# 1. 시스템 감시

# 1) 특정 프로세스 감시

> ps와 주기처리를 이용하여
> 특정 프로세스를 감시할 수 있음

예 test10-1.sh

```
#!/bin/sh
while :
do
        ps -ef|grep $1
        echo "======================="
        sleep 1
done
exit 0
```

〈출처 : 교수자 제작물〉

# 1) 특정 프로세스 감시

```
root@ubuntu:/home/shtest/test10# sh test10-1.sh apache2
root 1050 1 0 21:19 ? 00:00:00 /usr/sbin/apache2 -k start
www-data 1053 1050 0 21:19 ? 00:00:00 /usr/sbin/apache2 -k start
www-data 1054 1050 0 21:19 ? 00:00:00 /usr/sbin/apache2 -k start
www-data 1055 1050 0 21:19 ? 00:00:00 /usr/sbin/apache2 -k start
www-data 1056 1050 0 21:19 ? 00:00:00 /usr/sbin/apache2 -k start
www-data 1057 1050 0 21:19 ? 00:00:00 /usr/sbin/apache2 -k start
root 1863 1398 0 21:57 pts/0 00:00:00 sh test10-01.sh apache2
root 1865 1863 0 21:57 pts/0 00:00:00 grep apache2
=======================
root 1050 1 0 21:19 ? 00:00:00 /usr/sbin/apache2 -k start
www-data 1053 1050 0 21:19 ? 00:00:00 /usr/sbin/apache2 -k start
www-data 1054 1050 0 21:19 ? 00:00:00 /usr/sbin/apache2 -k start
www-data 1055 1050 0 21:19 ? 00:00:00 /usr/sbin/apache2 -k start
www-data 1056 1050 0 21:19 ? 00:00:00 /usr/sbin/apache2 -k start
www-data 1057 1050 0 21:19 ? 00:00:00 /usr/sbin/apache2 -k start
root 1863 1398 0 21:57 pts/0 00:00:00 sh test10-01.sh apache2
root 1868 1863 0 21:57 pts/0 00:00:00 grep apache2
^C
root@ubuntu:/home/shtest/test10#
```

〈출처 : 교수자 제작물〉

# 2) 접속 단말수 세기

사용자가 단말에 **telnet이나 ssh로 접속**하는 경우

sshd 또는 bash 프로세스 실행

사용자가 **웹 서버에 접속**하는 경우

httpd 혹은 apache2 실행

- bash의 감시 ps -ef|grep bash (sshd보다 bash가 더 정확하게 계수 가능)
- 사실 ps -ef|grep bash의 자신의 명령어도 체크됨
- wc -l을 이용하여 해당 출력된 줄 수를 카운트

# 2) 접속 단말수 세기

✔ sshd보다 bash가 더 정확하게 계수 가능 -〉 현재는 아래와 같이 나오지 않음

```
root@ubuntu:~# ps -ef|grep sshd
root      1043     1  0 14:38 ?        00:00:00 /usr/sbin/sshd -D
root      1387  1043  0 14:47 ?        00:00:00 sshd: root@pts/0
root      1708  1043  0 16:24 ?        00:00:00 sshd: root@pts/1
root      1907  1043  0 17:11 ?        00:00:00 sshd: root@pts/2
root      1996  1779  0 17:11 pts/1    00:00:00 grep --color=auto sshd
root@ubuntu:~#
```

〈출처 : 교수자 제작물〉

```
root@ubuntu:~# ps -ef|grep sshd|wc -l
5
root@ubuntu:~#
```

〈출처 : 교수자 제작물〉

# 2) 접속 단말수 세기

예 test10-2.sh

```
#!/bin/sh
while :
do
          i_bash=`ps -ef|grep bash|wc -l`
          i_bash=`expr $i_bash - 1`
          i_http=`ps -ef|grep apache2|wc -l`
          i_http=`expr $i_http - 1`
          echo ================================
          echo ' CONNECTED USERS : ' $i_bash
          echo ' WEB USERS : ' $i_http
          echo ================================
          sleep 2
done
exit 0
```

〈출처 : 교수자 제작물〉

# 2) 접속 단말수 세기

예 test10-2.sh

```
root@ubuntu:/home/shtest/test10# sh test10-2.sh
======================
CONNECTED USERS : 6
WEB USERS : 6
======================
======================
CONNECTED USERS : 6
WEB USERS : 6
======================
======================
CONNECTED USERS : 6
WEB USERS : 6
======================
^C
root@ubuntu:/home/shtest/test10#
```

〈출처 : 교수자 제작물〉

# 2. 시스템 모니터링

# 1) 크기가 얼마 이상인 파일 찾기

## 원리

- cut을 이용한 출력 줄에서 필요한 부분 자르기
- ls -l /etc/*.conf을 대상으로 실습
- 아래 숫자는 임의로 작성 (0부터 시작할 것)

```
root@ubuntu:/home/shtest/test03# ls -l /etc/*.conf
-rw-r--r-- 1 root root 2981 Dec 19 11:21 /etc/adduser.conf
-rw-r--r-- 1 root root  321 Mar 30 2012  /etc/blkid.conf
-rw-r--r-- 1 root root 8457 Dec 19 13:04 /etc/ca-certificates.conf
-rw-r--r-- 1 root root 2969 Mar 15 2012  /etc/debconf.conf
-rw-r--r-- 1 root root  604 Oct 20 2011  /etc/deluser.conf
-rw-r----- 1 root fuse  280 May 24 2013  /etc/fuse.conf
01234567890123456789012345678901234567890123456789012345678901234567890123456789
```

〈출처 : 교수자 제작물〉

# 1) 크기가 얼마 이상인 파일 찾기

## 원리

- 이는 ls -l 명령으로 볼 때, 크기 부분은 25~29이며 파일 이름은 43 이후 문자열임
- ls_list=`ls -l /etc/*.conf|cut -c25-29`
- $ls_list는 파일의 크기가 저장
- file_list=`ls -l /etc/*.conf|cut -c43-`
- $file_list는 파일이름이 저장

# 1) 크기가 얼마 이상인 파일 찾기

## 원리

예 test10-3.sh

```
#! /bin/sh
ls_list=`ls -l /etc/*.conf|cut -c25-29`
file_list=`ls -l /etc/*.conf|cut -c43-`
cnt=0
for ls_one in $ls_list   ;do
        if [ $ls_one -gt 100 ] ;   then
                echo -n "OK :"
                fcnt=0
                for filename in $file_list ;   do
                        if [ $cnt -eq $fcnt ];   then
                                printf "file [%s] size[%d]₩n" $filename  $ls_one
                        fi
                        fcnt=$(($fcnt+1))
                done
        fi
        cnt=$(($cnt+1))
done
```

〈출처 : 교수자 제작물〉

# 1) 크기가 얼마 이상인 파일 찾기

## 원리

```
root@ubuntu:/home/shtest/test10# sh test10-3.sh
OK :file [/etc/adduser.conf] size[2981]
OK :file [/etc/blkid.conf] size[321]
OK :file [/etc/ca-certificates.conf] size[8457]
OK :file [/etc/debconf.conf] size[2969]
OK :file [/etc/deluser.conf] size[604]
OK :file [/etc/fuse.conf] size[280]
OK :file [/etc/gai.conf] size[2584]
OK :file [/etc/hdparm.conf] size[4781]
OK :file [/etc/inetd.conf] size[1187]
OK :file [/etc/insserv.conf] size[771]
OK :file [/etc/kernel-img.conf] size[144]
OK :file [/etc/libaudit.conf] size[191]
OK :file [/etc/logrotate.conf] size[703]
OK :file [/etc/ltrace.conf] size[4867]
OK :file [/etc/mke2fs.conf] size[956]
OK :file [/etc/nsswitch.conf] size[475]
OK :file [/etc/pam.conf] size[552]
OK :file [/etc/popularity-contest.conf] size[350]
OK :file [->] size[1320]
OK :file [../run/resolvconf/resolv.conf] size[2084]
OK :file [/etc/rsyslog.conf] size[1260]
OK :file [/etc/sysctl.conf] size[321]
OK :file [/etc/ucf.conf] size[289]
root@ubuntu:/home/shtest/test10#
```

〈출처 : 교수자 제작물〉

# 1) 크기가 얼마 이상인 파일 찾기

## awk를 이용한 간단한 방법

> 추출 필드를 지정하고 형식을 주어서 출력할 수 있는 awk명령어를 사용하면 해당 내용이 간단해 짐

1. ls -l /etc/*.conf | awk '{ print "size : " $5 " , name : " $9}'
2. ls -l /etc/*.conf 결과를 awk명령으로 다시 처리
3. awk '{ print ~~}' 명령은 형식을 지정하여 출력함
4. size : 이라고 먼저 출력하고 $5 출력

# 1) 크기가 얼마 이상인 파일 찾기

## awk를 이용한 간단한 방법

**5** $5는 ls -l /etc/*.conf에서 얻어진 출력에 대하여 띄어쓰기(blank)를 기준으로 5번째 항목을 출력

```
root@ubuntu:/home/shtest/test03# ls -l /etc/*.conf
-rw-r--r-- 1 root root 2981 Dec 19 11:21 /etc/adduser.conf
-rw-r--r-- 1 root root  321 Mar 30  2012 /etc/blkid.conf
-rw-r--r-- 1 root root 8457 Dec 19 13:04 /etc/ca-certificates.conf
-rw-r--r-- 1 root root 2969 Mar 15  2012 /etc/debconf.conf
-rw-r--r-- 1 root root  604 Oct 20  2011 /etc/deluser.conf
-rw-r----- 1 root fuse  280 May 24  2013 /etc/fuse.conf
-rw-r--r-- 1 root root 2584 Oct 11  2012 /etc/gai.conf
-rw-r--r-- 1 root root 4781 Nov 16  2013 /etc/hdparm.conf
[ 1      ] 2 [ 3] [ 4] [ 5] [6]  7 [  8] [ 9                    ]
```

〈출처 : 교수자 제작물〉

# 1) 크기가 얼마 이상인 파일 찾기

## awk를 이용한 간단한 방법

6 이어서 name : 라고 먼저 출력하고 $9 를 출력

7 ls -l /etc/*.conf | awk '{ print "size : " $5 " , name : " $9}' 결과는 아래와 같음

```
root@ubuntu:/home/shtest/test10# ls -l /etc/*.conf | awk '{ print "size : " $5 " , name : " $9}'
size : 2981 , name : /etc/adduser.conf
size : 321 , name : /etc/blkid.conf
size : 8457 , name : /etc/ca-certificates.conf
size : 2969 , name : /etc/debconf.conf
size : 604 , name : /etc/deluser.conf
size : 280 , name : /etc/fuse.conf
size : 2584 , name : /etc/gai.conf
[ 1] 2 [ 3 ] 4 [ 5 ] 6 [ 7                    ]
```

〈출처 : 교수자 제작물〉

# 1) 크기가 얼마 이상인 파일 찾기

## awk를 이용한 간단한 방법

- ls -l /etc/*.conf | awk '{ print "size : " $5 " , name : " $9}' 결과 중 3번째 필드가 크기임
- awk명령어는 간단한 비교가 가능
- 해당 결과의 3번째 필드에 대하여 크기가 100이상인 경우만 출력함

# 1) 크기가 얼마 이상인 파일 찾기

## awk를 이용한 간단한 방법

예 test10-4.sh

```
#! /bin/sh
ls -l /etc/*.conf | awk '{ print "size : " $5 " , name : " $9}'|
awk '$3 >= 100 '
```

〈출처 : 교수자 제작물〉

# 1) 크기가 얼마 이상인 파일 찾기

## awk를 이용한 간단한 방법

**예** test10-4.sh

```
root@ubuntu:/home/shtest/test10# sh test10-4.sh
size : 2981 , name : /etc/adduser.conf
size : 321 , name : /etc/blkid.conf
size : 8457 , name : /etc/ca-certificates.conf
size : 2969 , name : /etc/debconf.conf
size : 604 , name : /etc/deluser.conf
size : 280 , name : /etc/fuse.conf
size : 2584 , name : /etc/gai.conf
size : 4781 , name : /etc/hdparm.conf
size : 1187 , name : /etc/inetd.conf
size : 771 , name : /etc/insserv.conf
size : 144 , name : /etc/kernel-img.conf
size : 191 , name : /etc/libaudit.conf
~~
```

〈출처 : 교수자 제작물〉

# 3. 시스템 백업 응용

# 1) 파일명을 찾아 날짜 형식의 디렉토리로 백업하기

## 원리

- date를 이용하여, **날짜를 디렉토리 명으로 디렉토리 생성**
  - 예제는 /etc/*.conf파일들을 당일자의 디렉토리 명으로 디렉토리를 만들고 해당 파일을 복사(백업)

- /etc 디렉토리의 확장자가 conf 인 파일을 백업하듯,
  중요한 파일을 골라서 일자별 디렉토리에 복사(백업)하는 것을 가정함

# 1) 파일명을 찾아 날짜 형식의 디렉토리로 백업하기

## 원리

예 test10-5.sh

```
#! /bin/sh
d_date=`date +%y%m%d`
mkdir /home/$d_date
mkdir /home/$d_date/etc
for file in $(ls /etc/*.conf)
do
        dest=`echo /home/$d_date$file`
        echo $file "===>" $dest
        cp $file $dest
done
exit 0
```

〈출처 : 교수자 제작물〉

# 1) 파일명을 찾아 날짜 형식의 디렉토리로 백업하기

## 원리

**예** test10-5.sh

```
root@ubuntu:/home/shtest/test10# sh test10-5.sh
/etc/adduser.conf ===> /home/160119/etc/adduser.conf
/etc/blkid.conf ===> /home/160119/etc/blkid.conf
/etc/ca-certificates.conf ===> /home/160119/etc/ca-certificates.conf
/etc/debconf.conf ===> /home/160119/etc/debconf.conf
~~
root@ubuntu:/home/shtest/test03# ls /home/16*/etc/
adduser.conf deluser.conf host.conf ld.so.conf mke2fs.conf resolv.conf updatedb.conf
blkid.conf fuse.conf inetd.conf libaudit.conf nsswitch.conf rsyslog.conf xinetd.conf
ca-certificates.conf gai.conf insserv.conf logrotate.conf pam.conf sysctl.conf
debconf.conf hdparm.conf kernel-img.conf ltrace.conf popularity-contest.conf ucf.conf
root@ubuntu:/home/shtest/test10#
```

〈출처 : 교수자 제작물〉

# 4. 시스템 감시 실습

# 1) 실습하기

실습내용

(1) 시스템 감시

# 5. 시스템 모니터링, 백업 실습

# 1) 실습하기

실습내용

(1) 시스템 모니터링
(2) 시스템 백업 응용

# 학습활동

* 일시정지 버튼을 클릭하고 학습활동에 참여해 보세요.

Q 이번 시간에는 리눅스에서 모니터링 쉘 프로그래밍을 알아보았습니다. 리눅스 환경에서 프로세스를 확인하거나, cpu상황을 확인하기 위한 주요 명령어들을 조사해 봅시다.

# 학습활동에 대한 해설

**Q** 이번 시간에는 리눅스에서 모니터링 쉘 프로그래밍을 알아보았습니다. 리눅스 환경에서 프로세스를 확인하거나, cpu상황을 확인하기 위한 주요 명령어들을 조사해 봅시다.

**A**

- 리눅스 서버에서 프로세스를 확인하거나 cpu상태를 확인하는 경우는 시스템 관리자에게는 빈번한 작업입니다.
- 프로세스를 확인하는 명령으로 jobs, ps, pstree, top, ulimit가 있습니다.
- cpu를 확인하는 명령으로 vmstat, sar가 있습니다.
- 이와 같은 내용을 검색해 보고 실제 현장에서 많이 활용하도록 합시다.