인덕대학교

운영체제 보안
- 프로세스 제한1

컴퓨터소프트웨어학과

김병국 교수

학습목표

- 프로세스를 제어하기 위한 몇가지 명령어들을 사용할 수 있다.
- 계정 별 프로세스의 기능을 제한할 수 있다.

목차

- 실습 환경 구축
- 프로세스 제어
- 프로세스 제한

1. 실습 환경 구축 (1/2)

❑ 사용자 계정 추가

- 계정 이름: test
  - 계정의 프로세스를 제어하고 기능을 제한하기 위한 목적
  - 관련 명령어: adduser

1. 실습 환경 구축 (2/2)

❑코드 작성

```c
#include <stdio.h>
#include <unistd.h>

int main(void)
{
    for (int i = 0;; i++)
    {
        printf("\r");
        switch (i % 4)
        {
            case 0:
                printf("-");
                break;
            case 1:
                printf("\\");
                break;
            case 2:
                printf("|");
                break;
            case 3:
                printf("/");
                break;
        }
        fflush(stdout);
        if (i == 0x7FFFFFFF)
            i = 0;
    }
    return 0;
}
```

[파일명: loop.c]

2. 프로세스 제어 (1/3)

❑ 프로세스 모니터링

- 명령어: top
  - 프로세스의 상태정보를 정리된 화면으로 출력
  - 명령어 ps와 동일한 기능을 수행하나, 꾸준한 모니터링 및 사용자 요구에 맞는 형태로 계속 볼 수 있음

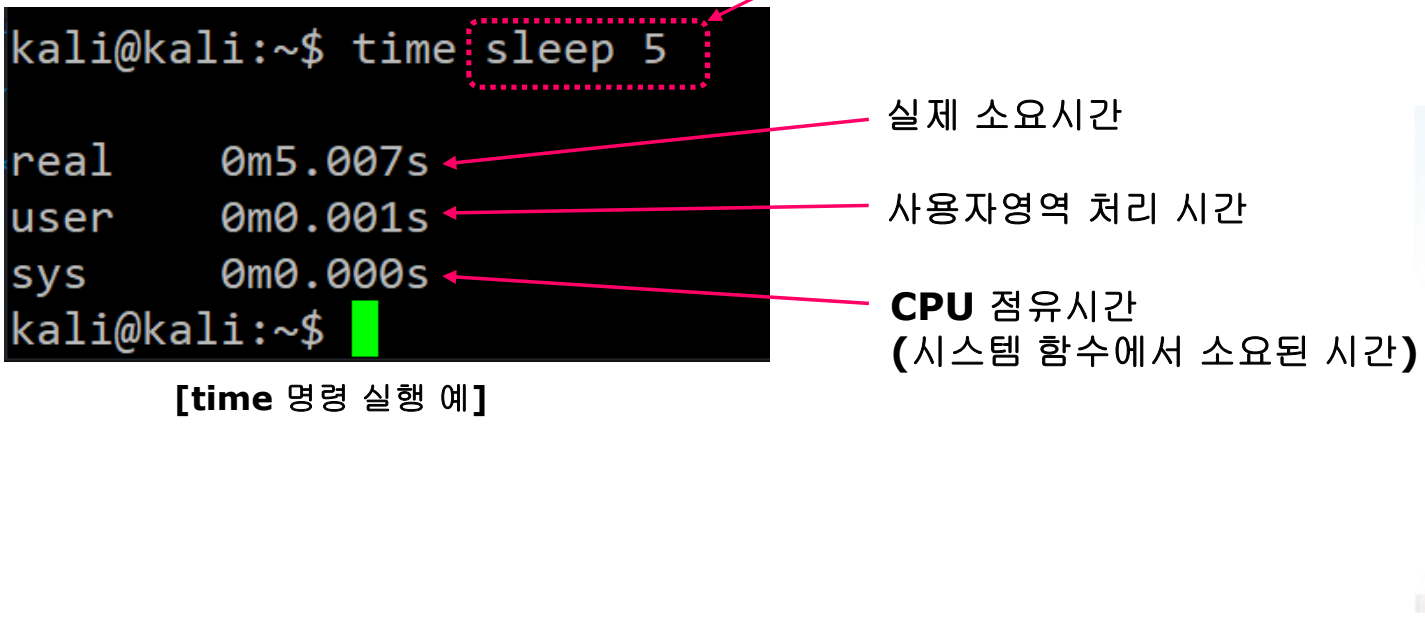
```
top - 12:31:35 up  1:18,  2 users,  load average: 0.00, 0.19, 0.47
Tasks: 177 total,   1 running, 176 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.1 us,  0.4 sy,  0.0 ni, 99.0 id,  0.0 wa,  0.0 hi,  0.4 si,  0.0 st
MiB Mem :   3933.0 total,   2803.8 free,    470.9 used,    658.3 buff/cache
MiB Swap:    975.0 total,    975.0 free,      0.0 used.   3207.4 avail Mem

    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
    966 kali      20   0  218412   2884   2424 S   1.0   0.1   0:29.15 VBoxClient
    752 root      20   0 1196648 107556  44952 S   0.5   2.7   0:19.88 Xorg
      1 root      20   0  164048  10544   7744 S   0.0   0.3   0:02.65 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-events_highpri
      7 root      20   0       0      0      0 I   0.0   0.0   0:02.06 kworker/0:1-events
      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   0:00.00 rcu_tasks_rude_
     11 root      20   0       0      0      0 S   0.0   0.0   0:00.00 rcu_tasks_trace
     12 root      20   0       0      0      0 S   0.0   0.0   0:00.04 ksoftirqd/0
     13 root      20   0       0      0      0 I   0.0   0.0   0:01.20 rcu_sched
     14 root      rt   0       0      0      0 S   0.0   0.0   0:00.10 migration/0
     15 root      20   0       0      0      0 S   0.0   0.0   0:00.00 cpuhp/0
```

[top명령 실행 예]

- 주요 키:
  - <space bar> : 리스트 갱신
  - <home>, <end>, <pg-up>, <pg-dn> : 페이지 전환
  - u <username> : 지정한 사용자의 프로세스들만 표시 (단, 공백→ 모든 사용자)
  - z : 컬러모드 전환
  - V : 트리형태로 구성(v(소문자): 트리의 마지막 노드만 보임)
  - s <#> : 갱신주기 설정
  - k <#> : 프로세스에게 시그널 전송

❑ 프로세스 동작시간 파악

- 명령어: time [명령어]
  - 지정한 명령어에 대한 동작 시간을 측정



[time 명령 실행 예]

2. 프로세스 제어 (3/3)

❑ 프로세스 동작시간 설정

- 명령어: timeout {-옵션} [시간] [명령어]
  - 지정한 시간동안에만 프로세스가 수행
  - 시간 경과 후 강제 종료

```
kali@kali:~$ timeout -k3 2s sleep 50
```

[timeout 명령 실행 예]

- 주요 옵션:
  - -s<#> : 지정한 타임아웃시간에 <#> 시그널을 전송
  - -k<#> : 타임아웃 이후에도 running 상태이면, <#>이후 KILL(9)신호를 보냄
- 시간 표현:
  - #s : 초단위
  - #m : 분단위
  - #h : 시간단위
  - #d : 일단위

3. 프로세스 제한 (1/5)

❑ 프로세스 제한

- 사용자의 명령들은 쉘(shell)을 통해 실행
- 쉘은 제한된 범주에서만 프로세스에게 자원을 할당
- 제한 설정 파일: /etc/security/limits.conf

❑ 제한 확인

- 명령어: ulimit {-옵션}
  - 제한 상태를 확인
  - 주요 옵션: -S (소프트), -H(하드), -a(모두 출력)

```
kali@kali:~$ ulimit -SHa
real-time non-blocking time  (microseconds, -R) unlimited
core file size              (blocks, -c) 0
data seg size               (kbytes, -d) unlimited
scheduling priority                 (-e) 0
file size                   (blocks, -f) unlimited
pending signals                     (-i) 15459
max locked memory           (kbytes, -l) 503420
max memory size             (kbytes, -m) unlimited
open files                          (-n) 1024
pipe size                (512 bytes, -p) 8
POSIX message queues         (bytes, -q) 819200
real-time priority                  (-r) 0
stack size                  (kbytes, -s) 8192
cpu time                   (seconds, -t) unlimited
max user processes                  (-u) 15459
virtual memory              (kbytes, -v) unlimited
file locks                          (-x) unlimited
kali@kali:~$
```

[ulimits 명령 실행 예]

3. 프로세스 제한 (2/5)

❑ 설정 파일내 구성

- 형태: <domain> <type> <item> <value>
- <domain> : 사용자 계정(사용자 명) 및 그룹(@그룹이름)을 지정
- <type> : soft와 hard로 구분
- <item> : 제한 기능 선택
- <value> : 제한 기능별 값

```
22 #<item> can be one of the following:
23 #        - core - limits the core file size (KB)
24 #        - data - max data size (KB)
25 #        - fsize - maximum filesize (KB)
26 #        - memlock - max locked-in-memory address space (KB)
27 #        - nofile - max number of open file descriptors
28 #        - rss - max resident set size (KB)
29 #        - stack - max stack size (KB)
30 #        - cpu - max CPU time (MIN)
31 #        - nproc - max number of processes
32 #        - as - address space limit (KB)
33 #        - maxlogins - max number of logins for this user
```

[item 예]

3. 프로세스 제한 (3/5)

❑최대 로그인 개수 제한

- 계정별 로그인 쉘의 개수를 제한

```
# /etc/security/limits.conf
#
#Each line describes a limit for a user in the form:
#
#<domain>        <type>  <item>  <value>
#
#Where:
#<domain> can be:
#        - a user name
#        - a group name, with @group syntax
#
#

test            soft    maxlogins       2

# End of file
```

[최대 로그인 개수 지정]

```
[test@kali ~]$w
 14:38:05 up  3:24,  4 users,  load average: 0.08, 0.08, 0.02
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
kali     tty7     :0               11:13    3:24m 33.72s  0.19s xfce4-session
kali     pts/3    192.168.0.2      14:17   14:21   0.02s  0.02s -bash
test     pts/5    -                14:37   29.00s  0.04s  0.01s -bash
test     pts/6    -                14:38    5.00s  0.07s  0.00s w
[test@kali ~]$
```

```
kali@kali:~$ sudo login test
[sudo] password for kali:
Password:
Linux kali 5.10.0-kali3-amd64 #1 SMP Debian 5.10.13-1kali1 (2021-02-08) x86_64

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
There were too many logins for 'test'.
Last login: Thu May 13 14:38:04 KST 2021 on pts/6

Permission denied
kali@kali:~$
```

[로그인 시도]

3. 프로세스 제한 (4/5)

❑ CPU 점유시간 제한

- 특정 계정에 대한 CPU의 점유 시간을 제한

※ sudo에서는 적용되지 않는 이유는?

```
1 # /etc/security/limits.conf
2 #
3 #Each line describes a limit for a user in the form:
4 #
5 #<domain>        <type>  <item>  <value>
6 #
7 #Where:
8 #<domain> can be:
9 #        - a user name
56
57 test            soft    cpu             1
58
59 # End of file
```

[CPU 점유시간 할당]

3. 프로세스 제한 (5/5)

❑ 프로세스 개수 제한

- 특정 계정에 대한 프로세스의 개수를 제한

```
53 #ftp            -       chroot          /ftp
54 #@student       -       maxlogins       4
55 #
56 test            soft    nproc           10
57
58 # End of file
```

[프로세스 개수 제한 설정 예]

```c
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>

int main(int argc, char *argv[])
{
    int nPid = 0;
    int nCount = 100;

    if (argc == 2)
        nCount = atoi(argv[1]);

    for (int i = 0; i < nCount; i++)  {
        nPid = fork();
        if (nPid == 0) {
            printf("[%02d] I'm a child process with %d.\n", i, getpid());
            sleep(100);
            break;
        }
    }

    return 0;
}
```

[파일명: **forkcount.c**]

수고하셨습니다.