





CRON

Unix 기반 OS에서 미리 작업을 백그라운드에서 수행하도록 예약할 수 있는 **Daemon**
(시간을 뜻하는 그리스어 chronos에서 기원)

작업을 고정된 시간, 날짜, 간격에 주기적으로 실행할 수 있도록 스케줄링하기 위해 사용
Ex) 주기적으로 메일 서버에서 메일을 받아오는 작업, 주기적으로 자료 백업

Daemon

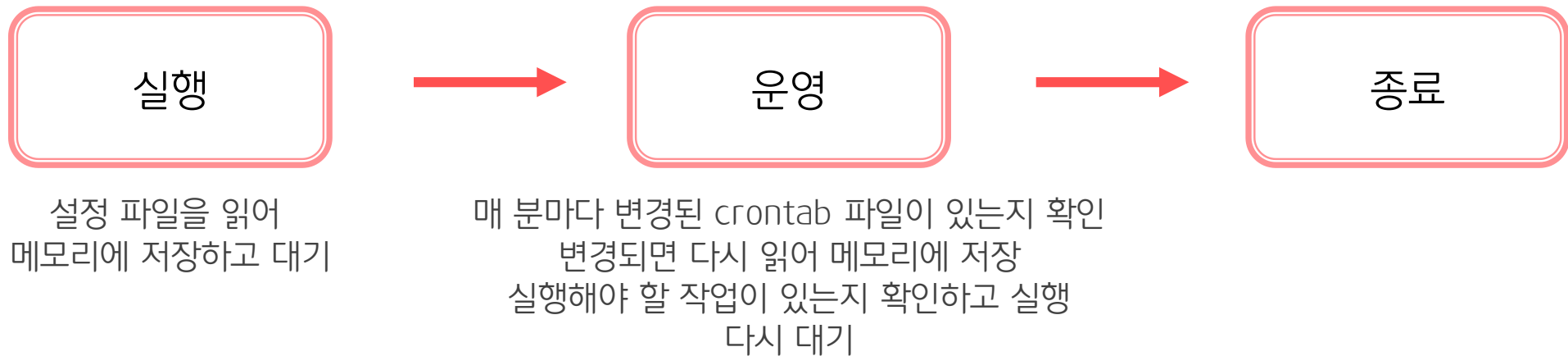
사용자가 직접적으로 제어하지 않고, 백그라운드에서 돌면서 여러 작업을 하는 프로그램
'd'를 이름 끝에 달고 있음
데몬은 시스템의 첫 프로세스인 init의 바로 하위 프로세스
/etc/init.d에 위치하며 service, systemctl으로 제어



CRON

Unix 기반 OS에서 미리 작업을 백그라운드에서 수행하도록 예약할 수 있는 **Daemon**
(시간을 뜻하는 그리스어 chronos에서 기원)

작업을 고정된 시간, 날짜, 간격에 주기적으로 실행할 수 있도록 스케줄링하기 위해 사용
Ex) 주기적으로 메일 서버에서 메일을 받아오는 작업, 주기적으로 자료 백업





CRON 시작하기

Cron 설치	<code>apt install cron</code>
Cron 시작	<code>service cron start</code> (or <code>/etc/init.d/cron start</code>)
Cron 종료	<code>service cron stop</code> (or <code>/etc/init.d/cron stop</code>)
Cron 재시작	<code>service cron restart</code> (or <code>/etc/init.d/cron restart</code>)
Cron 작동 확인	<code>Service cron status</code>
Cron 실행 확인	<code>ps aux grep cron</code>



CRONTAB



Crontab(Cron Table)은 cron을 통해 실행할 작업들을 관리
(cron은 실행, crontab은 설정이라고 생각하면 쉬울 것 같습니다.)

Cron은 하나지만 user마다 crontab 존재



CRONTAB 저장 경로

`/var/spool/cron/crontabs/[사용자 이름]` 각 사용자를 위한 crontab

`/etc/crontab` root 관리자를 위한 system crontab

`/etc/cron.d` 아래의 네 가지 주기가 아닌 경우 -> 일반적으로 주기적인 패키지 설치를 위해 사용

`/etc/cron.daily, /etc/cron.hourly, /etc/cron.monthly, /etc/cron.weekly`

직접 sh 파일을 등록시켜 놓으면 정해진 시간에 수행
`/etc/crontab`에 각 폴더에 대한 스케줄 설정이 되어있음

`/var/log/cron` Cron 실행이 기록되는 로그 파일



CRONTAB 저장 경로

`/var/spool/cron/crontabs/[사용자 이름]` 각 사용자를 위한 crontab

`/etc/crontab` root 관리자를 위한 system crontab

```
$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 * * * * root    cd / && run-parts --report /etc/cron.hourly
25 6 * * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6 * * 7 root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6 1 * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
#
```

이를 위해 사용



CRONTAB 명령어

<code>crontab -e</code>	예약 작업 작성 및 수정
<code>crontab -r</code>	예약 작업 삭제
<code>crontab -l</code>	예약된 작업 목록



CRONTAB 작성

환경변수

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/
```

```
# For details see man 4 crontabs
```

```
# Example of job definition:
# ----- minute (0 - 59)
# | ----- hour (0 - 23)
# | | ----- day of month (1 - 31)
# | | | ----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | ----- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# | | | | |
# * * * * * user-name command to be executed
```

crontab fromat



CRONTAB 형식

* * * * * [command]

```
# Example of job definition:
# ----- minute (0 - 59)
# |----- hour (0 - 23)
# | |----- day of month (1 - 31)
# | | |----- month (1 - 12) OR jan,feb,mar,apr ...
# | | |----- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# | | | |
# * * * * * user-name command to be executed
```

'*' 해당 필드의 모든 시간
';' 여러 시간 지정
'-' 시간을 범위로 지정
'/' 시간의 주기를 지정

<http://www.cronmaker.com>

* * * * * /home/script/test.sh

매분 test.sh 실행

45 5 * * 5 /home/script/test.sh

매주 금요일 오전 5시 45분에 test.sh 실행

0,20,40 * * * * /home/script/test.sh

매일 매시간 0분, 20분, 40분에 test.sh 실행

0-30 1 * * * /home/script/test.sh

매일 1시부터 1시 30분까지 매분 test.sh 실행

*/10 * * * * /home/script/test.sh

매 10분마다 test.sh 를 실행

*/10 2,3,4 5-6 * * /home/script/test.sh

5일에서 6일까지 2시, 3시, 4시에 매 10분마다 실행



CRONTAB 환경변수

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/
```

shell에서는 돌아가는 스크립트가 crontab에 등록하면 정상적인 동작을 하지 않는 경우

=> 환경 변수의 문제일 수 있다

=> Crontab 설정의 실행 명령에 환경변수를 포함하여 실행!

변수명 = 값		
SHELL		cron이 돌아가는 shell (Default : /bin/sh)
PATH		cron 프로그램 경로 (Default : /usr/bin:/bin)
MAILTO		cron 실행 결과를 받아볼 메일 주소 (" "으로 두면 메일이 오지 않는다) -> sendmail과 같은 메일발송 daemon이 필수
HOME		cron에 쓰일 홈 디렉토리 (\$HOME Default : /etc/passwd의 값)
LONGNAME		crontab의 소유주



ANACRON

Q. 만약 11시에 예약된 작업이 있는데,

10시 55분 ~ 11시 5분 사이에 있던 서버 보수 작업으로 시스템이 꺼져 있었다면?

A. 수행이 당연히 안된다. Cron은 다음 주기에 해당 작업을 수행할 것이다.

anacron

해당 작업이 정해진 시간 내에 실행된 적이 있는지 점검하고 수행한 적이 없다면 해당 작업을 실행
Daemon 프로세스가 존재하지 않으므로 정기적으로 anacron 명령어를 실행하는 구성이 따로 필요
Anacron config는 /etc/anacrontab에 저장됨



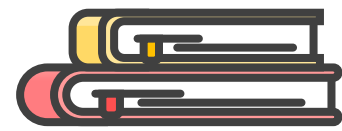
실습 1

1분마다 date를 출력해서 .txt 파일에 넣기 (10번 이상)



실습 2

1분마다 구구단 한 줄씩 .txt 파일에 추가하기



Q&A