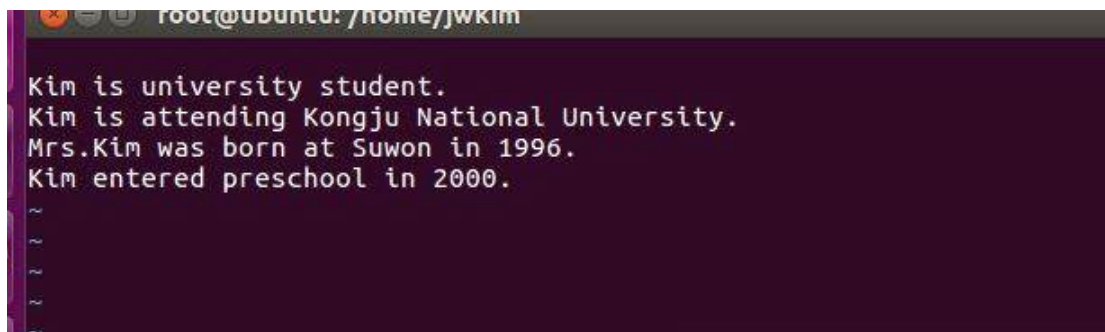
2. 수행 내역 & 수행 결과

-vi 실습

A terminal window titled 'root@ubuntu: /home/jwkim' showing the contents of a file named 'test1' edited with vi. The text in the file is: 'Kim is 21 years old woman. Kim is university student. Kim is attending Kongju National University. Kim is university student. Kim was born at Suwon in 1996. Kim was born at Suwon in 1996. Kim was born at Suwon in 1996. Kim was born at Suwon in 1996. Kim entered elementary school in 2015. Kim entered middle school in 2009. Kim entered high school in 2012.'

```
root@ubuntu: /home/jwkim
Kim is 21 years old woman. Kim is university student.
Kim is attending Kongju National University.
Kim is university student.
Kim was born at Suwon in 1996.
Kim was born at Suwon in 1996.
Kim was born at Suwon in 1996.
Kim was born at Suwon in 1996.
Kim entered elementary school in 2015.
Kim entered middle school in 2009.
Kim entered high school in 2012.
```

파일 test1의 결과창입니다.

A terminal window titled 'root@ubuntu: /home/jwkim' showing the contents of a file named 'test2' edited with vi. The text in the file is: 'Kim is university student. Kim is attending Kongju National University. Mrs.Kim was born at Suwon in 1996. Kim entered preschool in 2000.' followed by several empty lines.

```
root@ubuntu: /home/jwkim
Kim is university student.
Kim is attending Kongju National University.
Mrs.Kim was born at Suwon in 1996.
Kim entered preschool in 2000.
~
~
~
~
~
```

파일 test2의 결과창입니다.

-텔넷 서버 구축

윈도우 운영체제에서 다수의 명령창을 띄운 후 각각telnet으로 우분투에 접속하는 방법으로 다수의 우분투 작업을 동시에 수행할 수 있는 방법

설명

1)먼저 호스트에 텔넷 클라이언트 기능이 동작하는지 확인해야 함.
제어판의 "windows 기능 사용/사용 안함" 에서 '텔넷 클라이언트'에 체크함.
=> 윈도우에서 텔넷 클라이언트 기능 활성화됨.

2)우분투에서 텔넷 서버의 설치.

```
sudo apt-get install xinetd
```

```
sudo apt-get install telnetd
```

 를 통해 텔넷 서버를 설치 할 수 있음.

3)cd /etc/xinetd.d/ -> 디렉토리 이동.

```
vi telnet
```

 -> 파일 생성.

```
service telnet
```

```
{  
disable = no  
flags = REUSE  
socket_type = stream  
wait = no  
user = root  
server = /usr/sbin/in.telnetd  
log_on_failure += USERID  
}
```

-> telnet 파일에 작성하고 저장해줌.

```
service xinetd restart
```

 명령어를 수행시켜 변경사항이 반영되도록 함.

4)root 계정은 기본적으로 텔넷 접속이 차단되어 있음. 이를 해결하려면.

vi /etc/securetty

“console” 줄 다음에 아래에 pts 줄들을 추가

```
jwkim@ubuntu: /etc
/etc/securetty: list of terminals on which root is allowed to login.
# See securetty(5) and login(1).

console

pts/0
pts/1
pts/2
.....
pts/100
```

“service xinetd restart” 명령어를 수행

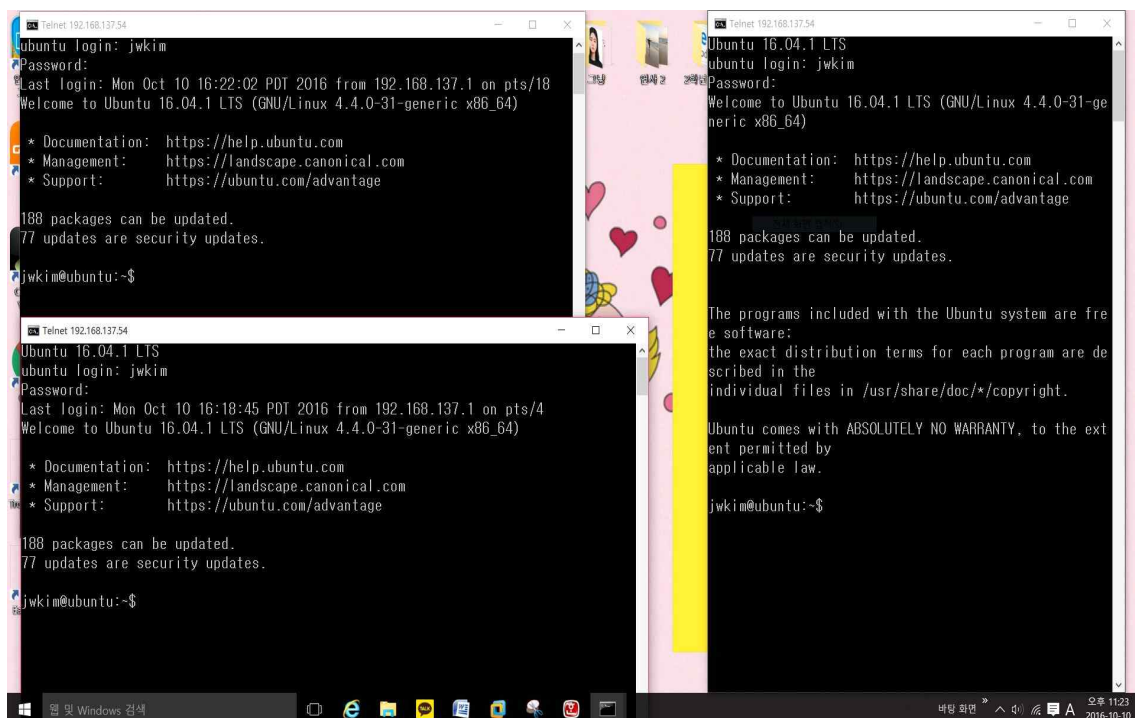
여기서 100은 텔넷 접속 수임. 필자는 100으로 지정.

5)cmd (명령 프롬프트 실행)

telnet IP주소 -> 로그인창 실행

이때 IP주소는 우분투에 할당된 것을 말함.

(우분투에서 ifconfig 를 통해서 IP주소 확인가능함)



3개의 윈도우 명령창을 띄우고 각각의 창을 통해 우분투에 로그인한 결과 화면입니다.

-FTP 서버 구축

설명

1)우분투에 vsftp라는 패키지를 설치하면 바로 우분투가 ftp 서버로 동작하므로 윈도우7에서 필요한 파일을 업로딩, 다운로드 할 수 있음.

vsftp 패키지 설치

(root 계정에서 수행할 경우 sudo 명령어는 생략)

```
$ sudo apt-get install vsftpd
```

2) 필요한 환경을 설정

vi vsftpd.conf -> vsftpd.conf 파일 읽기.

"write_enable=YES"가 주석처리 되어 있어 업로딩이 차단되어 있으므로 주석을 제거해줌.

vi /etc/ftpusers ->/etc/ftpusers 파일 읽기.

root앞에 주석 "#"를 추가.

리눅스 시스템을 다루는 본 실습은 root 계정을 사용하는 것이 편리하므로 root 계정으로 원격지에서 우분투 ftp 서버를 접근할 수 있도록해주는 것임.

"service vsftpd restart" 명령어 수행시켜줌.

3) 알ftp(알드라이브) 설치

윈도7에서 구축한 리눅스 ftp 서버에 접속되는지 확인.

The screenshot shows a '접속하기' (Connect) dialog box with the following details:

- Protocol:** FTP (File Transfer Protocol)
- Host:** 192.168.137.85
- Port:** 21
- Protocol Type:** Use plain FTP
- Username:** root
- Password:** Masked with dots (••••)
- Server Root Path:** /root

The '연결' (Connect) button is highlighted in yellow.

파일을 간단히 업로딩, 다운로드를 실행할 수 있음.

-웹사이트 구축

웹 서버를 제대로 구축하기 위해서는 웹 서버 뿐만 아니라 관련 소프트웨어인 데이터베이스 관리시스템, PHP 등 다양한 패키지를 설치해야 한다. 여기서는 아파치 웹 서버만 설치하여 기본적인 웹 서비스만 구축하고 확인한다.

1)

`su root` -> 우분투에서 root계정으로 로그인 함.

`sudo apt-get install apache2` ->아파치 웹서버 구축.

`sudo /etc/init.d/apache2 restart` -> 설치된 웹서버를 가동.

2)

cd /etc/apache2/sites-available/ -> 디렉터리 이동.

vi 000-default.conf ->파일로 이동함.

"DocumentRoot /var/www"를 찾아서 DocumentRoot /root로 바꾸어 줌.

cp test1 /root -> test1 root디렉터리에 복사해줌.

```
cd /root
```

mv test1 index.html ->test1을 index.html로 이름을 바꿔
줌.

```
cd /etc/apache2
```

```
vi apache2.conf -> apache2.conf 폴더로 이동.
```

```
chmod 755 /root
```

```
chmod 755 /root/index.html
```

```
sudo /etc/init.d/apache2 restart
```

3)웹브라우저에서 http://우분투 IP주소를 입력.

-> <http://192.158.137.85>



=> 1)~3)의 결과창입니다.

4) 작업한 초기문서를 USB에 백업.

"fdisk -l"

현재 시스템에 있는 파티션, 즉 논리 디스크 드라이브를 list하여 장착한 USB의 디바이스 이름(일반적으로 /dev/sdb1)을 파악.

USB 파일 시스템을 마운트 할 디렉토리인 /mnt/usb1가 없을 경우

"mkdir /mnt/usb1" 명령어로 생성.

"mount /dev/sdb1 /mnt/usb1" 의 명령어로 USB의 파티션을 /mnt/usb1 디렉토리에 접속.

"ls /mnt/usb1" 명령어로 USB 에 있는 내용물을 확인. 위에서 작성한 홈페이지 초기문서 파일을 USB에 복사함


```
root@ubuntu: ~  
I/O size (minimum/optimal): 4096 bytes / 4096 bytes  
  
Disk /dev/ram14: 64 MiB, 67108864 bytes, 131072 sectors  
Units: sectors of 1 * 512 = 512 bytes  
Sector size (logical/physical): 512 bytes / 4096 bytes  
I/O size (minimum/optimal): 4096 bytes / 4096 bytes  
  
Disk /dev/ram15: 64 MiB, 67108864 bytes, 131072 sectors  
Units: sectors of 1 * 512 = 512 bytes  
Sector size (logical/physical): 512 bytes / 4096 bytes  
I/O size (minimum/optimal): 4096 bytes / 4096 bytes  
  
Disk /dev/sda: 20 GiB, 21474836480 bytes, 41943040 sectors  
Units: sectors of 1 * 512 = 512 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disklabel type: dos  
Disk identifier: 0x3173daad  
  
Device      Boot      Start      End  Sectors  Size Id Type  
/dev/sda1   *                2048 39845887 39843840  19G 83 Linux  
/dev/sda2                39847934 41940991 2093058 1022M  5 Extended  
/dev/sda5                39847936 41940991 2093056 1022M 82 Linux swap / Solaris  
  
root@ubuntu:/home/jwkin# mkdir /mnt/usb1  
root@ubuntu:/home/jwkin# mount /dev/sda1 /mnt/usb1  
root@ubuntu:/home/jwkin# ls /mnt/usb1  
bin  cdrom  etc  initrd.img  lib  lost+found  mnt  proc  run  snap  sys  usr  vmlinuz  
boot  dev  home  initrd.img.old  lib64  media  opt  root /sbin  srv  tmp  var  vmlinuz.old  
root@ubuntu:/home/jwkin# cp index.html /mnt/usb1  
cp: cannot stat 'index.html': No such file or directory  
root@ubuntu:/home/jwkin# cd ~  
root@ubuntu:~# cp index.html /mnt/usb1  
root@ubuntu:~# ls /mnt/usb1  
bin  cdrom  etc  index.html  initrd.img.old  lib64  media  opt  root /sbin  srv  tmp  var  vmlinuz  vmlinuz.old  
boot  dev  home  initrd.img  lib  lost+found  mnt  proc  run  snap  sys  usr  vmlinuz  
root@ubuntu:~#
```

=> 4)의 결과창입니다.

-계정 관리과 프로세스 관리

설명

1) useradd로 계정을 추가함.
(참고로 계정관리는 root 계정만 할 수 있음.)

```
#useradd -m jwkim  
#useradd -m 201501480  
#useradd -m kim1480
```

2)계정을 추가하면 /home 디렉토리에 그 계정명과 동일한 디렉토리가 생김.
이를 그 계정의 홈 디렉토리라 함.

```
#cd /home  
#ls -l
```

jwkim, 201501480, kim1480 라는 하부 디렉토리가 생성된 것을 알 수 있음.

3)계정이 추가되면 그 계정의 동작환경이 설정되며, 동작환경에 관한 정보들이 홈 디렉토리에 숨겨진 파일로 저장됨. ls -al (a 옵션은 숨겨진 파일 보기임)을 사용.

```
#cd jwkim  
#ls -al
```

```
drwxr-xr-x 16 jwkim jwkim 4096 Oct 11 08:03 jwkim
root@ubuntu:/home# useradd -m jwkim
useradd: user 'jwkim' already exists
root@ubuntu:/home# userdel jwkim
userdel: user jwkim is currently used by process 1417
root@ubuntu:/home# cd jwkim
root@ubuntu:/home/jwkim# cd ..
root@ubuntu:/home# cd 201501480
bash: cd: 201501480: No such file or directory
root@ubuntu:/home# useradd -m 201501480
root@ubuntu:/home# useradd -m kim1480
root@ubuntu:/home# cd /home
root@ubuntu:/home# ls -l
total 12
drwxr-xr-x  2 201501480 201501480 4096 Oct 11 08:16 201501480
drwxr-xr-x 16 jwkim jwkim 4096 Oct 11 08:03 jwkim
drwxr-xr-x  2 kim1480 kim1480 4096 Oct 11 08:16 kim1480
root@ubuntu:/home# cd jwkim
root@ubuntu:/home/jwkim# ls -al
total 136
drwxr-xr-x 16 jwkim jwkim 4096 Oct 11 08:03 .
drwxr-xr-x  5 root root 4096 Oct 11 08:16 ..
-rw-r--r--  1 jwkim jwkim  57 Oct 10 17:34 .bash_history
-rw-r--r--  1 jwkim jwkim 220 Oct 10 13:50 .bash_logout
-rw-r--r--  1 jwkim jwkim 3771 Oct 10 13:50 .bashrc
-rwx----- 11 jwkim jwkim 4096 Oct 10 16:18 .cache
drwx----- 14 jwkim jwkim 4096 Oct 10 13:55 .config
drwxr-xr-x  2 jwkim jwkim 4096 Oct 10 13:53 Desktop
-rw-r--r--  1 jwkim jwkim  25 Oct 10 13:53 .dmrc
drwxr-xr-x  2 jwkim jwkim 4096 Oct 10 13:53 Documents
drwxr-xr-x  2 jwkim jwkim 4096 Oct 10 13:53 Downloads
-rw-r--r--  1 jwkim jwkim 8980 Oct 10 13:50 examples.desktop
drwx-----  2 jwkim jwkim 4096 Oct 10 13:54 .gconf
drwx-----  3 jwkim jwkim 4096 Oct 11 08:03 .gnupg
-rw-r--r--  1 jwkim jwkim 1272 Oct 11 08:03 .ICEauthority
drwxr-xr-x  2 root root 4096 Oct 10 15:51 jeewon
drwx-----  3 jwkim jwkim 4096 Oct 10 13:53 .local
drwxr-xr-x  2 jwkim jwkim 4096 Oct 10 13:53 Music
```

=> 1)~3) 과정의 결과창입니다.

4) jwkim, 201501480, kim1480 계정이 생성되었지만 패스워드가 없어 아직 사용할 수 없음. passwd라는 명령어를 사용하여 암호를 부여.

```
#passwd jwkim
```

```
#passwd 201501480
```

```
#passwd kim1480
```

(암호정책에 관한 확인 및 설정은 chage로 할 수 있음.)

```
root@ubuntu:/home/jwkim# passwd jwkim
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
root@ubuntu:/home/jwkim# passwd 201501480
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
root@ubuntu:/home/jwkim# passwd kim1480
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
```

=> 4) 과정의 결과창입니다.

5) 계정과 암호 등에 관한 정보는 /etc/passwd 파일에 저장된다. cat, tail 명령어로 내용을 확인.

```
#cd /etc
```

```
#cat passwd
```

```
#tail -10 passwd
```

(tail 명령어의 경우 마지막 10줄만 보여줌)

```
root@ubuntu:/home/jwkim# cd /etc
root@ubuntu:/etc# cat passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
syslog:x:104:108:/:/home/syslog:/bin/false
_apt:x:105:65534:/:/nonexistent:/bin/false
messagebus:x:106:110:/:/var/run/dbus:/bin/false
uuidd:x:107:111:/:/run/uuidd:/bin/false
lightdm:x:108:114:Light Display Manager:/var/lib/lightdm:/bin/false
whoopsie:x:109:116:/:/nonexistent:/bin/false
avahi-autoipd:x:110:119:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/bin/false
avahi:x:111:120:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/bin/false
dnsmasq:x:112:65534:dnsmasq,,,:/var/lib/misc:/bin/false
colord:x:113:123:colord colour management daemon,,,:/var/lib/colord:/bin/false
speech-dispatcher:x:114:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/false
hplip:x:115:7:HPLIP system user,,,:/var/run/hplip:/bin/false
kernoops:x:116:65534:Kernel Oops Tracking Daemon,,,:/bin/false
pulse:x:117:124:PulseAudio daemon,,,:/var/run/pulse:/bin/false

root@ubuntu:/etc# tail -10 passwd
kernoops:x:116:65534:Kernel Oops Tracking Daemon,,,:/bin/false
pulse:x:117:124:PulseAudio daemon,,,:/var/run/pulse:/bin/false
rtkit:x:118:126:RealtimeKit,,,:/proc:/bin/false
saned:x:119:127:/:/var/lib/saned:/bin/false
usbmux:x:120:46:usbmux daemon,,,:/var/lib/usbmux:/bin/false
jwkim:x:1000:1000:Kim Jee Won,,,:/home/jwkim:/bin/bash
telnetd:x:121:129:/:/nonexistent:/bin/false
ftp:x:122:130:ftp daemon,,,:/srv/ftp:/bin/false
201501480:x:1001:1001:/:/home/201501480:
kim1480:x:1002:1002:/:/home/kim1480:
```

=> 5) 과정의 결과창입니다.

6) 계정을 삭제는 userdel로 할 수 있음. (그 계정의 홈디렉토리의 파일 내용까지 모조리 삭제하고자 하는 경우는 -r 옵션을 사용하고, 계정만 삭제하고자 할 경우는 기본값을 사용함.)

```
#userdel -r kim1480
```

```
root@ubuntu:/etc# userdel -r kim1480
userdel: kim1480 mail spool (/var/mail/kim1480) not found
root@ubuntu:/etc# cd /home;ls -l
total 8
drwxr-xr-x  2 201501480 201501480 4096 Oct 11 08:16 201501480
drwxr-xr-x 16 jwkim      jwkim      4096 Oct 11 08:03 jwkim
```

kim1480의 디렉토리가 삭제되었는지 확인.

```
#cd /home;ls -l
```

“more /etc/passwd” 명령어로 패스워드 파일에 kim1480의 자료가 삭제되었는지 확인

```
root@ubuntu:/home# more /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin)/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
syslog:x:104:108:/:/home/syslog:/bin/false
_apt:x:105:65534:/:/nonexistent:/bin/false
messagebus:x:106:110:/:/var/run/dbus:/bin/false
uidd:x:107:111:/:/run/uidd:/bin/false
lightdm:x:108:114:Light Display Manager:/var/lib/lightdm:/bin/false
whoopsie:x:109:116:/:/nonexistent:/bin/false
avahi-autoipd:x:110:119:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/bin/false
avahi:x:111:120:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/bin/false
dnsmasq:x:112:65534:dnsmasq,,,:/var/lib/misc:/bin/false
colord:x:113:123:colord colour management daemon,,,:/var/lib/colord:/bin/false
speech-dispatcher:x:114:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/false
hplip:x:115:7:HPLIP system user,,,:/var/run/hplip:/bin/false
kernoops:x:116:65534:Kernel Oops Tracking Daemon,,,:/bin/false
pulse:x:117:124:PulseAudio daemon,,,:/var/run/pulse:/bin/false
rtkit:x:118:126:RealtimeKit,,,:/proc:/bin/false
saned:x:119:127:/:/var/lib/saned:/bin/false
usbmux:x:120:46:usbmux daemon,,,:/var/lib/usbmux:/bin/false
```

=> 6) 의 결과창입니다.

7) root 계정의 명령창에서 "who" 명령어 입력

현재 사용하고 있는 계정과 터미널 이름 및 접속시간 확인.

(터미널 이름은 해당 명령창을 지원하는 우분투 접속 단말을 의미하는 것으로 자체 접속의 경우 ttyn, 텔넷 접속의 경우 pts/n, 여기서 n은 번호로 표시됨.)

"whoami" 명령어 입력

현재 명령창을 이용하는 계정 확인.

```
root@ubuntu:/home# who
jwkim    tty7          2016-10-11 08:03 (:0)
root@ubuntu:/home# whoami
root
```

=> 7) 의 결과창입니다.

8) root 계정에서 "ps" 명령어 입력

그 계정에 대한 프로세스만 나열.

전체 사용자의 프로세스를 알아보려면 옵션을 사용.

옵션은 -aux, -eal 등 있음.

"ps -eal" 명령어를 입력하면 여러 페이지의 프로세스 관련 정보가 혹 지나가므로 제대로 읽을 수 가 없음.

"ps -eal|more"를 입력하면 스페이스바 키를 이용하여 한페이지씩 읽기 가능.

```

root@ubuntu:/home# ps
  PID TTY          TIME CMD
 2073 pts/17    00:00:00 su
 2075 pts/17    00:00:00 bash
 2326 pts/17    00:00:00 ps
root@ubuntu:/home# ps -eal
 F S      UID          PID    PPID    C    PRI   NI     ADDR   SZ    WCHAN    TTY          TIME CMD
 4 S      0              1        0      0   80    0      - 46349  ep_pol ?         00:00:02 systemd
 1 S      0              2        0      0   80    0      -      0 kthrea ?         00:00:00 kthreadd
 1 S      0              3        2      0   80    0      -      0 smpboo ?         00:00:00 ksoftirqd/0
 1 S      0              4        2      0   80    0      -      0 worker ?         00:00:00 kworker/0:0
 1 S      0              5        2      0   60   -20   -      0 worker ?         00:00:00 kworker/0:0H
 1 S      0              7        2      0   80    0      -      0 rcu_gp ?         00:00:00 rcu_sched
 1 S      0              8        2      0   80    0      -      0 rcu_gp ?         00:00:00 rcu_bh
 1 S      0              9        2     -40   -      0 smpboo ?         00:00:00 migration/0
 5 S      0             10        2     -40   -      0 smpboo ?         00:00:00 watchdog/0
 5 S      0             11        2      0   80    0      -      0 devtmp ?         00:00:00 kdevtmpfs
 1 S      0             12        2      0   60   -20   -      0 rescue ?         00:00:00 netns
 1 S      0             13        2      0   60   -20   -      0 rescue ?         00:00:00 perf
 1 S      0             14        2      0   80    0      -      0 watchd ?         00:00:00 khungtaskd
 1 S      0             15        2      0   60   -20   -      0 rescue ?         00:00:00 writeback
 1 S      0             16        2      0   85    5      -      0 ksm_sc ?         00:00:00 ksm
 1 S      0             17        2     99   19      0 khugep ?         00:00:00 khugepaged
 1 S      0             18        2      0   60   -20   -      0 rescue ?         00:00:00 crypto
 1 S      0             19        2      0   60   -20   -      0 rescue ?         00:00:00 kintegrityd
 1 S      0             20        2      0   60   -20   -      0 rescue ?         00:00:00 bioset
 1 S      0             21        2      0   60   -20   -      0 rescue ?         00:00:00 kblockd
 1 S      0             22        2      0   60   -20   -      0 rescue ?         00:00:00 ata_sff
 1 S      0             23        2      0   60   -20   -      0 rescue ?         00:00:00 md
 1 S      0             24        2      0   60   -20   -      0 rescue ?         00:00:00 devfreq_wq
 1 S      0             28        2      0   80    0      -      0 kswapd ?         00:00:00 kswapd0
 1 S      0             29        2      0   60   -20   -      0 rescue ?         00:00:00 vmstat
 1 S      0             30        2      0   80    0      -      0 fsnoti ?         00:00:00 fsnotify_mark
 1 S      0             31        2      0   80    0      -      0 ecrypt ?         00:00:00 ecryptfs-kthrea
 1 S      0             47        2      0   60   -20   -      0 rescue ?         00:00:00 kthrotld
 1 S      0             48        2      0   60   -20   -      0 rescue ?         00:00:00 acpi_thermal_pm
 1 S      0             49        2      0   60   -20   -      0 rescue ?         00:00:00 bioset
 1 S      0             50        2      0   60   -20   -      0 rescue ?         00:00:00 bioset
 1 S      0             51        2      0   60   -20   -      0 rescue ?         00:00:00 bioset

```

```

root@ubuntu:/home# ps -eal|more
 F S      UID          PID    PPID    C    PRI   NI     ADDR   SZ    WCHAN    TTY          TIME CMD
 4 S      0              1        0      0   80    0      - 46349  ep_pol ?         00:00:02 systemd
 1 S      0              2        0      0   80    0      -      0 kthrea ?         00:00:00 kthreadd
 1 S      0              3        2      0   80    0      -      0 smpboo ?         00:00:00 ksoftirqd/0
 1 S      0              4        2      0   80    0      -      0 worker ?         00:00:00 kworker/0:0H
 1 S      0              5        2      0   60   -20   -      0 worker ?         00:00:00 kworker/0:0H
 1 S      0              7        2      0   80    0      -      0 rcu_gp ?         00:00:00 rcu_sched
 1 S      0              8        2      0   80    0      -      0 rcu_gp ?         00:00:00 rcu_bh
 1 S      0              9        2     -40   -      0 smpboo ?         00:00:00 migration/0
 5 S      0             10        2     -40   -      0 smpboo ?         00:00:00 watchdog/0
 5 S      0             11        2      0   80    0      -      0 devtmp ?         00:00:00 kdevtmpfs
 1 S      0             12        2      0   60   -20   -      0 rescue ?         00:00:00 netns
 1 S      0             13        2      0   60   -20   -      0 rescue ?         00:00:00 perf
 1 S      0             14        2      0   80    0      -      0 watchd ?         00:00:00 khungtaskd
 1 S      0             15        2      0   60   -20   -      0 rescue ?         00:00:00 writeback
 1 S      0             16        2      0   85    5      -      0 ksm_sc ?         00:00:00 ksm
 1 S      0             17        2     99   19      0 khugep ?         00:00:00 khugepaged
 1 S      0             18        2      0   60   -20   -      0 rescue ?         00:00:00 crypto
 1 S      0             19        2      0   60   -20   -      0 rescue ?         00:00:00 kintegrityd
 1 S      0             20        2      0   60   -20   -      0 rescue ?         00:00:00 bioset
 1 S      0             21        2      0   60   -20   -      0 rescue ?         00:00:00 kblockd
 1 S      0             22        2      0   60   -20   -      0 rescue ?         00:00:00 ata_sff
 1 S      0             23        2      0   60   -20   -      0 rescue ?         00:00:00 md
 1 S      0             24        2      0   60   -20   -      0 rescue ?         00:00:00 devfreq_wq
 1 S      0             28        2      0   80    0      -      0 kswapd ?         00:00:00 kswapd0
 1 S      0             29        2      0   60   -20   -      0 rescue ?         00:00:00 vmstat
 1 S      0             30        2      0   80    0      -      0 fsnoti ?         00:00:00 fsnotify_mark
 1 S      0             31        2      0   80    0      -      0 ecrypt ?         00:00:00 ecryptfs-kthrea
 1 S      0             47        2      0   60   -20   -      0 rescue ?         00:00:00 kthrotld
 1 S      0             48        2      0   60   -20   -      0 rescue ?         00:00:00 acpi_thermal_pm
 1 S      0             49        2      0   60   -20   -      0 rescue ?         00:00:00 bioset
 1 S      0             50        2      0   60   -20   -      0 rescue ?         00:00:00 bioset
 1 S      0             51        2      0   60   -20   -      0 rescue ?         00:00:00 bioset
 1 S      0             52        2      0   60   -20   -      0 rescue ?         00:00:00 bioset
 1 S      0             53        2      0   60   -20   -      0 rescue ?         00:00:00 bioset
 1 S      0             54        2      0   60   -20   -      0 rescue ?         00:00:00 bioset
 1 S      0             55        2      0   60   -20   -      0 rescue ?         00:00:00 bioset
 1 S      0             56        2      0   60   -20   -      0 rescue ?         00:00:00 bioset
 1 S      0             57        2      0   60   -20   -      0 rescue ?         00:00:00 bioset
 1 S      0             58        2      0   60   -20   -      0 rescue ?         00:00:00 bioset
 1 S      0             59        2      0   60   -20   -      0 rescue ?         00:00:00 bioset

```

=> 8) 의 결과창입니다.

9)프로세서 족보

"more" 명령어가 보여주는 자료는 필요없는 정보까지 섞여있어 필요한 정보를 알아 내기가 쉽지 않음. 이럴때 "grep"사용.

알고자 하는 것은 root, tjkim 및 20121234 계정이 사용하는 터미널, 예로 pts/1, tty2, pts/2 의 이름을 이용하여 이들 이름이 들어가 있는 줄만 찾아 나열하면 알고자 하는 족보를 더 쉽게 파악할 수 있음.

"ps -eal|grep pts/1"

"ps -eal|grep tty2"

"ps -eal|grep pts/2"


```
root@ubuntu:/home# ps -eal|grep pts/1
0 S 1000 2031 2024 0 80 0 - 7373 wait pts/17 00:00:00 bash
4 S 0 2073 2031 0 80 0 - 15361 wait pts/17 00:00:00 su
4 S 0 2075 2073 0 80 0 - 7078 wait pts/17 00:00:00 bash
4 S 0 2349 2075 0 80 0 - 15361 wait pts/17 00:00:00 su
4 S 1001 2350 2349 0 80 0 - 7373 wait pts/17 00:00:00 bash
4 S 0 2362 2350 0 80 0 - 15361 wait pts/17 00:00:00 su
4 S 0 2365 2362 0 80 0 - 7073 wait pts/17 00:00:00 bash
4 R 0 2428 2365 0 80 0 - 8996 - pts/17 00:00:00 ps
0 S 0 2429 2365 0 80 0 - 5323 pipe_w pts/17 00:00:00 grep
root@ubuntu:/home# ps -eal|grep tty2
root@ubuntu:/home# ps -eal|grep pts/2
root@ubuntu:/home#
```

=> 9) 의 결과창입니다.

3. 수행 방법

-텔넷 서버 구축

설명

1)먼저 호스트에 텔넷 클라이언트 기능이 동작하는지 확인해야 함.
제어판의 "windows 기능 사용/사용 안함" 에서 '텔넷 클라이언트'에 체크함.
=> 윈도우7에서 텔넷 클라이언트 기능 활성화됨.

2)우분투에서 텔넷 서버의 설치.

```
sudo apt-get install xinetd
```

```
sudo apt-get install telnetd
```

 를 통해 텔넷 서버를 설치 할 수 있음.

3)cd /etc/xinetd.d/ -> 디렉토리 이동.

```
vi telnet
```

 -> 파일 생성.

```
service telnet
```

```
{
```

```
disable = no
```

```
flags = REUSE
```

```
socket_type = stream
```

```
wait = no
```

```
user = root
```



```
server = /usr/sbin/in.telnetd
log_on_failure += USERID
}
```

-> telnet 파일에 작성하고 저장해줌.

service xinetd restart 명령어를 수행시켜 변경사항이 반영되도록 함.

4)root 계정은 기본적으로 텔넷 접속이 차단되어 있음. 이를 해결하려면.

vi /etc/securetty

"console" 줄 다음에 아래에 pts 줄들을 추가

"service xinetd restart" 명령어를 수행

여기서 100은 텔넷 접속 수임. 필자는 100으로 지정.

5)cmd (명령 프롬프트 실행)

telnet IP주소 -> 로그인창 실행

이때 IP주소는 우분투에 할당된 것을 말함.

(우분투에서 ifconfig 를 통해서 IP주소 확인가능함)

3개의 윈도우 명령창을 띄우고 각각의 창을 통해 우분투에 로그인한 결과 화면입니다.

-FTP 서버 구축

설명

1)우분투에 vsftp라는 패키지를 설치하면 바로 우분투가 ftp 서버로 동작하므로 윈도우7에서 필요한 파일을 업로딩, 다운로드 할 수 있음.

vsftp 패키지 설치

(root 계정에서 수행할 경우 sudo 명령어는 생략)

```
$ sudo apt-get install vsftpd
```

2) 필요한 환경을 설정

vi vsftpd.conf -> vsftpd.conf 파일 읽기.

"write_enable=YES"가 주석처리 되어 있어 업로딩이 차단되어 있으므로 주석을 제거해줌.

vi /etc/ftpusers ->/etc/ftpusers 파일 읽기.

root앞에 주석 "#"를 추가.

리눅스 시스템을 다루는 본 실습은 root 계정을 사용하는 것이 편리하므로 root 계정으로 원격지에서 우분투 ftp 서버를 접근할 수 있도록해주는 것임.

"service vsftpd restart" 명령어 수행시켜줌.

3) 알ftp(알드라이브) 설치

윈도7에서 구축한 리눅스 ftp 서버에 접속되는지 확인.

파일을 간단히 업로딩, 다운로드를 실행할 수 있음.

-웹사이트 구축

웹 서버를 제대로 구축하기 위해서는 웹 서버 뿐만 아니라 관련 소프트웨어인 데이터베이스 관리시스템, PHP 등 다양한 패키지를 설치해야 한다. 여기서는 아파치 웹 서버만 설치하여 기본적인 웹 서비스만 구축하고 확인한다.

1)

su root ->우분투에서 root계정으로 로그인 함.

sudo apt-get install apache2 ->아파치 웹서버 구축.

sudo /etc/init.d/apache2 restart ->설치된 웹서버를 가동.

2)

cd /etc/apache2/sites-available/ ->디렉터리 이동.

vi 000-default.conf ->파일로 이동함.

"DocumentRoot /var/www"를 찾아서 DocumentRoot /root로 바꾸어 줌.

cp test1 /root -> test1 root디렉터리에 복사해줌.

cd /root

mv test1 index.html ->test1을 index.html로 이름을 바꿔 줌.

cd /etc/apache2

vi apache2.conf ->apache2.conf 폴더로 이동.

chmod 755 /root

chmod 755 /root/index.html

```
sudo /etc/init.d/apache2 restart
```

3)웹브라우저에서 http://우분투 IP주소를 입력.

->http://192.158.137.85

4) 작업한 초기문서를 USB에 백업.

```
"fdisk -l"
```

현재 시스템에 있는 파티션, 즉 논리 디스크 드라이브를 list하여 장착한 USB의 디바이스 이름(일반적으로 /dev/sdb1)을 파악.

USB 파일 시스템을 마운트 할 디렉토리인 /mnt/usb1가 없을 경우

```
"mkdir /mnt/usb1" 명령어로 생성.
```

```
"mount /dev/sdb1 /mnt/usb1" 의 명령어로 USB의 파티션을 /mnt/usb1 디렉토리에 접속.
```

"ls /mnt/usb1" 명령어로 USB 에 있는 내용물을 확인. 위에서 작성한 홈페이지 초기문서 파일을 USB에 복사함

-계정 관리과 프로세스 관리

설명

1) useradd로 계정을 추가함.

(참고로 계정관리는 root 계정만 할 수 있음.)

```
#useradd -m jwkim
```

```
#useradd -m 201501480
```

```
#useradd -m kim1480
```

2)계정을 추가하면 /home 디렉토리에 그 계정명과 동일한 디렉토리가 생김.

이를 그 계정의 홈 디렉토리라 함.

```
#cd /home
```

```
#ls -l
```

jwkim, 201501480, kim1480 라는 하부 디렉토리가 생성된 것을 알 수 있음.

3) 계정이 추가되면 그 계정의 동작환경이 설정되며, 동작환경에 관한 정보들이 홈 디렉토리에 숨겨진 파일로 저장됨. ls -al (a 옵션은 숨겨진 파일 보기임)을 사용.

```
#cd jwkim
#ls -al
```

4) jwkim, 201501480, kim1480 계정이 생성되었지만 패스워드가 없어 아직 사용할 수 없음. passwd라는 명령어를 사용하여 암호를 부여.

```
#passwd jwkim
#passwd 201501480
#passwd kim1480
```

(암호정책에 관한 확인 및 설정은 chage로 할 수 있음.)

5) 계정과 암호 등에 관한 정보는 /etc/passwd 파일에 저장된다. cat, tail 명령어로 내용을 확인.

```
#cd /etc
#cat passwd
#tail -10 passwd
```

(tail 명령어의 경우 마지막 10줄만 보여줌)

6) 계정을 삭제는 userdel로 할 수 있음. (그 계정의 홈디렉토리의 파일 내용까지 모조리 삭제하고자 하는 경우는 -r 옵션을 사용하고, 계정만 삭제하고자 할 경우는 기본값을 사용함.)

```
#userdel -r kim1480
```

kim1480의 디렉토리가 삭제되었는지 확인.

```
#cd /home;ls -l
```

"more /etc/passwd" 명령어로 패스워드 파일에 kim1480의 자료가 삭제되었는지 확인.

7) root 계정의 명령창에서 "who" 명령어 입력

현재 사용하고 있는 계정과 터미널 이름 및 접속시간 확인.

(터미널 이름은 해당 명령창을 지원하는 우분투 접속 단말을 의미하는 것으로 자체 접속의 경우 ttyn, 텔넷 접속의 경우 pts/n, 여기서 n은 번호로 표시됨.)

"whoami" 명령어 입력

현재 명령창을 이용하는 계정 확인.

8) root 계정에서 "ps" 명령어 입력

그 계정에 대한 프로세스만 나열.

전체 사용자의 프로세스를 알아보려면 옵션을 사용.

옵션은 -aux, -eal 등 있음.

"ps -eal" 명령어를 입력하면 여러 페이지의 프로세스 관련 정보가 훑 지나가므로 제대로 읽을 수 가 없음.

"ps -eal|more"를 입력하면 스페이스바 키를 이용하여 한페이지씩 읽기 가능.

9) 프로세서 족보

"more" 명령어가 보여주는 자료는 필요없는 정보까지 섞여있어 필요한 정보를 알아 내기가 쉽지 않음. 이럴때 "grep"사용.

알고자 하는 것은 root, tjkim 및 20121234 계정이 사용하는 터미널, 예로 pts/1, tty2, pts/2 의 이름을 이용하여 이들 이름이 들어가 있는 줄만 찾아 나열하면 알고자 하는 족보를 더 쉽게 파악할 수 있음.

"ps -eal|grep pts/1"

"ps -eal|grep tty2"

"ps -eal|grep pts/2"

4. 요구등급과 근거

저는 A등급을 받기를 희망합니다.

왜냐하면 이 과제가 과제1보다 너무 어려워서 친구 노한슬과 황혜진과 같이 하면서 과제를 끝낼 수 있었습니다. 그리고 원본 레포트의 오류들도 인터넷 검색으로 다 찾으면서 과제를 하나하나 이해하며 수행하였습니다.

그리고 이제는 처음부터 다시 하게 된다면 혼자서 할 수 있을 정도로 익히게 되었습니다. 그리고 다른 유사한 과제도 할 수 있게 되었습니다.

5. 작업 일지

일시 (소요시간)	작업명	작업 상세 내용	비고
10월10일 (30m)	vi실습	vi 명령어를 실행해 봄.	독자수 행
10월10일 (5m)	텔넷서버구축	텔넷 클라이언트 : "windows 기능 사용/사용 안 함" 체크해주기. 우분투에서 텔넷 서버의 설치.	독자수 행
10월10일 (20m)	텔넷서버구축	cd /etc/xinetd.d/ -> 디렉토리 이동. vi telnet -> 파일 생성. pts 줄들을 추가 명령 프롬프트 실행	독자수 행
10월10일 (30m)	FTP 서버 구축	vsftp 패키지 설치 필요한 환경을 설정 알ftp(알드라이브) 설치	독자수 행
10월11일 (15m)	웹사이트 구축	su root ->우분투에서 root계정으로 로그 인 함. sudo apt-get install apache2 ->아파치 웹서 버 구축. sudo /etc/init.d/apache2 restart ->설치된 웹서 버를 가동.	독자수 행
10월11일 (1h)	웹사이트 구축	cd /etc/apache2/sites-available/ vi 000-default.conf "DocumentRoot /var/www"를 찾아서 DocumentRoot /root로 바꾸어 줌. cp test1 /root cd /root mv test1 index.html cd /etc/apache2 vi apache2.conf chmod 755 /root chmod 755 /root/index.html sudo /etc/init.d/apache2 restart 웹브라우저에서 http://우분투 IP주소를 입력.	황혜진 과 노한슬 의 도움을 받아 수행.
10월11일 (20m)	웹사이트 구축	작업한 초기문서를 USB에 백업. "fdisk -l" "mkdir /mnt/usb1" "mount /dev/sdb1 /mnt/usb1" "ls /mnt/usb1"	독자수 행
10월11일	계정 관리과	.useradd로 계정을 추가함	독자수

(10m)	프로세스 관리	ls -l,ls -al 로 디렉토리가 생성된 것을 알 수 있음 passwd라는 명령어를 사용하여 암호를 부여	행
10월11일 (10m)	계정 관리과 프로세스 관리	userdel로 계정을 삭제. cd /home;ls -l : 로 디렉터리 삭제 확인. "who" 명령어 입력. "whoami" 명령어 입력.	독자수 행
10월11일 (15m)	계정 관리과 프로세스 관리	"ps -eal" : 페이지의 프로세스 관련 정보 확인. "ps -eal grep pts/1" "ps -eal grep tty2" "ps -eal grep pts/2" 프로세스 관련 정보 다시 한번 확인	독자수 행