

UNIX 및 실습

# 8장. 프로세스와 사용자 명령 익히기

# 학습목표

- ▶ 유닉스에서 프로세스가 무엇인지 그 개념을 이해한다.
- ▶ 프로세스와 관련된 유닉스 명령의 사용 방법을 익힌다.
- ▶ 포그라운드 처리와 백그라운드 처리의 차이를 이해한다.
- ▶ 사용자 정보를 보는 명령의 사용 방법을 익힌다.

# 03. 포그라운드와 백그라운드 프로세스

- ▶ UNIX는 다중 작업을 지원하는 운영체제이다.
  - ▶ 동시에 여러 개의 작업을 수행할 수 있다.
- ▶ 포그라운드 처리(전위 처리)
  - ▶ 사용자가 명령을 입력한 후 결과가 출력될 때까지 기다려야 하는 경우
  - ▶ 보통의 명령처리 방법

```
ssh lily.mmu.ac.kr
$ find / -name passwd
```

- ▶ 백그라운드 처리(후위 처리)
  - ▶ 명령의 처리결과 출력과 관계없이 곧바로 프롬프트가 출력되어 다른 작업을 계속 할 수 있는 경우
  - ▶ 명령 실행시 마지막에 &를 붙임

```
ssh lily.mmu.ac.kr
$ find / -name passwd &
[1] 5699
$
```

# 포그라운드와 백그라운드 작업제어

- ▶ 작업과 프로세스(job & process)
  - ▶ job은 셸이 관리할 수 있는 프로세스이다.
  - ▶ 셸은 job을 시작시키고 제어한다.
  - ▶ job은 프로세스이므로 각 job은 PID를 가지고 있고
  - ▶ 또한 셸이 할당한 일련번호인 job ID도 가지고 있다.
  - ▶ 셸은 동시에 여러 개의 job이 동작하도록 할 수 있다.
  - ▶ 포그라운드 작업 -> 포그라운드 프로세스
  - ▶ 백그라운드 작업 -> 백그라운드 프로세스
- ▶ 작업제어
  - ▶ 포그라운드 작업 -> 백그라운드 작업으로 전환
  - ▶ 백그라운드 작업 -> 포그라운드 작업으로 전환
  - ▶ 작업목록 보기
  - ▶ 작업 정지/종료/재동작

# 작업제어 명령 – jobs [1/2]

## jobs [%작업번호]

- ▶ 백그라운드 작업을 모두 출력
- ▶ 특정 작업번호를 지정할 경우 해당 작업의 정보만 출력
- ▶ 작업번호
  - ▶ %번호 : 해당 번호의 작업 정보를 출력
  - ▶ %+ 또는 %% : 작업순서가 +인 작업 정보를 출력
  - ▶ %- : 작업 순서가 -인 작업 정보를 출력

### ▶ 사용법

```
ssh lily.mmu.ac.kr
$ jobs
[1] +  실행중  sleep 100 &
[2] -  실행중  sleep 200 &
$ jobs %1
[1] +  실행중  sleep 100 &
$
```

# 작업제어 명령 – jobs [2/2]

## ▶ jobs 명령 출력 항목

항목	출력예제	의미
작업번호	[1]	작업번호로 백그라운드로 실행시킬 때마다 순차적으로 증가([1],[2],[3]...)
작업순서	+, -	작업순서를 표시 • + : 가장 최근에 접근한 작업 • - : + 작업보다 바로 전에 접근한 작업 • 공백 : 그 외의 작업
상태	실행중	작업의 상태를 표시 • 실행중(Running) : 현재 실행중 • 완료됨(Done) : 작업이 정상적으로 종료 • 종료됨(Terminated) : 작업이 비정상적으로 종료 • 정지(Stopped) : 작업이 잠시 중단됨.
명령	sleep 100&	실행중인 명령

# 작업제어 명령 - 작업전환 [1/2]

## ▶ 작업전환 및 종료 명령

명령	기능
bg [%작업번호]	현재 작업이나 특정 작업을 백그라운드로 전환시켜 실행
fg [%작업번호]	현재 작업이나 특정 작업을 포그라운드로 전환시켜 실행
ctrl+z	포그라운드 작업을 중지시키고, 백그라운드의 중지된 목록으로 보냄
stop %작업번호	백그라운드에서 수행중인 특정 작업을 중지
kill %n	특정 작업을 종료

# 작업제어 명령 - 작업전환 [2/2]

## ▶ 사용예

1

```
ssh lily.mmu.ac.kr
$ sleep 100
^Z
[1] + 정지 (SIGTSTP) sleep 100
$ bg %1
[1] sleep 100&
$
```

• 포그라운드 작업을 ctrl-z로 중지시키고 백그라운드로 전환시킨다.

2

```
ssh lily.mmu.ac.kr
$ jobs
[1] + 실행중 sleep 100&
$ fg
sleep 100
-
```

• jobs 명령으로 현재 작업을 확인한다.  
• 1번 job : 실행중인 상태  
• 포그라운드로 다시 전환하면 프로세스가 끝날 때까지 기다려야 한다.



# [실습하기] 작업제어 명령

## ▶ 실습하기

```
1) vi /etc/hosts
   • ctrl+z
2) sleep 300&
3) jobs
4) kill %2
5) jobs
6) fg
7) vi 저장후 종료
```

```
8) sleep 150 &
9) sleep 200 &
10) jobs
11) fg %1
12) ^C
13) jobs
14) kill %2
15) jobs
```

# 작업제어 명령 - nohup

## nohup 백그라운드명령

- ▶ 백그라운드 작업을 실행시킨 단말기가 종료되거나 사용자가 로그아웃하면 실행 중이던 백그라운드 작업은 함께 종료
  - ▶ 로그아웃한 다음에도 백그라운드 작업은 작업이 완료될 때까지 실행하도록 해야 할 때 nohup 명령 사용
- ▶ 명령의 실행결과와 오류메시지는 현재 디렉토리에 nohup.out파일로 자동적으로 저장
- ▶ 사용예

```
ssh lily.mmu.ac.kr
$ nohup find / -name passwd &
[1] 16454
$ exit
```

nohup.out파일에  
결과 저장.  
다시 로그인해서  
nohup.out파일 확인

# 04. 사용자 정보보기

- ▶ 로그인한 사용자 정보 보기
  - ▶ users
  - ▶ who
  - ▶ w
- ▶ 사용자 자신의 정보 보기
  - ▶ who am I
  - ▶ whoami
  - ▶ id

# 사용자명 출력하기 - users

users

- ▶ 현재 시스템에 로그인하고 있는 사용자명을 출력
- ▶ 사용예

```
ssh lily.mmu.ac.kr
```

```
$ users  
root user1  
$
```

# 사용자정보 출력하기 - who

## who [옵션]

- ▶ 시스템을 사용하고 있는 사용자의 정보를 출력
- ▶ 옵션
  - ▶ -q : 사용자명만 출력한다.
  - ▶ -H : 출력항목의 제목도 함께 출력한다.
  - ▶ -b : 마지막으로 재부팅한 날짜와 시간을 출력한다.

### ▶ 사용예

```
ssh lily.mmu.ac.kr
```

```
[kgu@lily ~]$ who
(unknown) tty1          2013-04-11 17:58 (:0)
kgu          pts/0        2013-04-29 09:01 (220.68.173.204)
race0       pts/2          2013-04-28 23:23 (112.164.86.181)
```

```
ssh lily.mmu.ac.kr
```

```
[kgu@lily ~]$ who -q
(unknown) kgu race0 kgu race0 ww7737 kgu
# users=7
```

```
ssh lily.mmu.ac.kr
```

```
[kgu@lily ~]$ who -b
                system boot 2013-04-11 17:57
```

# 사용자 작업정보 출력하기- w

## w [사용자명]

- ▶ 로그인한 사용자정보와 현재 하고 있는 작업정보를 출력
- ▶ 사용예

```
ssh lily.mmu.ac.kr
$ w
 09:13:36 up 17 days, 15:16,  7 users,  load average: 0.15, 0.13, 0.14
USER      TTY      FROM          LOGIN@      IDLE        JCPU       PCPU WHAT
kgu       pts/0    220.68.173.204 09:01       0.00s      0.15s      0.00s w
race0     pts/2    112.164.86.181 23:23       9:31m      0.17s      0.17s -bash
```

# 사용자 자신의 로그인 정보 - who am i

who am i

- ▶ who 명령의 결과 중 자신에 대한 정보만 출력
- ▶ 사용예

```
ssh lily.mmu.ac.kr
```

```
[kgu@lily ~]$ who am i
```

```
kgu pts/0 2013-04-29 09:01 (220.68.173.204)
```

# 자신의 로그인 사용자명 출력하기 – whoami

whoami

- ▶ 사용자의 로그인ID를 출력
  - ▶ BSD계열 명령
  - ▶ /usr/ucb 디렉토리에 있음
- ▶ 사용예

```
ssh lily.mmu.ac.kr
```

```
[kgu@lily ~]$ whoami  
kgu
```



# 현재 사용자명과 그룹정보 출력하기 - id

## id [옵션]

- ▶ 사용자의 로그인ID와 그룹 정보를 출력
- ▶ 옵션
  - ▶ -a : 기본 그룹 외에 2차 그룹 정보도 출력
- ▶ 사용예

```
ssh lily.mmu.ac.kr
```

```
[kgu@lily ~]$ id race0
```

```
uid=10044(race0) gid=10000(students) groups=10000(students)
```

```
[kgu@lily ~]$ id -a
```

```
uid=1000(kgu) gid=1000(prof) groups=1000(prof),3(sys),4(adm),10043(svn)
```

```
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

# [실습하기] 사용자정보 보기

## ▶ 실습하기

- 1) who
- 2) who -m
- 3) who -q
- 4) who -H
- 5) w
- 6) who am i
- 7) whoami
- 8) id
- 9) id -a

# 사용자 전환하기 - su

**su** [ - ] [ 사용자명 ]

- ▶ 사용자 계정을 변경한다
  - ▶ 아무 옵션 없이 사용하면 : root 계정으로 변경
  - ▶ 사용자 명 지정 : 지정한 사용자로 변경
  - ▶ - : root 혹은 지정한 사용자의 초기화 파일이 실행되면서 작업 디렉토리로 사용자의 홈 디렉토리로 변경됨

```
ssh lily.mmu.ac.kr
```

```
$ su user2
Password:
$ whoami
user2
$ pwd
/home/user1
$
```

```
ssh lily.mmu.ac.kr
```

```
$ su - user2
Password:
$ whoami
user2
$ pwd
/home/user2
$
```

```
ssh lily.mmu.ac.kr
```

```
$ su
Password:
#whoami
root
#
```