

15-2. Process 관리 명령

Process 목록 보기

- **ps 명령의 옵션**
 - **유닉스(SVR4) 옵션:** 묶어서 사용할 수 있고, 불임표로 시작한다. (예: -ef)
 - **BSD 옵션:** 묶어서 사용할 수 있고, 불임표로 시작하지 않는다. (예: aux)
 - **GNU 옵션:** 불임표 2개로 시작한다. (예: --pid)
- **ps**

기능	현재 실행 중인 프로세스의 정보를 출력한다.		
형식	ps 옵션		
옵션	<p><유닉스 옵션></p> <p>-e: 시스템에서 실행 중인 모든 프로세스의 정보를 출력</p> <p>-f: 프로세스의 자세한 정보를 출력</p> <p>-u UID: 특정 사용자에게 대한 모든 프로세스의 정보를 출력</p> <p>-p PID: pid로 지정한 특정 프로세스의 정보를 출력</p> <p><BSD 옵션></p> <p>a: 터미널에서 실행한 프로세스의 정보를 출력</p> <p>u: 프로세스의 소유자 이름, CPU 사용량, 메모리 사용량 등 상세 정보를 출력</p> <p>x: 시스템에서 실행 중인 모든 프로세스의 정보를 출력</p> <p><GNU 옵션></p> <p>--pid PID목록: 목록으로 지정한 특정 PID의 정보를 출력</p>		
사용 예	ps	ps -ef	ps aux

현재 단말기의 프로세스 목록 출력하기: ps

- ps 명령을 옵션 없이 사용하면 현재 셸이나 터미널에서 실행한 사용자 프로세스의 정보를 출력한다.
 - PID: 프로세스 번호
 - TTY: 현재 터미널 번호
 - TIME: 해당 프로세스가 사용한 CPU 시간의 양
 - CMD: 프로세스가 실행 중인 명령

전체 프로세스의 목록 출력하기(유닉스 옵션): -ef 옵션

UID	프로세스를 실행한 사용자 ID	STIME	프로세스의 시작 날짜나 시간
PID	프로세스의 번호	TTY	프로세스가 실행된 터미널의 종류와 번호

PPID	부모 프로세스 번호	TIME	프로세스 실행 시간
C	CPU 사용량(% 값)	CMD	실행되고 있는 프로그램의 이름(명령)

```

lima@itserver: ~
lima@itserver:~$ ps
  PID TTY          TIME CMD
 1946 pts/4    00:00:00 bash
 2049 pts/4    00:00:00 ps
lima@itserver:~$ ps -ef | more
UID          PID    PPID  C   STIME TTY          TIME CMD
root         1        0   0   16:43 ?        00:00:00 /sbin/init splash
root         2        0   0   16:43 ?        00:00:00 [kthreadd]
root         4        2   0   16:43 ?        00:00:00 [kworker/0:0H]
root         6        2   0   16:43 ?        00:00:00 [mm_percpu_wq]
root         7        2   0   16:43 ?        00:00:00 [ksoftirqd/0]
root         8        2   0   16:43 ?        00:00:00 [rcu_sched]
root         9        2   0   16:43 ?        00:00:00 [rcu_bh]
root        10        2   0   16:43 ?        00:00:00 [migration/0]
root        11        2   0   16:43 ?        00:00:00 [watchdog/0]
root        12        2   0   16:43 ?        00:00:00 [cpuhp/0]
root        13        2   0   16:43 ?        00:00:00 [kdevtmpfs]
root        14        2   0   16:43 ?        00:00:00 [netns]
root        15        2   0   16:43 ?        00:00:00 [rcu_tasks_kthre]
root        16        2   0   16:43 ?        00:00:00 [kauditd]
root        17        2   0   16:43 ?        00:00:00 [khungtaskd]
root        18        2   0   16:43 ?        00:00:00 [oom_reaper]
root        19        2   0   16:43 ?        00:00:00 [writeback]
root        20        2   0   16:43 ?        00:00:00 [kcompactd0]
root        21        2   0   16:43 ?        00:00:00 [ksmd]
root        22        2   0   16:43 ?        00:00:00 [khugepaged]
root        23        2   0   16:43 ?        00:00:00 [crypto]
root        24        2   0   16:43 ?        00:00:00 [kintegrityd]
root        25        2   0   16:43 ?        00:00:00 [kblockd]
root        26        2   0   16:43 ?        00:00:00 [ata_sff]
root        27        2   0   16:43 ?        00:00:00 [md]
root        28        2   0   16:43 ?        00:00:00 [edac-poller]
root        29        2   0   16:43 ?        00:00:00 [devfreq_wq]

```

전체 프로세스의 목록 출력하기(BSD 옵션): aux 옵션

```

lima@itserver: ~
lima@itserver:~$ ps aux | more
USER          PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  0.0  0.2 119648 5788 ?        Ss   16:43   0:00 /sbin/init spla
sh
root           2  0.0  0.0      0     0 ?        S    16:43   0:00 [kthreadd]
root           4  0.0  0.0      0     0 ?        I<   16:43   0:00 [kworker/0:0H]
root           6  0.0  0.0      0     0 ?        I<   16:43   0:00 [mm_percpu_wq]
root           7  0.0  0.0      0     0 ?        S    16:43   0:00 [ksoftirqd/0]
root           8  0.0  0.0      0     0 ?        I    16:43   0:00 [rcu_sched]
root           9  0.0  0.0      0     0 ?        I    16:43   0:00 [rcu_bh]
root          10  0.0  0.0      0     0 ?        S    16:43   0:00 [migration/0]
root          11  0.0  0.0      0     0 ?        S    16:43   0:00 [watchdog/0]
root          12  0.0  0.0      0     0 ?        S    16:43   0:00 [cpuhp/0]
root          13  0.0  0.0      0     0 ?        S    16:43   0:00 [kdevtmpfs]
root          14  0.0  0.0      0     0 ?        I<   16:43   0:00 [netns]
root          15  0.0  0.0      0     0 ?        S    16:43   0:00 [rcu_tasks_kthr
e]
root          16  0.0  0.0      0     0 ?        S    16:43   0:00 [kauditd]
root          17  0.0  0.0      0     0 ?        S    16:43   0:00 [khungtaskd]
root          18  0.0  0.0      0     0 ?        S    16:43   0:00 [oom_reaper]
root          19  0.0  0.0      0     0 ?        I<   16:43   0:00 [writeback]
root          20  0.0  0.0      0     0 ?        S    16:43   0:00 [kcompactd0]
root          21  0.0  0.0      0     0 ?        SN   16:43   0:00 [ksmd]
root          22  0.0  0.0      0     0 ?        SN   16:43   0:00 [khugepaged]
root          23  0.0  0.0      0     0 ?        I<   16:43   0:00 [crypto]
root          24  0.0  0.0      0     0 ?        I<   16:43   0:00 [kintegrityd]
root          25  0.0  0.0      0     0 ?        I<   16:43   0:00 [kblockd]
root          26  0.0  0.0      0     0 ?        I<   16:43   0:00 [ata_sff]
root          27  0.0  0.0      0     0 ?        I<   16:43   0:00 [md]
root          28  0.0  0.0      0     0 ?        I<   16:43   0:00 [edac-poller]
root          29  0.0  0.0      0     0 ?        I<   16:43   0:00 [devfreq_wq]
root          30  0.0  0.0      0     0 ?        I<   16:43   0:00 [watchdogd]

```

USER	사용자 계정 이름	VSZ	사용하고 있는 가상 메모리의 크기(KB)
%CPU	CPU 사용량	RSS	사용하고 있는 물리적 메모리 크기(KB)
%MEM	물리적 메모리 사용량	START	프로세스의 시작 시간

특정 Process 정보 검색하기

ps 명령을 이용해 특정 프로세스 정보 검색하기

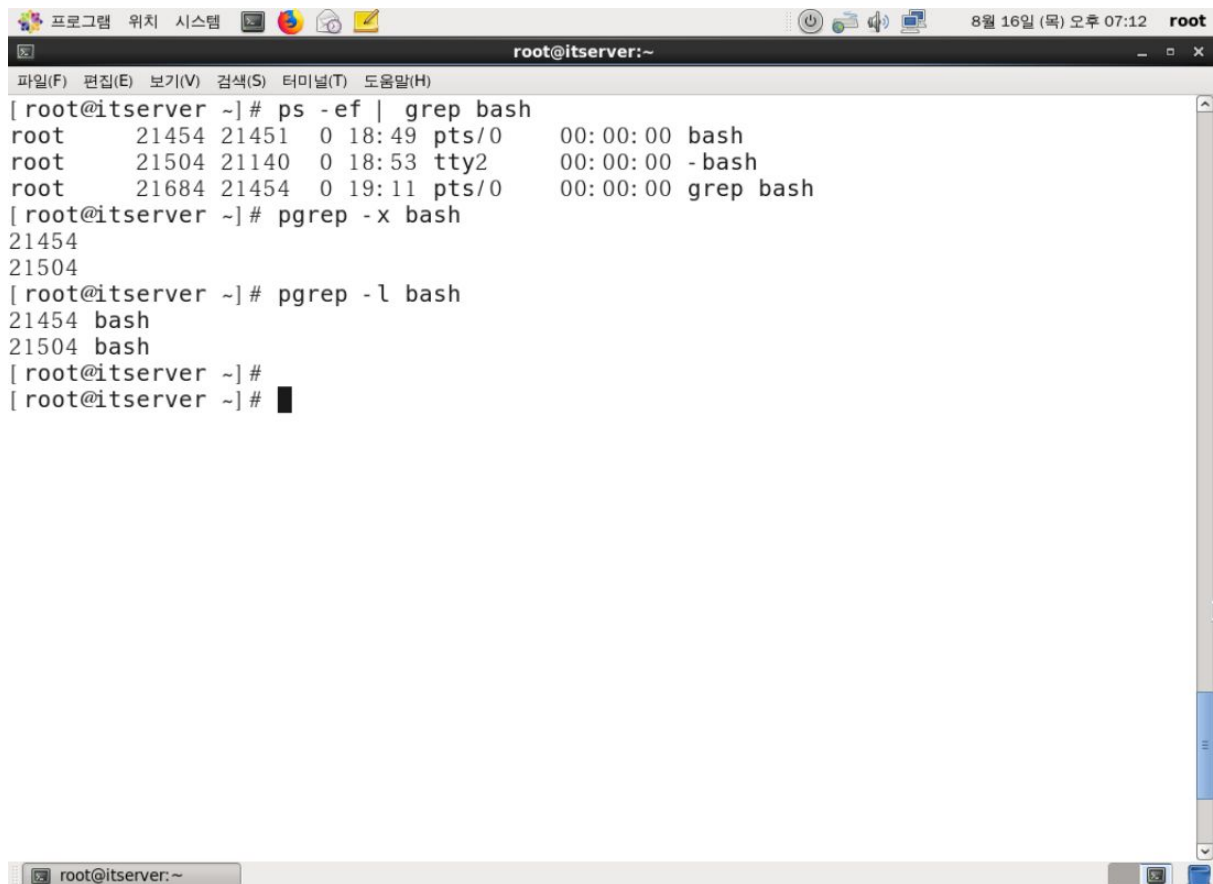
- ps -ef | grep 명령
 - ps 명령과 grep 명령을 |(파이프)로 연결하여 특정 프로세스를 검색할 수 있다.

[user1@itserver ~]\$ ps -ef | grep bash

pgrep 명령을 이용해 특정 프로세스 정보 검색하기

- pgrep
 - = ps + grep

기능	지정한 패턴과 일치하는 프로세스의 정보를 출력한다.
형식	pgrep 옵션 패턴
옵션	-x: 패턴과 정확히 일치하는 프로세스의 정보를 출력 -n: 패턴을 포함하고 있는 가장 최근의 프로세스 정보를 출력 -u 사용자명: 특정 사용자에게 대한 모든 프로세스를 출력 -l: PID와 프로세스의 이름을 출력 -t term: 특정 단말기와 관련된 프로세스의 정보를 출력
사용 예	pgrep bash



```
root@itserver:~  
[root@itserver ~]# ps -ef | grep bash  
root      21454 21451  0 18:49 pts/0    00:00:00 bash  
root      21504 21140  0 18:53 tty2     00:00:00 -bash  
root      21684 21454  0 19:11 pts/0    00:00:00 grep  bash  
[root@itserver ~]# pgrep -x bash  
21454  
21504  
[root@itserver ~]# pgrep -l bash  
21454 bash  
21504 bash  
[root@itserver ~]#  
[root@itserver ~]#
```

- 더 자세한 정보를 검색하려면 pgrep 명령과 ps 명령을 연결해서 사용한다.
 - **ps -fp \$(pgrep -x bash)**
 - pgrep으로 검색하려는 프로세스의 PID를 찾아 ps 명령으로 자세한 정보를 확인하는 것
 - **ps -fp \$(pgrep -u user1 bash)**
 - -u 옵션으로 사용자명을 지정하여 그 사용자의 프로세스 정보만 자세하게 검색하는 것

Process 종료하기

시그널

- 어떤 이벤트가 발생했다는 것을 프로세스에게 알리기 위해 전달하는 메시지이다. = 소프트웨어 인터럽트
- 사용자가 특정 키를 입력했을 때나, kill 같은 시스템 호출, 알람 같은 소프트웨어 조건, 하드웨어의 예외 상황에서 발생
- 시그널 종류
 - 2. SIGINT: Ctrl + c
 - 3. SIGQUIT: Ctrl + w
 - 9. SIGKILL: 프로세스 종료
 - 10. SIGUSR1: 사용자 정의 시그널
 - 15. SIGTERM: kill 시스템 호출

kill 명령을 이용해 프로세스 종료하기

- **kill**

- kill 명령은 인자로 지정한 프로세스에 시그널을 전달한다.
- 시그널은 프로세스에 보내는 간단한 숫자 메시지이다.

기능	지정한 시그널을 프로세스에 보낸다.
형식	kill 시그널 PID ...
시그널	-l(소문자 L): 시그널 목록 출력 -2: 인터럽트 시그널 -9: 프로세스 강제 종료 -15: 프로세스가 관련된 파일을 정리하고 프로세스를 종료. 종료되지 않는 프로세스가 있을 수도(기본값)
사용 예	kill 1001 kill -15 1001 kill -9 1001

pkill 명령을 이용해 프로세스 종료하기

- pkill은 kill 명령과 같이 시그널을 보내는데, PID가 아니라 프로세스의 명령 이름으로 프로세스를 찾아 종료한다.
- kill 명령과의 차이점은 명령 이름으로 찾아 종료하므로 같은 명령이 여러 개 검색될 경우 한번에 모두 종료한다는 점이다.
- 물론 자신이 소유한 프로세스만 종료할 수 있다.
- **pkill -x 명령어**

Process 관리 도구

top 명령

- **top**
 - 현재 실행 중인 프로세스에 대한 정보를 주기적으로 출력하는데, 프로세스의 자세한 요약 정보를 상단에 출력하고 각 프로세스의 정보를 하단에 출력한다.
 - top 명령의 출력 정보

PID	프로세스 ID	SHR	프로세스가 사용하는 공유 메모리 크기
USER	사용자 계정	%CPU	CPU 사용량
PR	우선순위	%MEM	메모리 사용량
NI	Nice 값	TIME+	CPU 누적 이용 시간
VIRT	프로세스가 사용하는 가상 메모리 크기	COMMAND	명령 이름
RES	프로세스가 사용하는 메모리 크기		

```
lima@itserver: ~  
top - 17:11:30 up 28 min, 1 user, load average: 0.01, 0.02, 0.05  
Tasks: 146 total, 1 running, 113 sleeping, 0 stopped, 0 zombie  
%Cpu(s): 0.7 us, 0.0 sy, 0.0 ni, 99.3 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st  
KiB Mem : 2041400 total, 913720 free, 544376 used, 583304 buff/cache  
KiB Swap: 998396 total, 998396 free, 0 used. 1321524 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1626	lima	20	0	1261132	165912	93556	S	0.7	8.1	0:28.11	compiz
1939	lima	20	0	708452	50320	40732	S	0.3	2.5	0:00.52	gnome-termi+
1	root	20	0	119648	5788	4016	S	0.0	0.3	0:00.91	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_wq
7	root	20	0	0	0	0	S	0.0	0.0	0:00.04	ksoftirqd/0
8	root	20	0	0	0	0	I	0.0	0.0	0:00.13	rcu_sched
9	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_bh
10	root	rt	0	0	0	0	S	0.0	0.0	0:00.00	migration/0
11	root	rt	0	0	0	0	S	0.0	0.0	0:00.00	watchdog/0
12	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/0
13	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kdevtmpfs
14	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	netns
15	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu_tasks_k+
16	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kauditd
17	root	20	0	0	0	0	S	0.0	0.0	0:00.00	khungtaskd
18	root	20	0	0	0	0	S	0.0	0.0	0:00.00	oom_reaper
19	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	writeback
20	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kcompactd0
21	root	25	5	0	0	0	S	0.0	0.0	0:00.00	ksmd
22	root	39	19	0	0	0	S	0.0	0.0	0:00.00	khugepaged
23	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	crypto
24	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kintegrityd
25	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kblockd
26	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	ata_sff