

Section 1. 리눅스의 개요

리눅스의 특징 및 장단점

특징

- 오픈 소스 운영체제이다.
- 멀티유저(다중사용자), 멀티테스킹(다중작업) 운영체제이다.
- 다중스레드를 지원하는 네트워크 운영체제이다.
- 여러 종류의 파일시스템을 지원하는 운영체제이다.

장단점

- 리눅스는 유닉스와 완벽하게 호환 가능하다.
- 리눅스는 PC용 운영체제보다 안정적이다.
- 하드웨어 기능을 효과적으로 사용한다.
- 리눅스는 오픈 소스 운영체제이다.
- 공개 운영체제이기 때문에 문제점 발생 시 기술지원을 받기 어렵다.
- 한글 지원이 미흡하다.

리눅스 디렉터리 종류와 특징

디렉터리	저장 내용
/	파일 시스템이 있는 최상위 디렉터리로 루트 디렉터리라고 함 모든 디렉터리의 출발점인 동시에 다른 시스템과의 연결점이 되는 디렉터리
/boot	부트 디렉터리로 부팅 시 커널 이미지의 부팅 정보 저장 파일
/proc	시스템 정보 디렉터리이며 커널 기능을 제어하는 역할 현재 실행되는 프로세스와 실제로 사용되는 장치, 하드웨어 정보 저장
/lib	공유 라이브러리 디렉터리 커널 모듈 파일들과 프로그램 실행을 지원해 주는 라이브러리 저장
/bin	기본적인 명령어가 저장된 디렉터리 root 사용자와 일반 사용자가 함께 사용할 수 있는 명령의 디렉터리
/dev	시스템 디바이스 파일들을 저장하는 디렉터리 하드디스크 장치 파일, CD-ROM 장치파일 같은 파일 저장
/etc	시스템 환경 설정 파일 저장 디렉터리
/root	시스템 관리자용 홈 디렉터리

/sbin	관리자용 시스템 표준 명령 및 시스템 관리와 관련된 실행 명령어 저장
/usr	사용자 디렉터리로 사용자 데이터나 애플리케이션 저장
/home	사용자 계정 디렉터리로 계정들의 홈 디렉터리가 위치 일반 사용자들이 로그인 시 처음으로 위치하게 되는 디렉터리
/var	가변 자료 저장 디렉터리로 로그 파일이나 메일 데이터 저장
/tmp	각종 프로그램이나 프로세스 작업을 할 때 임시로 생성되는 파일 저장 모든 사용자에게 대해서 읽기와 쓰기가 허용 시티키 비트 설정으로 파일의 소유자만이 자신의 소유 파일을 지울 수 있음
/mnt	파일 시스템을 일시적으로 마운트 할 때 사용
/lost+found	결함이 있는 파일에 대한 정보가 저장되는 디렉터리

리눅스 배포판

종류

- **슬랙웨어 리눅스** : 배포판 가운데 가장 먼저 대중화된 배포판으로 1992년 패트릭 볼커딩에 의해 출시되었다.
- **데비안** : 데비안 프로젝트에서 만들어 배포하는 공개 운영체제로 GNU의 공식적인 후원을 받고있는 유일한 배포판이다.
- **우분투** : 데비안GNU/리눅스에 기초한 운영체제이다.
- **레드햇** : 미국의 레드햇사가 개발하던 리눅스 배포판이다.
- **RHEL** : 레드햇이 개발하여 판매하고 있는 상용 리눅스 배포판이다.
- **페도라** : 리눅스 커널에 기반한 운영체제와 레드햇의 후원과 개발 공동체의 지원 아래 개발된 배포판이다.
- **CentOS** : 업스트림 소스인 레드햇 엔터프라이즈 리눅스와 완벽하게 호환되는 무료 기업용 컴퓨팅 운영체제이다.
- **수세** : 독일에서 출시된 배포판으로 유럽에서 인기를 누리고 있다.

Section 2. 리눅스 역사

넘어감

Section 3. 리눅스 라이선스

종류

- **GNU** : GNU 는 유닉스가 아니다 (GNU's Not UNIX) 의 약자이다.
- 자유 소프트웨어 재단 : FSF(Free Software Foundation) 으로 1985년 리처드 스톨만이 설립한 재단이다.
- 오픈 소스 소프트웨어 : Open Source Software 1998 년 일부 커뮤니티에서 '자유 소프트웨어' 대신 '오픈 소스 소프트웨어' 라는 용어를 사용하기 시작했다.
- GNU GPL(General Public License) : GPL 은 자유 소프트웨어 재단에서 만든 Free 소프트웨어 라이선스다.
- GNU LGPL(Lesser General Public License) : LGPL 은 GPL 보다는 훨씬 완화된 조건의 공개 소프트웨어 라이선스이다. LGPL 이 적용된 라이브러리를 이용하여 개발하였을 경우 프로그램 소스코드는 공개하지 않아도 된다.
- BSD(Berkeley Software Distribution) 라이선스 : 버클리 캘리포니아 대학의 자유소프트웨어 저작권의 한 가지이다.
- **아파치(Apache) 라이선스** : 아파치 소프트웨어 재단에서 자체적으로 만든 소프트웨어에 대한 라이선스 규정이다.
- MIT(Massachusetts Institute of Technology) 라이선스 : BSD 라이선스를 기초로 작성된 BSD 계열 라이선스 중의 하나이다.
- MPL(Mozilla Public License) : MPL 의 특징은 소스코드와 실행파일의 저작권을 분리했다는 점이다.

Section 4. 기본 설치 및 유형

리눅스 설치의 개요

- 리눅스 설치 파일은 해당 배포본의 홈페이지에서 다운로드 받을 수 있다.
- 리눅스는 단 하나의 제품 또는 한 종류의 제품군만 있는 것이 아니다.
- 리눅스 배포판마다 설치 환경과 설치 과정이 다르다.
- 리눅스 설치 유형은 배포판마다 다르지만 패키지에 따라 데스크탑형, 서버형, 사용자 정의형으로 구분한다.
- 설치 전에 시스템이 있는 모든 파일을 백업해 둔다.
- 멀티 부팅 시스템을 만든다면, 현재 운영체제의 배포 미디어를 가지고 있어야 한다.
- 부팅 드라이브를 다시 파티션 하는 경우라면, 운영체제의 부트로드를 다시 설치해야 할

수도 있고, 더 많은 경우에 운영체제 전체를 해당 파티션에 다시 설치해야한다.

리눅스 설치를 위한 하드웨어 정보 파악

- 하드웨어 정보
- 하드웨어 호환성
- 네트워크 설정

리눅스 설치하기

- 1단계 : 설치 초기화면 – Test this media & install CentOS7, Install CentOS8
- 2단계 : 설치 초기화면 – Rescue a CentOS system, Run a memory test
- 3단계 : 언어선택
- 4단계 : 설치 요약 확인
- 5단계 : 날짜와 시간 설정
- 6단계 : 설치 소스 – 저장소(repository) 서버 주소 변경
- 7단계 : 설치 소스 – 설치와 관련된 특수 저장 장치 설정
- 8단계 : 설치 소스 – 기본 저장 장치 하드디스크 파티션 설정
- 9단계 : 소프트웨어 선택
- 10단계 : Kdump 설정
- 11단계 : 네트워크 설정과 호스트명 지정
- 12단계 : 패스워드 지정

Section 5. 파티션

파티션 특징과 종류

- 파티션이란, 하나의 물리적 디스크를 여러 개의 논리적인 디스크로 분할하는 것이다.
- 파티션은 주 파티션, 확장 파티션, 논리 파티션, 스왑 파티션으로 구분된다.

주 파티션	부팅이 가능한 기본 파티션 하나의 하드디스크에 최대 4개의 주 파티션 분할 가능 하드디스크를 4개 이상의 파티션으로 사용해야 할 때 하나의 확장 파티션
-------	--

	을 설정하여 확장 파티션 안에 여러 개의 논리 파티션을 분할하여 데이터 저장
확장 파티션	주 파티션 내에 생성, 하나의 물리적 디스크에 1개만 생성 파티션 번호는 1~4번이 할당 데이터 저장 영역을 위한 것이 아니라 논리 파티션을 생성
논리 파티션	확장 파티션 안에 생성되는 파티션 논리 파티션은 12개 이상 생성하지 않는 것을 권고하며 5번 이후의 번호가 붙여짐
스왑 파티션	하드디스크의 일부를 메모리처럼 사용하는 영역 주 파티션 또는 논리 파티션에 생성 프로그램 실행 시 부족한 메모리 용량을 하드디스크로 대신 리눅스 설치 시에 반드시 설치되어야 하는 영역 스왑 영역의 크기는 메모리의 2배를 설정하도록 권고

디스크와 장치명

- 분할된 파티션은 디스크의 장치 파일명 뒤에 숫자를 붙인다.
- 리눅스에서 파티션 만들고 마운트할 때 지정된 디바이스명을 사용한다.

파일 시스템

- 파일 시스템은 운영체제가 파일을 시스템의 디스크 파티션상에 구성하는 방식이다.
- 일정한 규칙을 가지고 파일을 저장하도록 규칙 방식을 제시한다.
- 파티션에 파일 시스템이 없으면, 파일 시스템 생성을 거쳐야 사용이 가능하다.
- 리눅스는 고유의 파일 시스템뿐만 아니라 다양한 파일 시스템을 지원하고 있다.

LVM(Logical Volumn Manager)

- 여러 개의 하드디스크를 합쳐서 사용하는 기술로 한 개의 파일 시스템을 사용한다.
- 작은 용량의 하드디스크 여러 개를 큰 용량의 하나의 하드디스크처럼 사용한다.
- 서버를 운영하면서 대용량의 별도 저장 공간이 필요할 때 활용된다.
- 다수 개의 디스크를 묶어서 사용함으로써 파티션의 크기를 줄이거나 늘릴 수 있다.

RAID

- RAID 는 복수 배열 독립 디스크의 약자이다.
- 여러 개의 물리적 디스크를 하나의 논리적 디스크로 인식하여 작동하게 하는 기술이다.

- 여러 개의 하드디스크에 일부 중복된 데이터를 나눠서 저장하는 기술이다.
- RAID 종류는 하드웨어 RAID 와 소프트웨어 RAID 로 나뉜다.
- 데이터를 저장하는 다양한 방법이 존재하며 이 방법들을 레벨이라 한다.
- 레벨에 따라 저장 장치의 신뢰성을 높이거나 전체적인 성능을 향상시키는 다양한 목적을 만족시킨다.
- 각 레벨의 장점을 합친 RAID 구성으로는 대표적으로 RAID 0+1과 RAID 1+0 이 있다.

파티션 분할

- fdisk 는 파티션 테이블을 관리하는 명령으로 리눅스의 디스크 파티션을 생성, 수정, 삭제 할 수 있는 일종의 유틸리티이다.
- fdisk 명령어 : a, l, n, t, w, p, q

Section 6. 부트 매니저

부트로드

- 부트스크랩 로더의 준말로 컴퓨터를 사용자가 사용할 수 있도록 디스크나 플래시에 저장된 운영체제를 읽어 주기억장치에 적재해 주는 프로그램이다.
- 부트로더는 운영체제가 시동되기 이전에 미리 실행되면서 커널이 올바르게 시동되기 위해 필요한 모든 관련 작업을 마무리하고 최종적으로 운영체제를 시동시키기 위한 프로그램이다.
- 임베디드 시스템 부트로더란, PC 의 BIOS 와 OS Loader 의 기능을 수행하는 프로그램으로 시스템이 부팅할 때 가장 먼저 수행된다.
- 운영체제 실행에 필요한 환경을 설정하고 운영체제 이미지를 메모리에 복사한다.
- 부트로더는 부트매니저라고도 부르며 크기가 512바이트로 하드디스크의 첫번째 섹터인 MBR(Master Boot Record) 에 위치한다.
- 주 파티션마다 부트섹터가 할당된다.
- 분할된 주 파티션들은 자신의 부트 레코드를 MBR에 기록하여 실행된다.
- 한 컴퓨터에 다수개의 운영체제가 설치되어 있는 경우 작업 운영체제를 선택하여 부팅할 수 있게 한다.

런레벨

- 리눅스 부팅 시 작동하는 서비스들이 있다. 런레벨에 따라 작동하는 서비스를 조정 가능하다.
- 런레벨은 0 에서 6까지 총 7가지이다.
- 7가지 런레벨 중 리눅스가 가동 시 특정 모드의 레벨을 디폴트로 할 경우 파일 /etc/inittab 에 설정한다.
- /etc/inittab 파일 형식은 '코드 런레벨:행동:명령어'이다.
- 현재 실행되는 런레벨을 확인하는 명령어는 runlevel 이다.

로그인과 로그아웃

로그인

- 리눅스는 X윈도우상에서의 로그인/로그아웃과 콘솔상에서의 로그인/로그아웃이 있다.
- 로그인 과정 : 입력한 패스워드와 파일 /etc/passwd 필드 비교 -> 셸 설정 파일 실행 -> 로그인 셸 실행

로그아웃

- 로그아웃은 logout, exit 또는 조합키 Ctrl+D 를 사용한다.
- 관리자는 일정시간 동안 작업을 수행하지 않는 모든 사용자들을 강제로 로그아웃할 수 있다.

Section 7. 사용자 생성 및 계정 관리

리눅스 명령어

which

- 명령어의 경로를 확인하는 명령어이다.
- 명령어의 위치를 찾아주거나 alias 를 보여주는 명령어이다.

alias

- 자주사용하는 명령어를 특정 문자로 입력해 두고 명령어 대신 해당 문자를 사용할 수 있게 하는 명령어이다.

unalias

- alias 기능을 해제한다.

환경변수 PATH

- PATH 는 실행 파일들의 디렉터리 위치를 저장해 놓는 환경 변수이다.
- 명령어 echo \$PATH 는 지정된 PATH 값을 확인할 수 있다.
- 기존의 PATH 에 새로운 경로를 추가하는 방법에는 명령어 PATH 나 홈 디렉터리의 .bash_profile 에 PATH 를 추가한다. 기존 경로와 새 경로는 : 으로 구분한다.

리눅스 도움말

man

- 리눅스에서 사용하는 명령어들의 매뉴얼을 제공한다.

info

- 리눅스 명령어의 사용 방법, 옵션 등을 나타낸다.

whatis

- 명령어에 대한 기능을 간략하게 나타낸다.
- 완전히 키워드가 일치해야만 해당 명령어의 기능을 확인할 수 있다.

manpath

- man 페이지의 위치 경로를 검색하여 표시해 주는 명령어이다.

whereis

- 찾고자 하는 명령어의 실행 파일 절대 경로와 소스코드, 설정 파일 및 매뉴얼 페이지를 찾아 출력하는 명령어이다.

apropos

- man 페이지 설명에서 지정한 키워드를 포함하고 있는 명령어이다.

사용자 생성 명령어

useradd

- 계정을 생성하는 명령어로 명령어 adduser 와 동일한 기능을 갖는다.
- 계정자의 홈 디렉터리는 '/home/계정명' 이다.
- 생성된 계정자 정보는 파일 /etc/passwd, /etc/shadow, /etc/group 에 저장된다.

passwd

- 생성된 계정자의 패스워드를 입력 및 변경하는 명령어이다.
- 생성된 계정자의 패스워드는 /etc/shadow 파일 안에 기록된다.

su

- su 는 switch user 의 줄임말이다.
- 현재의 사용자 계정에서 로그아웃하지 않고 다른 사용자 계정으로 로그인하여 해당 사용자의 권한을 획득하는 명령어이다.

사용자 관련 파일

/etc/default/useradd

- 명령어 useradd 로 사용자 계정을 추가할 때 사용되는 정보를 읽어오는 파일이다.

/etc/passwd

- 계정자의 정보를 가지고 있는 파일로 리눅스에 로그인할 때 사용된다.

/etc/shadow

- 계정자의 패스워드 정보가 암호화되어 있는 파일로 암호화 패스워드 및 계정의 유효 기간 등을 기록하고 있는 파일이다.

/etc/login.defs

- 사용자 계정 설정과 관련된 기본값을 정의한 파일이다.
- 새로운 계정을 생성할 때 반드시 참조하는 파일이다.

사용자 계정 관리

usermod

- 디렉터리 /home 에 위치한 사용자들의 정보를 변경하는 명령어이다.
- 사용자의 홈 디렉터리 변경, 그룹 변경, 유효기간 등을 변경한다.

userdel

- 기존 계정 정보를 삭제하는 명령어이다.
- 사용자의 홈 디렉터리 변경, 그룹 변경, 유효기간 등을 변경한다.
- 옵션 없이 userdel 을 사용하면 /etc/passwd, /etc/shadow, /etc/group 에서 해당 계정자의

정보가 삭제된다.

chage

- 패스워드의 만료 정보를 변경하는 리눅스 명령어이다.

그룹관리

/etc/group

- 사용자 그룹에 대해 정의되어 있는 파일
- 모든 계정은 한 개 이상의 그룹에 포함되어 있다.

/etc/gshadow

- 그룹의 암호를 MD5 로 하여 저장하며 그룹의 소유주, 구성원 설정이 가능하다.

groupadd

- 새로운 그룹을 생성하는 명령어이다.

groupdel

- 기존의 그룹을 삭제하는 명령어이다.
- 그룹 안에 소속되어 있는 계정명이 있을 경우 해당 그룹은 삭제되지 않는다.

groupmod

- 그룹의 설정을 변경하는 명령어이다.

사용자 조회 명령어

users

- 시스템에 로그인한 사용자 정보를 출력하는 명령어이다.

who

- 현재 시스템에 접속해 있는 사용자들을 조회하는 명령어이다.
- 사용자 계정명, 터미널정보, 접속시간, 접속한 서버 정보 등을 확인할 수 있다.
- 관리자 root 와 일반 사용자 모드 사용이 가능하다.
- 명령어 'who am I' 또는 'whoami' 는 자신의 정보를 조회할 수 있다.

w

- 현재 접속 중인 사용자들의 정보를 나타내는 명령어이다.
- 확인 가능한 정보는 서버의 현재 시간 정보, 서버 부팅 후 시스템 작동 시간, 서버 접속자의 총 수, 접속자별 서버 평균 부하율, 접속자별 서버 접속 계정명, TTY명, 로그인 시간 정보 등이다.
- JCPU 는 w TTY 필드의 장치명에서 사용되는 모든 프로세스의 CPU 사용 시간이다.
- PCPU 는 해당 프로세스 결과값에서 WHAT 필드에 나타나는 프로세스명에서 사용하는 CPU 총시간이다.

id

- 사용자 계정의 uid, gid, group 을 확인하는 명령어이다.

groups

- 사용자 계정이 속한 그룹 목록을 확인하는 명령어이다.

Section 8. 디렉터리 및 파일

디렉터리 관리 명령어

pwd

- 현재 작업 중인 디렉터리의 위치를 나타내는 명령어이다.

cd

- 디렉터를 이동할 때 사용하는 명령어이다.
- 절대경로는 시작 위치와 상관없이 경로에 모든 디렉터를 표시하며, 절대 경로의 시작은 / 에서부터 시작한다.
- 상대 경로는 현재 작업 중인 디렉터를 기준으로 표시하는 경로이다.

mkdir

- 새로운 디렉터를 생성할 때 사용하는 명령어이다.

rmdir

- 디렉터리만 삭제하는 명령어로 디렉터리 안에 파일이 존재하는 경우 삭제되지 않는다.

파일 관리 명령어

ls

- 현재 위치한 디렉터리의 파일 목록들을 나타내는 명령어이다.

cp

- 파일 또는 디렉터를 복사하는 명령어이다.

rm

- 파일 또는 디렉터를 삭제하는 명령어이다.

mv

- 파일 또는 디렉터를 이동하거나 파일명을 변경할 때 사용하는 명령어이다.

touch

- 파일 크기가 0바이트인 빈 파일을 생성한다.
- 서버의 현재 시간으로 파일의 최근 사용한 시간과 최근 수정 시간 등 타임스탬프를 변경한다.

file

- 파일 종류 및 파일 속성값을 나타내는 명령어이다.

find

- 현재 디렉터리에서부터 하위 디렉터리까지 주어진 조건의 파일을 찾아 해당 경로를 표시한다.

locate

- 파일 위치를 찾는 명령어이다.

텍스트 파일 관련 명령어

cat

- 파일의 내용을 출력하는 명령어이다.

head

- 파일의 앞부분을 지정한 만큼 출력하는 명령어이다.

tail

- 파일의 마지막 행을 기준으로 지정한 행까지의 파일 내용 일부를 출력하는 명령어이다.

more

- 파일을 확인하는 명령어로 파일을 읽어 화면에 화면 단위로 끊어서 출력하는 명령어이다.
- 위에서 아래 방향으로만 출력되기 때문에 지나간 내용을 다시 볼 수 없다.

less

- 텍스트 파일을 한 번에 한 화면씩 나타내는 명령어이다.
- 기능적으로 more를 확장한 것으로 커서를 파일의 상하좌우로 이동할 수 있다.

grep

- 파일에서 특정한 패턴 또는 정규 문자식으로 나타낸 단어를 찾는 명령어이다.

wc

- 파일의 라인 수, 단어 수, 알파벳 수를 알려주는 명령어이다.

sort

- 명령어 결과나 문서 내용을 정렬하는 명령어이다.

cut

- 파일에서 특정 필드를 추출해 낸다. 필드는 구분자로 구분할 수 있다.

split

- 하나의 파일을 여러 개의 작은 파일로 분리할 때 사용한다.
- 파일의 내용을 라인 수로 분할할 수도 있고, 용량 단위로 분할할 수도 있다.
- 주로 디스켓에 파일을 나누어 복사하거나 백업할 때 CD-RW 용량 단위로 분할할 경우 유용하다.

파일 비교 명령어

diff

- 두 개의 파일을 행 단위로 비교하여 다른 부분을 출력하는 명령어이다.
- 두 개의 파일명을 매개변수로 사용하여 화면에 차이점을 나열한다.

cmp

- 두 개의 파일을 바이트 단위로 비교하여 출력하는 명령어이다.

comm

- 두 개의 파일의 행과 행을 비교하여 출력하는 명령어이다.

리다이렉션과 정규 표현식

리다이렉션

- 표준 입력과 표준 출력의 방향을 재지정하는 것이다.
- 표준 입력/출력/에러가 화면이 아닌 파일로 대체한다. 즉, 모니터로 출력이 파일로 재지정한다.
- 표준 입력 장치는 키보드, 표준 출력 장치는 모니터, 표준 에러 장치는 모니터이다.

파이프

- 둘 이상의 명령을 함께 묶어 출력 결과를 다른 프로그램이 입력으로 전환하는 기능이다.
- 현재 명령의 표준 출력을 다음 명령의 표준 입력으로 사용하는 것이다.
- 명령어와 명령어의 연결은 | 기호를 사용한다.
- 명령어1의 출력 결과는 명령어2의 입력으로 처리된다.
- 더 이상 처리할 명령어가 없으면 표준 출력 장치인 화면으로 출력한다.

정규표현식

기호	의미
^	라인의 첫 글자
\$	라인의 끝 글자
.	한 글자
*	* 기호 바로 이전의 글자는 정규 표현식이 0회 이상 반복
[]	대체 글자 목록을 [] 에 나열
[^]	대체 못할 글자 목록을 [^] 에 나열
+	+ 기호 바로 이전 글자나 정규 표현식이 1회 이상 반복
?	? 기호 바로 이전 글자나 정규 표현식이 없거나 1회만 존재
()	부분 정규 표현식의 시작과 끝을 표시
	로 구분된 단어들 중 최소 하나 존재
{m, n}	{ } 기호 바로 이전 글자나 정규 표현식이 m개 이상 n개 이하 반복

Section 9. 기타 명령어

네트워크 관련 명령어

ping

- 외부 호스트에 신호를 보내며 신호를 받은 호스트는 응답을 주면서 서로 네트워크가 연결되어 있음을 확인시켜주는 명령어이다.

traceroute

- 목적지 호스트까지의 경로를 표시하고 그 구간의 정보를 기록하는 명령어이다.
- 목적지 호스트까지의 패킷 전송 지역을 측정하거나 목적지 호스트로 향하는 경로상에 어떤 장애가 있는 경우 위치를 파악할 수 있다.

nslookup

- 도메인명으로 IP 주소를 조회하거나 또는 IP 주소로 도메인명을 조회하는 명령어이다.

dig

- 명령어 nslookup 과 유사한 기능을 가진 명령어로 호스트명에 대한 IP 주소 정보 또는 IP 주소에 대한 호스트명을 조회하는 명령어이다.
- 서버명은 확인하고자 할 네임 서버를 지정하는 것이며 지정하지 않을 경우 /etc/resolv 에 등록된 네임 서버를 이용하여 루트 서버를 조회하게 된다.

host

- 호스트명을 알고 있는데 IP 주소를 모르거나 그 반대의 경우에 사용하는 명령어이다.
- 호스트명을 이요하면 IP 주소뿐만 아니라 하위 호스트명도 조회할 수 있다.
- 호스트는 시스템에 등록된 DSN 서버를 이용하여 검색하는데 다른 DNS 서버를 이용해서 따로 지정할 수 있다.

hostname

- 시스템 이름을 확인하거나 변경할 때 사용하는 명령어이다.

시스템 종료 명령어

shutdown

- 시스템을 종료하거나 재부팅하는 명령어이다.
- 현재 수행 중인 프로세스들을 종료하며 sync 를 수행하여 저장되지 않는 데이터를 디스크에 저장하고 모든 파일 시스템을 mount 시킨 후에 시스템을 종료한다.
- root 사용자만이 권한을 가지고 있는 명령어이다.

init

- shutdown 명령어 동일한 기능을 가진 명령어이다.

reboot

- 시스템을 재부팅하는 명령어이다.

halt

- 시스템을 종료하는 명령어이다.

기타 명령어

cal

- 시스템에 설정된 달력을 출력하는 명령어이다.

date

- 시스템의 날짜와 시간을 표시하거나 변경한다.

clear

- 터미널의 내용을 지우는 명령어이다.

tty

- 현재 사용하고 있는 단말기 장치의 경로명과 파일명을 나타낸다.
- 텔넷 등에서 동일한 계정으로 여러 개 로그인한 경우 확인 시 유용하다.

time

- 프로그램이 수행되는데 걸리는 시간을 측정하여 출력하는 명령어이다.
- 세 가지 시간 결과 real, user, sys를 보여준다. real 을 총 수행시간, user 는 CPU가 사용자 영역에서 보낸 시간, sys 는 시스템 호출 실행에 걸린 시간이다.

wall

- 모든 로그인된 사용자들에게 터미널을 통해 메시지를 전달받는 명령어이다.

write

- 해당 사용자에게 메시지를 전달하는 명령어이다.

mesg

- write 를 사용해서 들어오는 메시지 수신 여부를 확인하고 제어하는 명령어이다.

Section 1. 권한 및 그룹 설정

- 리눅스 시스템은 모든 파일과 디렉터리에는 접근권한과 소유권이 부여된다.
- 명령어 'ls -l' 은 파일 속성을 나타낸다.
- 속성 필드 중 첫 번째 필드는 파일이나 디렉터리의 허가권, 세번째와 네번째 필드는 파일이나 디렉터리의 소유권을 나타낸다.
- **파일의 허가권이나 소유권을 설정하는 명령어는 chmod, chown, chgrp, umask 등이 있다.**

소유권 관련 명령어

- 소유권은 임의의 파일 또는 디렉터리에 대한 사용자와 그룹들의 소유 권한을 나타낸 것이다.
- 그룹은 사용자들의 시스템 운영 특성에 따라 묶어 놓은 것으로, 같은 그룹에 속한 사용자들은 파일 또는 디렉터리에 대해 동일한 소유권과 직접 권한을 갖는다.

chown

- 파일과 디렉터리의 **사용자 소유권과 그룹 소유권**을 변경한다.

chgrp

- 파일이나 디렉터리의 **그룹** 소유권을 변경한다.

허가권 관련 명령어

- 명령어 'ls -l' 으로 파일 유형과 허가권을 알 수 있다.
- 파일 허가권의 첫 번째 자리는 파일 유형을 기호로 정의한다.
- 파일은 일반 파일, 디렉터리 파일, 특수 파일로 나뉜다.
- 파일 권한을 읽기(read), 쓰기(write), 실행(execute)이 있다.
- 읽기, 쓰기 또는 실행의 접근 **제한 표시는 하이픈(-) 으로** 나타낸다.

chmod

- 파일이나 디렉터리의 접근 **허가권을 변경하는 명령어이다.**

umask

- **새로 생성되는 파일이나** 디렉터리의 기본 허가권 값을 지정한다.

- 파일의 기본 권한은 666, 디렉터리의 기본 권한은 777 이다.
- 파일이나 디렉터리 생성 시 디폴트 권한 값에서 설정한 umask 를 뺀 값을 기본 허가권으로 설정한다.

```
umask
0002
mkdir AAA
ls -l
```

0777-0002 = 0775

특수 권한

SetUID 와 SetGID

- 프로세스가 실행되는 동안 해당 프로세스의 root 권한을 임시로 가져오는 기능이다.
- 프로세스가 사용자보다 높은 수준의 접근을 요구할 때 파일 접근 제한 때문에 원활한 기능을 제공할 수 없기 때문에 이러한 문제점을 해결하기 위한 방법이다.
- SetUID의 경우 사용자가 사용할 때만 소유자 권한으로 파일을 실행시키고, SetGID 의 경우 사용자가 사용할 때만 그룹 권한으로 파일을 실행한다.

Sticky bit

- 일반적으로 공용 디렉터리를 사용할 때 sticky bit 를 설정하여 사용한다.
- 사용자 권한을 지정하기 어려운 프로그램들이 일시적으로 특정 디렉터리에 파일을 생성하고 삭제하도록 이용된다.
- 설정된 디렉터리에는 누구든 접근 가능하고 파일을 생성할 수 있다.
- Sticky bit 가 설정되어 있는 디렉터리 안의 내용은 해당 파일의 소유자나 root 만이 변경이 가능하다.

디스크 쿼터

- 파일 시스템마다 사용자나 그룹이 생성할 수 있는 파일의 용량 및 개수를 제한하는 것이다. 보통 블록 단위의 용량 제한과 inode의 개수를 제한한다.
- 사용자나 그룹이 가질 수 있는 inode의 수, 사용자나 그룹에게 할당된 디스크 블록 수를 제한한다.
- 쿼터는 사용자별, 파일 시스템별로 동작된다.
- 그룹 단위로도 용량을 제한할 수 있으며 웹호스팅 서비스를 하는 경우에 유용하다.

디스크 쿼터 지정 단계

- 단계 1: 파일 /etc/fstab 에 디스크 쿼터 관련 설정
- 단계 2: 재마운팅 실행 후 확인
- 단계 3: 마운트 된 쿼터를 끄고 생성된 쿼터 파일 삭제
- 단계 4: 쿼터 데이터베이스 생성
- 단계 5: 사용자별 쿼터 지정
- 단계 6: 쿼터 현재 상태 점검

디스크 쿼터 관련 명령어

- **quotaoff** : 쿼터 서비스를 비활성화 한다.
- **quotacheck** : 파일 시스템의 디스크 사용 상태를 검색한다.
- **edquota** : 편집기를 이용하여 사용자나 그룹에 디스크 사용량을 할당하는 명령어이다.
- **setquota** : 편집기가 기반이 아닌 명령행에서 직접 사용자나 그룹에 디스크 사용량을 할당하는 명령어이다.

Section 2. 파일 시스템의 관리

파일 시스템의 개요와 종류

개요

- 운영체제가 파일을 시스템의 디스크상에 구성하는 방식이다.
- 컴퓨터에서 파일이나 자료를 쉽게 발견 및 접근할 수 있도록 보관 또는 조직하는 체제이다.
- 하드디스크나 CD-ROM 과 같은 물리적 저장소를 관리한다.
- 파일 서버상의 자료로의 접근을 제공하는 방식과 가상의 현태로서 접근 수단만이 존재하는 방식도 파일 시스템의 범위에 포함된다.

종류

- 리눅스 전용 디스크 기반 파일 시스템

파일 시스템	설명
ext	리눅스 초기에 사용되던 파일 시스템이며 호환성이 없음

	ext2의 원형 2GByte 의 데이터와 파일명을 255자까지 지정 가능
ext2	고용량 디스크 사용을 염두하고 설계된 파일 시스템 쉽게 호환되며 업그레이드도 쉽게 설계되어 있음
ext3	리눅스의 대표적인 저널링을 지원하도록 확장된 파일 시스템 ACL(Access control List) 를 통한 접근 제어 지원
ext4	파일에 디스크 할당 시 물리적으로 연속적인 블록을 할당 64비트 기억 공간 제한을 없앴 16 TeraByte 의 파일을 지원

- 저널링 파일 시스템

파일 시스템	설명
JFS	Journaling File System 의 약자 IBM 사의 독자적인 저널링 파일 시스템 GPL로 공개하여 현재 리눅스용으로 개발
xfs	eXtended File System 고성능 저널링 시스템 64비트 주소를 지원하며 확장성이 있는 자료 구조와 알고리즘 사용 데이터 읽기/쓰기 트랜잭션으로 성능 저하를 최소화 64비트 파일 시스템으로 큰 용량의 파일도 다룰 수 있음
ReiserFS	독일의 한스 라이저가 개발한 파일 시스템 모든 파일 객체들을 B트리에 저장, 간결한 색인화 된 디렉터리 지원

- 네트워크 파일 시스템

파일 시스템	설명
SMB	Server Message Block 삼바 파일 시스템을 마운트 지정 윈도우 계열 OS 환경에서 사용되는 파일/프린터 공유 프로토콜 리눅스, 유닉스 계열 OS와 윈도우 OS와의 자료 및 하드웨어 공유
CIFS	Common Internet File System SMB를 확장한 파일 시스템 SMB를 기초로 응용하여 라우터를 뛰어넘어 연결할 수 있는 프로토콜
NFS	Network File System 썬마이크로시스템이 개발한 네트워크 공유 프로토콜 파일 공유 및 파일 서버로 사용됨 공유된 영역을 마운트할 때 지정 하드웨어, 운영체제 또는 네트워크 구조가 달라도 공유 가능 NFS 서버의 특정 디렉터리를 마운트하여 사용할 수 있음

- 기타 지원 가능한 파일 시스템

파일 시스템	설명
FAT	Windows NT가 지원하는 파일 시스템 중 가장 간단한 시스템 FAT 로 포맷된 디스크는 클러스터 단위로 할당 클러스터 크기는 볼륨 크기에 따라 결정 읽기 전용, 숨김, 시스템 및 보관 파일 특성만 지원 삼바 파일 시스템을 마운트 지정
VFAT	Virtual FAT FAT 파일 시스템이 확장된 것으로 FAT 보다 제한이 적음 파일 이름도 최고 255자까지 만들 수 있음 공백이나 여러 개의 구두점도 포함
FAT32	SMB를 확장한 파일 시스템 32GB 보다 큰 파티션을 만들 수 없고 파티션에 4GB를 초과하는 파일을 저장할 수 없음
NTFS	윈도우에서 사용하는 파일 시스템 안정성이 뛰어나고 대용량 파일도 저장 파일 크기 및 볼륨은 이론상으로 최대 16EB 이나 실질적으로는 2TB 로 한계가 있음
ISO 9660	CD-ROM 의 표준 파일 시스템 1988년에 제정된 표준
UDF	Universal Disk Format 의 약자로 최신 파일 시스템 형식 광학 매체용 파일 시스템 표준 ISO 9660파일 시스템을 대체하기 위한 것으로 대부분 DVD에서 사용
HPFS	OS/2 의 운영체제를 위해 만들어진 파일 시스템

관련 명령어

mount 와 umount

- 마운트는 특정 디바이스를 특정 디렉터리처럼 사용하기 위해 장치와 디렉터리를 연결한다.
- 리눅스는 PnP 기능을 지원하지만 지원하는 하드웨어가 많지 않으므로 시스템 부팅후에 수동으로 마운트해서 사용을 하고 사용이 끝난 후에는 언마운트를 시킨다.
- 파일 /etc/mtab 은 현재 마운트된 블록 시스템 정보를 표시한다.

eject

- 이동식 보조기억장치등과 같은 미디어를 해제하고 장치를 제거하는 명령어이다.

fdisk

- 새로운 파티션의 생성, 기존 파티션의 삭제, 파티션의 타입 결정 등의 작업을 수행할 수

있다.

- 한 번에 한 디스크에 대해서만 작업을 수행한다.

mkfs

- 리눅스 파일 시스템을 생성한다.
- fdisk로 하드디스크를 파티션을 나눈 후 해당 파티션에 맞는 파일 시스템을 생성한다.

mke2fs

- ext2, ext3, ext4 타입의 리눅스 파일 시스템을 생성하는 명령어이다.

fsck

- 파일 시스템의 무결성을 점검하고 대화식으로 복구하는 명령어이다.
- 디렉터리 /lost+found 는 fsck 에서 사용하는 디렉터리이다.

e2fsck

- ext2, ext3, ext4 타입의 리눅스 파일 시스템을 복구하는 명령어이다.

du

- Disk Usage의 약자로 디렉터리별로 디스크 사용량을 확인할 수 있다.

df

- 시스템에 마운트된 하드 디스크의 남은 용량을 확인할 때 사용하는 명령어이다.
- 기본적으로 1024 Byte 블록 단위로 출력한다.

Section 3. 셸 개념 및 종류

개념

- 명령어 해석기이다.
- 로그인할 때 실행되어 사용자별로 사용 환경 설정을 가능하게 한다.
- 강력한 스크립트 언어이다.
- 입출력 방향 재지정과 파이프 기능을 제공한다.
- 포어/백그라운드 프로세스를 실행한다.

종류

- 본셸계열과 C셸 계열로 나뉜다.
- 사용자 프롬프트가 \$ 이면 본셸 계열, % 이면 C 셸 계열을 사용하고 있다는 것이다.
- 대부분의 셸은 본셸 계열의 기능을 포함하여 확대 발전한 형태이다.
- C셸은 본셸의 모든 기능과 명령어 히스토리, 별명, 작업 제어 기능을 추가로 가지고 있다.

셸 확인 및 변경

로그인 셸 확인

- 파일 /etc/shell 에서 사용할 수 있는 셸들을 확인할 수 있다.
- 파일 /etc/passwd 파일에서 계정마다 할당된 셸을 확인할 수 있다.
- 명령어 `echo $SHELL` 은 현재 로그인한 사용자가 사용하고 있는 셸을 확인할 수 있다.

셸 변경

- 로그인 셸 변경은 반영구적인 셸 변경 방법으로 관리자가 셸 변경 후 다음 변경을 하기 전까지 지정된 셸을 사용한다.
- 명령어 `chsh`: 일반 사용자 환경에서 셸 변경 시 사용한다.
- 명령어 `usermod`: 관리자 환경에서 지정된 계정자의 정보를 변경할 때 사용하는 명령어이다.

Section 4. 셸 환경 설정

환경 변수와 셸 변수

환경 변수

- 전체 셸에서 사용 가능한 전역 변수
- 서브 셸에 기능 상속 가능
- 환경 변수 확인 명령 `env`

셸 변수 (지역 변수)

- 현재 로그인 셸에서만 사용 가능한 지역 변수
- 서브 셸에 기능 상속 불가능

- 셸 변수 확인 명령 set

환경 설정 파일

- 셸 시작 시 자동으로 실행되는 고유의 시작 파일이 있다. 이 파일은 사용자 운영환경을 설정한다.
- 배쉬셸의 시작 파일은 /etc/profile, /etc/bashrc, ~/.bash_profile, ~/.bashrc 이다.
- 셸 파일은 전역적 파일과 지역적 파일로 나뉜다.
- 파일 /etc/profile.d 는 몇몇 응용 프로그램들이 시작 시 자동 실행할 스크립트 파일 경로를 넣어둔다.

배쉬셸의 주요 기능

History 기능

- 일정 개수 이상 사용했던 명령어를 .bash_history 에 저장해 두고 다시 불러서 사용할 수 있게 하는 기능이다.
- 대부분의 셸은 이전에 입력했던 명령어를 반복하거나 약간 변형하여 다시 사용할 수 있도록 하는 기능이다.

alias 기능

- 자주 사용하는 명령어를 특정 문자로 입력해 두고 간편하게 사용할 수 있게 하는 기능이다.

Section 5. 프로세스 개념 및 유형

개념

- 프로세스는 CPU와 메모리를 할당받아 실행 중인 프로그램이다.
- 프로세스들마다 고유의 프로세스 ID를 할당받는다.

프로세스의 유형

포어그라운드 프로세스

- 사용자와 상호작용하는 프로세스
- 터미널에 직접 연결되어 입출력을 주고받는 프로세스

- 명령 입력 후 수행 종료까지 기다려야 하는 프로세스
- 화면에서 실행되는 것이 보이는 프로세스
- 응용프로그램이나 명령어 등

백그라운드 프로세스

- 사용자와 직접적인 대화를 하지 않고 뒤에서 실행되는 프로세스
- 사용자의 입력에 관계없이 실행되는 프로세스
- 실행은 되지만 화면에 나타나지 않고 실행되는 프로세스
- 시스템 프로그램, 데몬 등

fork

- 새로운 프로세스를 만들 때 기존 프로세스를 복제하는 방식을 사용한다.
- 새로운 프로세스를 위한 메모리를 할당한다.
- 새로 생성된 프로세스는 원래의 프로세스와 똑 같은 코드를 가지고 있다.
- 원본 프로세스를 부모 프로세스라 부르고, 새로 복제된 프로세스를 자식 프로세스라고 부른다.

exec

- 호출하는 프로세스가 새로운 프로세스로 변경되는 방식이다.
- 새로운 프로세스를 위한 메모리를 할당하지 않는다.
- 호출한 프로세스의 메모리에 새로운 프로세스의 코드를 덮어씌워 버린다.

데몬

- 리눅스 시스템이 부팅 시 **자동으로 실행되는 백그라운드 프로세스이다.**
- 메모리에 상주하면서 사용자의 특정 요청이 오면 즉시 실행되는 대기 중인 서버 프로세스이다.
- 주기적이고 지속적인 서비스 요청을 처리하기 위해 사용된다.
- 사용자들은 이 프로세스들을 볼 수 있는 권한이 없다.

Section 6. 프로세스 유틸리티

프로세스 관련 명령어

ps

- 현재 실행중인 프로세스의 상태를 보여주는 명령어이다.
- CPU사용도가 낮은 순서로 출력된다.

pstree

- 실행 중인 프로세스들을 트리구조로 나타낸다.

jobs

- 작업이 중지된 상태, 백그라운드로 진행 중인 상태, 변경되었지만 보고되지 않은 상태 등을 표시한다.
- 백그라운드로 실행중인 프로세스를 확인한다. [숫자] 는 작업번호이다.
- 출력된 목록에서 +는 현재 작업 실행, -는 앞으로 실행될 작업을 나타낸다.

bg와 fg

- 포어그라운드에서 백그라운드로 전환: bg %작업번호 또는 bg 작업번호
- 백그라운드에서 포어그라운드로 전환: fg %작업번호 또는 fg 작업번호

kill

- 프로세스를 종료시킨다.

killall

- 같은 데몬의 여러 프로세서를 한 번에 종료시킬 때 사용한다.
- 프로세스명으로 연관된 프로세스들을 종료시킨다.

nice

- 프로세스 사이의 우선순위를 확인하고 우선순위를 변경할 수 있는 명령어이다.
- 조정할 수 있는 NI 값의 범위는 -20 ~ 19 (우선순위 높음 -> 낮음)이다.
- 우선순위 0의 값을 가지며 값이 작을수록 우선순위가 높다.
- 옵션 -n을 사용하지 않으면 디폴트는 10을 사용한다.
- 조정수치가 생략되면 명령의 우선권은 10만큼 증가한다.
- 명령어 'nice -10 bash' 는 bash 프로세스 NI 값을 10만큼 증가시키는 것이다. 값이 증가

한다는 것은 우선순위를 낮추는 것이다. 우선순위를 높이는 명령어는 'nice --10 bash' 이다. 이것은 NI 값을 -10만큼 감소시켜 우선순위를 높인다.

- nice [옵션] 프로세스명

renice

- 이미 실행중인 프로세스의 우선순위를 변경한다.
- nice는 기존 NI값을 증감하지만 명령어 renice는 지정한 NI값을 설정한다.
- nice는 양수값은 -를, 음수값은 --를 사용하지만, renice는 양수값에 -를 사용하지 않는다.
- renice [옵션] NI값 PID

top

- 리눅스 시스템의 운영 상태를 실시간으로 모니터링하거나 프로세스 상태를 확인할 수 있다.

nohup

- 프로세스가 중단되지 않고 백그라운드로 작업을 수행할 수 있게 한다.
- 사용자가 로그아웃하거나 작업 중인 터미널 창이 닫혀도 실행 중인 프로세스를 백그라운드 프로세스로 계속 작업할 수 있도록 한다.
- 용량이 큰 데이터 압축 해제와 같은 실행 시간이 오래 걸리는 프로세스들에 대해 nohup으로 처리하여 작업하면 작업 중단 없이 해당 업무를 완료할 수 있다.
- 백그라운드로 실행될 수 있도록 명령행 뒤에 &를 명시한다.

tail

- 파일의 마지막 행을 기준으로 지정한 행까지 파일 내용의 일부를 출력한다.
- 기본값으로 마지막 10줄을 출력한다.

스케줄링과 cron

- 주기적으로 반복되는 일은 자동적으로 실행될 수 있도록 설정한다.
- 스케줄링 데몬은 crond이며 관련 파일은 /etc/crontab이다.
- 파일 /etc/crontab은 7개의 필드로 구성되어 있다.
- 명령어 crontab은 사용자가 주기적인 작업을 등록할 수 있게 한다.

Section 7. 에디터 종류

개요

- 리눅스에서 지원하는 편집기로는 `vi`, `emacs`, `pico`, `gedit`, `xedit` 등이 있다.
- 리눅스 편집기는 편집기를 통해 파일을 수정한다.

종류

pico

- 워싱턴 대학의 Aboil Kasar가 개발한 유닉스 기반의 텍스트 에디터이다.
- 메뉴 선택 방식의 텍스트 편집기로 기본 인터페이스가 윈도우의 메모장과 유사하여 간단하다.
- 자유 소프트웨어 라이선스가 아니기 때문에 소스 수정이 불가능하다.
- 다른 편집기에 비해 사용하기 쉽고 사용하기 편리하지만 기능이 부족하고 업데이트가 잘 되지 않는다.
- GNU프로젝트에서는 pico의 복제 버전 에디터인 nano를 개발하였다.
- vi편집기처럼 입력모드와 명령모드가 존재하지 않고 바로 텍스트 입력이 가능하다.

emacs

- 리처드 스톨만이 매크로 기능이 있는 텍스트 교정 및 편집기로 개발하였다.
- 최초의 개발자는 리처드 스톨만이며, 이후 제임스 고슬링이 LISP언어를 기반으로 하여 emacs에 다양한 기능을 개발하여 추가하였다.
- LISP에 기반을 둔 환경 설정 언어를 가지고 있다.

vi

- 1976년 빌 조이가 초기 BSD 릴리즈에 포함될 편집기로 만들었다.
- 리눅스 배포판과 유닉스에 기본적으로 포함되어 있다.
- 유닉스 환경에서 가장 많이 쓰이는 문서 편집기이다.
- 다른 편집기들과 다르게 모드형 편집기이다.
- 명령모드, 입력모드, 편집모드로 구성되어 있다.

vim

- 브람 무레나르가 만든 편집기이다.
- **vi 편집기와** 호환되면서 독자적으로 다양한 기능을 추가하여 만든 편집기이다.
- 편집 시 다양한 색상을 이용하여 가시성을 높일 수 있다.
- 패턴 검색 시 하이라이트 기능을 제공하여 빠른 검색을 가능하게 해준다.
- ex모드에서 히스토리 기능을 제공한다.
- 확장된 정규 표현식 문법과 강력한 문법 강조 기능을 갖는다.

gedit

- 그놈 데스크톱 환경으로 개발된 자유 소프트웨어 텍스트 편집기이다.
- 마이크로소프트, 윈도, 맥OS X에서도 사용할 수 있다.
- UTF-8과 호환하며 텍스트 문서를 편집하는 용도에 중점을 두었다.
- X-윈도우 시스템에 맞춰 개발되었다.
- GTK+와 그놈 라이브러리를 이용하여 개발되었다.
- 텔넷 접속 시나 텍스트 기반 콘솔 창에서는 사용할 수 없다.

Section 8. 에디터 활용

에디터 기초 사용법

pico

메뉴	기능
[Ctrl] + [O]	파일 저장
[Ctrl] + [X]	파일 종료, 종료 시 저장이 안 되어 있으면 저장할 것인지 물어봄
[Ctrl] + [R]	현재 커서 위치에 다른 파일을 불러옴
[Ctrl] + [A]	현재 행의 맨 앞으로 이동
[Ctrl] + [E]	현재 행의 맨 끝으로 이동
[Ctrl] + [V]	이전 페이지로 이동
[Ctrl] + [Y]	다음 페이지로 이동
[Ctrl] + [C]	현재 커서의 위치를 표시
[Ctrl] + [T]	영문자의 철자를 확인
[Ctrl] + [W]	키를 누르고 문자열을 입력하면 원하는 문자열을 찾음
[Ctrl] + [K]	현재 라인을 삭제
[Ctrl] + [U]	마지막으로 삭제된 라인을 복구

[Ctrl] + [I]	화면 갱신
--------------	-------

emacs

메뉴	기능
[Ctrl] + [X] [Ctrl] + [S]	파일 저장
[Ctrl] + [X] [Ctrl] + [C]	편집 종료
마크 설정 후 [Ctrl] + [W]	잘라내기
[Ctrl] + [K]	커서 뒤에 있는 한 줄이 모두 지워짐
[Ctrl] + [A]	커서를 줄의 맨 앞으로 이동
[Ctrl] + [E]	커서를 줄의 맨 뒤로 이동
[Ctrl] + [N]	커서를 한 줄 아래로 이동
[Ctrl] + [S] 찾을 문자열	커서의 아랫부분에서 찾을 문자열을 검색
[Ctrl] + [R] 찾을 문자열	커서의 윗부분에서 찾을 문자열을 검색
[Ctrl] + [G]	진행되고 있는 명령을 끄

Section 9. 소프트웨어 프로그램 설치

계열

데미안 계열

- 배포 업체: Debian, Ubuntu, Xandros, Linspire
- 패키지 툴: dpkg, apt-get, optitude

레드햇 계열

- 배포 업체: Fedora, CentOS, RHEL, openSUSE, Mandirva
- 패키지 툴: rpm, yum

rpm

- 레드햇사에서 만들어낸 패키지 관리 툴이다.
- 새로운 패키지를 설치하거나 업그레이드, 삭제 시 사용한다.

- Windows의 setup.exe와 유사하게 만든 프로그램이다.
- 레드햇 계열의 패키지 파일 확장명은 *.rpm 이다.

yum

- 네트워크를 통해 기존 RPM 패키지 파일의 업데이트 자동 수행, 새로운 패키지 설치 및 제거를 수행한다.
- RPM의 의존성 문제를 해결하기 위한 유틸리티이다.
- 인터넷을 기반으로 설치하므로 네트워크가 정상적으로 연결된 상태여야만 한다.
- YUM은 페도라 22버전 이후부터 YUM의 문제점을 보완한 DNF로 전환되고 있다.

dpkg

- 데비안의 저레벨 패키지 관리 툴이다.
- deb 패키지의 설치, 삭제, 정보 제공을 위해 사용된다.
- 확장자 deb 파일은 데비안 패키지 파일이다.
- 패키지 설치 및 제거 시 RPM과 같은 의존성 문제를 일으킨다.
- 명령어 'dpkg -s 패키지'는 지정된 패키지에 대한 자세한 정보를 나타낸다.

apt-get

- 데비안 리눅스에서 소프트웨어 설치와 제거를 위한 패키지 관리 유틸리티이다.
- 패키지 관련 정보를 확인하거나 패키지 설치 시 발생할 수 있는 의존성과 충돌문제를 해결하기 위해 /etc/apt/source.list 파일을 참조한다.

aptitude

- 우분투 패키지 관리 유틸리티로 APT처럼 패키지를 관리를 자동화한다.

Section 10. 소스 파일 설치

파일 아카이브와 압축

파일 아카이브

- 아카이브는 다수 개의 파일이나 디렉터리를 하나의 파일로 묶는 것이다.
- 아카이브 파일은 다른 시스템으로 다수 개의 파일을 한 번에 전송하거나 파일 백업용으로 사용한다.

파일압축과 해제

- 대표적인 파일 압축 명은 `compress`, `gzip`, `bzip2`, `xz`가 있다.
- 일반적으로 많이 사용되는 압축 명령어는 `gzip`과 `bzip2`이다.
- 압축률이 가장 낮은 것은 명령어 `compress`이며, 압축률이 가장 높은 것은 명령어 `xz`이다.

소스 코드 설치

- 소스 코드를 압축 해제 후 컴파일 순서에 따라 프로그램을 설치한다.
- 컴파일 순서는 설치 파일의 환경설정, 컴파일, 파일 설치이다.
- 1단계 환경설정: `./configure` 프로그램 설치 과정에서 필요로 하는 환경파일 `makefile` 생성
- 2단계 컴파일: `make` `makefile`을 기반으로 소스 파일을 컴파일
- 3단계 파일 설치: `make install` 컴파일 된 실행파일을 지정된 속성으로 지정된 디렉터리에 설치

Section 11. 주변 장치 연결 및 설정

프린터 인쇄 시스템 설치 및 설정

LPRng

- 리눅스 초기에 사용되었던 인쇄 시스템이다.
- 버클리 프린팅 시스템으로 BSD 계열 유닉스에서 사용하기 위해 개발되었다.
- 라인 프린터 데몬 프로토콜을 사용하여 프린터 스폰링과 네트워크 프린터 서버를 지원한다.
- LPRng설정 파일은 `/etc/printcap`이다.

CPUS

- 애플이 개발한 오픈 소스 프린팅 시스템이다.
- 유닉스 계열 운영체제의 시스템을 프린터 서버로 사용 가능하게 해준다.
- 매킨토시나 윈도우 등 시중에 시판되는 대부분의 프린트를 지원한다.
- HTTP 기반의 IPP(Internet Printing Protocol)를 사용하여 프린터를 웹 기반으로 제어한다.
- CPUS 설정 파일은 `/etc/cpus`이다.

- 사용자 및 호스트 기반의 인증을 제공한다.
- CPUS 관련 파일은 cpusd.conf, printers.conf, classes.conf, cpused 등이 있다.

프린터 설정

- 일반적으로 X-Windows상에서 '프린터 설정 도구'로 프린터를 설치한다.
- '로컬 접속'으로 프린터를 직접 연결할 수 있다.
- 네트워크 프린터를 설정할 경우 5가지 방법을 제공하고 있다.

AppSocket/HP jecDirect	프린터가 컴퓨터에 연결되어 있지 않고 네트워크에 연결된 경우 사용
LPD/LPR 호스트 또는 프린터	IPP 프로토콜 기반의 프린터 설정 시 사용
Windows Printer via SAMBA	윈도우 시스템에 연결된 프린터 설정 시 사용 삼바 기반의 SMB 프로토콜 사용
인터넷 프린터 프로토콜 https	https 프로토콜 기반의 프린터 설정 시 사용
인터넷 프린터 프로토콜 ipp	IPP 프로토콜 기반의 프린터 설정 시 사용

사운드 카드 설치 및 설정

OSS(Open Sound System)

- 리눅스 및 유닉스 계열 운영체제에서 사운드를 만들고 캡처하는 인터페이스이다.
- 표준 유닉스 장치 시스템콜에 기반을 둔 것이다.
- 현재 리눅스 커뮤니티에서는 ALSA로 대체되었다.

ALSA(Advanced Linux Sound Architecture)

- 사운드 카드용 장치 드라이버를 제공하기 위한 리눅스 커널 요소이다.
- GPL 및 LGPL 라이선스 기반으로 배포되고 있다.
- 사운드 카드를 자동으로 구성하고 시스템에 여러 개의 사운드 장치를 관리하는 것이 목적이다.
- OSS의 지원을 받아서 하드웨어 기반 미디합성, 다중 채널 하드웨어 믹싱, 전이중 통신, 다중 프로세서와의 조화, 스레드 안전장치 드라이브 등의 기능을 지원한다.
- 현재 설정 파일은 /etc/asound.state이다.

스캐너 설치 및 설정

SANE(Scanner Access Now Easy)

- 평판 스캐너, 핸드 스캐너, 비디오 캠 등 이미지 관련 하드웨어를 제어하는 API이다.
- GPL 라이선스, 리눅스 및 유닉스 계열, OS2, Windows도 지원한다.

XSANE(X based interface for the SANE)

- SANE 스캐너 인터페이스를 이용하여 X-Windows 기반의 스캐너 프로그램이다.
- 스캐너, 디지털 카메라, 디지털 캠 등 다양한 장치에서 사용이 가능하다.
- 스캔 작업뿐만 아니라 캡처한 이미지에 수정 작업을 할 수도 있다.
- GPL 라이선스, 리눅스 및 유닉스 계열, OS2, Windows도 지원한다.

Section 12. 주변 장치 활용

프린터 설치 및 설정

BSD 계열 프린터 명령어들

- `lpr`: 프린터 작업 요청을 한다.
- `lpq`: 프린터 큐에 있는 작업 목록을 출력한다.
- `lprm`: 프린터 큐에 대기 중인 작업을 삭제한다. 취소할 프린트 작업 번호를 입력한다.
- `lpc`: 라인 프린터 컨트롤 프로그램이다.

System V 계열 프린터 명령어들

- `lp`: 프린터 작업 요청(명령어 `lpr`과 유사한 기능)을 한다.
- `lpstat`: 프린터 큐의 상태를 확인한다.
- `cancel`: 프린트 작업을 취소한다. 취소할 요청 ID를 `lpstat`로 확인 후 삭제한다.

사운드 카드 관련 명령어

- `alsactl`: ALSA 사운드 카드를 제어한다.
- `alsamixer`: 커서라이브러리 기반의 오디오 프로그램이다.
- `cdparanoia`: 오디오 CD로부터 음악 파일을 추출 시 사용한다.

스캐너 관련 명령어

- sane-find-scanner: SCSI 스캐너와 USB 스캐너 관련 장치 파일을 찾아주는 명령어이다.
- scanimage: 이미지를 스캔한다.
- scanadf: 자동 문서 공급 장치가 장착된 스캐너에서 여러 개의 사진을 스캔한다.
- xcam: GUI 기반으로 평판 스캐너나 카메라로부터 이미지 스캔한다.

Section 1. X-windows 개념 및 사용법

X-윈도우의 특징과 구성 요소

개념과 특징

- 리눅스 환경의 각종 애플리케이션과 유틸리티에 대해 그래픽 사용자 인터페이스를 제공한다.
- 플랫폼과 독립적으로 작동하는 그래픽 시스템이다.
- X-윈도우는 X11, X, X Windows System이라 한다.
- 네트워크 기반의 **그래픽 환경을 지원한다.**
- 이기종 시스템 사이에서도 사용 가능하다.
- 스크롤바, 아이콘, 색상 등 그래픽 환경 자원들이 특정 형태로 정의되어 있지 않다.
- 디스플레이 장치에 의존적이지 않으며 원하는 인터페이스를 만들 수 있다.
- X-윈도우는 네트워크 프로토콜 기반의 클라이언트/서버 시스템이다.
- 서버 프로그램과 클라이언트 프로그램으로 나누어 작동한다.
- 서버는 클라이언트들의 디스플레이에 관한 접근 허용, 클라이언트 간의 자원 공유, 네트워크 메시지 전달, 클라이언트와 입출력 기기와의 중계를 담당한다.
- 클라이언트는 애플리케이션으로 X 서버가 제공하는 기능들을 이용한다.
- 오픈 데스크톱 환경으로 KDE, Gnome, XFCE 등이 있다.

구성요소의 종류

- Xprotocol
- Xlib
- XCB
- Xtoolkit
- XFree86
- XF86Config

X-윈도우 설정과 실행

파일 /etc/inittab

- **init 프로세스가 읽는 파일로**, init 프로세스가 무엇을 해야 할 것인가를 결정한다.
- 리눅스 사용 환경을 초기화한다.

X-윈도우 실행

- 그래픽 환경이 아닌 터미널 윈도우로 로그인한 경우에는 몇 개의 프로그램을 실행해야 한다.
- 텀 | 널 윈도우의 명령어 프롬프트상에서 다음의 명령어를 실행시켜야 한다.
- startx는 X-윈도우를 실행하는 스크립트로 시스템 환경을 초기화하고 xinit을 호출한다.
- startx 실행 시 인자값을 xinit에 전달하는 옵션은 '--' 이다.

환경변수 DISPLAY

- 환경 변수는 프로세스가 컴퓨터에서 동작하는 방식에 영향을 주는 동적인 값이다.
- 셸에서 정의되고 실행하는 동안 프로그램에 필요한 변수이다.
- 환경 변수 DISPLAY는 현재 X-윈도우 Display 위치를 지정할 수 있다.
- 형식: export DISPLAY=IP주소:디스플레이번호.스크린번호

윈도우 매니저와 데스크톱 환경

윈도우 매니저

- 윈도우 매니저는 X window상에서 창의 배치와 표현을 담당하는 시스템 프로그램이다.
- 창 열기와 닫기, 창의 생성 위치, 창 크기 조정, 창의 외양과 테두리를 변화시킬 수 있다.
- 라이브러리는 Xlib와 XCB를 사용한다.
- 윈도우 매니저의 대표적인 종류로는 fwm, twm, mw, windowMaker, AfterStep 등이 있다.

데스크톱 환경

- GUI사용자에게 제공하는 인터페이스 스타일로 데스크톱 관리자라고도 한다.
- 윈도우 매니저, 파일 관리자, 도움말, 제어판 등 다양한 도구를 제공하는 패키지 형태의 프로그램이다.
- 아이콘, 창, 도구모음, 폴더, 배경화면, 데스크톱 위젯도 제공한다.

- 드래그 앤 드롭과 프로세스 간의 통보 기능을 지원한다.
- 대표적인 데스크톱 환경에는 KDE, GNOME, LXDE, Xfce 등이 있다.

디스플레이 매니저

- X window system 상에서 작동하는 프로그램이다.
- 디스플레이 매니저 종류들로는 XDM, GDM, KDM 등이 있다.
- 로컬 또는 리모트 컴퓨터의 X server의 접속과 세션 시작을 담당한다.
- 사용자에게 그래픽 로그인 화면을 띄워주고 아이디와 패스워드를 입력받아 인증을 진행하고 인증이 정상적으로 완료되면 세션을 시작한다.

Section 2. X-윈도우 활용

원격지에서 X 클라이언트 이용

xhost

- 명령어 xhost는 X 서버에 접속할 수 있는 클라이언트를 지정하거나 해제한다.
- X 서버에게 디스플레이를 요청 시 해당 요청에 대해 허용 여부를 호스트 단위로 제어한다.
- xhost [+|-] [IP|도메인명]
- 환경변수 DISPLAY로 X 서버 프로그램이 실행될 때 표시되는 클라이언트 주소를 지정한다.

xauth

- .Xauthority 파일의 쿠키 내용을 추가, 삭제, 리스트를 출력하는 유틸리티이다.
- xhost가 호스트 기반 인증 방식을 사용하기 위해 필요한 유틸리티라면 xauth는 MMC방식의 인증방식을 사용하기 위한 필수 유틸리티이다.
- 원격지에서 접속하는 X클라이언트를 허가할 때 IP 주소나 호스트명이 아닌 X-윈도우 실행 시에 생성되는 키 값으로 인증할 때 사용한다.
- 사용자 인증 기반을 지원하기 위해 각 사용자에게 네트워크화 된 홈 디렉터리에 파일 \$HOME/.Xauthority에 대해 읽기 및 쓰기 권한이 있어야 한다.

X-윈도우 응용 프로그램

오피스

- LibreOffice: 오피스 프로그램 패키지
- gedit: 텍스트 편집 프로그램
- kwrite: KDE 기반의 텍스트 편집기

그래픽

- GIMP: 이미지 편집 프로그램
- ImageMagick: 이미지 생성 및 편집을 지원하는 프로그램
- eog: GNOME의 이미지 뷰어 프로그램
- kolourpaint: Ubuntu 이미지 편집 프로그램
- gThumb: GNOME 데스크톱 이미지 뷰어 프로그램
- gwenview: KDE의 기본 이미지 뷰어

멀티미디어

- Totem: GNOME 기반의 사운드 및 비디오 재생 프로그램
- RHYTHMBOX: 통합형 음악 관리 프로그램
- CHEESE: GNOME 기반의 카메라 동영상 프로그램

개발

- ECLIPSE: 통합 개발 환경으로 자바를 비롯한 다양한 언어를 지원

기타

- Dolphin: KDE용 파일 관리자
- KSnapshot: 스크린샷 프로그램

Section 3. 인터넷 활용

네트워크 분류

lan(Local Area Network)

- 근거리 통신망을 연결하는 네트워크이다.
- 학교나, 회사등 가까운 지역을 묶는 소규모 네트워크이다.

man(MetroPolice Area Network)

- 도시권 통신망을 연결하는 네트워크이다.
- LAN과 WAN의 중간크기이다.

wan(Wide Area Network)

- 국가, 대륙 등과 같이 넓은 지역을 연결하는 네트워크이다.
- 거리의 제한은 없으나 다양한 경로를 경유해 도달하므로 속도가 느리고 전송 에러율도 높다.

san(Storage Area Network)

- 스토리지를 위해 고안된 스토리지 전용 고속 네트워크이다.
- 파이버 채널을 이용하여 구성되는 저장장치 네트워크이다.
- 호스트 컴퓨터의 종류에 구애받지 않고 별도의 연결된 저장장치 사이에 대용량의 데이터를 전송시킬 수 있는 고속 네트워크이다.
- 서버가 클라이언트로부터 받은 파일 I/O 요청을 직접 블록 I/O로 전환하여 SAN에 연결된 스토리지로 저장한다.

네트워크 개요

LAN 토폴로지

- 토폴로지는 호스트 및 장비들의 물리적인 배치 형태이다.
- 토폴로지는 성형, 망형, 버스형, 링형, 트리형 등이 있다.

성형

- 중앙 컴퓨터에 여러 대의 컴퓨터가 허브 또는 스위치와 같은 장비로 연결
- 중앙 집중식 형태로 네트워크 확장에 용이
- 고속의 대규모 네트워크에 적합
- 관리하는 중앙 컴퓨터 고장 시 전체 네트워크 사용이 불가능



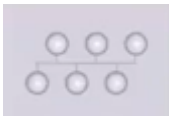
망형

- 모든 노드가 서로 일대일로 연결된 형태
- 대량의 데이터를 송수신할 경우 적합
- 장애 발생 시 다른 시스템에 영향이 적고 우회할 수 있는 경로가 존재하여 가장 신뢰성이 높은 방식
- 회선 구축 비용이 많이 듦



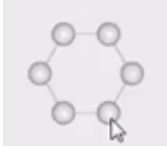
버스형

- 하나의 통신회선에 여러 컴퓨터를 연결해서 전송
- 연결된 컴퓨터 수에 따라 네트워크 성능이 변동
- 단말기 추가 및 제거가 용이하며 설치 비용이 저렴
- 노드 수 증가 시 트래픽 증가로 병목현상 발생, 네트워크 성능 저하 초래
- 문제가 발생한 노드의 위치를 파악하기 어려움



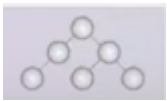
링형

- 각 노드가 좌우의 인접한 노드와 연결되어 원형을 이룬 형태
- 앞의 컴퓨터로부터 수신한 내용을 다음컴퓨터로 재전송하는 방법
- 토큰패싱이라는 방법을 통해 데이터 전송
- 고속네트워크로 자주 네트워크 환경이 바뀌지 않는 경우 구성
- 분산제어와 검사 및 회복이 가능
- 네트워크 전송상의 충돌이 없고 노드 숫자가 증가하더라도 망 성능의 저하가 적음
- 논리적인 순환형 토폴로지로 하나의 노드장애가 전체 토폴로지에 영향
- 노드의 추가 및 삭제가 용이하지 않음



트리형

- 버스형과 성형 토폴로지의 확장 형태
- 백본과 같은 공통 배선에 적절한 분기장치를 사용하여 링크를 덧붙여 나갈 수 있는 구조
- 트래픽 양 증가 시 병목 현상의 가능성 증대



매체 접근 제어 방식

- CSMA/CD: 단말기가 전송로의 신호 유무를 조사하고 다른 단말기가 신호를 송출하는지 확인한다.
- 토큰패싱: 토큰의 흐름에 의해 전송 순서가 결정된다. 토큰패싱은 free token과 busy token을 이용하여 매체 접근을 제어한다.

네트워크 장비

케이블

- 보호 외피나 외장 안에 두 개 이상의 전선이나 광섬유로 묶여 있는 것이다.
- TP 케이블, 동축 케이블, 광섬유 케이블 등을 사용할 수 있다.

리피터

- 신호의 재생 및 증폭기능을 수행하여 물리적인 거리를 확장시킨다.

허브

- 신호를 노드에 전달해 주는 장비이다.
- 네트워크 확장, 다른 허브와의 상호 연결, 신호의 증폭 등의 기능을 제공한다.

LAN카드

- 네트워크에 접속할 수 있도록 컴퓨터 내에 설치되는 확장 카드이다.
- 전기신호로부터 데이터를 송신하고 변환하며, 데이터를 전기신호로 변환하여 송신한다.
- MAC 주소를 이용하여 데이터의 수신 여부를 판별한다.

브릿지

- 모든 수신 프레임을 일단 버퍼에 저장하고, 주소에 따라 목적지 포트로 프레임을 전달하는 장비이다.
- 큰 네트워크를 단일 네트워크상의 트래픽 감소 등과 같은 작고 관리하기 쉬운 Segment로 나눌 필요가 있을 때 사용한다.
- 전기적으로 신호의 재생 및 패킷의 송수신 어드레스를 분석하여 패킷의 통과 여부를 판정하는 필터링을 작용한다.

스위치

- 브릿지와 비슷한 기능을 갖는 장비이다.
- 소프트웨어 기반인 처리 방식으로 브릿지보다 빠르게 데이터를 전송한다.
- 맥 주소 테이블을 기반으로 프레임 전송한다.
- 스위칭 허브는 전용매체교환 기술을 이용하여 트래픽 병목 현상을 제거하고, 포트별로 속도가 전용으로 보장된다.

라우터

- OSI모델의 물리 계층, 데이터링크 계층, 네트워크 계층의 기능을 지원하는 장치이다.
- 서로 다른 통신망과 프로토콜을 사용하는 네트워크 간의 통신을 가능하게 한다.
- LAN을 연결시켜주는 망 연동 장치로서 통신망에서 정보를 전송하기 위해 경로를 설정하는 역할을 제공하는 핵심적인 통신장비이다.
- 목적지로 향하는 최적의 경로 설정 데이터를 목적지까지 전달하는 기능을 수행한다.

게이트웨이

- 서로 다른 형태의 네트워크를 상호 접속하는 장치이다.
- 서로 다른 통신망이나 프로토콜을 사용하는 네트워크 간의 통신을 가능하게 하는 장비를 통칭한다.
- 데이터 포맷 등 두 개의 시스템 사이에서 중계자 역할을 수행한다.

UTP 케이블링

- 다이렉트 케이블
- 크로스오버 케이블

프로토콜 개요와 기능

OSI7 모델과 TCP/IP모델

OSI 7 모델		기능	TCP/IP 모델
7	응용 계층	사용자에게 다양한 네트워크 서비스를 제공하기 위해 User Interface를 제공, UI로 데이터를 생성	응용 계층
6	표현 계층	부호화, 압축, 암호화 기능	
5	세션 계층	종단 간 애플리케이션들의 연결 설정, 유지, 해제	
4	전송 계층	종단 간 연결, 응용 계층 사이에 논리적인 통로 제공	전송 계층
3	네트워크 계층	논리적인 주소를 사용 경로 관리, 최적 경로 결정	인터넷 층
2	데이터링크 계층	데이터 전송을 위한 형식 결정 데이터 전송을 위하여 Media에 접근하는 방법 제공 오류 검출 기능 제공	네트워크 인터페이스 층
1	물리 계층	물리적인 연결, 전기적, 기계적, 기능적 절차적인 수단 제공	

계층별 프로토콜

- 프로토콜은 특정 통신 기능을 수행하기 위한 규약이다. 이것은 두 노드 사이의 정보 교환 시 발생하는 통신상의 에러를 피하기 위하여 합리적인 통제를 한다.
- 프로토콜 구성요소는 형식, 의미, 순서이다.
 - 형식: 데이터 포맷, 부호화 및 신호 레벨 등
 - 의미: 특정 패턴을 어떻게 해석하고, 어떤 동작을 할 것인가 결정
 - 순서: 속도 일치 및 순서 제어 등
- 인터넷/네트워크 계층 프로토콜: IP, ICMP, IGMP, ARP, RARP
- 전송 계층 프로토콜: TCP, UDP
- 응용 계층 프로토콜: SMTP, POP, Telnet, SSH, FTP, HTTP, SNMP, TFTP, DHCP

IP주소와 도메인

IPv4 주소 체계

- IPv4 주소는 4개의 옥텟으로 구성된다.
- 각 옥텟은 8비트이므로 IPv4는 총 32비트이다.
- 클래스 E는 240~255 사이의 대역에 있으며 IP 주소 부족을 위해 예약해 놓은 것이다.
- IP주소는 네트워크 ID와 호스트 ID로 구성되어 있다.
- 서브넷 마스크는 네트워크 부분과 호스트 부분을 구분해주는 값이다. 이것은 효율적인 네트워크 분리를 가능하게 한다.

서브네팡팅

- 서브넡은 특정 네트워크를 여러 개의 네트워크, 브로드캐스트 도메인을 나누는 것이다.
- IP주소의 부족 현상을 해소하기 위한 방안이다.
- 서브넡팅은 디폴트 서브넡 마스크를 기준으로 해서 네트워크 ID비트수를 늘리고 호스트 ID 비트수를 줄이는 것이다. 이때 기준에서 늘어난 네트워크 ID비트는 서브넡ID라 부른다.

IPv6 주소 체계

구분	IPv4	IPv6
IP주소 비트수	32비트	64비트
IP주소 방식	지정 주소 방식 일반주소 브로드캐스트주소	자동 설정 주소 방식 유니캐스트주소 멀티캐스트주소 애니캐스트주소
IP헤더 길이	20바이트 ~ 60바이트	40바이트와 확장필드 이용
기능	IP헤더를 이용한 기본 기능	암호 및 인증 가능 QoS 관련 기능 보안 기능

Section 4. 인터넷 서비스의 종류

WWW 서비스

- 프로토콜 HTTP를 기반으로 한 멀티미디어와 하이퍼텍스트를 통합한 정보 검색 시스템이다.
- 다양한 그래픽 유저 인터페이스를 사용하는 것이 가능하다.
- WWW 서비스는 분산 클라이언트-서버 모델을 기반으로 한다.
- 표준 웹 프로토콜(HTTP, XML, SOAP, WSDL, UDDI)을 기본으로 하여 서로 다른 개발 환경과

운영체제에서도 상호 통신이 가능하다.

메일 서비스

- 전자 메일 시스템은 컴퓨터 사용자끼리 편지를 주고받는 서비스이며 MTA, MUA, MDA로 구성된다.
- 메일 클라이언트에서 송신은 SMTP, 수신은 POP3 또는 IMAP4를 이용한다.
- 메일을 보내거나 메일 서버간의 메시지 교환은 SMTP프로토콜을 사용하고, 메일 서버에 도착한 메일을 사용자 컴퓨터에서 확인할 때에는 POP3과 IMAP를 사용한다.
- MIME(Multipurpose Internet Mail Extension)은 멀티미디어 전자우편을 위한 표준으로, 멀티미디어 데이터를 ASCII 형식으로 변환할 필요 없이 인터넷 전자 우편으로 송신하기 위한 SMTP의 확장 규격이다.

FTP 서비스

- FTP(File Transfer Protocol) TCP/IP에 의해 제공되는 호스트 간의 파일 복사를 위한 프로토콜이다.
- FTP의 통신 모드는 패시브 모드와 액티브 모드로 구분한다.
- FTP는 20번(일반 데이터 전송용)과 21번(제어 데이터 전송용) 포트 번호를 사용한다.
- FTP는 사용자계정을 가진 사용자들의 접속과 익명의 로그인을 허용하고 있다.
- 익명은 공개 소프트웨어를 제공하는 FTP서버에 접속할 때 입력할 수 있는 계정이다.

DNS(Domain Name System) 서비스

- 호스트 이름을 기반으로 IP주소를 변환하거나 IP주소를 기반으로 호스트 이름을 변환시켜 주는 프로토콜이다.
- DNS에서는 도메인명을 분산된 트리 형태의 계층적 구조로 관리한다.

Telnet과 SSH 서비스

- 네트워크상에 있는 다른 컴퓨터에 로그인하거나 원격 시스템에서 명령 실행, 파일 복사 등을 제공하는 서비스이다.
- Telnet과 ssh는 사용자가 서버에 접속하여 서버 관리, 파일 편집 등을 Text모드 환경에서 시스템 명령을 실행하고 결과를 화면을 통해 볼 수 있다.
- Telnet은 서버와 주고받는 정보를 Byte 스트림 형식으로 전송하고, ssh는 DES, RSA 등의

암호화기법을 사용해서 전송한다. 따라서 Telnet보다 ssh가 안전한 데이터 전송을 보장한다.

- ssh는 암호화뿐만 아니라 압축 기술도 제공한다. 암호화를 통해서 늘어난 데이터의 크기를 압축하여 사용자와 서버가 송수신하는 것을 가능하게 한다.

NFS(Network File System)

- 네트워크 기반에 다른 시스템과 파일 시스템을 공유하기 위한 클라이언트/서버 프로그램이다.
- 1984년 썬 마이크로시스템즈 사에서 개발하였다.
- 원격지에 있는 리눅스 서버의 특정 디렉터리를 로컬 시스템의 하위 디렉터리처럼 사용할 수 있다.
- 다른 컴퓨터의 파일 시스템을 마운트하고 공유하여 자신의 디렉터리인 것처럼 사용할 수 있게 해준다.
- NFS는 portmap이 먼저 수행되어 있어야만 NFS 서비스가 실행된다. portmap은 NIS, NFS 등 RPC(Remote Procedure Call) 연결에 관여하는 데몬이다.
- NFS 서비스는 nfsd, rpc.mounted, rpc, statd, rpc.rockd, rpc.rquotad 데몬들이 구동된다.

RPC(Remote Procedure Call)

- 동적으로 서비스와 포트를 연결할 때 사용하는 방법이다.
- 기본적으로 포트와 서비스가 정적으로 구성될 때는 /etc/services 파일을 참조하지만 동적으로 포트를 할당받아 사용할 때는 RPC 인 rpcbind를 사용한다.

Section 5. 인터넷 서비스 설정

네트워크 인터페이스 설정

- 리눅스는 다양한 네트워크를 지원한다.
- 일반적으로 네트워크 인터페이스는 자동으로 인식되지만 자동으로 인식되지 않을 경우 수동으로 설정해야 한다.
- 네트워크 인터페이스 수동 설정 방법은 컴파일된 인터페이스 모듈을 커널에 적재하는 것이다.

네트워크 설정 파일들

- `/etc/sysconfig/network`: 네트워크의 기본 정보가 설정되어 있는 파일이다.
- `/etc/sysconfig/network-scripts/ifcfg-ethX`: 지정된 네트워크 인터페이스의 네트워크 환경 설정 정보가 저장된다.
- `/etc/resolv.conf`: 기본적으로 사용할 도메인명과 네임서버를 설정한다.
- `/etc/hosts`: IP 주소와 도메인 주소를 1:1로 등록하여 도메인에 대한 IP 주소를 조회하도록 한다.
- `/etc/host.conf`: DNS 서비스를 제공할 때 먼저 이 파일을 검사하여 파일의 설정에 따라 서비스를 한다.

네트워크 설정

IP 주소 설정

- 네트워크 설정 파일로 주소 설정: 파일 `/etc/sysconfig/network` 또는 `/etc/sysconfig/network-scripts/ifcfg-ethX`로 IP 주소를 설정한다.
- 명령어를 이용한 주소 설정: 명령어 `ifconfig`를 이용하여 IP주소를 할당한다.
- 유틸리티를 이용한 주소 설정: `netconfig`, `system-config-network`, `redhat-config-network` 등의 다양한 유틸리티를 이용하여 주소를 설정한다.

라우팅 테이블 설정 및 관리

- 라우팅이란 송신 패킷이 목적지까지 전송할 수 있도록 **경로를 설정하는 작업이다.**
- 송신 패킷은 라우팅 테이블에 목적지 경로 정보가 있다면 해당 경로로 패킷을 전송한다.
- **명령어 `route`는** 라우팅 테이블을 설정하거나 확인한다.
- 목적지 경로가 라우팅 테이블에 없다면 디폴트게이트웨이로 트래픽을 전송할 수 있게 라우팅 테이블을 설정할 수 있다.

네트워크 관련 명령어

- TCP/IP주소 설정 정보 확인: `ifconfig`, `nslookup`
- 네트워크 경로 상태 확인: `ping`, `traceroute`
- 네트워크 연결 상태 확인: `netstat`

- 라우팅 테이블 확인: route
- NIC 상태 확인: ethtool, mii-tool, arp

Section 6. 기술 동향

리눅스 동향

- 1991년, 초기 리눅스는 주로 서버로 이용되었다.
- 현재의 리눅스 활용 분야는 크게 서버, 데스크톱 및 개발, 임베디드 분야로 나눌 수 있다.

리눅스 관련 기술

클러스터링

- 여러 개의 시스템이 하나의 거대한 시스템으로 보이게 만드는 기술이다.
- 여러 개의 컴퓨터를 네트워크를 통해 연결하여 하나의 컴퓨터처럼 동작하도록 제작한 시스템이다.
- 클러스터 노드와 클러스터 관리자로 구성된다.
- 고계산용 클러스터, 부하분산 클러스터, 고가용성 클러스터 종류가 있다.

임베디드 시스템

- 컴퓨터의 하드웨어 제어인 프로세스, 메모리 입출력장치와 하드웨어를 제어하는 소프트웨어가 조합되어 특정한 목적을 수행하는 시스템이다.
- 미리 정해진 특정한 기능을 수행하도록 프로그램이 내장되어 있는 시스템이다.
- 하드웨어와 소프트웨어를 포함하는 특정한 응용시스템이다.
- 개인 휴대 정보 단말, 지리 정보 시스템, 정보가전, 게임기기 등의 시스템을 통칭한다.
- 하드웨어로는 프로세서/컨트롤러, 메모리, I/O장치, 네트워크 장치, 센서 등이다.
- 소프트웨어로는 운영체제, 시스템 S/W, 응용 S/W 이다.
- 실시간 처리를 지원한다.
- 소형, 경량 및 저전력을 지원한다.

Section 7. 활용 기술

리눅스 서버 분야

서버 가상화

- 서버를 구성하는 모든 자원의 가상화를 의미한다.
- 하나의 물리적 서버 호스트에서 여러 개의 서버 운영 체제를 게스트로 실행할 수 있게 해주는 소프트웨어 아키텍처이다.
- 서버들을 하나의 서버로 통합하여 가상환경을 구동시킴으로써 물리서버 및 공간을 절감한다.
- 서버 자원 통합 운영으로 하드웨어 가용성을 증가시킨다.
- 손쉬운 이중화구성과 유연한 자원 할당으로 시스템 가용성과 안정성을 확보한다.
- 통합 구축, 공동 활용, 유지 관리, 전력 및 관리 비용 등 중복 투자 방지 및 예산을 절감한다.

클라우드 컴퓨팅

- 공유 구성이 가능한 컴퓨팅 리소스의 통합으로 어디서나 간편하게 요청에 따라 네트워크를 통해 접근하는 것을 가능하게 한다.
- 사업자와 직접 상호 작용하지 않고 사용자의 개별 관리 화면을 통해 서비스를 이용할 수 있다.
- 사업자의 컴퓨팅 리소스를 여러 사용자가 공유하는 형태로 이용한다.
- 필요에 따라 필요한 만큼의 스케일업과 스케일다운이 가능하다.

IaaS(Infrastructure as a Service): 서버나 스토리지 같은 하드웨어 자원만을 임대해 주는 클라우드 서비스이다.

Paas(Platform as a Service): 소프트웨어 서비스를 개발하기 위한 플랫폼을 제공하는 클라우드 서비스이다.

SaaS(Software as a Service): 클라우드 환경에서 동작하는 응용 프로그램을 서비스 형태로 제공한다.

- 사설 클라우드, 공유 클라우드, 하이브리드 클라우드 모델이 있다.

빅데이터

- 기존 데이터베이스 관리 도구의 데이터 수집, 저장, 관리, 분석 역량을 넘어서는 데이터이다.

- 다양한 종류의 대규모 데이터로부터 저렴한 비용으로 가치를 추출하고 데이터의 빠른 수집, 발굴 분석을 지원할 수 있도록 고안된 차세대 기술 및 아키텍처이다.
- 볼륨, 속도, 다양성의 3대 요소가 있다.

임베디드 시스템

모바일

- 스마트폰의 OS란 스마트폰을 구성하고 있는 하드웨어 부품인 메모리, LCD, CPU 등의 기계적인 부품들을 효율적으로 관리 및 구동하게 하며, 사용자와의 편리한 의사소통을 위해 만들어진 다양한 프로그램들이 구동될 수 있도록 하는 소프트웨어 플랫폼이다.
- 대표적으로 안드로이드와 iOS가 있다.
- 리눅스 기반의 공개형 운영체제로 마에모, 모블린, 미고, 리모, 타이벤, iOS 가 있다.

스마트TV

- 텔레비전에 인터넷 접속 기능이 결합되어, 각종 애플리케이션을 설치하여 TV방송 시청이외의 다양한 기능을 활용할 수 있는 다기능 TV이다.
- 인터넷 TV 또는 커넥티드 TV라고도 불린다.

IVI(In Vehicle Infotainment)

- 인포테인먼트란 정보와 오락의 합성어로 정보전달에 오락성을 가미한 시스템이다.
- 내비게이션이나 계기판, AV 시스템, DMB, MP3, 오디오 및 외부 기기와의 연결까지 가능한 통합적인 차량 내부 시스템을 포함한다.
- GENIVI 표준 플랫폼 기반의 인포테인먼트 시스템이다.
- 운전자 편의성 및 안전성을 증대시킨다.
- GENIVI(GENEVA In Vehicle Infotainment)는 오픈 소스 기반 플랫폼 얼라이언스로 차량 멀티미디어 플랫폼 표준화 활동이다.