



---

# 운영체제 과제 1

---

과목명	운영체제
교수명	김철홍 교수님
학 과	IT대학 컴퓨터학부
학 번	20172655
이 름	이강산
제출일	2021.09.19.

## < 목차 >

### 1. 리눅스 커널 컴파일 및 설치

- 1) 현재 커널 버전 확인
- 2) 커널 소스 다운로드
- 3) 필요 툴 설치 및 커널 컴파일
- 4) 커널 버전 변경 확인

### 2. 리눅스 시스템 상태 정보 조사

- 1) CPU 정보
- 2) 메모리 정보
- 3) 저장장치 정보
- 4) I/O장치 정보
- 5) 동작중인 프로세스 정보
- 6) 파일시스템 정보
- 7) 기타 OS에서 관리 중인 컴퓨터 시스템 정보

### 3. 과제를 통해 배운 점

## 1. 리눅스 커널 컴파일 및 설치

### 1) 현재 커널 버전 확인

```
san@linux:~$ cat /etc/os-release
NAME="Ubuntu"
VERSION="20.04.3 LTS (Focal Fossa)"
```

etc폴더의 os-release파일을 읽어 현재 동작중인 운영체제의 정보를 파악한다. 리눅스 커널 컴파일 및 설치가 진행될 환경은 Ubuntu 20.04 LTS이다.

```
san@linux:~$ uname -sr
Linux 5.4.0-84-generic
```

uname 명령어를 통해 현재 커널 버전을 확인한다. s옵션은 커널 이름을, r옵션은 커널 릴리즈를 출력한다. 5.4 버전의 커널은 2019년 11월 24일에 배포된 버전으로, 2021년 9월 18일 기준 가장 최신의 리눅스 커널은 5.14.6 버전이다.

### 2) 커널 소스 다운로드

```
san@linux:~$ wget https://www.kernel.org/pub/linux/kernel/v5.x/linux-5.14.6.tar.xz
```

wget 명령어를 사용하여 [www.kernel.org](https://www.kernel.org) 로부터 5.14.6 버전 커널을 다운로드 받는다.

```
root@linux:~# mkdir /usr/src/linux
root@linux:~# cp /home/san/linux-5.14.6.tar.xz /usr/src/linux
root@linux:~# cd /usr/src/linux
root@linux:/usr/src/linux# ls
linux-5.14.6.tar.xz
root@linux:/usr/src/linux# tar -xf linux-5.14.6.tar.xz
```

mkdir 명령어를 이용해 커널 소스를 컴파일 linux 디렉토리를 생성한다. 커널 소스를 linux 폴더에 복사한 후, tar 명령어를 이용해 압축을 해제한다.

### 3) 필요 툴 설치 및 커널 컴파일

```
root@linux:/usr/src/linux# apt-get install vim make gcc kernel-package libncurses5-dev bison flex libssl-dev
```

Linux 커널 컴파일에 필요한 라이브러리들 설치한다.

```
root@linux:/usr/src/linux# uname -sr
Linux 5.4.0-84-generic
root@linux:/usr/src/linux# cd linux-5.14.6/
root@linux:/usr/src/linux/linux-5.14.6# cp /boot/config-5.4.0-84-generic ./.config
root@linux:/usr/src/linux/linux-5.14.6# make menuconfig
```

현재 커널의 configuration 파일을 복사하여 .config란 이름으로 복사한다. 복사한 .config파일을 바탕으로 make 하여 새롭게 설치할 커널의 설정을 진행한다.

```
root@linux:/usr/src/linux/linux-5.14.6# make-kpkg --initrd --revision=1.0 kernel_i
mage
```

make-kpkg 명령어를 이용하여 커널을 컴파일을 해 이미지 파일을 생성한다.

```
root@linux:/usr/src/linux# ls
linux-5.14.6  linux-5.14.6.tar.xz  linux-image-5.14.6_1.0_amd64.deb
root@linux:/usr/src/linux# dpkg -l linux-image-5.14.6_1.0_amd64.deb
```

컴파일이 완료되어 deb파일이 생성되었다. dpkg 명령어를 이용하여 새로운 커널 이미지로 부팅한다.

#### 4) 커널 버전 변경 확인

- 변경 전

```
san@linux:~$ uname -sr
Linux 5.4.0-84-generic
```

- 변경 후 리눅스 커널이 5.14.6 버전으로 변경되었음을 알 수 있다.

```
san@linux:~$ uname -r
5.14.6-051406-generic
```

## 2. 리눅스 시스템 상태 정보 조사

- 리눅스 시스템 상태 정보는 proc file system내에 위치한 파일을 직접 cat하거나, 해당 장치의 정보를 알려주는 명령어를 입력하거나, DMI테이블 내용을 번역해주는 dmidecode명령어를 사용하는 등 다양한 방법을 통해 얻을 수 있다.

#### 1) CPU 정보

- 현재 시스템에서 사용중인 CPU는 Intel Core i7 7500U로 2개의 코어로 동작하는 CPU이다. CPU장치에 대한 정보는 lscpu 명령어를 통해 알아볼 수 있다.

```
san@linux:~$ lscpu
Architecture:            x86_64
CPU op-mode(s):          32-bit, 64-bit
Byte Order:               Little Endian
CPU(s):                   2
On-line CPU(s) list:     0,1
Thread(s) per core:       1
Core(s) per socket:       1
Socket(s):                2
NUMA node(s):            1
Vendor ID:                GenuineIntel
CPU family:               6
Model:                   142
Model name:               Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz
Stepping:                 9
CPU MHz:                  2904.003
BogoMIPS:                 5808.00
Hypervisor vendor:        VMware
Virtualization type:      full
L1d cache:                32K
L1i cache:                32K
L2 cache:                 256K
L3 cache:                 4096K
```

- lscpu 명령어를 실행시킨 결과이다. 모델명(i7-7500u), 아키텍처(64비트), 바이트 오더 방식(little endian), 코어 수(2), CPU 클럭(2900MHz), 계층 별 캐시메모리 크기 등의 정보를 확인 할 수 있다.

## 2) 메모리 정보

- 메모리 용량 정보는 proc file system의 meminfo 파일을 참조하였다. 가상머신에 설치 당시 4GB만큼의 용량을 할당하였고, cat 명령어를 통해 확인하였다.

```
san@linux:~$ cat /proc/meminfo | grep 'MemTotal'
MemTotal:      4001708 kB
```

- free 명령어를 사용하면 현재 사용 중인 메모리의 정보를 파악할 수 있다.

```
san@linux:~$ free
              total        used         free       shared    buff/cache   available
Mem:           4001708      1410748       585368         9440       2005592       2320636
스왑:              969960           0        969960
```

## 3) 저장장치 정보

- 디스크 관련 정보는 df 명령어를 통해 알 수 있다. h 옵션은 바이트 단위로 표현되는 size를 더 큰 단위로 표현한다. /dev/sda1의 size는 50GB로 가상머신에 설치 당시 할당한 용량과 동일하며, 현재 33GB(70%)만큼 사용 중임을 알 수 있다.

```
san@linux:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            1.9G   0    1.9G   0% /dev
tmpfs           391M  2.0M  389M   1% /run
/dev/sda1       50G   33G   15G   70% /
tmpfs           2.0G   0    2.0G   0% /dev/shm
tmpfs           5.0M  4.0K  5.0M   1% /run/lock
tmpfs           2.0G   0    2.0G   0% /sys/fs/cgroup
/dev/loop0      2.3M  2.3M   0 100% /snap/gnome-system-monitor/157
/dev/loop1      640K  640K   0 100% /snap/gnome-logs/106
/dev/loop2      242M  242M   0 100% /snap/gnome-3-38-2004/70
/dev/loop3      2.5M  2.5M   0 100% /snap/gnome-calculator/884
/dev/loop5       62M   62M   0 100% /snap/core20/1081
/dev/loop4       66M   66M   0 100% /snap/gtk-common-themes/1515
/dev/loop6       56M   56M   0 100% /snap/core18/2128
/dev/loop7      219M  219M   0 100% /snap/gnome-3-34-1804/72
/dev/loop8      768K  768K   0 100% /snap/gnome-characters/726
/dev/loop9      2.5M  2.5M   0 100% /snap/gnome-calculator/748
/dev/loop10     219M  219M   0 100% /snap/gnome-3-34-1804/66
/dev/loop11      56M   56M   0 100% /snap/core18/2066
/dev/loop12     640K  640K   0 100% /snap/gnome-logs/103
/dev/loop13     384K  384K   0 100% /snap/gnome-characters/708
/dev/loop14      33M   33M   0 100% /snap/snapd/12159
/dev/loop15      65M   65M   0 100% /snap/gtk-common-themes/1514
/dev/loop17     2.5M  2.5M   0 100% /snap/gnome-system-monitor/163
/dev/loop16      33M   33M   0 100% /snap/snapd/12883
tmpfs           391M   16K  391M   1% /run/user/121
tmpfs           391M   32K  391M   1% /run/user/1000
```

#### 4) I/O장치 정보

- iostat 명령어를 통해 디스크에서의 입출력 통계 및 처리량, 대기열 길이 등을 확인할 수 있다. 표현된 정보는 다음과 같다.

tps : 초당 입출력 작업 건수

KB\_read/s, KB\_wrtn/s : 초당 읽거나 쓴 양(kB)

KB\_read, KB\_wrtn : 읽거나 쓴 양(kB)

```
san@linux:~$ iostat
Linux 5.14.6-051406-generic (linux)      2021년 09월 19일      _x86_64_      (2 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           1.79    0.14    1.23    0.07    0.00   96.76

Device            tps    kB_read/s    kB_wrtn/s    kB_dscd/s    kB_read    kB_wrtn    kB_dscd
d
loop0              0.02         0.16         0.00         0.00        346         0
loop1              0.02         0.17         0.00         0.00        359         0
loop10             0.01         0.03         0.00         0.00         59         0
loop11             0.02         0.06         0.00         0.00        132         0
loop12             0.02         0.06         0.00         0.00        128         0
loop13             0.02         0.06         0.00         0.00        137         0
loop14             0.03         0.50         0.00         0.00       1081         0
loop15             0.23         7.07         0.00         0.00      15218         0
loop16             0.02         0.16         0.00         0.00        347         0
loop17             0.03         0.49         0.00         0.00       1065         0
loop18             0.01         0.01         0.00         0.00         14         0
loop2              0.02         0.17         0.00         0.00        365         0
loop3              0.03         0.50         0.00         0.00       1075         0
loop4              0.02         0.50         0.00         0.00       1082         0
loop5              0.02         0.06         0.00         0.00        128         0
loop6              0.03         0.51         0.00         0.00       1092         0
loop7              0.01         0.03         0.00         0.00         57         0
loop8              0.01         0.03         0.00         0.00         68         0
loop9              0.01         0.03         0.00         0.00         68         0
sda                12.37        419.43       139.03         0.00     903117     299360
scd0               0.01         0.00         0.00         0.00         2         0
```



## 5) 동작중인 프로세스 정보

- 현재 동작 중인 프로세스들의 정보는 ps 명령어를 통해 알아낼 수 있다. aux 옵션을 주어 순서대로 다음과 같은 정보를 알 수 있다.

USER : 소유자명

PID : 식별번호

%CPU : CPU 점유율

%MEM : 메모리 점유율

VSZ, RSS : 실제 메모리 사용량

TTY : 프로세스 동작 중인 터미널 번호

STAT : 현재 상태

START : 시작 시간

TIME : 총 사용 시간

COMMAND : 프로세스 실행시킨 명령어

```
san@linux:~$ ps -aux
```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.7	0.2	102072	11432	?	Ss	21:44	0:01	/sbin/init auto noprompt
root	2	0.0	0.0	0	0	?	S	21:44	0:00	[kthreadd]
root	3	0.0	0.0	0	0	?	I<	21:44	0:00	[rcu_gp]
root	4	0.0	0.0	0	0	?	I<	21:44	0:00	[rcu_par_gp]
root	5	0.1	0.0	0	0	?	I	21:44	0:00	[kworker/0:0-events]
root	6	0.0	0.0	0	0	?	I<	21:44	0:00	[kworker/0:0H-events_highp]
root	7	0.0	0.0	0	0	?	I	21:44	0:00	[kworker/0:1-events]
root	8	0.0	0.0	0	0	?	I	21:44	0:00	[kworker/u256:0-events_unb]
root	9	0.0	0.0	0	0	?	I<	21:44	0:00	[mm_percpu_wq]
root	10	0.0	0.0	0	0	?	S	21:44	0:00	[rcu_tasks_rude_]
root	11	0.0	0.0	0	0	?	S	21:44	0:00	[rcu_tasks_trace]
root	12	0.1	0.0	0	0	?	S	21:44	0:00	[ksoftirqd/0]
root	13	0.1	0.0	0	0	?	I	21:44	0:00	[rcu_sched]
root	14	0.0	0.0	0	0	?	S	21:44	0:00	[migration/0]

- top 명령어를 통해 운영체제의 작업 내역을 모니터링할 수 있다. 시스템 프로세스들의 CPU, Memory 점유율을 실시간으로 나타낸다.

```
top - 20:59:27 up 1:33, 1 user, load average: 0.02, 0.02, 0.01
```

Tasks: 317 total, 3 running, 245 sleeping, 0 stopped, 0 zombie  
 %Cpu(s): 2.0 us, 1.7 sy, 0.0 ni, 96.1 id, 0.0 wa, 0.0 hi, 0.2 si, 0.0 st  
 KiB Mem : 4001708 total, 113124 free, 1419632 used, 2468952 buff/cache  
 KiB Swap: 969960 total, 969692 free, 268 used, 2306416 avail Mem

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
3159	_apt	20	0	91328	9324	8168	R	3.0	0.2	2:02.16	http
1424	san	20	0	472532	77552	37412	S	2.7	1.9	1:14.68	Xorg
1537	san	20	0	3528100	200160	93524	S	2.3	5.0	1:07.82	gnome-shell
1851	san	20	0	819216	45468	31508	S	1.7	1.1	0:36.67	gnome-terminal-
3488	san	20	0	45656	4396	3496	R	0.3	0.1	0:00.07	top
1	root	20	0	159652	9020	6696	S	0.0	0.2	0:02.56	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.01	kthreadd
3	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_gp
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_par_gp
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H-kb
9	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_wq
10	root	20	0	0	0	0	S	0.0	0.0	0:00.17	ksoftirqd/0
11	root	20	0	0	0	0	R	0.0	0.0	0:03.04	rcu_sched
12	root	rt	0	0	0	0	S	0.0	0.0	0:00.03	migration/0
13	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/0
14	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/0
15	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/1
16	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/1
17	root	rt	0	0	0	0	S	0.0	0.0	0:00.13	migration/1
18	root	20	0	0	0	0	S	0.0	0.0	0:03.25	ksoftirqd/1

## 6) 파일시스템 정보

- 디스크가 사용 중인 파일시스템의 정보는 df 명령어에 타입을 나타내는 T 옵션을 주어 확인한다. 현재 디스크(/dev/sda1)는 파일시스템으로 ext4 방식을 사용 중이다.

```
san@linux:~$ df -Th
Filesystem      Type      Size  Used Avail Use% Mounted on
udev            devtmpfs  1.9G   0    1.9G   0% /dev
tmpfs           tmpfs     390M   1.8M 388M   1% /run
/dev/sda1       ext4      50G   34G   14G   72% /
tmpfs           tmpfs     2.0G   0    2.0G   0% /dev/shm
tmpfs           tmpfs     5.0M   4.0K 5.0M   1% /run/lock
tmpfs           tmpfs     2.0G   0    2.0G   0% /sys/fs/cgroup
/dev/loop0      squashfs  56M    0     0 100% /snap/core18/2066
/dev/loop1      squashfs  56M    0     0 100% /snap/core18/2128
/dev/loop2      squashfs  62M    0     0 100% /snap/core20/1081
/dev/loop3      squashfs 219M   219M   0 100% /snap/gnome-3-34-1804/72
/dev/loop5      squashfs 2.5M   2.5M   0 100% /snap/gnome-calculator/748
/dev/loop4      squashfs 219M   219M   0 100% /snap/gnome-3-34-1804/66
/dev/loop6      squashfs 242M   242M   0 100% /snap/gnome-3-38-2004/70
/dev/loop7      squashfs 384K   384K   0 100% /snap/gnome-characters/708
/dev/loop8      squashfs 768K   768K   0 100% /snap/gnome-characters/726
/dev/loop10     squashfs 640K   640K   0 100% /snap/gnome-logs/103
/dev/loop9      squashfs 640K   640K   0 100% /snap/gnome-logs/106
/dev/loop11     squashfs 2.5M   2.5M   0 100% /snap/gnome-calculator/884
/dev/loop12     squashfs 2.3M   2.3M   0 100% /snap/gnome-system-monitor/157
/dev/loop13     squashfs 2.5M   2.5M   0 100% /snap/gnome-system-monitor/163
/dev/loop14     squashfs 66M    66M   0 100% /snap/gtk-common-themes/1515
/dev/loop15     squashfs 33M    33M   0 100% /snap/snapd/12883
/dev/loop16     squashfs 33M    33M   0 100% /snap/snapd/12159
/dev/loop17     squashfs 65M    65M   0 100% /snap/gtk-common-themes/1514
tmpfs           tmpfs     390M   20K 390M   1% /run/user/1000
```

## 7) 기타 OS에서 관리 중인 컴퓨터 시스템 정보

- 네트워크 정보

```
san@linux:~$ ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.119.134  netmask 255.255.255.0  broadcast 192.168.119.255
    inet6 fe80::9c:91fa:78b0:6ba7  prefixlen 64  scopeid 0x20<link>
    ether 00:0c:29:70:72:14  txqueuelen 1000  (Ethernet)
    RX packets 996403  bytes 1503081363 (1.5 GB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 475244  bytes 28863910 (28.8 MB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
```

- 시스템의 네트워크 인터페이스 상태를 확인하기 위해선 ifconfig 명령어를 사용한다. 현재 ip(192.168.119.134), 송수신한 패킷 정보(RX, TX packets) 등을 확인할 수 있다.



- vmstat 명령어를 통해 시스템의 리소스 상황(CPU, I/O, Memory) 모니터링할 수 있다. 표기된 내용은 다음과 같은 정보를 의미한다.

r : 실행중인 프로세스 수                      b : I/O 처리 중 블록 처리된 프로세스  
swpd : 사용중인 가상 메모리 양              free : 사용가능한 가상 메모리 양  
buff : 버퍼로 사용 중인 메모리 양          cache : 캐시로 사용 중인 메모리 양  
si : 디스크에서 메모리로 스왑된 메모리 용량  
so : 디스크로 스왑되어 나간 메모리 용량  
bi : 초당 블록 디바이스로 보내는 블록 수  
bo : 초당 블록 디바이스로부터 받은 블록 수  
in : 초당 인터럽트                              cs : 초당 context switch  
us : 사용자의 CPU 사용 시간 비율(%)      sy : 시스템의 CPU 사용 시간 비율(%)  
id : CPU idle time(%)                          wa : 입출력 대기

```
san@linux:~$ vmstat
procs -----memory----- --swap-- -----io----- -system-- -----cpu-----
 r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa  st
 3   0    268 127412 72816 2381420   0   0   78  196 225 336  2  1 97  0  0
san@linux:~$
```

### 3. 과제를 통해 배운 점

리눅스 커널 컴파일 및 설치 과제를 통해, 리눅스 커널 소스를 어디에서 다운받는 지, 빌드하려는 커널의 세부 설정은 어떻게 하는 지, 그리고 빌드는 어떻게 해야 하는 지에 대해 알게 되었다. 또한 수 시간이 걸리는 리눅스 커널 빌드 과정에서 make 유틸리티의 필요성을 다시 한 번 절감했다.

운영체제는 다양한 시스템 리소스를 효율적으로 관리하기 위해 노력하며, 또한 관리 중인 리소스에 대한 정보를 제공하기 위해 다양한 명령어들을 지원한다. CPU 관련 정보에는 lscpu, iostat 등의 명령어가 사용되며, 디스크에 대한 정보는 df, iostat 등의 명령어를 통해 얻을 수 있다. 이외에도 운영체제는 프로세스 관리, 메모리 관리, 파일 시스템 등의 정보를 다양한 방법으로 사용자에게 제공한다.