

정리하기

1. 시스템 측정 및 모니터링 명령어

- ⑧ **sar** (System Activity Reporter): 여러 항목에 대한 모니터링을 하려면 그들 CPU 값을 기본으로 출력
↳ 처음 실행시 apt install sysstat으로 설치 가능함

- **Sar -N M**: N초마다 M번까지 시스템 모니터링 정보 출력

출력 내용 1) %user: 사용자로부터 받은 CPU의 비율

2) %nice: 우선순위가 낮은 일의 사용자로부터 받은 CPU 비율

3) %system: 시스템(커널)이 사용하는 CPU의 비율

4) %iowait: 디스크 등의 I/O 바느기 리오 CPU가 wait 상태에 있는 비율

5) %steal: 프세서가 CPU를 바꾸거나, 무스승의 변위로 다른 안트 변변화명사 사용된 시간의 비율

6) %idle: CPU가 free한 상태의 비율

- **Sar -d N M**: N초마다 M번까지 디스크 상태를 출력

출력 내용 1) tps: 초당 I/O 횟수가 발생하러 램

2) kb-ses/s: 초당 디바이스가 read하는 데이터량

3) kb-ws-ses/s: 초당 디바이스가 write한 데이터량

4) avarq-sz: 디바이스에서 발생한 I/O량의 평균 사이즈

5) avarq-sz: 디바이스에서 발생한 I/O량의 평균 사이즈

6) avarq-sz: 디바이스에서 I/O량을 처리하는데 걸리는 평균시간

7) %util: 디바이스에서 I/O량이 발생한 동안 걸리는 시간의 비율

- ⑨ **top**: 현재 CPU에 사용되고 있는 프로세스 상태를 CPU를 많이 사용하는 순서대로 자동으로 계속 보여주는 명령어

↳ 메모리 상황, swap 상황, 실행중의 프로세스, 바이트메모리, 자원 상태 등 출력시켜 "각 프로세스당 처리하는 프로세스의 상태를" 확인할 수 있다. * **htop**도 있다. 화면 내미는 함수 클리드 가능

* 끝났기때 없이 계속 실행을 할때는 프로세스는 hang(무엇에 묶여있음) 상태일 수 있음. ctrl+c, job no.로 종료

- ⑩ **fscck** (file system check): 파일시스템의 이거나 크래시를 복구. 명령어 사용시엔 **df -k** 명령어 명령어를 실행하고있는 디렉토리 등이 unmount 상태인지와 연 사용된 보드의 후진 후 실행해야 하라 (이제 안하면 재설치 필요)

- ⑪ **du** (disk usage): 파일시스템에서 사용량을 보여주는 명령어.
- **du -a**: 현재 디렉토리 하위 모든 파일의 디스크 사용량을 보여줌 (모든)
- **du -k**: 파일시스템이 1024 단위로 출력되는 것은 **du -k**로 실행 (디렉토리나 하위)

- ⑫ **vmstat**: CPU의 호출을 상용화, CPU 리미트 다양한 정보 출력 가능

- ⑬ **vi /proc/cpuinfo**: 현재 설치된 CPU의 자세한 상황 출력 (원래 디바이스의 정보 등)

- ⑭ **free**: 메모리와 swap메모리 상황 확인 명령어.

- ⑮ **df -k**: 파일시스템별 사용된 상황 및 배분 여부 디스크 크기 출력

- ⑯ **iostat**: 현재 I/O 발생상황 출력

- ⑰ **jobs**: 현재 tc prompt에서 실행되는 job 확인. 수행하는 job이 없으면 시 현재는 없음.
"명령어 실행을 백에서 중단하기 한 후 job에서 확인해 볼 수 있었다."

- ⑱ **ps (ps -l)**: 각종 프로세스 상황 확인

(ps -a: 모든 프로세스도 확인 가능)

- ⑲ **ps tree**:
↳ 아래 가계도 ps 상황 확인 가능

- ⑳ **winter -a**: 프로세스 자원 관리 확인

- ㉑ **/proc 디렉토리**: 현재 실행중인 프로세스 정보는 /proc 디렉토리에 저장한다. 해당 디렉토리으로 이동하여 ls -al 하면 목록 확인이 가능하다.

- ㉒ **ifconfig**: 랜카드 등의 인터페이스 상태를 출력 (ip netmask 등등의 ipinfo)

- ㉓ **ping**: 네트워크 연결상태 (ping -pong 반응 확인) 보는 명령어 옵션 -t, -c, -n 등이 있다.

- ㉔ **netstat -na**: network status를 상세히 보여줘 옵션 등을 ip 나 숫자로 보여주는 명령어 실행.

- 2. Task 4에 CPU, Memory, Disk 상황들을 구할 수 있는 식이 있는 것 모르면 코드를 짜다가 중반에 알아채는데, 그대 편 코드가 아가워서 Memory는 그대로 실행했다.
그런데 역시 코드를 짤고 당겨 간혹 하는 게 개발자의 과제 중 하나이기 때문에, 코드를 복기 보단,

free | grep Mem | awk '{print "MEM_FREE=" \$4 "; MEM_USED=" \$3 "; }'

이렇게 다른 값을 구하는 것처럼 하면 더 좋을 것 같다

여유가 있으면 아래쪽에 RGB 값을 조금 만져서 색깔도 바꿔 설정해도 좋다.

그렇지만 awk 명령어는 처음 봤다 조금 생소한 감이 있었는데, 프렌치 기본 인자를 받아 표현하기 하는 명령어 아니었던가. 좀더 쉽게 잘 사용할 것 같으니까. 가령 나만 잘 기억해 두어야겠다.

그런 파일을 찾거나 계속해서 여러로 파일을 새로 만든 것은 못하는데, 새로운 방법을 알게되었다. 모두 커맨드들 내에서 문법해 지어할 뿐이었지만, 이번 친구프로그래밍을 하면서 더 좋은 코드를 짜 나간 것 같다.

3. 이해하기

1. 시스템 측정 및 모니터링 명령

시스템 서버가 정상적으로 운영되고 있는지 점검을 위하여 수시로 모니터링을 수행하여야 한다. 이를 위하여 시스템을 모니터링 하는 방법에 대하여 하나하나 배워 보도록 한다.

1) sar

sar(sar(System Activity Reporter) 명령어는 유닉스,리눅스등에서 유용하게 쓰는 시스템모니터링 명령어이다.

①sar명령어 역할

- 모니터링 대상이상당히 넓은 편이며기본값은 CPU활동에대한 통계를 출력
- 각종 활동에대한 통계를 다른 프로그램(sysstat)을 이용하여파일로 저장하고통계치를 리포팅하는 기능을 제공.

②sar 설치

- sar 명령어가실행되지 않는 경우 sysstat를 설치하여야함

```
root@kopocte:~# sar 1 100
The program 'sar' can be found in the following packages:
  sysstat
  atsar
Try: apt-get install <selectedpackage> root@kopocte:~# apt install sysstat
```

<그림 IV-25> sysstat 설치

④sar 보여주는 항목

- %user : user level (application) 이 사용하는 cpu의 양(%)
- %nice : nice priority(우선순위가 지정된 job)를 가진 user level이 사용하는 cpu의 양(%) **낮을수록 좋음**
- %system : system level(kernel)이 사용하는 cpu의 양(%)

3. 이해하기

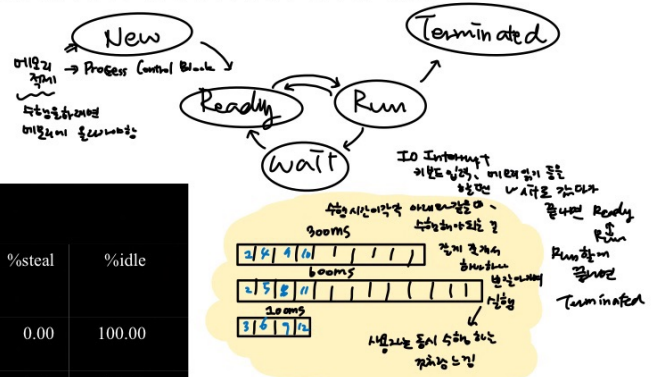
·%iowait : 디스크등의 i/o의 속도차로 인하여 cpu가 쉬게 되는 시간 비율(%)

·%steal : 프로세스(cpu상 하나의 실행단위)가 cpu들을 바꾸거나, 잡 우선순위에 의하여 다른 잡이 들어오거나 하는 순간 때 문에 사용된 시간 비율(%)

·%idle : cpu가 널널한 비율 (100- (%user+%nice+%system)) **cpu 유휴**

④sar 사용

- sar 1 5 : cpu 사용량을 1초간격으로 5번 조사



```
root@kopocte:~# sar 1 5
Linux 3.13.0-74-generic (kopocte) 01/13/2025 _i686_ (1 CPU)
```

04:57:35 PM	C P U	%user	%nic e	%syst em	%iowai t	%steal	%idle
04:57:36 PM	al l	0.00	0.00	0.00	0.00	0.00	100.00
04:57:37 PM	al l	0.00	0.00	0.00	0.00	0.00	100.00
04:57:38 PM	al l	0.00	0.00	1.00	0.00	0.00	99.00
04:57:39 PM	al l	0.00	0.00	0.00	0.00	0.00	100.00
04:57:40 PM	al l	0.00	0.00	0.00	0.00	0.00	100.00
Average:	al l	0.00	0.00	0.20	0.00	0.00	99.80

```
root@kopoct c:~#
```

3. 이해하기

2) top

ktop도 있음

top 명령어는 유닉스, 리눅스 등에서 유용하게 쓰는 시스템 모니터링 명령어로 다시 복습해 본다.

① top 명령 : 현재 cpu에 수행되고 있는 프로세스상황을 cpu자원을 많이 사용하는 순서대로 자동으로 계속 보여줌

② 메모리상황, swap상황, 실행 우선순위, 버추얼 메모리, 리소스, 등을 보여줌

③ 가장 중요한 것은 처리시간을 얼마나 사용하고 있는가를 모니터링 하는 것임

④ 만일 계속하여 처리시간을 잡아먹고 있는 프로세스가 있다면, 그 프로세스는 hang인 상태인 경우(무한루프)일 수 있으며 이 프로세스는 kill -9 job#로 처리해주는 경우가 있을 수 있음

top - 17:08:57 up 26 min, 1 user, load average: 0.00, 0.01, 0.05
Tasks: 80 total, 1 running, 79 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.3 us, 0.3 sy, 0.0 ni, 99.3 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem: 766352 total, 354120 used, 412232 free, 32000 buffers

3. 이해하기

```
top - 17:08:57 up 26 min, 1 user, load average: 0.00, 0.01, 0.05
Tasks: 80 total, 1 running, 79 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.3 us, 0.3 sy, 0.0 ni, 99.3 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem: 766352 total, 354120 used, 412232 free, 32000 buffers

 260 root      2  0   0   0   0 S   0  0  0:01.36 kworker/0:2
 1162 tomcat7     2  0 2819 639 9648 0 8 0:04.62 java
      (      20 60 S   3 3
 1386 kopoctc   2  0 1119 169 948 0 0 0:00.16 sshd
      (      6 6 S   3 2
 1 root        2  0 4324 232 1392 0 0 0:00.88 init
      (      8 S   3
 2 root        2  0   0   0   0 S   0  0 0:00.00 kthreadd
      (      0
 3 root        2  0   0   0   0 S   0  0 0:00.01 ksoftirqd/0
      (      0
 4 root        2  0   0   0   0 S   0  0 0:00.00 kworker/0:0
      (      0
 5 root        0  2   0   0   0 S   0  0 0:00.00 kworker/0:0H
      0
 6 root        2  0   0   0   0 S   0  0 0:00.16 kworker/u2:0
      (      0
 7 root        2  0   0   0   0 S   0  0 0:00.16 rcu_sched
```

● 3. 이해하기

3) fsck, du

fsck, du 명령어는 유닉스, 리눅스 등에서 유용하게 쓰는 파일 관련 모니터링 명령어이다.

① fsck

· fsck : file system check

· 파일시스템의 에러나 크래시(파일시스템의 부서짐)를 복구.

*mount가 실행중이면 강제 종료
(리눅스에 리스가 돌을때만 가능)*

· 각 파일시스템을 unmount하고 단일사용자 모드에서 하는 것이 효과적(장애 때)

사실 보통은 두개 둘은 이해한 개월식 있음

· 디스크 작업 시 본 명령 실행 시 디스크 크래시 주의 (디스크가 사용되고 있는 상황에서 본 명령이 실행되는 경우 파일 시스템의 복구 불능 장애 발생)

② du

· du (disk usage): 파일 스페이스의 사용량을 보여줌

· du -a 현재 디렉토리 하위 모든 파일의 스페이스 사용량을 보여줌. ls에서 보이는 실크기와 디스크에 할당된 크기는 차이가 있음

· 파일 시스템이 1024단위로 할당되지 않은 경우 du -k로 실행

1<10byte

```
root@kopoct:/home# du
4  ./kopoct/.cache
32  ./kopoct
48  .
root@kopoct:/home# du -k
4  ./kopoct/.cache
32  ./kopoct
48  .
root@kopoct:/home#
```

<그림 IV-29> du 명령어

● 3. 이해하기

Disk 확인	df -k	파일 시스템별 마운트 상황, 배분 크기가 보임 <i>전체 용량, 남는 양</i>
	du -a	파일별 스페이스 사용량 보임
	iostat	현재 io 발생 상황을 보여줌
네트워크 관련	ifconfig	랜카드 등 인터페이스 상황을 보여줌
	ping	네트워크 연결 설정을 보는 기본 명령
	netstat -na	Tcp/ip 어플리케이션 차원의 서비스 상태를 보여줌

<표 IV-5> 주요 모니터링 명령어

Tip

아래 명령어 들은 시스템 모니터링을 위하여 엄선된 명령어이다.

옵션을 잘 활용함에 따라 모니터링의 활용도가 높아진다. 반드시 man을 통하여 명령어를 찾아보자