

# 목차

00. 개요

01. 프로세스의 개념

02. 프로세스 관리 명령

03. 포그라운드, 백그라운드 프로세스와 작업 제어 (저번 시간 이어  
서)

04. 작업 예약

## 04 작업 예약

### ■ 특정한 시간에 작업을 수행하도록 예약할 수 있는 두 가지 방법

- 정해진 시간에 한 번만 수행
- 정해진 시간에 반복 수행

### ■ 정해진 시간에 한 번 실행

#### at

**기능** 예약한 명령을 정해진 시간에 실행한다.

**형식** at [옵션] 시간

**옵션**

- l : 현재 실행 대기 중인 명령의 전체 목록을 출력한다(atq 명령과 동일).
- r 작업 번호 : 현재 실행 대기 중인 명령에서 해당 작업 번호를 삭제한다(atrm과 동일).
- m : 출력 결과가 없더라도 작업이 완료되면 사용자에게 메일로 알려준다.
- f 파일 : 표준 입력 대신 실행할 명령을 파일로 지정한다.

**사용 예** at -m 0730 tomorrow                      at 10:00 pm                      at 8:15 am May 30

## 04 작업 예약

### ■ at 명령 설치 : `sudo apt-get install at`

```
user1@myubuntu:~/linux_ex/ch6$ at 09:00 am
warning: commands will be executed using /bin/sh
at>
```

### ■ at 명령 설정하기

- at 명령을 사용하여 정해진 시간에 명령을 실행하도록 예약하려면 at 명령 뒤에 시간을 명시

- 시간을 지정하는 형식

- at 4pm + 3 days : 지금부터 3일 후 오후 4시에 작업을 수행한다.
- at 10am Jul 31 : 7월 31일 오전 10시에 작업을 수행한다.
- at 1am tomorrow : 내일 오전 1시에 작업을 수행한다.
- at 10:00am today : 오늘 오전 10시에 작업을 수행한다.

- at로 실행할 명령은 기본적으로 표준 입력으로 지정: 명령의 입력을 마치려면 `ctrl+d` 입력

```
user1@myubuntu:~/linux_ex/ch6$ at 09:00 am
at> ls -l ~user1
at> <EOT>
job 1 at Tue Feb 25 09:00:00 2014
user1@myubuntu:~/linux_ex/ch6$
```

시간을 지정 한다.  
실행할 명령을 지정 한다.  
Ctrl+d를 입력하여 종료한다.  
작업 예약이 완료되었다.

## 04 작업 예약

### ■ at 작업 파일 확인하기

- at로 생성된 작업 파일은 /var/spool/at 디렉터리에 저장

```
user1@myubuntu:~/linux_ex/ch6$ at 11:10 am
warning: commands will be executed using /bin/sh
at> ls
at> <EOT>
job 6 at Tue Feb 25 11:10:00 2014
user1@myubuntu:~/linux_ex/ch6$
```



```
user1@myubuntu:~/linux_ex/ch6$ sudo ls -l /var/spool/cron/atjobs
합계 4
-rwx----- 1 user1 daemon 2417  2월 24 14:25 a0000601625542
user1@myubuntu:~/linux_ex/ch6$
```

- root 사용자만 /var/spool/at 디렉터리 내용 확인 가능

## 04 작업 예약

### ■ at 작업 목록 확인하기 : -l 옵션, atq

- at 명령으로 설정된 작업의 목록은 -l 옵션으로 확인

```
user1@myubuntu:~/linux_ex/ch6$ at -l
6          Tue Feb 25 11:10:00 2014 a user1
user1@myubuntu:~/linux_ex/ch6$
```

- atq 명령으로도 확인 가능

```
user1@myubuntu:~/linux_ex/ch6$ atq
6          Tue Feb 25 11:10:00 2014 a user1
user1@myubuntu:~/linux_ex/ch6$
```

## 04 작업 예약

### ■ at 작업 삭제하기 : -d 옵션, atrm

- at 명령으로 설정한 작업이 실행되기 전에 삭제하려면 -d 옵션을 사용하고 삭제할 작업 번호를 지정
- 작업예약

```
user1@myubuntu:~/linux_ex/ch6$ at 1am tomorrow
warning: commands will be executed using /bin/sh
at> ls
at> <EOT>
job 7 at Tue Feb 25 01:00:00 2014
user1@myubuntu:~/linux_ex/ch6$
```

- 설정된 작업 확인

```
user1@myubuntu:~/linux_ex/ch6$ atq
7          Tue Feb 25 01:00:00 2014 a user1
6          Tue Feb 25 11:10:00 2014 a user1
user1@myubuntu:~/linux_ex/ch6$
```

- 작업 삭제

```
user1@myubuntu:~/linux_ex/ch6$ at -d 7
user1@myubuntu:~/linux_ex/ch6$ atrm 6
user1@myubuntu:~/linux_ex/ch6$ atq
user1@myubuntu:~/linux_ex/ch6$
```

## 04 작업 예약

### ■ at 명령 사용 제한하기

- 관련된 파일: /etc/at.allow와 /etc/at.deny
  - /etc/at.allow 파일과 /etc/at.deny 파일에는 한 줄에 사용자 이름을 하나씩만 기록
  - /etc/at.allow 파일이 있으면 이 파일에 있는 사용자만 at 명령을 사용할 수 있다. 이 경우에 /etc/at.deny 파일은 무시된다.
  - /etc/at.allow 파일이 없으면 /etc/at.deny 파일에 지정된 사용자를 제외한 모든 사용자가 at 명령을 사용할 수 있다.
  - 만약 두 파일이 모두 없다면 root만 at 명령을 사용할 수 있다.
  - 한 사용자가 두 파일 모두에 속해 있다면 그 사용자는 at 명령을 사용할 수 있다. /etc/at.allow 파일이 적용되기 때문이다.
  - /etc/at.deny를 빈 파일로 두면 모든 사용자가 at 명령을 사용할 수 있는데, 이것이 초기 설정이다.
- 
- at.deny 파일에 user1 사용자가 기록되어 있다면 at 명령을 실행했을 때 사용 권한이 없다는 메시지가 출력

```
user1@myubuntu:~/linux_ex/ch6$ at
You do not have permission to use at.
user1@myubuntu:~/linux_ex/ch6$
```

## 04 작업 예약

### ■ 정해진 시간에 반복 실행

#### crontab

**기능** 사용자의 crontab 파일을 관리한다.

**형식** crontab [-u 사용자 ID] [옵션] [파일 이름]

**옵션** -e : 사용자의 crontab 파일을 편집한다.

-l : crontab 파일의 목록을 출력한다.

-r : crontab 파일을 삭제한다.

**사용 예** crontab -l                  crontab -u user1 -e                  crontab -r

분(0~59)	시(0~23)	일(1~31)	월(1~12)	요일(0~6)	작업 내용
---------	---------	---------	---------	---------	-------

30    23    1    \*    \*    /usr/bin/ls -l ~user1 > ~user1/cron.out

↓    ↓    ↓    ↓    ↓    ↓

**1** 분   **2** 시   ↓   ↓   ↓   **6** 작업 내용

**3** 일   **4** 월   **5** 요일

         crontab 파일의 형식



## 04 작업 예약

### ■ crontab 파일 생성하고 편집하기 : crontab -e

■ crontab -e 명령으로 편집한 파일을 저장하면 자동적으로 /var/spool/cron/crontabs 디렉터리에 사용자 이름으로 생성

### ■ crontab 파일 내용 확인하기 : crontab -l

```
user1@myubuntu:~/linux_ex/ch6$ sudo ls -l /var/spool/cron/crontabs
[sudo] password for user1:
합계 4
-rw----- 1 user1 crontab 1137  2월 24 14:55 user1
user1@myubuntu:~/linux_ex/ch6$
```

### ■ crontab 파일 삭제하기 : crontab -r

```
user1@myubuntu:~/linux_ex/ch6$ crontab -l
30 23 1 * * /bin/ls -l ~user1 > ~user1/cron.out
user1@myubuntu:~/linux_ex/ch6$
```

```
user1@myubuntu:~/linux_ex/ch6$ crontab -r
user1@myubuntu:~/linux_ex/ch6$ crontab -l
no crontab for user1
user1@myubuntu:~/linux_ex/ch6$
```

## 04 작업 예약

### ■ crontab 명령 사용 제한하기

- /etc/cron.allow, /etc/cron.deny 파일
- cron.deny 파일은 기본적으로 있지만 cron.allow 파일은 관리자가 만들어야 함
- 두 파일이 적용되는 기준
  - /etc/cron.allow 파일이 있으면 이 파일 안에 있는 사용자만 crontab 명령을 사용할 수 있다.
  - /etc/cron.allow 파일이 없고 /etc/cron.deny 파일이 있으면 이 파일에 사용자 계정이 없어야 crontab 명령을 사용할 수 있다.
  - /etc/cron.allow 파일과 /etc/cron.deny 파일이 모두 없다면 시스템 관리자만 crontab 명령을 사용할 수 있다.
- 두 파일이 모두 없는데 일반 사용자가 crontab 명령을 사용하려고 하면 다음과 같은 메시지가 출력

```
user1@myubuntu:~/linux_ex/ch6$ crontab -e
You (user1) are not allowed to use this program (crontab)
See crontab(1) for more information
user1@myubuntu:~/linux_ex/ch6$
```