

리눅스 마스터 2급 정리

by minusi.tistory.com



본 문서를 이용하실 때 위 라이선스를 준수하는 내에서 자유롭게 사용하셔도 됩니다.

네트워크 명령어

arp

- ▶ IP주소를 MAC주소로 변환
 - ▶ 다른 호스트의 MAC주소 확인 가능
- : arp [opt] [ip주소] [mac주소]

-a	ARP Cache에 있는 호스트 정보 출력
-s	특정 ip주소에 대한 MAC 주소 변경
-d	ARP Cache에 저장된 특정 ip주소에대한 MAC 주소 삭제
-i	특정 Ethernet의 ARP 확인

route

- ▶ 라우팅 테이블 관련 정보 조회

ipconfig

- ▶ 네트워크 인터페이스를 설정
 - ▶ service network restart하면 초기 IP주소로 되돌림
 - ▶ 관련 파일
 - /etc/sysconfig/network : 호스트명, 게이트웨이
 - /etc/resolv.conf : DNS 설정
 - /etc/hosts : 캐시된 DNS 설정
 - /etc/sysconfig/network-script/ifcfg-eth0
- : ipconfig [인터페이스명] [opt] [주소] [up/down]

netmask [넷마스크]	서브넷마스크 지정
-------------------	-----------

broadcast [주소]	브로드캐스트 주소 지정
-------------------	--------------

up/down	활성화/비활성화
---------	----------

netstat

- ▶ 네트워크 연결상태, 라우팅테이블, 인터페이스 상태 등을 보여줌
- : netstat [opt]

-a	모든 네트워크 상태 출력
-r	라우팅 테이블 출력
-t	TCP 프로토콜만 출력
-u	UDP 프로토콜만 출력
-A protocol	프로토콜 별로 보기 (inet, unix, ipx, ax25, netrom, ddp)

-c	매초마다 실행
-n	다른 pc와 연결되어 있는 포트번호 확인
-e	랜카드에서 송수신한 패킷의 용량 및 종류 확인
-s	IP, ICMP, UDP 프로토콜별 상태
-r	라우팅 테이블 확인 및 커넥션 포트번호 확인

ethtool

▶ 네트워크 인터페이스 카드의 물리적 연결 여부 확인 및 변경 가능
: ethtool [opt] [ethx] [opt]
>> ethtool -s eth0 speed 10 duplex full autoneg on

-s 속도, 전송방식(Duplex)여부 등 변경가능

mii-tool

▶ 네트워크 장치가 정상적으로 인식되는지 확인
▶ link ok가 나오면 무조건 mii-tool

ss(Socket Statistics)

▶ netstat 명령 대체
▶ 서버에 접속한 클라이언트 IP 주소 및 포트 번호 확인

-a 모든 소켓 정보

ip

▶ ip 주소 정보 조회 및 ip 설정
▶ 자신의 MAC 주소 확인 가능
: p [option] [대상] [층]
>> ip addr show (ip 정보 출력)
>> ip addr/del add 192.168.5.10/24 dev dth0
(eth0 인터페이스에 ip 설정/삭제)
>> ip link set eth0 up/down
(eth0 인터페이스 활성화/비활성화)

>> ip route show (라우팅 정보 출력)
>> ip route add/del default via 192.168.5.1
(게이트웨이 설정/삭제)
>> ip route add/del 10.20.12.0/24
via 192.168.5.1 dev eth0

(정적 라우팅 설정/삭제)	
ip a	네트워크 인터페이스 정보(IP, MAC, DNS 서버, = ip address [show])
mail	
▶ 메일클라이언트 사용 : mail [opt] >> mail -s "제목" [수신자주소] < [파일명]	
-s	메일 전송(send)

telnet

▶ 텔넷 클라이언트 실행. 호스트에 원격 접속 수행
▶ 관련파일
- /etc/motd : 로컬접속, 원격접속 모두 메시지 출력
- /etc/issue : 로컬 접속시 보여줄 메시지 출력
- /etc/issue.net : 원격 접속시 보여줄 메시지 출력
: telnet [IP주소 | 도메인명 | 호스트명]

-l 유저ID	텔넷 서버 시스템에 접속할 계정 지정
-a	현재 사용자 이름을 ID로 사용

ftp

▶ file transfer protocol을 사용한 대용량 파일 송수신 가능
▶ 익명 계정으로 접속가능
(ID:Anonymous, PW:이메일주소)
▶ Active / Passive 모드 지원
▶ FTP 서버가 있어야 함.
▶ 경량화 버전의 저용량 데이터용 TFTP가 있음

open	호스트 이름, IP로 접속
close	접속 연결 끊고 ftp 명령어 모드로
ascii	ASCII 모드로 파일 송수신
binary	Binary 모드로 파일 송수신
(m)get	파일 수신
(m)put	파일 송신
hash	파일 전송 상태를 # 문자로 확인
delete	파일 삭제

ssh

▶ ssh 클라이언트 실행. 다른 컴퓨터에 로그인하여 원격 시스템에서 명령 실행 및 다른 시스템으로 파일 전송

일 복사 가능 ▶ ssh 관련 서비스 <ul style="list-style-type: none"> - rsh(remote shell) - scp(원격 파일 로컬로 복사해오기) - sftp(ssh ftp, ftp의 보안 연결) ▶ SSH 관련 파일 <ul style="list-style-type: none"> - ~/.ssh/authorized_keys ▶ ECDSA(Elliptic Curve Digital Signature Algorithm), RSA Key 등 보안 알고리즘 보이면 ssh : ssh [opt] [user@]서버주소	
-l 유저ID	ssh에 로그인할 계정 지정 [user@]서버주소와 같이 로그인됨
-p port	지정된 포트로 연결 수행
startx	
▶ run level 3에서 x윈도우 실행하는 명령 ▶ 최종적으로 xinit 수행 ▶ [ctrl] + [alt] + [backspace] : x-window 종료 : startx [[client] [opt]] [-- [server] [display] [opt]]	
--depth	농도(비트단위, 8 : 256)
--dpi	dpi

명	
-P 패키지명	해당 패키지 및 설정파일 모두 삭제
APT(Advanced Packaging Tool)	
▶ 데비안 GNU/리눅스 계열에서 소프트웨어 설치, 제거 수행 ▶ 저장소 기반 패키지 관리 ▶ 실질적으로 dpkg와 함께 작동 ▶ 관련 파일 - /etc/apt/sources.list : 의존성, 충돌성 해결 위해 참조되는 파일	
apt-get install [패키지명]	패키지 설치
apt-get update [패키지명]	패키지 업데이트
apt-get upgrade [패키지명]	패키지 업그레이드
apt-get dist-upgrade [패키지명]	패키지 추가 업그레이드
apt-get remove [패키지명]	패키지 제거
apt-cache search [패키지명]	패키지 설치여부 검색
apt-cache show [패키지명]	설치패키지 정보조회
rpm(Redhat Package Manager)	
▶ 레드햇 계열 패키지 관리, 저장소 기반 ▶ rpm은 패키지파일로 압축파일이 아님 ▶ rpm 패키지 이름 - 패키지명, 버전, 릴리즈, 아키텍처(패버릴라로 암기) ex) 패키지명.버전.릴리즈.아키텍처로 이름 구성	
-v	설치과정 확인

-V	<p>rpm 패키지 검증/비교 >> S 5.... T. c /etc/mail/sendmail.cf S : 파일크기 변경됨 L : 심볼릭링크 변경됨 T : 파일수정일 변경됨 D : 장치파일 변경됨 U : 파일사용자/소유자 변경됨 G : 파일 그룹 변경됨 M : 파일 모드(권한,형태)가 변경됨 5 : MD5체크섬이 변경됨 ?: 원인 모름</p>
-h	설치과정 #으로 출력
-i	새 패키지 설치
-e	<p>패키지 삭제 --nodeps : 의존성 있는 있는 패키지 가 존재해도 제거</p>
-U	패키지 업그레이드
-F	이전 버전 있을 때만 업그레이드 (freshen)
-ivh	새 패키지 설치
-Uvh	업그레이드 / 새 패키지 설치
-Fvh	이전 패키지 업그레이드
-q	패키지 설치 되었는지 확인
-qa	시스템에 설치된 모든 패키지 목록
-qd	관련 문서 및 man 페이지 파일 정보 출력
-qf	설치 패키지를 지정한 포맷으로 출력
-qi	패키지 이름과 버전만 표시하여 설치된 패키지 정보 출력
-ql	패키지에서 설치한 모든 파일 정보 출력
-qR	패키지 설치에 필요한 의존성 목록 확인 가능
yum(Yellowdog Updater Modified)	
<ul style="list-style-type: none"> ▷ Duke 대학교에서 rpm 명령을 개선한 패키지 ▷ rpm 명령의 패키지 의존성 문제 해결 ▷ 저장소 기반 패키지 관리 	

<p>▷ yum 파일 - /etc/yum.conf : yum 동작 방식 설정 파일 - /etc/yum.repos.d : 원격지 저장소 정보 저장 - /etc/yum/pluginconf.d/ : 플러그인 정보 저장 ▷ 이후, 페도라22에서부터 yum의 문제점을 보완한 울가 사용됨</p>	
install	설치
check-update	업데이트 가능 목록
update	패키지 업데이트
remove erase -e	삭제
search	패키지 설치 가능 확인
info	패키지 정보
groupinfo	해당 패키지 그룹명과 관련된 패키지 정보 보여줌
list	전체 패키지에 대한 정보 출력
history	yum 관련 작업이력 출력
synaptic	
<ul style="list-style-type: none"> ▷ 데비안 계열 ▷ APT 패키지 관리 시스템으로 GTK+ 기반의 GUI 도구 ▷ APT(Advanced Packaging Tool)의 GUI이면서 프론트엔드 격. ▷ 일반적으로 .deb 패키지 기반에서 이용하지만 rpm 베이스에서도 사용가능 	
aptitude	
<ul style="list-style-type: none"> ▷ 데비안 계열 ▷ 고수준 패키지 매니저 인터페이스 	

압축 관련 명령어
tar
<ul style="list-style-type: none"> ▷ tape archive. 파일아카이브 생성/관리/해제
:tar [opt] [파일들]

▷ 압축 효율 : xz > bzip2 > gzip > Z

-J	xz 압축 적용 옵션
-j	bzip2 압축 적용 옵션
-z	gzip 압축 적용 옵션
-Z	Z 압축 적용 옵션
-c	tar 아카이브 생성. 덮어쓰기
-u -r	tar 아카이브 마지막에 파일추가
-t	tar 아카이브 파일 내용 확인
-x	tar 아카이브 파일 압축해제
-v	처리 과정 상세히 출력
-f tar이름	대상 tar 아카이브 지정 파일명 대신 -를 쓰면 표준 입출력
-C 디렉토리	디렉토리 변경시 사용
xz / unxz	
<ul style="list-style-type: none"> ▶ .xz 확장자로 압축, 압축해제 명령어 ▶ LZW 알고리즘 사용 리눅스 계열 OS에서 자주 사용 	
-z	기본 운영모드로 압축
-d	압축 해제
-t	무결성 검사
bzip2 / bunzip2	
<ul style="list-style-type: none"> ▶ .bz2 확장자로 압축, 압축해제 명령어 ▶ 블록정렬알고리즘 + 허프만 코딩 사용 	
gzip / gunzip	
<ul style="list-style-type: none"> ▶ .gz 확장자로 압축, 압축해제 명령어 ▶ LZ77 알고리즘, 허프만 코드 이용 <p>: gzip [opt] [파일명]</p>	
-n	압축 정도(1~9, 낮을수록 압축률 감소)
-c	압축결과 출력, 원본파일은 그대로
-d	압축 해제
-l	압축 전 파일명 크기, 압축률 정보 출력
-f	강제 압축
-r	디렉토리 파일도 압축

compress / uncompress	
	▶ .Z 확장자로 압축, 압축해제 명령어
<ul style="list-style-type: none"> ▶ LZW 알고리즘 사용 	
zcat	
<ul style="list-style-type: none"> ▶ 압축파일을 표준 출력으로 출력함 ▶ .gz 및 .Z 파일의 내용 볼 수 있음 <p>: zcat [opt] [파일명]</p>	
-t	압축된 파일을 테스트
-d	압축 해제

스케줄링 관련 명령어	
cron	
<ul style="list-style-type: none"> ▶ 원하는 시간에 작업을 수행하는 데몬 	
<ul style="list-style-type: none"> ▶ 관련파일 	
-z	/var/spool/cron : 시스템 개별사용자를 위한 crontab 파일.
-d	/etc/cron.d : 소프트웨어 패키지 설치시 필요한 주기적 작업을 등록하는 공간(디렉토리).
-t	/etc/crontab : 관리자가 직접 지정한 작업들을 설정, 임의의 사용자 권한으로 실행 가능. 시스템 작업 등록해서 사용하는 곳
-v	/etc/cron.allow : 이 파일이 있을 때 파일 내 지정된 사용자만 crontab 등록 가능
-w	/etc/cron.deny : 이 파일이 있을 때 파일 내 지정된 사용자는 crontab 등록 불가능. allow나 deny 둘 다 없으면 루트 관리자만 등록 가능
crontab	
<ul style="list-style-type: none"> ▶ 정해진 시간에 지정 작업을 수행하도록 하는 명령 	
<p>: crontab [-u 사용자ID] 파일</p> <p>: crontab [opt] [-u 사용자ID]</p>	
-e	설정된 파일 편집 – 처음에 편집기 고를 수 있음
-l	등록된 리스트 출력
-u 사용자	사용자가 등록한 crontab(다른 옵션과

	병행)
-r	crontab에 등록된 작업목록 삭제

프로세스 관련 명령어	
jobs	
▶ 진행중인 작업 상태를 보여줌 : jobs [-l, PID까지 보여줌]	
free	
▶ 메모리 사용량을 확인하는 명령어 : free [opt]	
ps	
▶ 프로세스 확인 명령어	
▶ ps 상태코드(STAT 필드)	
- P : 수행 가능 수행 중	
- T : 정지된 상태(Traced or stopped)	
- D : 인터럽트할 수 없는 대기상태	
- S : 20초 미만 잠듦(특정 이벤트가 끝나길 대기)	
- L : 20초 이상 길게 잠듦	
- W : 페이지 상태	
- Z : 좀비 프로세스 상태	
: ps [opt]	
-a	다른 사용자들의 프로세스도 보여줌
-u	특정 소유자의 프로세스 출력
-x	데몬 프로세스처럼 터미널에 종속되지 않는 프로세스 출력
-p PID	특정 PID 프로세스 출력
-e	모든 프로세스 출력
-f	프로세스 풀 포맷으로 출력
-l	긴 포맷 출력(F, S, UID, PPID, C, PRI, NI 등)
no 옵션	PID : 프로세스 ID TTY : 프로세스가 연결된 제어 터미널 TIME : CPU 사용 시간 시:분 CMD : 실행된 프로세스명

	no 옵션에 더해서
	USER :
	%CPU : CPU 사용비율
	%MEM : 메모리 사용 비율
	VSZ : 사용중인 가상 메모리 크기
	RSS : 사용중인 실제 메모리 크기
	STAT : 프로세스 상태
-aux	<ul style="list-style-type: none"> - R : 실행중 - S : sleeping - I : idle - Z : Zombie - W : swapped out - N : nice에 의해 순위 낮아짐 - T : 종료
	START : 프로세스 시작 시간 날짜
-ef	<ul style="list-style-type: none"> 기존 + UID : 프로세스 소유자명 PPID : 부모 프로세스 ID C :
	STIME : 프로세스 시작 시간 날짜
top	
▶ CPU, MEM, Process 등 시스템 상태를 실시간으로 모니터링하거나 프로세스 관리할 수 있음(우선순위 변경, 종료 등)	
: top [opt]	
-n	top 실행 주기 설정
명령 모드	
t	프로세스와 CPU 항목 on/off
c	Command line 옵션 on/off
shift + p	CPU 사용률 내림차순
shift + m	메모리 사용률 내림차순
shift + t	프로세스 동작 시간순
k PID	특정 프로세스 kill
nice	
▶ 프로세스 실행시 실행 우선순위 조정(NI값변경)	
▶ 컴파일과 같이 CPU/MEM 많이 쓸 때 시스템 속도가 저하되므로 다른 프로세스에게 우선순위 줄 때 사용	

▶ 기본값 0, -20 ~ 19까지 조정 가능 : nice [-[증가값 n 지정값], 디폴트 10] [프로세스명] >> nice -10 : 우선순위 기준 + 10 증가 >> nice --10 : 우선순위 기준 - 10 증가 : nice [프로세스명] (기준값에서 10 증가)	
renice	
▶ 실행중인 프로세스에 대한 nice값 변경 ▶ nice는 프로세스명으로 우선순위 조정 및 명령 실행시 새 프로세스 발생 ▶ renice는 PID로 우선순위 조정 및 기존 프로세스 우선순위 값을 바로 수정 : renice [opt] [NI값] [PID]	
-u	사용자 이름 지정
-g	프로세스 그룹 ID 지정
-p	해당프로세스 ID 지정
kill	
▶ 지정한 프로세스에 지정한 시그널을 보냄 1. SIGHUP : 연결끊기. 프로세스 설정파일 다시 읽음 2. SIGINT : 인터럽트(Ctrl + C) 3. SIGQUIT : 종료(Ctrl + #) 9. SIGKILL : 강제 종료 15.SIGTERM : 소프트웨어 종료 시그널(디폴트) 20. SIGTSTP : 키보드에 의해 발생함 (Ctrl + Z) : kill [opt or signal PID]	
-l	시그널 종류 출력
killall	
▶ 프로세스명으로 프로세스 종료 : killall [opt] [프로세스명]	
-l	시그널 리스트 출력
-s 시그널	SIGTERM 대신 해당 시그널 전송
-i	종료 전 확인
-y 시간	지정 시간 이전 종료(--young)
-o 시간	지정 시간 이후 종료(--old)
-u 사용자	해당 사용자 모든 프로세스 종료

-g 그룹	해당 그룹 프로세스 종료
nohup	
▶ 프로세스 중단을 무시하고 명령 실행 ▶ 표준출력, 에러는 \$HOME/nohup.out 파일에 기록(쓰기 불가 작업 역시 여기에 기록) ▶ 터미널창이 닫혀도 실행중인 프로세스를 백그라운드로 작업	
디스크 관련 명령어	
df(disk free)	
리눅스 시스템의 전체 디스크 사용량 표기	
-h	사람이 보기 편하게 단위별 사용량
-T	Type. 파일시스템 유형 표시
du(disk usage)	
리눅스의 디렉토리별 용량을 확인할 수 있음 : du [opt] [dir]	
-a	모든 파일 정보 표시
-s	총사용량만 표시
-c	모든 파일의 디스크 사용정보 제공
-b	byte
-k	kilobyte
-h	K, M ,G 로 표시
-sh	디렉토리 전체 사용량 MB단위로
-h	하위 디렉토리 사용량, 합계 출력
-ah	하위 파일, 디렉토리 사용량, 합계 출력
fdisk	
▶ 하드 디스크 초기화 및 파티션 생성 : fdisk [opt] [장치명]	
-n	새 파티션 생성
-d	파티션 제거
-s 장치명	장치 크기 출력
-p	파티션 설정 상태 출력
-q	변경된 파티션 정보 저장하지 않고 종

	료
-I	현재 파일 시스템 목록 확인
특정 파티션 RAID로 하려면 >> fdisk [장치명] 이후 t 누르고 fd 누르면 됨	
	mkfs
▶ 파티션에 파일시스템 생성 : mkfs [opt] [장치명]	
-t filesystem	파일시스템 지정
-T largefile	I-node 크기를 1MB로 지정
-v	현재 진행사항 출력
-I	파일로부터 초기 bad block
-c	Bad Block 검사, 초기화
	mke2fs
▶ mkfs 확장 명령어 ▶ mke2fs 파일 - 명령어 위치 : /sbin/mke2fs - 설정파일위치 : /etc/mke2fs.conf : mke2fs [opt] 장치명 [블록수]	
-t filesystem	파일시스템 지정(기본 ext2)
-j	저널링 기술(ext3로 적용)
-b 숫자	블록크기 지정
-c	배드블록 검사
-i 숫자	inode당 크기
	mount
▶ 보조기억장치들의 연결에 사용 ▶ 단독으로 사용할 경우 현재 시스템에 마운트된 장비 정보 출력 : mount [opt] [장치] [디렉터리]	
-a	/etc/estab에 있는 파일 시스템 모두 마운트
-t filesystem	파일시스템 지정
-o adopt	추가 옵션 ▶ ro : ReadOnly ▶ remount : 리마운트 ▶ loop : iso, loop 디바이스, CD-ROM

	파일 마운트 ▶ [계정] [패스워드] : 마운트에 계정 인증이 필요할 때 ▶ acl : Access Control Lists 마운트시
	blkid
	▶ 파일시스템 타입이 어떻게 구성됐는지 확인 ▶ "장치명: UUID 파일시스템 타입" 형식으로 출력 : blkid [파티션명] : blkid [opt]
	fsck
	▶ 파일시스템의 무결성 검사 : fsck [opt] [장치명]
	▶ chkdsk는 윈도우의 검사이며 fat32, ntfs 등의 파일시스템만 검사 ▶ scandisk 역시 윈도우 명령어
-t filesystem	검사 수행할 파일시스템
-A	/etc/estab 파일에 기술된 파일시스템 전부 검사
-P	병렬 처리 수행하여 루트 시스템 점검
-R	루트 시스템 예외
	쿼터 관련 명령어
	▶ 디스크 쿼터는 파일시스템마다 사용자, 그룹이 생성할 수 있는 파일의 용량 및 개수를 제한
	quotacheck
	▶ 파일시스템을 점검하고 쿼터 설정에 대한 데이터베이스 파일 생성
	▶ aquota.user : 사용자 쿼터 데이터베이스 파일 ▶ aquota.group : 그룹 쿼터 데이터베이스 파일 ▶ DB 파일은 해당 파일 시스템 최상위에 생성
	cron을 통해 주단위로 quotacheck -a 실행 권장 : quotacheck [opt] [filesystem]
-a	모든 파일시스템 체크
-u	사용자 쿼터 관련 체크

-g	그룹 쿼터 관련 체크
-m	리마운트(remount) 생략
-p	처리 결과 출력
-n	첫 번째 검색된 것 사용
setquota	
쿼터를 지정하는 명령	
: setquota [opt] [이름]	
[Block soft limit] [Block hard limit]	
[inode soft limit] [inode hard limit]	
[파티션명]	
-u	사용자 쿼터 설정
-g	그룹 쿼터 설정
-a	해당 시스템 모든 설정
-t	유예 기간 설정(초 단위)
edquota	
사용자, 그룹에 쿼터를 설정	
: edquota [opt] [user group]	
-u	사용자 쿼터 설정
-g	그룹에 대한 쿼터 설정
-p [원본] [복사]	'원본' 쿼터를 복사해서 '복사'에 복사 함
-t	soft limit 초과 후 시간제한
quota	
▶ 쿼터 정보 출력(repquota와 같지만 -u, -g 뿐)	
: quota [opt] [사용자명 그룹명]	
repquota	
▶ 사용자별 현재 사용량 확인할 수 있는 명령어	
▶ 쿼터설정 및 용량 제한내용 리포팅	
: repquota [opt] [사용자명 그룹명]	
-a	/etc/mtab에 있는 모든 파일시스템에 관한 할당량 출력
-c	결과물을 콜론으로 분리
-g	그룹 할당량만 출력
-u	사용자 할당량만 출력

-v	사용량이 전혀없는 경우를 포함하여 모든 쿼터내용 출력
-l	긴이름도 출력
quotaon / quotaoff	
파일 시스템의 쿼터 기능 활성화	
-a	전체 파일시스템 쿼터 활성화/비활성화
-u	사용자 쿼터 활성화/비활성화
-g	그룹 쿼터 활성화/비활성화
-v	명령 진행 상황 상세 출력
권한관리 명령어	
umask	
▶ 파일, 디렉토리 default 권한 지정	
▶ 파일 : 666, 디렉토리 :777	
:umask [-S]	
-S	umask를 u/g/o 및 r/w/x로 표시
chmod	
▶ 파일 및 디렉토리에 권한 부여	
▶ setuid = 4, setgid = 2, stickybit = 1	
▶ sticky bit는 디렉토리에만 효과	
▶ t는 디렉토리 실행가능, T는 실행불가	
▶ 디렉토리 r는 디렉토리 내부 내용 볼 수 있음	
▶ 디렉토리 w는 내부 파일 생성/삭제 가능	
▶ 디렉토리 x는 내부로 접근할 수 있음	
▶ sticky bit 적용된 대표적인 디렉토리 : /tmp	
▶ set-Bit가 적용된 파일 : /usr/bin/passwd	
: chmod [숫자] [파일 디렉토리]	
: chmod [u g o] [+ - =] [r w x t] [파일 디렉토리]	
chown	
▶ 파일에 대한 사용자, 그룹 변경	
: chown [opt] [UID:GID] [디렉 파일]	
-R	하위 디렉토리 모든 권한 변경

-c	사용자, 그룹 변경 결과 출력
chgrp	
▶ 파일이나 디렉토리 소유그룹 변경	
: 총게 [opt] [그룹파일]	
-R	하위 디렉토리 역시 변경
-c	변경 결과 출력
-h	심볼릭 링크 자체의 그룹 변경

사용자 관련 명령어	
chsh	
▶ 일반사용자 환경에서 셸 변경시 사용	
: chsh [opt] [shell]	
-s shell	로그인 셸 변경
-l	사용가능한 셸 목록 표시
chfn	
▶ 사용자의 정보를 변경하는 명령어	
usermod	

환경변수/히스토리 관련 명령어	
env	
▶ (글로벌)환경변수를 보여주거나, 설정 혹은 삭제하는 명령	
▶ 진짜 환경변수	
: env [opt] [name=값] [명령 [인수]]	
: printenv	
-i	현재 환경 무시하고 지정변수 값 따름
-u	지정한 변수(name) 제거
set	
▶ (로컬)환경변수를 조회하는 리눅스 명령어	
▶ 정확히는 쉘 변수를 조회	
: set [opt] [name=값]	
export	
▶ 쉘 변수를 환경변수로 저장할 수 있음	
: NAME=VALUE	

: export [NAME]	
history	
▶ 사용자가 입력한 명령어 이력 출력	
▶ !! : 마지막에 사용한 명령실행	
▶ !n : 최근 명령어부터 n번째 명령 실행	
▶ !a : 가장 최근에 사용한 a로 시작하는 명령 실행	
▶ !?a? : 가장 최근에 사용한 a를 포함하는 명령 실행	
: history [n] (최근 실행한 n개의 명령어 실행)	
last	
▶ 사용자가 언제, 어떤 ip로 접속했는지 확인	
▶ 관련파일	
- /var/log/wtmp : 모든 계정사용자 접속정보 기록	
: last [opt] [유저명]	

소스파일 관련 명령어	
make	
▶ 파일 관리 유틸리티	
▶ 파일 간의 종속관계를 파악하여 Makefile에 적힌 대로 컴파일러에 명령하여 SHELL 명령이 순차적으로 실행될 수 있게 하는 명령어	
install	make를 통해 만들어진 설치파일을 설치
clean	컴파일되었던 .o를 제거
all	make clean과 make를 같이해줌
dep	컴파일 이전에 의존성 검사

장치 관련 명령어	
lspci	
▶ 메인보드 주변 장치를 확인	
▶ 주로 설치된 RAID카드, HBA 카드 종류 확인을 위해 사용	
▶ 설치시 yum install pciutils	
-t	장치 번호를 트리로 보여줌

-vv	제조사, 모델, 비디오카드 등 자세히
lp	
▶ 프린터 작업요청 : lp -n 매수 (lpr은 -# 매수임)	
lpr	
▶ 프린터 작업요청 : lpr [opt] [파일명] >> lpr -# 3 -P lp ihd.txt	
▶ lp 명령어는 다음과 같이 수행함 >> lp -d 3 ...	
-# 값	인쇄할 매수 지정(1-100)
-P 프린터명	프린터 지정
lpc	
▶ 라인 프린트 컨트롤 프로그램	
cancel	
▶ 프린터 작업 삭제 : cancel [-a]	
lprm	
▶ 프린터 큐에 대기중인 작업 삭제 : lprm – 로 모든 인쇄작업 취소	
scanimage	
▶ 이미지를 스캔하는 명령	
-d	SANE의 장치 파일명을 적는 옵션
--format	이미지 형식 지정(pnm, tiff) 기본적으로 pnm 형식으로 지정
-L	사용 가능한 스캐너 목록 출력
scanadf	
▶ 자동 문서 공급 장치가 장착된 스캐너에서 여러 사진을 스캔할 때 사용	
--format	이미지 형식 지정(pnm, tiff) 기본적으로 pnm 형식으로 지정
-L	가능한 스캐너 장치 목록 출력
-d	SANE 장치 파일명 적는 옵션
alsactl	

▶ ALSA 사운드 드라이버에 대한 향상된 제어 명령	
-E	환경 변수 설정(--env)
-f	환경 설정 파일 지정 (디폴트 : /etc/asound.state)
-i	init을 위한 설정 파일 지정(--initfile) (디폴트 : /PREFIX/share/alsa/init/00main)
alsamixer	
▶ ALSA 사운드 카드 오디오 믹서 프로그램 ▶ 커서(ncurse) 라이브러리 기반	
cdparanoia	
▶ 오디오CD에서 음악파일 추출 가능(기본형식 : wav)	
sane-find-scanner	
▶ SCSI 및 USB 스캐너 디바이스 파일을 찾음 ▶ SCSI 스캐너 : /dev/sg0, /dev/scanner ▶ USB 스캐너:/dev/usb/scanner, /deb/usbscanner :sane-find-scanner [opt] [장치명]	
-p	병렬 포트에 연결된 스캐너 찾기
-q	스캐너 장치만 출력
-v	상세한 정보 출력

기타 명령어	
nm-connection-editor	GUI 기반 네트워크 설정
system-config-network	TUI 기반 네트워크 설정
xcam	GUI 기반 평판 스캐너나 카메라로부터 이미지 스캔

프로토콜 및 시스템 정리	
snmp(Simple Network Management Protocol)	
<ul style="list-style-type: none"> ▶ TCP/IP 프로토콜을 사용하는 인터넷에서 장치들을 관리하기 위한 프로토콜 ▶ UDP 프로토콜 사용 	
NFS	
<ul style="list-style-type: none"> ▶ 1984년 썬 마이크로시스템즈에서 개발한 네트워크 공유 기술 ▶ 네트워크 상 다른 컴퓨터 하드디스크를 내 하드디스크처럼 사용(RPC 기반) ▶ 공유하고자하는 서버에서 nfs 데몬 활성화 이후 디렉토리 설정. ▶ 상대방은 IP or 도메인 주소와 함께 mount 설정 ▶ Window와의 파일 시스템 공유의 경우 SAMBA를 많이 씀 ▶ 관계 프로토콜 : RPC, NIS ▶ 관계 서비스 : rpcbind, portmap 	
NIS	
<ul style="list-style-type: none"> ▶ 하나의 서버에 등록된 사용자 계정, 암호, 그룹정보 등을 공유하며 다른 시스템에 제공하는 서비스 ▶ 명령 실행, 파일 복사 등의 작업이 가능함 	
SAMBA	
<ul style="list-style-type: none"> ▶ 리눅스와 윈도우 간에 파일 및 프린터를 공유할 수 있게 해주는 프로그램. ▶ SMB / CIFS 프로토콜을 이용함으로써 가능함 ▶ 마운트 시 cifs 유형으로 지정 ▶ 인증을 통해 접근하므로 NFS에 비해 보안성 높음 	
SMB(Server Message Block)	
<ul style="list-style-type: none"> ▶ 마이크로소프트 사와 인텔이 윈도우 시스템이 다른 시스템의 디스크나 프린터와 같은 자원을 공유 할 수 있도록 개발됨. ▶ NFS, NIS, lpd와 같은 유닉스의 분산인증구조와 유사 	
프로토콜	TCP/IP, NetBIOS
SMTP	Simple Mail Transfer Protocol
CIFS(Common Internet File System)	
SMB 파일 공유 프로토콜의 확장 버전 윈도우와 유닉스 환경을 동시에 지원하는 인터넷	

표준 파일 규약	
프린팅 시스템 명령어 계열	
<ul style="list-style-type: none"> ▶ BSD 계열과 System V 계열로 나뉨 	
BSD	lpr, lpq, lprm, lpc
System V	lp, lpstat, cancel
CUPS(Common Unix Printing System)	
<ul style="list-style-type: none"> ▶ 애플이 개발한 오픈소스 ▶ 컴퓨터를 인쇄 서버로 기능하도록 해주는 유닉스 계열 os를 위한 프린팅시스템 ▶ HTTP 기반 IPP 프로토콜 사용, SMB 부분 지원 ▶ BSD, System V 계열 명령어 모두 사용 가능 ▶ 사용자 및 호스트 기반의 인증 제공 ▶ 관련 파일 <ul style="list-style-type: none"> - /etc/cups/cupsd.conf : 프린터 데몬 환경설정 - /etc/cups/printers.conf : 프린터 큐 환경설정 - /etc/cups/classes.conf : 프린터 데몬 클래스 설정 	
LPRng	
<ul style="list-style-type: none"> ▶ 버클리 프린팅 시스템 ▶ 리눅스에서 프린터를 지원해주는 초기 인쇄 시스템 ▶ 라인 프린터 데몬 프로토콜을 사용하여 프린터 스플링 및 네트워크 프린터 서버 지원 	
OSS(Open Sound System)	
<ul style="list-style-type: none"> ▶ 1992 Hannu Savolainen에 의해 개발 ▶ 2007.7 Linux의 GPL기반 라이선스로 공개 ▶ 2008.4 BSD라이선스 기반 소스 추가 공개 ▶ 리눅스 및 유닉스 os에서 사운드를 만들고 캡처하는 인터페이스 ▶ 표준 유닉스 장치(POSIX READ, WRITE, IOCTL 등)에 기반을 둠 ▶ 현재 리눅스 커뮤니티에서 ALSA로 대체 	
ALSA	
<ul style="list-style-type: none"> ▶ 사운드카드용 장치 드라이버 제공하기 위한 리눅스 커널 요소 ▶ 1988 Jaroslav Kysela에 의해 시작 ▶ GPL, LGPL 라이선스 기반으로 배포 	
SANE(Scanner Access Now Easy)	
<ul style="list-style-type: none"> ▶ 평판, 핸드스캐너, 비디오 캠 등 이미지 관련 하드 	

워어를 제어하는 API
▶ GPL 라이선스(오픈소스)
▶ 리눅스, 유닉스, OS2, Window 지원
XSANE(X based interface for the SANE)
▶ X Window 기반 스캐너 프로그램
▶ GTK+ 라이브러리로 개발
WWW(World-wide web)
▶ 멀티미디어와 하이퍼텍스트
GOPHER
▶ 인터넷 기반의 문서 검색 시스템
▶ 텍스트 기반의 메뉴 인터페이스 제공
▶ 터미널 환경 기반으로 대량 정보 검색 및 저장 위해 사용됨.
▶ 이후 www로 대체됨

nfs
▶ Network File System으로 원격서버 마운트시 사용
▶ Unix <-> Linux
ufs
▶ UNIX File System으로 UNIX System 5 표준파일시스템
sysv
▶ 유닉스 시스템 V 지원하기 위한 파일시스템
vfat, msdos, hpfs, ntfs, hfs
▶ fat : 윈도우 NT 지원하는 가장 간단한 파일시스템 (fat12, fat16, vfat, fat32, exfat)
▶ vfat : 윈도우 95.98.NT 지원 위한 파일시스템
▶ msdos : MS-DOS 파티션을 위한 파일시스템
▶ hpfs : HPFS 지원을 위한 파일시스템
▶ ntfs : 윈도우 NT, 2000 ntfs 지원 위한 파일시스템
▶ hfs : MAC 루 파일시스템 지원을 위한 파일시스템
udf(Universal Disk Format)
▶ 광 디스크 파일 저장을 위한 파일 시스템 규격
nfs(Network File System)
▶ NFS 서버에서 공유한 파일 시스템을 NFS클라이언트에서 마운트하여 사용하는 시스템
기타 네트워크 장치명
▶ 도커 기반 가상화 장치명 : docker0
▶ 가상화 장치명 : virbr0

파일 시스템 정리
ext2(extended file system 2)
▶ 단일파일 크기 2GB
▶ 파일명 256B
▶ 최대 파일시스템 크기 4TB
ext3(extended file system 3)
▶ 단일 파일 크기 4GB
▶ 파일명 256B
▶ 최대 파일 시스템 크기 16TB
▶ 저널링 파일 시스템
ext4(extended file system 4)
▶ 단일 파일 크기 16TB
▶ 최대 파일 시스템 크기 1EB
▶ ext2, 3 호환가능, 마운트 가능
XFS
▶ SGI가 만든 저널링 파일 시스템.
▶ 최대 파일 시스템 크기 8EB
▶ ReiserFS, JFS 역시 저널링 파일 시스템
iso9660
▶ DVD, CD-ROM 데이터 읽을 때 마운트시 사용하는 파일시스템. Read Only로 사용

셀 정리
셀 발전순서
▶ Borne shell : /bin/sh (POSIX shell)
▶ C shell : /bin/csh (Sun microsystem, BSD)
▶ Korn shell : /bin/ksh (유료)
▶ bash : /bin/bash
▶ zsh : /bin/zsh
배수 관련 파일
▶ /etc/profile : 모든사용자 로그인시 적용되는 스크립트

립트 정의

- ▶ /etc/bashrc : 모든 사용자에게 적용되는 bash 쉘 스크립트
- ▶ ~./bash_logout : 사용자 로그아웃시 실행명령저장
- ▶ ~./bashrc : run command 약자. 해당 사용자에게 적용되는 bash 쉘 스크립트
- ▶ ~./bash_profile : 경로 추가 등 사용자가 정의하는 환경변수들을 정의

셀 프롬프트 모양

₩h	호스트명
₩t	시간 24시 표기법
₩T	시간 12시 표기법
₩d	요일/월/일 형식
₩u	현재 사용자명 표시
₩w	절대 경로
₩W	상대 경로

환경변수

TERM	터미널 종류의 이름
LANG	프로그램 사용시 기본 지원 언어
DISPLAY	x-window 시스템에서 xterm과 같은 x 응용 프로그램이 화면 출력을 위해 접속할 때 x 서버 주소
PS1	SHELL 프롬프트 선언시 사용하는 변수
PS2	2차 SHELL 프롬프트 선언시 사용 변수

현재 사용중인 쉘 확인

- ▶ echo \$SHELL
- ▶ ps \$\$
- ▶ echo \$\$

bourne shell

tcsh

- ▶ 1975 켄 그리어가 테넥스 운영체제에 명령행 완

성 기능을 반영하면서 시작

- ▶ 명령행 편집 기능 추가 지원

C shell(csh)

- ▶ 빌 조이에 의해 제작
- ▶ c언어의 특징을 지님
- ▶ 히스토리, 별명, 작업제어 지능
- ▶ 다양한 프로그래밍 작성 가능
- ▶ 명령행 편집기능, 완성 기능 지원 x

Korn shell

- ▶ 데이비드 콘에 의해 제작
- ▶ 벨 연구소에서 본 셸을 확장해서 만듬
- ▶ 명령어 완성 기능, 히스토리

Bash

- ▶ 브라이언 폭스가 제작
- ▶ GNU 프로젝트에 의해 개발
- ▶ POSIX와 호환
- ▶ 기능
 - 명령어 완성
 - 명령어 히스토리
 - 명령행 편집
 - 키 조합 설정
 - 향상된 패턴 검색
 - 프롬프트 서식 설정

에디터 정리

vi

- ▶ 빌 조이가 개발함
- ▶ 문법 강조 기능
- ▶ 자동 들여쓰기 가능
- ▶ 문자열 치환 가능

- ▶ 관련 파일

.exrc : vi 실행시 자동으로 명령 실행

모드

- 입력 모드 : 문서를 편집하는 모드. 글자를 입력함
- 명령 모드 : 키입력을 통해 명령을 내리는 모드. 커서 이동, 삭제, 복사, 붙이기 등
- ex 모드 : 치환, 저장, 종료 등의 기능을 수행하는 모드

▶ 치환

- [% | 범위]s/[**₩<**]”원본”[**₩>**]”치환”/[gci]
- ▶ 파일 열면서 n번째로 커서 위치 : vi +n 파일명
- ▶ 약어
- :ab [약어] [문자열]
- ▶ 키매팅
- :map : Normal, Visual, Op-pending mode
- :vmap : Visual
- :nmap : Normal
- :omap : Operation-pending
- :map! : Insert, Command-line
- :imap : Insert
- :cmap : Command-line

o	[명령모드] 현재 커서 다음 줄로 입력모드(아래)
O	[명령모드] 현재 커서 이전 줄로 입력모드(위)
s	[명령모드] 커서 위치에 한 문자 지우고 입력모드
S	[명령모드] 커서가 위치한 줄 지우고 입력모드
x	[명령모드] 커서에 위치한 문자 삭제
p	[명령모드] 복사 내용 현재 라인 이후 에 붙여넣기
u	[명령모드] 되돌리기(undo)
dd	현재 커서 행 삭제
yy	현재 커서 라인 복사
:set nu	[EX모드] 행 번호 표시 (=:set number)
:set ic	[EX모드] 대소문자 무시(ignore case)
:set ai	[EX모드] 자동 들여쓰기
ctrl +f	커서가 위치한 부분부터 한 화면 아래 로

- ▶ vi + 파일명 : 파일의 마지막 행으로 이동
- ▶ vi -r 파일명 : 파일 복구 후 잘못된 동작 이후부
터
- ▶ vi -c 커맨드 : ex) vi -c set nu

vim

- ▶ 브람 무레나르가 만든 편집기
- ▶ 문법 강조 기능
- ▶ 문법 검사 기능
- ▶ 패턴 검색 하이라이트 기능
- ▶ 다중 되돌리기 기능

-r 파일명 편집 중이던 파일로 복구

gVim

- ▶ GUI 기반 에디터

pico

- ▶ 워싱턴 대학에서 aboil kasar가 개발한 유닉스용 텍스트 에디터
- ▶ Pine이라는 E-mail 클라 프로그램과 같이 배포됨
- ▶ 2005년 개발 중단으로 리눅스 배포판에 설치 힘듬
- ▶ 아파치 라이센스
- ▶ 단락 정의
- ▶ 맞춤법 검사
- ▶ 복사/붙여넣기

alt + d 커서가 위치한 부분부터 단어 삭제

alt + k 커서가 위치한 부분부터 문장전체 삭제

ctrl + k 현재 줄을 삭제

ctrl + g 도움말 출력

ctrl + x 프로그램 종료

ctrl + o 편집중인 문서 저장

ctrl + j 상위 줄과 결합하여 한 문단으로 만듬

ctrl + p 현재 커서를 한 줄 위로(previous)

ctrl + n 현재 커서를 한 줄 아래로(next)

ctrl + b 현재 커서를 뒤(왼쪽)으로(back)

ctrl + f 현재 커서를 앞(오른쪽)으로(front)

ctrl + a 커서가 위치한 줄의 맨 앞으로 이동

ctrl + e 커서가 위치한 줄의 맨 뒤으로 이동

nano

- ▶ 자동 들여쓰기 가능
- ▶ GPL 라이선스

emacs(editor macros)	
<ul style="list-style-type: none"> ▶ 리처드 스톤만이 제작 ▶ 매크로 기능이 있는 텍스트 교정 및 편집기 ▶ LISP 언어 	
커서 이동키가 pico와 같음	
ctrl + d	커서 위치의 글 삭제
ctrl + k	현재 라인 커서 뒤부터 모두 삭제
ctrl + x , f	파일 불러오기, 새 파일 열기
ctrl + x, c	종료
GNU emacs	
<ul style="list-style-type: none"> ▶ 리처드 스톤만이 'emacs LISP'라는 LISP에 기반을 두는 환경 설정 언어를 가지고 처음부터 다시 만든 버전 	
XEmacs	
<ul style="list-style-type: none"> ▶ GUI 기반 에디터 	
gedit	
<ul style="list-style-type: none"> ▶ GUI 기반 에디터 	
grep	
^, \$	문자열 행의 시작과 끝
{n,m}	문자열 n회 이상 m회 이하 일치

소스철치 정리	
소스파일 설치 순서	
<ol style="list-style-type: none"> 1. 압축해제(tar/unzip/gunzip/bunzip2/unxz등) 	
<ol style="list-style-type: none"> 2. 소스코드 컴파일 위한 설정(환경 설정단계) <ul style="list-style-type: none"> ▶ ./configure [opt] ▶ 이 때, Makefile이 생성됨. 	
<ol style="list-style-type: none"> 3. 컴파일 <ul style="list-style-type: none"> ▶ make [Makefile명] ▶ 타겟과 의존성 관련 작업 수행후 실행파일 생성 ▶ 이 작업이 완료되면 소스파일->setup파일로 전환 	
<ol style="list-style-type: none"> 4. 파일 설치 <ul style="list-style-type: none"> ▶ make install ▶ root권한 필요 ▶ 컴파일된 실행파일을 지정된 속성으로 지정된 디 	

렉토리에 설치
cmake
<ul style="list-style-type: none"> ▶ 크로스 플랫폼 빌드 지원 ▶ Microsoft .NET 지원 ▶ cmake 설치 과정 <ul style="list-style-type: none"> : cmake – make install (configure 과정 생략)
소스 파일 관련 프레임워크

패키지 정리
플랫폼별 패키지 관리
<ul style="list-style-type: none"> ▶ SUSE : yaST, 16zypper ▶ Debian : dpkg, apt, alien, dselect, aptitude, synaptic ▶ Redhat : rpm, yum
yum
<ul style="list-style-type: none"> ▶ 저장소 기반 패키지 관리
apt-get
<ul style="list-style-type: none"> ▶ 저장소 기반 패키지 관리
yaST
zypper
<ul style="list-style-type: none"> ▶ 저장소 기반 패키지 관리

X-window 정리
X-window System
<ul style="list-style-type: none"> ▶ 시스템 역사 : XFree86 – X.org Server - Wayland ▶ Bob Scheifler가 1986년 오픈소스 프로젝트로 ▶ X Protocol 사용 ▶ MIT 라이선스
<ul style="list-style-type: none"> ▶ 구성 <ul style="list-style-type: none"> - 윈도우 매니저 : 도크(Dock), 태스크바(Task bar),

프로그램 런처(Program launcher), 데스크톱 아이콘(Desktop Icon), 바탕화면(Wallpaper)와 같은 다양한 유틸리티 제공

- 데스크톱 환경 : GUI 환경을 이용하기 위해 사용자에게 제공되는 인터페이스 스타일. 아이콘, 창, 도구모음, 폴더, 배경화면 등도 제공
- 디스플레이 매니저 : XDM, GDM, KDM 등이 존재. 부팅시에 X윈도가 실행된 후 디스플레이 매니저가 실행되어 로그인 창이 나타나며, 로그인 이후 세션을 시작해주는 역할 수행

▶ 관련 파일

- /etc/inittab : 리눅스 부팅모드 설정(Run level 5로)

▶ XFree86

- IBM 호환 시스템을 사용하는 유닉스 계열 os를 위한 X 윈도 프로젝트로 1992년에 시작
- 초기에는 인텔 계열 플랫폼으로 개발되었으나 추후에 Macintosh, Alpha, SPARC 등으로 확대

▶ X.org

- X.Org 재단에 의해 구성된 X윈도 시스템용 디스플레이 서버 구현체
- 클라이언트 사이드 구현체로 Xlib 또는 XCB 사용 가능
- 현재 최신 버전X11R.7.7(릴리즈 1.20)
- ▶ Wayland
 - X.Org 재단 개발자들이 만든 디스플레이 기반의 서버 이름

▶ Xlib :

- C언어로 구현된 라이브러리로 X서버와 대화 가능하도록 해줌.
- 개발자가 자세히 알지 못해도 쉽게 사용 가능
- 저수준 인터페이스로 키보드, 마우스에 대한 반응 등의 단순한 기능만 가짐

▶ XCB : (X protocol C-language Binding)

- X 서버에 가까운 가장 저수준 X 클라이언트 라이브러리.
- Xlib를 대체하며 크기가 작고 단순하며 확장성이 뛰어남. 향상된 스레딩 기능을 지원

▶ X toolkit :

- 상위 라이브러리
- Xt Intrinsics, Xaw(Athena widget set), Motif, FLTK,

GTK+, Qt, Tk, SDL(Simple DirectMedia Layer) 등

데스크톱 환경 프로그램

Aqua	Apple의 데스크톱 환경
GNOME	
plasma	과거 KDE에서 사용된 데스크톱 환경
KDE	
LXDE	X 원도우에 사용되는 데스크톱 환경
XFce	KDE나 Gnome에 비해 유저가 적은 편

윈도우 매니저 프로그램

▶ 직접 비디오 카드, 마우스, 키보드에 접근하지 않고 디스플레이 서버를 통해서 접근
▶ GNOME에서 Mutter(GNOME 3ver) 및 Metacity(GNOME 2ver)가 있음.

mutter	GNOME 3 윈도우 매니저
metacity	GNOME 2 윈도우 매니저
kWin	KDE 윈도우 매니저
Window makek	깔끔한 인터페이스와 사용하기 편리함
twm	(Tab Window Manager) 텍스트 기반 간단한 윈도우 매니저
fwm	KDE 기본 윈도우 매니저
fvwm	(Feline Virtual Window Manager) twm에서 파생된 것으로 안정적이고 일반화되어 오랜 기간 많이 사용된 프로그램
Afterstep	fvwm을 기반 제작된 윈도우 매니저 넥스트스텝 gui와 유사한 UI 제공
Blackbox	넥스트스텝의 인터페이스 기반으로 한 윈도 매니저
Enlightenment	규칙에 따라 기능부분에서 겉모습까지 세밀한 부분까지 설정 가능. GNOME 기본 윈도우 매니저

프로그램

nautilus	파일 관리 프로그램
eog	이미지 뷰어 프로그램
GIMP	이미지 편집, 변환, 생성 프로그램
gThumb	이미지 뷰어, 편집 프로그램
Cheese	웹캠 사용한 사진, 비디오 추출 프로그램
Totem	Movie Player (미디어 플레이어)
evolution	GNOME 데스크톱의 일부 개인 정보 관리(이메일, 주소, 일정, 작업 목록 등)
konqueror	웹 브라우저 및 파일 관리 시스템
Dolphin	파일 관리자
Okular	문서 뷰어 프로그램
kwrite	텍스트 편집기
Gwenview	이미지 뷰어
KSnapshot	스크린 캡처 프로그램
KMid	KDE4 미디어 플레이어
Dragon Player	미디어 플레이어
KGet	다운로드 관리자
Krfb	원격 접속 프로그램
KMail	그룹웨어 Kontact의 구성 요소, 이메일 클라이언트 프로그램
Image MagicK	비트맵 이미지를 보여주고 생성 및 편집이 가능하도록 지원해줌
thunderbird	모질라 재단에서 개발한 E-mail 및 뉴스그룹 클라이언트 프로그램
GNOME(GNU Network Object Model Environment)	
<ul style="list-style-type: none"> ▷ 컴퓨터를 쉽게 사용할 수 있게 하는 데스크톱 환경 ▷ 전용 윈도우 관리자가 없으며 선택하여 사용 ▷ 라이선스 정책 <ul style="list-style-type: none"> - GPL : 라이브러리/프로그램 모두 적용 - LGPL : 라이브러리/데스크톱에 적용 	

그래픽라이브러리	GTK+
설정항목 수	필요한 것만
텍스트에디터	gedit
기본 브라우저	Web
토렌트 클라	Transmission
파일 탐색기	Nautilus
KDE	
그래픽라이브러리	Qt
설정항목 수	최대한 많이
텍스트에디터	kate
기본 브라우저	Konquerer
토렌트 클라	kTorrent
파일 탐색기	Konquerer
X-Window 기반 명령어	
system-config-printer	X-Windows 환경에서 프린터 설정
xhost	X server에서 접근 가능한 IP 주소 및 호스트명 확인 :xhost + ip로 허용 가능
xauth	>> xauth list \$DISPLAY >> 결과 : www/unix:0 MIT-MAGIC-COOKIE(MMC) b989... ipv6같이 16비트
oowriter	워드프로세서 프로그램 LibreOffice Writer 실행
ooimpress	프레젠테이션 프로그램 LibreOffice Impress 실행
oocalc	스프레드시트 프로그램 LibreOffice Calc 실행
oodraw	드로잉 프로그램

LibreOffice Draw 실행

네트워크 관련	
물리	
CSMA/CD	단말기가 전송로의 신호유모 조사 및 다른 단말기가 신호 송출하고 있는지 확인
FDDI	전송매체를 광섬유 케이블을 사용해 링 구조의 통신망, ISO 규격으로 승인
DQDB	<ul style="list-style-type: none"> - IEEE 제정한 MAN 표준 규격 IEEE 802.6에 채용된 다중 접속 프로토콜 - 버스 방식 LAN형 프로토콜 - 회선/패킷 교환 모두 가능, 데이터/음성/비디오 전송 지원
ATM	<ul style="list-style-type: none"> - 비동기 전송모드 약어 - 전송로 1개를 여러 대 단말이 공용해 통신 가능. - 전화선으로 음성, 데이터, 동영상 동시에 보낼 수 있음
X.25	<ul style="list-style-type: none"> - 회선 종단장치(DCE) 및 데이터 단말장치(DTE) 사이에 이루어지는 상호작용 규정 - 물리, 데이터링크, 네트워크까지 규정 - X.25에서 사용하는 프레임을 간소화하여 성능을 향상시킨 서비스를 프레임 릴레이(Frame relay)라고 함

ITU(International Telecommunication Union)

- ▶ 국제 전기통신 연합
- ▶ ITU-T(전기통신 및 관련 표준화)
- ▶ ITU-R(무선 전파 통신)
- ▶ ITU-D(정보통신 개발 부문) : 인프라, 보안 등
- ▶ 5G 제정

IEEE(Institute of Electrical and Electronics Engineers)

- ▶ 전기전자 기술자 협회
- ▶ DQDB 제정

IANA(Internet Assigned Numbers Authority)

- ▶ 인터넷 할당 번호 관리기관
- ▶ IP주소 및 포트번호 관련 제정

ICANN(Internet Corporation of Assigned Names and Numbers)

- ▶ 인터넷 주소 관리 기구
- ▶ IP 및 도메인 관리하는 국제기관
- ▶ 국내기관은 : KISA

국내 기관

- ▶ KISA(Korea Internet & Security Agency)
: 한국인터넷진흥원
- ▶ NIPA(National IT Industry Promotion Agency)
: 정보통신산업진흥원

가상화 관련

VirtualBox

- ▶ GNU GPL 라이선스
- ▶ VMware의 VMDK 이미지 지원
- ▶ Microsoft Virtual PC 이미지 VHD 지원

서버 가상화 종류

VMware	VSphere
Redhat	XVM 기반의 RHEV
Microsoft	Hyper-V 기반의 Windows Server

리눅스 가상화

- | | |
|-----|--|
| XEN | <ul style="list-style-type: none"> ▶ CPU 전/반가상화 모두 지원 ▶ 물리적 서버 대비 효율성 좋음 |
|-----|--|

	<ul style="list-style-type: none"> ▶ 베어메탈방식의 하이퍼바이저 ▶ 전가상화 구성시 QEMU기반 동작 ▶ 반가상화 구성시 호스트와 다른 아키텍처 게스트 실행 불가
KVM	<ul style="list-style-type: none"> ▶ CPU 전가상화만 지원 ▶ 이더넷카드, DISK I/O, VGA 등은 반 가상화도 지원 ▶ 상용제품 : RHEV
Docker	<ul style="list-style-type: none"> ▶ 경량화된 가상화 기술 ▶ 하이퍼바이저나 게스트OS 없이 서버 운영에 필요한 프로그램과 라이브러리만 격리해서 설치 및 운용 가능
OpenStack	<ul style="list-style-type: none"> XEN, KVN 등으로 나눠서 운영되는 가상화 환경 통합 관리에 유용 ▶ CloudStack, OpenNebula가 비슷함

<ul style="list-style-type: none"> - 사용자명:패스워드:uid:gid:이름:홈디렉토리:셀 ▶ /etc/shadow - 사용자명:패스워드:최종수정일 :변경최소일:변경최대일:만료경고기간 :파기기간:계정만료기간:예약필드 ▶ /etc/shell - 사용 가능한 셸 목록 표시
디스크 파일
<ul style="list-style-type: none"> ▶ /etc/inittab - init 데몬이 이 정보를 읽고 프로세스 시작 - id : rstate : action : process ▶ /etc/fstab - 부팅 시에 마운트 정보에 대한 정의 ▶ /etc/mtab - 현재 마운트된 정보를 담음

디렉토리 관련 정리	
네트워크 파일	
▶ /etc/sysconfig/network :	<ul style="list-style-type: none"> - 네트워크 사용유무 지정, 호스트명 지정, 게이트웨이주소 설정, 게이트웨이장치파일 설정, 도메인 이름 등이 기록
▶ /etc/sysconfig/network-scripts :	<ul style="list-style-type: none"> - 네트워크 인터페이스 환경설정 관련 파일들
▶ /etc/resolv.conf	<ul style="list-style-type: none"> - 질의할 DNS 서버 등록
▶ /etc/hosts	<ul style="list-style-type: none"> - IP, 호스트명, 도메인네임 맵핑
▶ /etc/host.conf	<ul style="list-style-type: none"> - 도메인 요청 시 검색 순서
▶ /etc/protocols	<ul style="list-style-type: none"> - 서비스 가능한 프로토콜 목록이 정의된 파일
▶ /etc/services	<ul style="list-style-type: none"> - 네트워크 서비스를 위한 모든 포트 목록
유저 파일	
▶ /usr/bin/passwd	<ul style="list-style-type: none"> - 계정 비밀번호 변경할 수 있는 실행파일
▶ /etc/passwd	

기타	
브라우저	
크롬	<ul style="list-style-type: none"> 웹카트 레이아웃 엔진으로 개발 최근에는 웹카트에서 포크된 블링크 사용
파이어폭스	게코 레이아웃 엔진
오페라	프레스토 레이아웃 엔진
사파리	웹카트 레이아웃 엔진으로 개발
메일 관련 프로토콜	
POP3	<ul style="list-style-type: none"> ▶ Post Office Protocol version ▶ 유저가 서버에서 메일을 받아오는 프로토콜 중 하나 ▶ 메일서버에 설정한 계정에 온 메일이 있는지 사용자 프로그램에서 확인 후 메일을 가져옴 ▶ 메일이 로컬 저장소에 저장됨(서버에서는 삭제된다고 하지만 요즘은 바로 삭제되지 않음) ▶ 메일서버 용량이 적을 때 사용
IMAP	<ul style="list-style-type: none"> ▶ Internet Message Access Protocol ▶ 사용자가 메일서버에서 메일 내려받는 방식 중 하나 ▶ 중앙 서버 동기화로 모든 장치에서

	<p>동일한 이메일 폴더 확인 가능 ▶ 메일을 가져가도 서버에 계속 남겨둠</p>
SMTP	<p>▶ Simple Mail Transfer Protocol ▶ 일반적으로 메일 서버간의 메시지 교환 위해 사용</p>
최상위 도메인 종류	
<p>▶ com, net, org, edu, gov, mil, int</p>	
잘 알려진 포트 번호	
ftp-data	20
ftp	21
ssh	22
telnet	23
dns	53
http	80
POP3	110
IMAP	143
HTTPS	443
클러스터	
고가용성	<p>하나의 작업을 여러 개로 구성된 노드들을 이용해서 처리 (Backup에 대한 내용 나오면 고가용성) HA(Higher Availability)</p>
고계산용	<p>고성능의 계산능력 제공을 위한 슈퍼 컴퓨터 구성에 주로 사용 HPC(High Performance Computing) GNU C Compiler PVM(Parallel Virtual Machine) MPI(Message Passing Interface)</p>
부하분산	<p>여러 개의 컴퓨터 자원들에 작업을 나누는 것. 가용성 및 응답시간을 최적화 LVS(Linux Virtual Server)</p>
베어울프	<p>다수 컴퓨터를 하나로 만든 후 병렬 프로그래밍을 통해 고성능 수치 연산 시스템 구축(슈퍼컴퓨터)</p>

채널 본딩	호스트 컴퓨터에 2개 이상 네트워크 인터페이스를 장착한 후에 안정성, 전송속도를 높이기 위해 구성하는 기술
OS별 기반	
QNX	유닉스 계열 서버 OS
webOS	리눅스 커널 기반 모바일 OS
Tizen	리눅스 커널 기반 OS
MeeGo	
Moblin	
ios	애플이 제작한 임베디드 OS
BlackBerry	Research in Motion에 의해 만들어진 OS
네트워크 프린트 설정 가능 환경	
<p>▶ 인터넷 프린터 프로토콜(ipp) ▶ 인터넷 프린터 프로토콜(http) ▶ AppSocket/HP jecDirect ▶ LPD/LPR 호스트 또는 프린터 ▶ Windows Printer via SAMBA</p>	
Run Level	
<p>0 : PROM 검사 단계 1 : 관리상태 단계, 단일 사용자 단계로 여러 fs 로드 2 : 공유 자원 가지지 않은 다중 사용자 단계 3 : 기본 실행단계, 공유자원 가진 다중 사용자 단계, 텍스트 유지 모드 5 : 3단계 이후 그래픽모드(X-window) 실행 (x server의 접속 및 세션시작 담당) 6 : 재부팅 단계, 실행단계 3으로 재부팅</p>	
프로세스 우선순위	
<p>▶ 우선순위 변경가능 명령어 : nice, renice, top ▶ NI 기본값 0, -20 ~ 19까지 지정가능 ▶ 일반사용자는 NI 증가만 가능, root만 감소 가능 ▶ NI값 설정시 리눅스는 상황에 따라 PRI(Priority) 값을 변경하여 우선순위 조정</p>	
빅데이터	
<p>▶ 대량의 정형/비정형 데이터 집합에서 가치 추출 및 결과 분석</p>	

- ▶ 데이터마이닝, 기계학습, 자연어처리, 패턴인식 등
- ▶ 데이터를 유연하고 빠르게 처리하기 위해 NoSQL 활용
- ▶ 분석된 데이터 시각적으로 표현하기 위해 R 사용

데몬

standalone	
inetd	커널 2.4버전부터 xinetd 대체
xinetd	환경 설정 파일을 이용해 자체적으로 접근 제어 가능

클라우드 서비스

- ▶ IaaS(Infrastructure as a Service)
 - : 공급업체를 통해 사용자가 컴퓨팅 리소스를 이용 할 수 있는 클라우드 오퍼링

지식의 파편...

Athena	Amazon의 표준 SQL 사용해 아마존 DB에 저장된 데이터를 분석할 수 있는 대화식 쿼리 서비스
--------	---

T568A : 녹파(흰주)주(흰파)갈

T568B : 주파(흰녹)녹(흰파)갈

실질적으로 1,2,3,6(흰주,주,흰녹,녹)만 사용