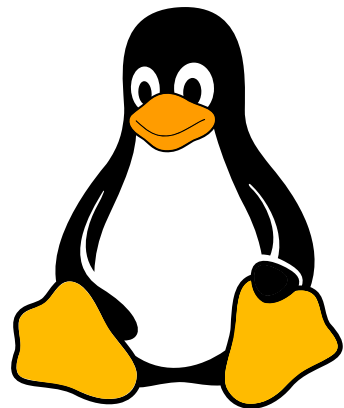


리눅스마스터 (2급)

김윤수
(kchris000@gmail.com)



※ 참고 (도움 주신 분들)

영진 닷컴 (<https://youngjin.com/>)

- ✓ 본 리눅스마스터 (2급) 자료는 영진 닷컴의 지원으로 만들어졌습니다.
- ✓ 본 문서의 수정/편집, 재배포 및 강의 등에 사용이 필요하실 경우
김윤수(kchris000@gmail.com) / 영진 닷컴 (<https://youngjin.com/>) 에게
문의 부탁드립니다.

※ 영진닷컴 (https://youngjin.com/)

이기적 시리즈

내 실력에 맞는 수험서를 찾아 공부하자!



기본서
기초부터 탄탄히!
초심자를 위한 완벽한 개론서

각급 실력: 초·중·고
문제 용이: 기본을 익히는 학생들에게
학습 기간: 기본을 익히는 단기간 동안

절대족보

작지만 강하다!
핵심 이론 + 기출문제

각급 실력: 초·중·고
문제 용이: 핵심 이론과 문제풀이를
학습 기간: 빠른 학습 효과를



최신문제집

문제 중심 학습!
핵심 이론과 다양한 학습자료 제공

각급 실력: 초·중·고
문제 용이: 다양한 문제와
학습 기간: 문제 풀이 중심

기출문제집

초단기 학습!
기출문제 풀이로 마무리 학습

각급 실력: 초·중·고
문제 용이: PDF로 제공되는
학습 기간: 기출문제 풀이를 통한



이기적만의 특별한 혜택

도서 구매 시 모두 제공!

• 제공되는 혜택은 도서별 상이합니다. 각 도서의 혜택을 확인해주세요.

1:1 질문답변



전문가의 1:1 일격 답변

동영상 강의



영문 강의의 영문 강의를 무제한으로!

놓칠 수 없는 이벤트



365 이벤트부터 리뷰 서평단까지
이기적이 제공하는 무궁한 선물!

CBT 온라인 문제집



PC와 모바일로 실전 연습!

자료실



재정 프로그램 추가 기출문제
핵심 요약집 제공

베타테스터, 관수진을 모집합니다!

베타테스터, 관수진으로 도서 제작에 참여해 주세요.



← 베타테스터 지원

관수진 지원 →



BOOK LIST

자격증은 이기적!

기본서

- 컴퓨터활용능력 1급, 2급
- 워드프로세서
- 정보처리기사 · 산업기사 · 기능사
- 정보보안기사/산업기사
- 빅데이터분석기사
- 사무자동화산업기사
- 정보기기운용기능사
- 컴퓨터그래픽스운용기능사
- 웹디자인기능사
- 리눅스마스터 1급, 2급
- ADsP 데이터분석 준전문가

완상의 콤비

- ITX(한글&엑셀&파워포인트)
- 정보처리기사 필기+실기
- 정보처리산업기사 필기+실기
- 정보보안기사 필기+실기
- 한식/양식조리기능사 필기+실기
- GTO(포토샵&일러스트) 1급, 2급

ITQ, GTQ

- ITQ 한글(NEO, 2020)
- ITQ 엑셀
- ITQ 파워포인트
- ITQ 액세스
- ITQ 인터넷
- GTQ 포토샵 1급, 2급(CS4, CC)
- GTQ 일러스트 1급, 2급(CS4, C56, CC)
- GTQ 인디자인 1급(CS6)

최신문제집

- 정보처리기사 필기
- 정보처리기사 실기
- 정보처리기사 실기 600제
- 워드프로세서 필기
- 컴퓨터활용능력 1급 필기
- 컴퓨터활용능력 2급 필기
- 바리스타 2급
- 한국실용글쓰기

기출문제집

- 한국사능력검정시험
- 운전면허 1 · 2종 공통
- 컴퓨터활용능력 1급, 2급
- 전산회계 1급

절대족보

- 정보처리기사 필기
- 사무자동화산업기사 필기
- 웹디자인기능사 필기
- 컴퓨터활용능력 1급 필기
- 컴퓨터활용능력 2급 필기
- 컴퓨터그래픽스운용기능사 필기

상시시험 공략집

- 컴퓨터활용능력 1급, 2급 필기+실기
- 워드프로세서 필기+실기

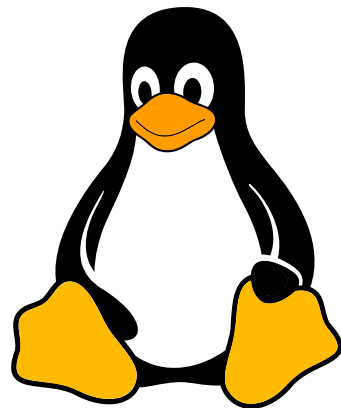
실용서 시리즈

- 한식 조리기능사
- 양식 조리기능사
- 중식/일식 조리기능사 실기
- 목재조기능사 필기+실기
- 제과/제빵기능사
- 바리스타 1급, 2급
- 전산회계 1급, 2급 이론+실무
- 회계원리(입문편)
- 전산회계운용사 2급 필기
- 한국사능력검정시험 실화/기본
- SW코딩자격 2급, 3급
- SQL 개발자
- 한국실용글쓰기
- 일러스트 기사 · 산업기사 필기
- NCS직업기초능력평가 입문서 (고졸채용&초보자용)
- 네트워크관리사 1 · 2급 필기+실기
- 글자/지체재문천기능사 필기
- 검색광고마스터 1급

리눅스마스터 (2급)

- 리눅스 일반 -

김윤수
(kchris000@gmail.com)



01. 리눅스 일반

리눅스 일반

- ✓ 리눅스의 이해
 - 리눅스 개요, 리눅스 배포판의 특징, 오픈소스 라이선스
- ✓ 리눅스 설치
 - 리눅스 설치, 파티션과 파일 시스템, 부트 매니저

01. 리눅스 일반 – 리눅스의 이해

리눅스 개요

✓ 리눅스의 특징

- 오픈소스 소프트웨어 → 소스코드와 관련 자료가 공개
- 멀티유저(다중사용자), 멀티태스킹(다중작업) → 가상 터미널을 이용한 다중 작업
- 다중 스레드 → 동시에 여러 개의 네트워크 작업, 인터넷 서비스
- 다양한 파일 시스템 : 리눅스 파일시스템, 윈도우 파일 시스템, 네트워크 파일 시스템

✓ 리눅스의 장점

- POSIX(Portable Operating System Interface) 표준 기반, 유닉스와 호환 가능
- 상대적으로 안정적인 동작, 하드웨어 등 시스템 자원 효율성 → 많은 사용자가 동시에 접속해서 사용하는 서비스 운영에 유리

✓ 리눅스의 단점

- 공개 운영체제로 기술 지원의 한계, 보안상 취약점 노출 가능성, 한글 지원 미흡 → 최근에는 많이 보완됨

01. 리눅스 일반 – 리눅스의 이해

리눅스 배포판의 특징

✓ 리눅스 배포판이란?

- 리눅스 커널, GNU 소프트웨어, 응용 프로그램 등을 함께 묶어서 구성한 리눅스 운영체제 → 목적에 따라 다양한 배포판이 존재

✓ 리눅스 배포판의 종류와 특징

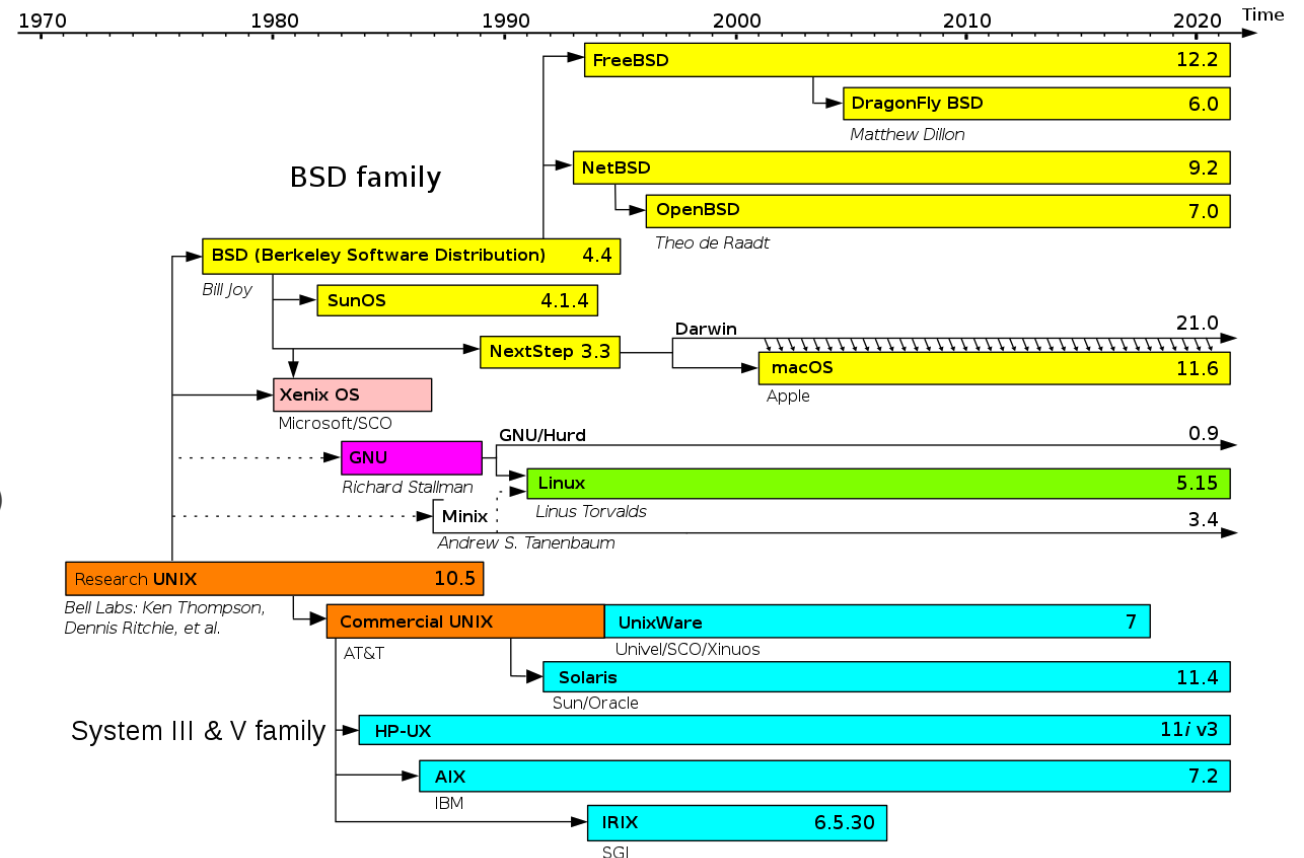
- 슬랙웨어 리눅스(Slackware Linux) : 1992년 패트릭 볼커딩, 가장 먼저 대중화
- 데비안(Debian) : 데비안 프로젝트, GNU의 공식 후원, apt 패키지 매니저를 이용한 소프트웨어 관리
- 우분투(Ubuntu) : 데비안 GNU/리눅스에 기초, 유니티(데스크톱 환경), 사용자 편의성에 초점
- 레드햇 : 미국 레드햇사, RHEL(기업용)과 페도라로 나뉨
- RHEL(Red Hat Enterprise Linux) : 레드햇 사용 리눅스, 서브스크립션 형태의 요금, 출시 후 7년간 기술지원
- 페도라 : 레드햇과 커뮤니티의 지원으로 개발, 6개월 마다 새로운 버전, 지원 기간은 13개월
- CentOS : RHEL과 완벽하게 호환, 무료 운영체제
- 수세(SuSe) : 독일에서 출시, 롤링 릴리즈 (Rolling Release)

01. 리눅스 일반 – 리눅스의 이해

리눅스의 역사

✓ 출제 예상 문제 대비

- 켄 톰슨 (UNIX)
 - 데니스 리치 (UNIX C언어로 재개발)
 - 리처드 스톨먼 (GNU 프로젝트, FSF, GPL)
 - 앤드루 타넨바움 (MINIX)
 - 리누스 토발즈
(MINIX 커널 수정, GNU 시스템 커널, 리눅스 커널)



〈출처 : https://commons.wikimedia.org/wiki/File:Unix_timeline.en.svg, public domain〉

01. 리눅스 일반 – 리눅스의 이해

오픈소스 라이선스

✓ GNU, FSF, 오픈소스 소프트웨어, 오픈소스 라이선스

- GNU(GNU's Not UNIX) : 리처드 스톨만의 자유 소프트웨어 재단 프로젝트, 완전한 유닉스 호환 시스템이 목적
- FSF(Free Software Foundation) : 리처드 스톨만이 설립, 실행/복제/배포/학습/개작/개선의 자유, 유료/무료 배포
- 오픈소스 소프트웨어 (OSS) : 자유 소프트웨어 (Free Software) 대신 사용하게 된 용어, 자유와 무료의 용어 혼돈을 고려
- 오픈소스 라이선스 : OSI(Open Source Initiative)가 주도, OSS 개발자와 이용자 간 사용 방법, 조건, 범위 등 명시한 계약

01. 리눅스 일반 – 리눅스의 이해

오픈소스 라이선스

✓ 오픈소스 라이선스의 종류와 특징

- GNU GPL (General Public License) : FSF가 만든 Free 소프트웨어 라이선스, GPL 코드 사용시 해당 프로그램이 GPL
- GNU LGPL (Lesser/Library GPL) : LGPL 코드/라이브러리의 단순한 이용 시 해당 프로그램은 소스코드 공개 의무 없음
- BSD (Berkeley Software Distribution) : 버클리 캘리포니아 대학, 소스코드 공개 의무 없음, 상업 소프트웨어 사용 가능, OpenCV 등
- 아파치 (Apache) 라이선스 : 아파치 소프트웨어 재단, 파생된 프로그램 제작, 저작권 양도, 전송 가능, 상업적 사용 가능
- MIT (Massachusetts Institute of Technology) 라이선스 : BSD 라이선스 계열, Node.js, X 윈도우 시스템 (X11) 등
- MPL (Mozilla Public License) : 모질라 재단, 소스코드와 실행파일의 저작권 분리, MPL 소스코드를 수정한 2차 소스코드는 MPL로 공개 및 원저작자에게 수정 부분 통지, 실행파일은 독점 라이선스 가능

01. 리눅스 일반 – 리눅스의 이해

리눅스 주요 디렉토리의 특징

디렉터리	저장 내용
/	<ul style="list-style-type: none">· 파일 시스템이 있는 최상위 디렉터리로 루트 디렉터리라고도 함· 모든 디렉터리의 출발점인 동시에 다른 시스템과의 연결점이 되는 디렉터리
/boot	부트 디렉터리로 부팅 시 커널 이미지와 부팅 정보 파일 저장
/proc	<ul style="list-style-type: none">· 시스템 정보 디렉터리이며 커널 기능을 제어하는 역할· 현재 실행되는 프로세스와 실제로 사용되는 장치, 하드웨어 정보 저장
/lib	<ul style="list-style-type: none">· 공유 라이브러리 디렉터리· 커널 모듈 파일들과 프로그램 실행을 지원해 주는 라이브러리 저장
/bin	<ul style="list-style-type: none">· 기본적인 명령어가 저장된 디렉터리· root 사용자와 일반 사용자가 함께 사용할 수 있는 명령어 디렉터리

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

01. 리눅스 일반 – 리눅스의 이해

리눅스 주요 디렉토리의 특징

디렉터리	저장 내용
/dev	<ul style="list-style-type: none">· 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	<ul style="list-style-type: none">· 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

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

01. 리눅스 일반 – 리눅스의 이해

리눅스 주요 디렉토리의 특징

디렉터리	저장 내용
/tmp	<ul style="list-style-type: none">· 각종 프로그램이나 프로세스 작업을 할 때 임시로 생성되는 파일 저장· 모든 사용자에게 대해서 읽기와 쓰기가 허용· 스티키 비트(Sticky bit) 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

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

01. 리눅스 일반 – 리눅스 설치

리눅스 설치

✓ 리눅스 설치의 개요

- 해당 배포판의 홈페이지에서 설치 파일 다운로드, 배포판에 따라 설치 환경 및 과정이 다름
- 리눅스 설치 유형 : Minimal (필수 패키지만), Desktop (개인용 패키지, 문서 작성 SW등),
Server (기본서버, DB서버, Web 서버), Laptop (노트북용 패키지), 가상 호스트 (KVM, XEN),
Software Development Workstation (소스코드 컴파일, 개발 도구), 사용자 설정 시스템
- 디스크 파티션 시 모든 파일 삭제 → 설치 전 모든 파일 백업 필요
- 부팅 드라이브 다시 파티션 → 부트로더 재설치, 운영체제 전체 재설치

✓ 리눅스 설치 위한 하드웨어 정보와 호환성 점검

- 하드웨어 정보 : CPU, Memory, HDD/SDD 및 Interface, 네트워크 장치 및 주소 설정 값, 모니터, 프린터, 키보드, 마우스 등
- 하드웨어 호환성 : 장치 웹사이트 (장치 드라이버, 설명서, 호환성 목록), PNP 기능 도착하지 않을 경우 대비
- 네트워크 설정 정보 : 호스트명, 도메인명, IP 주소, 서브넷 마스크, 게이트웨이 주소, DNS 서버 주소, 무선 네트워크 SSID/보안키

01. 리눅스 일반 – 리눅스 설치

리눅스 설치

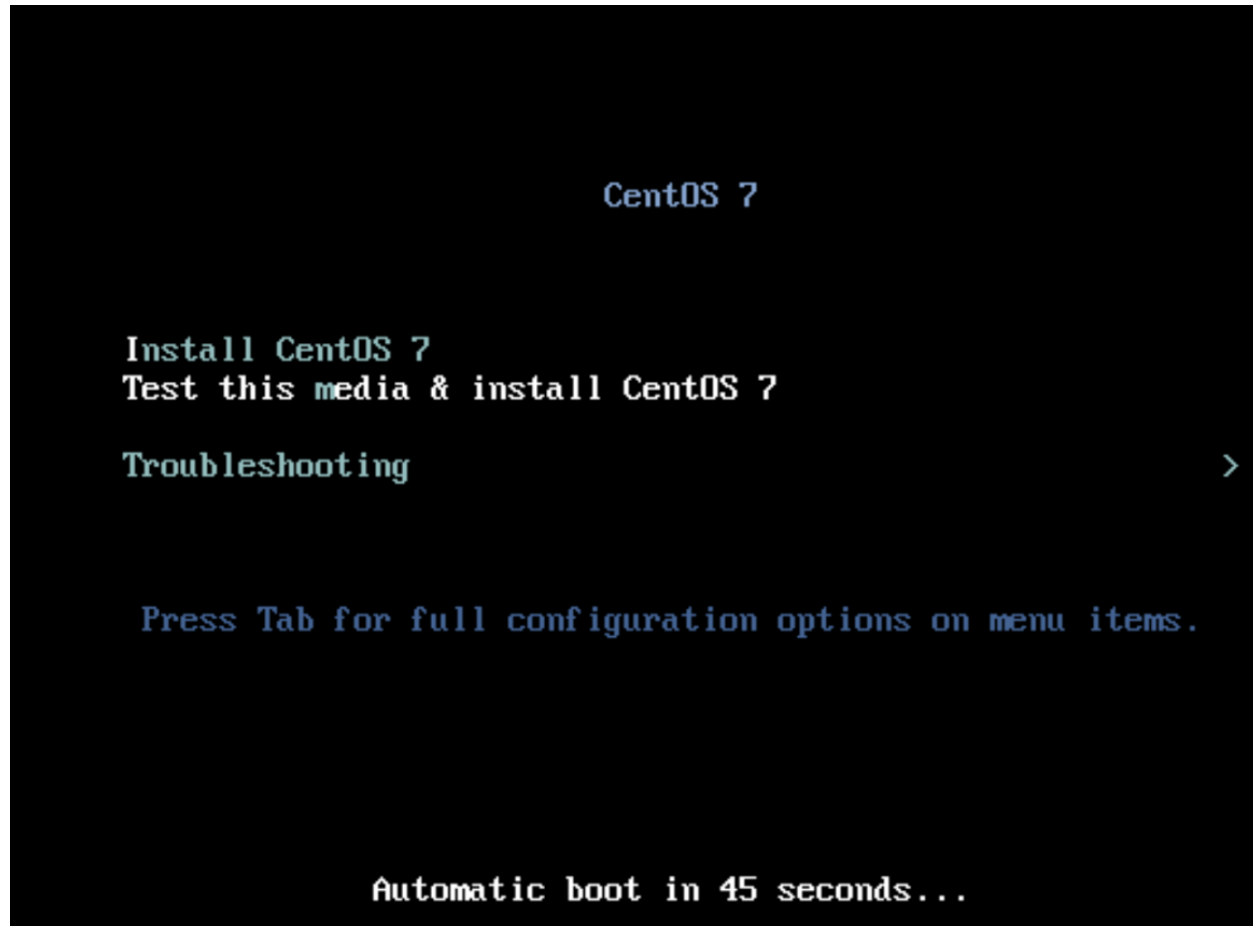
✓ 리눅스 설치 단계

① 설치 초기 화면 (기본)

- Install CentOS7
- Test this media & install CentOS7
(파일 이상여부 체크 후 설치)

② 설치 초기 화면 (Troubleshooting)

- Rescue a CentOS System
- Run a Memory test
- Boot from local drive



01. 리눅스 일반 – 리눅스 설치

리눅스 설치

✓ 리눅스 설치 단계 (계속)

- ③ 언어선택 : 한국어, 영어
- ④ 설치요약 확인 : 현지화(시간, 언어, 키보드), SECURITY, 소프트웨어 (설치소스, 소프트웨어 선택), 시스템 (파티션, 네트워크)
- ⑤ 날짜와 시간 : 수동 설정, NTP(Network Time Protocol) → 시간 동기화(일치) 필요 (장애대응, 분석/로깅 등에 필수)
- ⑥ 설치소스 : 저장소(Repository) 서버 주소 (선택사항), yum이 업데이트 파일을 다운받기 위한 저장소 서버 주소
- ⑦ 설치대상 : 설치 시 사용할 특수 저장장치 (선택사항), SAN / FCoE / iSCSI, Multipath 등
- ⑧ 수동으로 파티션 설정 : 기본 저장 장치 (HDD/SDD 파티션 설정 (필수사항), 자동 혹은 수동 파티션
→ 루트(/)와 swap(RAM의 2배)
- ⑨ 소프트웨어 선택 : 리눅스 사용 목적에 따른 환경 결정,
최소 설치 / 인프라 서버 / 기본 웹 서버 / 가상화 호스트 / GNOME 데스크톱 등 패키지 그룹 결정

01. 리눅스 일반 - 리눅스 설치

리눅스 설치

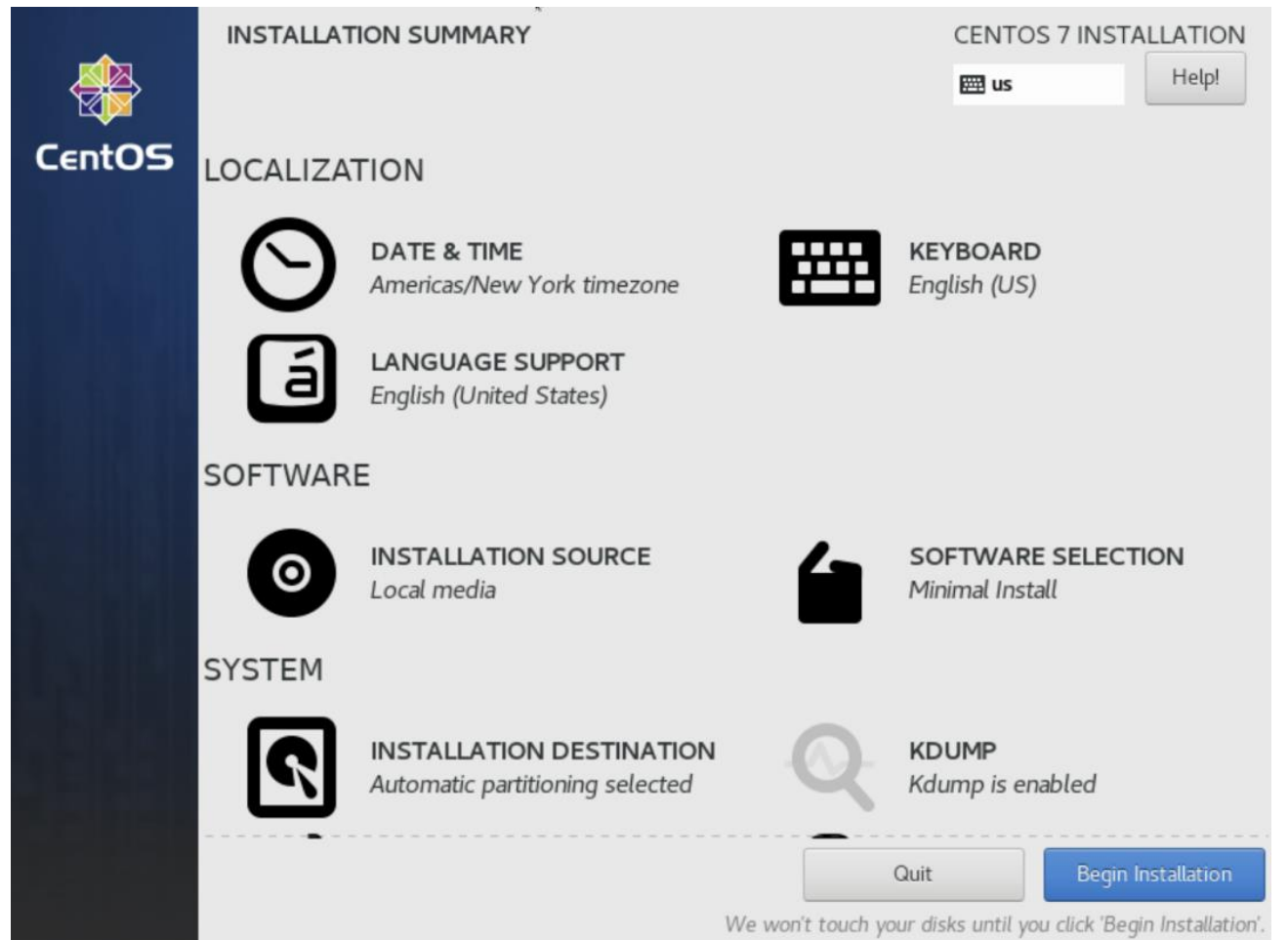
✓ 리눅스 설치 단계 (계속)

⑩ K DUMP 설정

- 커널 충돌(Crash) 발생 시 정보 수집 위한 데이터 저장 및 확인, 물리적 메모리 설정

⑪ 네트워크 설정 및 호스트명 지정

- 고정 주소 / 유동 주소, IP 주소, 서브넷 마스크, 게이트웨이 주소, DNS 서버 주소 등



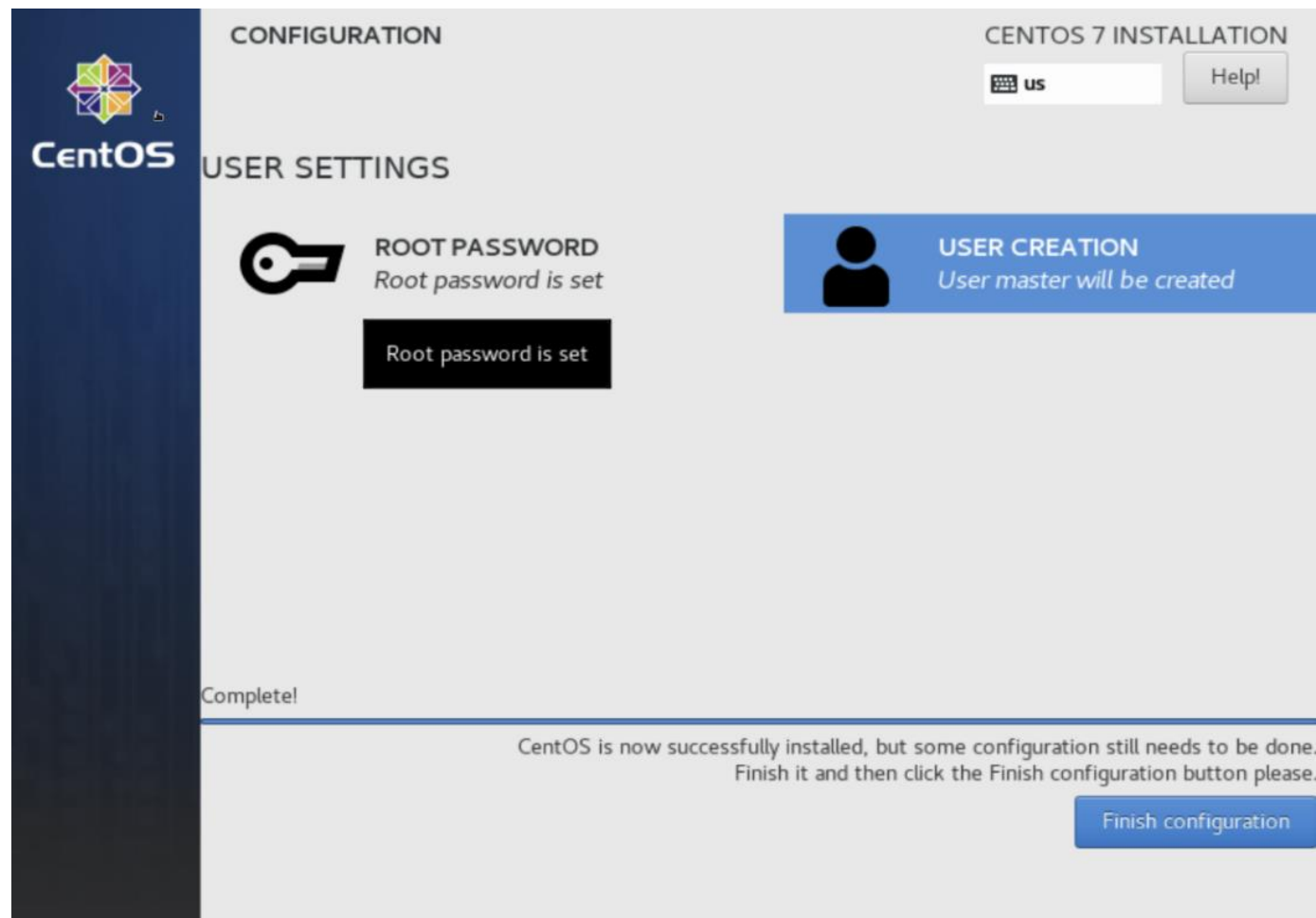
01. 리눅스 일반 - 리눅스 설치

리눅스 설치

✓ 리눅스 설치 단계 (계속)

⑫ 사용자 설정

- ROOT 암호
- 사용자 생성 (암호지정)



01. 리눅스 일반 – 리눅스 설치

파티션과 파일 시스템

✓ 파티션 (Partition)의 개요

- 파티션 : 물리적 디스크 → 여러 개의 논리적 디스크로 분할
- 독립적 파일 시스템 → 관리 효율성 (부팅 시간, 오류 점검 시간, 백업/복구), 관리 안전성 (시스템 손상 시 영향 최소화)
- 상태 정보 확인 : `/proc/partitions`

01. 리눅스 일반 – 리눅스 설치

파티션과 파일 시스템

✓ 파티션(Partition)의 구분

주 파티션	<ul style="list-style-type: none"> • 부팅이 가능한 기본 파티션 • 하나의 하드디스크에 최대 4개의 주 파티션 분할 가능 • 하드디스크를 4개 이상의 파티션으로 사용해야 할 때 하나의 확장 파티션을 설정하여 확장 파티션 안에 여러 개의 논리 파티션을 분할하여 데이터 저장
확장 파티션	<ul style="list-style-type: none"> • 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	<ul style="list-style-type: none"> • 확장 파티션 안에 생성되는 파티션 • 논리 파티션은 12개 이상 생성하지 않는 것을 권고하며 5번 이후의 번호가 붙여 짐
스왑 파티션	<ul style="list-style-type: none"> • 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

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

01. 리눅스 일반 – 리눅스 설치

파티션과 파일 시스템

✓ 디스크와 장치명

- 파티션 이름 : 디스크의 장치 파일명 + 숫자
- 디스크와 장치명의 예 : `/dev/hda3` → `/dev` + `/hd` + `a` + `3`
 - `/dev` : 시스템 디바이스 파일들이 저장되는 디렉터리
 - `/hd` : IDE 혹은 ATA(AT Attached) 방식
`/sd` : SCSI(Small Computer System Interface) 혹은 USB 방식
 - `a` : 하나의 케이블에 연결된 디스크의 우선순위
 - `3` : 파티션 번호
 - 1~4 : primary 혹은 extended 파티션
 - 5~ : logical 파티션
- 마운트(mount)시 지정된 장치명을 사용 : 예) `/dev/fd0`, `/dev/hda` 등

01. 리눅스 일반 – 리눅스 설치

파티션과 파일 시스템

✓ 파일 시스템 (File System)의 개요

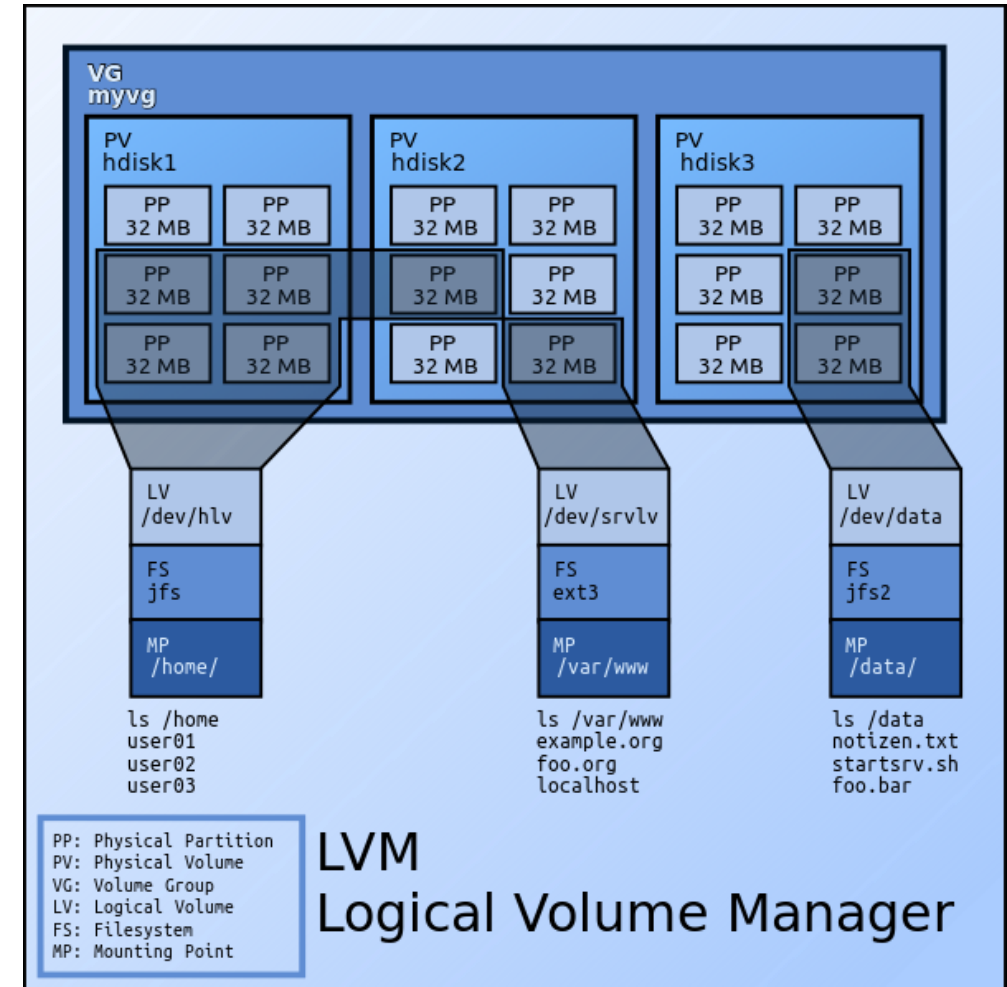
- 파일 시스템 : 디스크의 파티션에 파일을 사용할 수 있도록 구성해 놓은 것 혹은 구성 방식 → 규칙, 구조
- 파일 시스템의 유형
 - 리눅스 전용 : ext1, ext2, ext3, ext4
 - 저널링 파일 시스템 : JFS, XFS
 - 네트워크 파일 시스템 : SMB, NFS
 - 클러스터링 파일 시스템 : GFS
 - 장치 파일 시스템 : IS9660, UDF
 - 운영체제별 파일 시스템 : FAT32, NTFS, HPFS

01. 리눅스 일반 - 리눅스 설치

LVM (Logical Volume Manager)

✓ LVM 의 개요

- LVM : 다수의 물리적인 하드디스크를 논리적으로 통합하여 하나의 파일 시스템으로 구성하여 사용하는 방식
→ 큰 용량의 저장장치, 파티션 구성에 유연성
- LVM 구성 볼륨의 종류
 - 물리 볼륨 : 개별 물리적인 디스크
 - 볼륨 그룹 : 여러 개의 물리 볼륨을 하나의 그룹으로 구성
 - 논리 볼륨 : 볼륨 그룹의 일부를 논리 그룹으로 나누어 사용



〈출처 : <https://commons.wikimedia.org/wiki/File:Lvm.svg>〉

01. 리눅스 일반 – 리눅스 설치

RAID (Redundant Array of Independent Disks)

✓ RAID 의 개요

- RAID : 여러 개의 물리적 디스크를 하나의 논리적 디스크로 구성하여 사용하는 방법
→ 데이터 분할 저장, 데이터 중복 저장, 오류 관리 → 고용량, 신뢰성, 성능 향상
- RAID 구성 방식 (RAID 0 ~ RAID 6 등)
 - RAID 0 : 스트라이핑, 디스크에 나누어 저장 → 고용량, 빠른 I/O, 고장에 취약
 - RAID 1 : 미러링, 디스크에 중복 저장 → 결함허용, 2배의 저장 공간 필요
 - RAID 0 + 1 : 스트라이핑 후 미러링, 빠른 I/O, 느린 복구 시간
 - RAID 1 + 0 : 미러링 후 스트라이핑, 손실된 데이터의 빠른 복원

01. 리눅스 일반 - 리눅스 설치

fdisk - 파티션 관리 명령어

✓ fdisk - 파티션을 관리하는 명령어

- 주요 기능 : 파티션의 생성, 수정, 삭제
- 기본 사용법 : fdisk 장치명 → fdisk sdb
 - a : 부팅 파티션 설정
 - l : 지원하는 파티션 목록 표시
 - n : 새로운 파티션 생성
 - t : 파티션 종류 변경
 - w : 파티션 정보 저장
 - p : 파티션 정보 확인
 - q : 종료

```
Command (m for help): m
```

```
Command action
```

```
a  toggle a bootable flag
b  edit bsd disklabel
c  toggle the dos compatibility flag
d  delete a partition
g  create a new empty GPT partition table
G  create an IRIX (SGI) partition table
l  list known partition types
m  print this menu
n  add a new partition
o  create a new empty DOS partition table
p  print the partition table
q  quit without saving changes
s  create a new empty Sun disklabel
t  change a partition's system id
u  change display/entry units
v  verify the partition table
w  write table to disk and exit
x  extra functionality (experts only)
```

01. 리눅스 일반 – 리눅스 설치

리눅스 부팅 과정

✓ 리눅스의 부팅 과정의 주요 순서

- ① ROM-BIOS : POST (Power On Self Test), 부트로더 (LILO, GRUB) 로딩
- ② 부트로더 : 리눅스 커널 선택 및 로딩, 커널에 의한 부팅 시작, 스와퍼 프로세스 호출
- ③ 스와퍼 프로세스 : 장치 드라이브 초기화, init 프로세스 수행
- ④ init 프로세스 : /etc/inittab 의 내용에 따라 필요한 프로세스를 실행
- ⑤ 부팅 레벨 결정
- ⑥ rc.sysinit 스크립트 : /etc/rc.d/rc.sysinit 스크립트 실행 → 시스템 초기화 작업
- ⑦ rcX.d 스크립트 : /etc/rc.d/rcX.d 스크립트 실행 → 부팅 레벨에 따른 스크립트 순차 실행
- ⑧ X 윈도우 : 부팅 레벨이 5일 경우 X 윈도우 실행 (GUI)

01. 리눅스 일반 – 리눅스 설치

부트로더 (bootloader)

✓ 부트로더 (bootloader, bootstrap loader, boot manager)의 개요

- 운영체제를 주기억장치로 적재하는 프로그램 → 필요한 초기 작업, 운영체제(멀티부팅)를 주 메모리로 복사, 운영체제 실행
- 임베디드 시스템 부트로더 : PC BIOS와 OS Loader 기능 수행
- 부트로더의 저장 위치 : MBR(Master Boot Record), 하드디스크의 첫번째 섹터 (부트 섹터), 512 바이트

주 파티션마다 고유의 부트 섹터가 할당됨.

✓ 부트로더의 종류

- LILO(Linux Loader) : 리눅스 전용
- GRUB2(Grand Unified Bootloader version 2)
 - 편리한 설정 및 사용, 대화형 UI, /boot/grub2/grub.cfg 읽기전용 설정파일 (/etc/grub2.cfg로 링크)
 - /etc/default/grub과 /etc/grub.d 로 환경 설정 → grub2-mkconfig 로 적용
 - 부팅 시 부트 정보 수정 지원, 멀티부팅, 파일 시스템과 커널 실행 형식 인식 후 부팅

01. 리눅스 일반 – 리눅스 설치

런레벨 (Run Level)

✓ 런레벨의 개요

- 서비스(프로그램)의 실행을 단계별로 구분, 0~6의 총 7개의 레벨로 구성 → /etc/inittab에 설정, init 프로세스가 참고
 - 런레벨 0 : Halt, 시스템 중지, 기본값으로 설정 불가 (init 0 과 동일)
 - 런레벨 1 : Single User mode, 네트워크 등 사용하지 않음, 시스템 복구 등 관리 목적, root 계정으로 로그인
 - 런레벨 2 : Multi-User, without NFS, 네트워크를 사용하지 않음
 - 런레벨 3: Full Multi-User, CLI, 일반적인 다중 사용자 모드
 - 런레벨 4 : Unused, 사용자가 정의해서 사용
 - 런레벨 5 : X11, GUI 다중 사용자, 최근 리눅스의 기본값
 - 런레벨 6 : Reboot, 시스템 재 부팅, 기본값으로 설정 불가 (init 6 과 동일)
- /etc/inittab 형식
 - id(코드/이름) : runlevel : action(프로세스 실행 지시어) : process(실행 프로세스) → runlevel 명령으로 레벨 확인

01. 리눅스 일반 – 리눅스 설치

로그인과 로그아웃

✓ 리눅스의 로그인

- 로그인 과정
 - ① 패스워드 확인 → /etc/passwd의 패스워드 필드
 - ② Shell 설정 실행 → PATH경로, Terminal설정, 환경변수, 로그인 시 실행 명령, 로그인 메시지 출력
 - ③ 로그인 Shell 실행 → /etc/passwd의 마지막 필드
- 로그인 메시지 설정 파일
 - /etc/issue (로그인 전 콘솔 접속 시), /etc/issue.net (로그인 전 원격 접속 시)
 - /etc/motd (로그인 후 메시지)
- Shell의 종류와 프롬프트
 - sh (본셸, \$), csh (C셸, %), ksh (콘셸, \$), bash (배시셸, \$)
 - Shell 프로그램 저장 위치 : /bin/

01. 리눅스 일반 – 리눅스 설치

로그인과 로그아웃

✓ 리눅스의 로그아웃

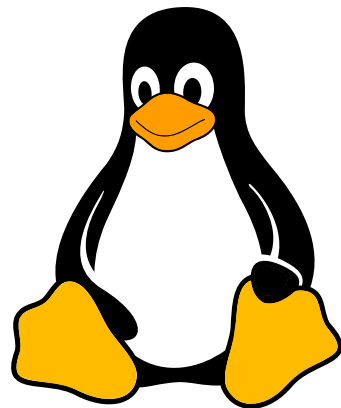
- 로그아웃 방법 : logout, exit, CTRL+D
- 전체 사용자 로그아웃 타임아웃 : /etc/profile 의 TMOUT 값 설정 (초)
- 특정 사용자 로그아웃 타임아웃 : ~/.bashrc 의 TMOUT 값 설정 (초)

→ source 명령으로 설정 값 적용

리눅스마스터 (2급)

- 리눅스 기본명령어 -

김윤수
(kchris000@gmail.com)



01. 리눅스 기본 명령어

리눅스 기본 명령어

- ✓ 사용자 생성 및 계정 관리
- ✓ 디렉터리 및 파일 관련 명령어
- ✓ 리다이렉션과 정규 표현식
- ✓ 기타 명령어

02. 리눅스 기본 명령어

기본 명령어

✓ which - 명령어 경로 확인

- PATH 환경변수의 경로에 따라 사용자가 실행할 명령어의 위치 혹은 alias를 표시
- which [명령어]

```
[LinuxMaster] #which bash  
/usr/bin/bash
```

```
[LinuxMaster] #which pwd  
/usr/bin/pwd
```

02. 리눅스 기본 명령어

기본 명령어

- ✓ whereis - 대상 명령어의 절대 경로, 소스 파일, 설정 파일, 메뉴얼 등 정보를 표시
 - whereis <-옵션> [명령어]
 - 옵션 : -b (바이너리만 검색), -m (지정한 메뉴얼 섹션만 검색), -M (메뉴얼 페이지 위치 제한), -u (특정 파일 제외)

```
[LinuxMaster] #whereis find
```

```
find: /usr/bin/find /usr/share/man/man1p/find.1p.gz /usr/share/man/man1/find.1.gz
```

02. 리눅스 기본 명령어

기본 명령어

✓ alias - 명령어의 별칭 만들기

- 명령어에 별칭을 부여하여 자주 사용하는 명령을 편리하게 사용
- alias [별칭 = '명령']

```
[LinuxMaster] #alias showall='ls -al'
```

```
[LinuxMaster] #showall
```

```
합계 1968
```

```
dr-xr-x---. 10 root root  4096 1월 5 13:28 .
```

```
dr-xr-xr-x. 19 root root  4096 9월 7 02:18 ..
```

```
drwx-----. 3 root root  4096 3월 9 2021 .ansible
```

```
-rw----- 1 root root 15821 1월 5 15:44 .bash_history
```

```
-rw-r--r--. 1 root root   18 12월 29 2013 .bash_logout
```

```
...
```

02. 리눅스 기본 명령어

기본 명령어

✓ unalias - 명령어의 별칭 삭제하기

- 만들어 놓은 별칭을 삭제

- unalias [별칭]

➤ unalias -a : 설정해 놓았던 별칭을 모두 삭제

```
[LinuxMaster] #unalias showall
```

```
[LinuxMaster] #showall
```

```
-bash: showall: command not found
```

02. 리눅스 기본 명령어

기본 명령어

- ✓ PATH 환경 변수 - 명령어(프로그램)가 위치한 경로를 찾는 순서 지정
 - PATH 환경 변수에 지정된 디렉터리를 순서대로 탐색하여 명령어(프로그램)을 찾고 실행
 - 확인 : echo \$PATH
 - 설정: export PATH = [설정값:설정값:...]
 - ~/.bash_profile 에 기본값 설정. (예) export PATH=\$PATH:/var/myfile

※ 기타 환경 변수의 확인 및 설정 방법도 동일함

```
[LinuxMaster]#echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin
[LinuxMaster]#export MYVAR=LinuxMaster
[LinuxMaster]#echo $MYVAR
LinuxMaster
```

02. 리눅스 기본 명령어

기본 명령어

- ✓ man - 리눅스 명령어의 메뉴얼(manual) 확인
 - 리눅스에서 사용할 수 있는 명령어들의 메뉴얼을 확인
 - man <섹션><-옵션> [명령어]
 - 섹션 : 1 (일반), 2 (시스템 호출), 3 (C 표준 함수), 4 (장치/특수 파일), 5 (특정한 파일), 6 (게임/화면보호기)
7 (리눅스 파일 표준, 프로토콜, 시그널 목록), 8 (시스템 관리 명령어, 데몬 정보), 9 (커널 관리 정보)
 - -h (사용법 표시), -a (모든 페이지 표시), -w (검색 문자의 메뉴얼 페이지의 위치 표시), -f (키워드 검색하여 페이지 표시)

```
[LinuxMaster]#man find
```

```
FIND(1)
```

```
General Commands Manual
```

```
FIND(1)
```

```
NAME
```

```
find - search for files in a directory hierarchy
```

```
...
```

02. 리눅스 기본 명령어

기본 명령어

- ✓ **apropos** - 명령어 메뉴얼에서 지정한 키워드가 포함된 것을 검색
 - **whatis** 데이터베이스를 참조하여 지정한 문자열을 포함한 것을 검색하여 표시
 - **apropos** [문자열]

```
LinuxMaster]#apropos find
```

```
ffs (3)      - find first bit set in a word
```

```
ffsl (3)     - find first bit set in a word
```

```
ffsll (3)    - find first bit set in a word
```

```
tfind (3)    - manage a binary tree
```

```
ttyslot (3)  - find the slot of the current user's terminal in some file
```

```
...
```

02. 리눅스 기본 명령어

기본 명령어

- ✓ **manpath** - man 명령이 참조하는 메뉴얼의 경로 표시
 - man 명령이 페이지를 찾기 위해 탐색하는 경로 정보를 표시
 - **manpath**

```
LinuxMaster]#manpath  
/usr/local/share/man:/usr/share/man/overrides:/usr/share/man
```


02. 리눅스 기본 명령어

기본 명령어

✓ info - 리눅스 명령어의 사용법, 옵션 등 표시

- 리눅스 명령어의 사용법, 옵션 등을 표시. man 명령보다 제한적인 기능 제공
- info [명령어]

```
[LinuxMaster]#info find
```

```
File: find.info, Node: Invoking find, Next: Invoking locate, Up: Reference
```

```
8.1 Invoking 'find'
```

```
=====
```

```
find [-H] [-L] [-P] [-D DEBUGOPTIONS] [-OLEVEL] [FILE...] [EXPRESSION]
```

```
...
```

02. 리눅스 기본 명령어

기본 명령어

- ✓ **whatis** - 명령어의 기능을 간략히 표시
 - **whatis** 데이터베이스에서 키워드를 검색하여 일치할 경우 명령어의 기능을 표시
 - **whatis** [명령어]

```
LinuxMaster]#whatis find
```

```
find (1p)      - find files
```

```
find (1)      - search for files in a directory hierarchy
```

02. 리눅스 기본 명령어

사용자 생성 및 계정 관리

✓ useradd , adduser - 사용자 계정 생성

- 사용자 계정과 관련 정보를 생성
- useradd <-옵션> [계정명]
 - 옵션 : -s (로그인 셸 지정), -d (홈 디렉터리 지정), -f (패스워드 만료 후 계정 영구 정지까지의 기간)
-e (계정의 유효 기간, 임시계정 생성), -c (/etc/passwd에 사용자 설명 추가), -G (소속할 그룹 추가)
- 기본 홈 디렉터리 : /home/계정명
- 계정 정보 파일 : /etc/passwd, /etc/shadow, /etc/group에 정보 추가

```
[LinuxMaster]#useradd -d /CHRIS chris
```

02. 리눅스 기본 명령어

사용자 생성 및 계정 관리

- ✓ passwd - 사용자 패스워드 변경
 - /etc/shadow 파일에 패스워드 변경 및 저장
 - passwd <-옵션> [계정명]
 - 옵션 : -S (상태표시), -d (삭제), -l (잠금), -u (잠금해제)
 - 상태표시 정보 : PS (정상), NP (패스워드 없음), LK (잠금 혹은 패스워드 없는 상태)

```
[LinuxMaster]#passwd -S chris
chris LK 2023-01-08 0 99999 7 -1 (비밀 번호 잠금.)
[LinuxMaster]#passwd chris
...
passwd: 모든 인증 토큰이 성공적으로 업데이트 되었습니다.
[LinuxMaster]#passwd -S chris
chris PS 2023-01-08 0 99999 7 -1 (비밀번호 설정, SHA512 암호화. )
```

02. 리눅스 기본 명령어

사용자 생성 및 계정 관리

✓ su - 사용자 전환 (switch user)

- 로그아웃 없이 다른 사용자의 계정으로 로그인, 다른 사용자의 권한을 획득

- su <-옵션> [계정명] <셸변수>

➤ 옵션 : - 혹은 -l 혹은 -login (사용자 환경변수 적용), -s (셸 지정), -c (계정 변환없이 지정한 명령어 실행)

```
[LinuxMaster] #su chris
```

```
[LinuxMaster] #pwd
```

```
/root
```

```
[LinuxMaster] #exit
```

```
exit
```

```
[LinuxMaster] #su - chris
```

```
[chris@118-27-119-212 ~]$ pwd
```

```
/CHRIS
```

02. 리눅스 기본 명령어

사용자 생성 및 계정 관리

✓ 사용자 설정 관련 환경 구성 파일

- /etc/default/useradd : 사용자를 추가할 때 기본 설정 정보로 활용
 - SKEL : 사용자 홈 디렉터리로 복사될 초기 환경 설정 파일이 저장된 디렉터리 지정
- 텍스트 편집기 (vi, nano 등) 혹은 'useradd -D 옵션'으로 변경

```
[LinuxMaster] #cat /etc/default/useradd
# useradd defaults file
GROUP=100
HOME=/home
INACTIVE=-1
SHELL=/bin/bash
SKEL=/etc/skel
...
```

02. 리눅스 기본 명령어

사용자 생성 및 계정 관리

✓ 사용자 설정 관련 환경 구성 파일 (계속)

- /etc/passwd : 사용자 계정 정보 저장
 - 파일 형식) username : password : uid : gid : comment : home-directory : shell
 - password : 패스워드 → pwconv 실행 (x로 표시, /etc/shadow에 패스워드 기록), pwunconv로 비활성화
- /etc/shadow : 사용자 패스워드 암호화 파일
 - 파일 형식) username : password : lastchange : mindays : maxdays : warndays : inactive : expire : flag
 - lastchange : 1970년 1월 1일 기준 최근 비밀번호 변경일
 - mindays : 비밀번호 변경 후 재설정 가능한 최소 기간
 - inactive : 비밀번호 만료 후 계정 사용 불가능 까지의 유예 기간
 - expire : 계정 사용 정지 날자 (1970년 1월 1일 기준으로 이후의 날자 수)
 - flag : 기본 0으로 설정

02. 리눅스 기본 명령어

사용자 생성 및 계정 관리

✓ 사용자 설정 관련 환경 구성 파일 (계속)

- /etc/login.defs : 사용자 계정 설정 시 기본값
 - 예) MAIL_DIR (메일 디렉터리), PASS_MAX_DAYS (패스워드 변경 없이 사용할 수 있는 최대 일 수)
PASS_MIN_LEN (패스워드 최소 길이), CREATE_HOME (홈 디렉터리 생성 여부) 등

```
[LinuxMaster]#cat /etc/login.defs
```

```
...
```

```
# PASS_MAX_DAYS Maximum number of days a password may be used.
```

```
# PASS_MIN_DAYS Minimum number of days allowed between password changes.
```

```
...
```

```
PASS_MAX_DAYS 99999
```

```
PASS_MIN_DAYS 0
```

```
...
```


02. 리눅스 기본 명령어

사용자 생성 및 계정 관리

- ✓ **usermod** - 사용자 계정 정보 수정
 - 사용자 홈 디렉터리, 그룹, 유효기간, 셸 등 정보를 변경
 - **usermod** <-옵션> [계정명]
 - 옵션 : -u (새로운 UID), -g (새로운 GID), -G (새로운 보조 그룹), -d (새로운 홈 디렉터리), -s (새로운 셸), -c (새로운 주석), -I (새로운 계정명)

```
[LinuxMaster] #usermod -h
```

```
Usage: usermod [options] LOGIN
```

Options:

```
-c, --comment COMMENT    new value of the GECOS field
```

```
...
```

02. 리눅스 기본 명령어

사용자 생성 및 계정 관리

- ✓ userdel - 사용자 계정 정보 삭제
 - /etc/passwd, /etc/shadow, /etc/group에서 사용자 정보 삭제
 - usermod <-옵션> [계정명]
 - 옵션 : -r (사용자의 메일 파일과 홈 디렉토리 모두 삭제)

```
[LinuxMaster] #userdel -h
```

```
Usage: userdel [options] LOGIN
```

```
Options:
```

```
-f, --force          force some actions that would fail otherwise
                     e.g. removal of user still logged in
                     or files, even if not owned by the user
```

```
...
```

02. 리눅스 기본 명령어

사용자 생성 및 계정 관리

✓ chage - 패스워드 만료 정보 변경

- chage <-옵션> [계정명]

➤ 옵션 : -l (계정 정보 표시), -m (최소 사용 일자), -M (사용 가능 일자), -E (만기일), -W (만기 전 지정 날짜 부터 경고)

```
[[LinuxMaster]#chage -l chris
```

```
마지막으로 암호를 바꾼 날           : 1월 08, 2023
```

```
암호 만료                           :안함
```

```
암호가 비활성화 기간                :안함
```

```
계정 만료                           :안함
```

```
암호를 바꿀 수 있는 최소 날 수      : 0
```

```
암호를 바꿔야 하는 최대 날 수      : 99999
```

```
암호 만료 예고를 하는 날 수        : 7
```

```
...
```

02. 리눅스 기본 명령어

사용자 생성 및 계정 관리

- ✓ groupadd - 그룹을 추가하는 명령어
 - groupadd <-옵션> [그룹명]
 - 옵션 : -g (GID 지정), -r (시스템 그룹 시 500번 이하의 값 중 가능 높은 범위 할당)
- ✓ groupdel - 그룹을 삭제하는 명령어 (단, 소속 계정이 있을 경우 삭제 안됨)
 - groupdel [그룹명]
- ✓ groupmod - 그룹 설정 변경
 - groupmod <-옵션> [그룹명]
 - 옵션 : -g (GID 변경), -n (새로운 그룹명 설정),

02. 리눅스 기본 명령어

사용자 생성 및 계정 관리

- ✓ 그룹 설정 관련 환경 구성 파일
 - /etc/group : 사용자가 소속된 그룹을 설정
 - 파일 형식) `groupname : password : gid : users`
 - `uses` : 콤마(,)로 구분된 사용자 계정
 - /etc/gshadow : 그룹의 패스워드를 MD5로 암호화 하여 저장
 - 파일 형식) `groupname : password : owner : users`
 - `password : !` 이면 암호가 없는 상태
 - `uses` : 콤마(,)로 구분된 사용자 계정

02. 리눅스 기본 명령어

사용자 생성 및 계정 관리

- ✓ users : 로그인한 사용자 정보 표시
 - users <-옵션>
 - 옵션 : --version (명령어 버전 정보)

```
[LinuxMaster] #users
```

```
root
```

02. 리눅스 기본 명령어

사용자 생성 및 계정 관리

✓ who : 접속한 사용자 정보 표시

- 계정명, 터미널 정보, 접속 시간, 접속 서버 등 표시

- who <옵션>

➤ 옵션 : -b (마지막 부팅 시간), -q (로그인 사용자, 사용자 수 표시), -r (현재 실행 레벨 표시)

```
[LinuxMaster] #who -q
```

```
root
```

```
# users=1
```

```
[LinuxMaster] #who -r
```

```
run-level 3 2022-09-07 02:20
```

02. 리눅스 기본 명령어

사용자 생성 및 계정 관리

✓ w : 접속한 사용자 정보 표시

- 서버 시간, 부팅 후 작동 시간, 접속자 수, 접속 자별 평균 부하율, 접속자별 계정명, TTY 이름, 로그인 시간 등
- w
 - JCPU : TTY 장치명에서 사용되는 모든 프로세스의 CPU 사용 시간
 - PCPU : WHAT에 표시된 프로세스의 CPU 사용 시간

```
[LinuxMaster] #w
```

```
17:43:20 up 123 days, 15:26, 1 user, load average: 0.06, 0.03, 0.05
```

```
USER  TTY  FROM      LOGIN@  IDLE   JCPU   PCPU   WHAT
```

```
root  pts/0  10.0.0.7  13:25   0.00s  0.50s  0.01s  w
```


02. 리눅스 기본 명령어

사용자 생성 및 계정 관리

✓ id : 접속한 사용자 정보 표시

- uid, gid, group 정보 표시
- id <-옵션> [계정명]

➤ 옵션 : -u (UID만 표시) - g (GID 만 표시), -G (사용자가 표시된 모든 그룹 표시)

```
[LinuxMaster] #id
```

```
uid=0(root) gid=0(root) groups=0(root)
```

02. 리눅스 기본 명령어

사용자 생성 및 계정 관리

✓ who am i , whoami: 접속해 있는 자신의 정보를 표시

- who am i 혹은 whoami

```
[LinuxMaster] #whoami
```

```
root
```

02. 리눅스 기본 명령어

사용자 생성 및 계정 관리

✓ groups : 사용자가 속한 그룹 목록 표시

- group <계정명>

```
LinuxMaster] #groups
```

```
root
```

```
[LinuxMaster] #groups chris
```

```
chris : chris
```

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ pwd : 현재 작업 디렉터리명 표시

- pwd

```
[LinuxMaster] #pwd
/root
[LinuxMaster] #cd /var/www/
[LinuxMaster] #pwd
/var/www
[LinuxMaster] #cd ~
[LinuxMaster] #pwd
/root
```

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ cd : 디렉터리 이동 (change directory)

- cd [경로명]

➤ 경로명 : ~ (홈 디렉터리), . (현재 디렉터리), .. (상위 디렉터리), / (루트 디렉터리)

```
[LinuxMaster] #pwd
```

```
/root
```

```
[LinuxMaster] #cd /var/www/
```

```
[LinuxMaster] #pwd
```

```
/var/www
```

```
[LinuxMaster] #cd ~
```

```
[LinuxMaster] #pwd
```

```
/root
```

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ mkdir : 디렉터리 만들기

- mkdir <옵션> [디렉터리명]

➤ 옵션 : -m (권한 설정, 기본 755), -p (상위 디렉터리 생성), -v (생성 디렉터리 메시지 표시)

```
[LinuxMaster] #pwd
```

```
/root/Test
```

```
[LinuxMaster] #mkdir One
```

```
[LinuxMaster] #mkdir -m 700 Two
```

```
[LinuxMaster] #ls -al
```

```
...
```

```
drwxr-xr-x  2 root root 4096  1월 8 18:00 One
```

```
drwx-----  2 root root 4096  1월 8 18:04 Two
```

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ rmdir : 디렉터리 삭제하기 (비어있는 디렉터리만 가능)

- rmdir [디렉터리명]

※ rm 명령으로 비어 있지 않은 디렉터리도 삭제할 수 있음.

```
[LinuxMaster] #ls
One Two
[LinuxMaster] #rmdir Two
[LinuxMaster] #ls
One
```

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ ls : 디렉터리의 파일 목록(정보) 표시

- ls <옵션> <디렉터리명>

➤ 옵션 : -a (숨김 파일 포함), -l (자세히 목록 보기) -d (디렉터리 지정), -r (역순으로 표시), -R (하위 디렉터리 포함)

```
LinuxMaster]#ls -al
```

```
합계 16
```

```
drwxr-xr-x  3 root root 4096 1월 8 18:11 .
```

```
dr-xr-x---. 11 root root 4096 1월 8 18:00 ..
```

```
drwxr-xr-x  2 root root 4096 1월 8 18:00 One
```

```
-rw-r--r--  1 root root   0 1월 8 18:11 hell.txt
```

```
-rw-r--r--  1 root root   7 1월 8 18:11 hi.txt
```


02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ cp : 파일, 디렉터리 복사

- cp <옵션> [소스] [타겟]

➤ 옵션 : -b (타겟 파일 존재 시 백업), -f (타겟 파일 존재 시 강제로 삭제 및 복사), -i (타겟 파일 존재 시 덮어쓰기 확인)
-r (하위 폴더 및 파일 포함)

```
[LinuxMaster] #ls
One hell.txt hi.txt
[LinuxMaster] #cp hell.txt hello.txt
[LinuxMaster] #ls
One hell.txt hello.txt hi.txt
```

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ rm : 파일, 디렉터리 삭제

- rm <옵션> [파일명|디렉터리명]

➤ 옵션 : -f (확인없이 삭제), -i (사용자에게 확인), -r (하위 디렉터리 포함, 모든 파일 제거)

```
[LinuxMaster] #ls
One hello.txt hi.txt
[LinuxMaster] #rm hi.txt
rm: remove 일반 파일 'hi.txt'? y
[LinuxMaster] #ls
One hello.txt
```

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ mv : 파일, 디렉터리 이동. 파일명 변경에서 사용

- mv <옵션> [소스] [타겟]

➤ 옵션 : -b (타겟 파일 존재 시 백업), -f (타겟 파일이 있어도 강제 적용), -v (진행 과정의 정보 표시)

```
[LinuxMaster] #ls
One hell.txt hello.txt
[LinuxMaster] #mv hell.txt hi.txt
[LinuxMaster] #ls
One hello.txt hi.txt
```

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

- ✓ touch : 비어있는 파일 생성 혹은 파일의 시간(time stamp) 변경
 - touch <옵션> [파일명]
 - 옵션 : -a (접근시간 변경), -m (수정시간 변경), -c (파일 시간을 현재 시간으로 변경)
-t (특정 파일의 시간 변경), -r 파일A 파일B (파일A의 시간을 파일B와 동일하게 변경)

```
[LinuxMaster] #ls -al
...
-rw-r--r--  1 root root  0 1월 8 18:21 empty.txt
[LinuxMaster] #touch -c empty.txt
[LinuxMaster] #ls -al
...
-rw-r--r--  1 root root  0 1월 8 18:23 empty.txt
```

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ file : 파일의 종류와 속성값 표시

- file <옵션> [파일명]

➤ 옵션 : -b (파일 형식만 표시), -f (지정한 파일만 대상으로 실행), -z (압축 파일 내용 표시), -i (MIME으로 출력)

```
LinuxMaster] #file hi.txt
```

```
hi.txt: ASCII text
```

```
[LinuxMaster] #file -i hi.txt
```

```
hi.txt: text/plain; charset=us-ascii
```

```
[LinuxMaster] #file -b hi.txt
```

```
ASCII text
```

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ find : 지정된 조건에 따라 파일을 검색하는 명령어 (지정한 경로 및 하위 경로를 대상으로 검색)

- find <경로> <-옵션> <정규표현식> <파일명>
 - 옵션 : -name (이름으로 검색), -user (사용자의 파일/디렉터리 검색), -perm (권한으로 검색),
-type (파일 유형으로 검색), -atime 숫자 (숫자 일수 이전에 접근한 파일 검색), -size (파일 크기를 지정하여 검색)
-exec (검색 파일을 대상으로 실행할 명령어)

```
[LinuxMaster] #pwd
```

```
/
```

```
[LinuxMaster] #find -name One
```

```
./root/Test/One
```

✓ locate : 파일의 위치 검색

- locate [파일명]

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ cat : 파일 내용을 출력

- cat <-옵션> [파일명]

➤ 옵션 : -n (행 번호 표시), -b (행 번호 표시, 비어 있는 행 제외), -s (2개 이상 빈 행을 하나로 표시)
-A (텍스트 파일의 개행 문자, 탭 문자 등 확인)

```
LinuxMaster] #cat -n -s /etc/httpd/conf/httpd.conf
```

```
1 #
```

```
2 # This is the main Apache HTTP server configuration file. It contains the
```

```
3 # configuration directives that give the server its instructions.
```

```
4 # See <URL:http://httpd.apache.org/docs/2.4/> for detailed information.
```

```
5 # In particular, see
```

```
6 # <URL:http://httpd.apache.org/docs/2.4/mod/directives.html>
```

```
...
```

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ head : 파일의 앞 부분만 출력

- head <-옵션> [파일명]

➤ 옵션 : -c (지정 숫자의 바이트 만큼 출력), -n (지정 줄 수 만큼 출력, 기본 10줄)

```
[LinuxMaster] #head -n 3 /etc/httpd/conf/httpd.conf
```

```
#
```

```
# This is the main Apache HTTP server configuration file. It contains the  
# configuration directives that give the server its instructions.
```

```
[LinuxMaster] #head -3 /etc/httpd/conf/httpd.conf
```

```
#
```

```
# This is the main Apache HTTP server configuration file. It contains the  
# configuration directives that give the server its instructions.
```


02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ tail : 파일의 마지막 부분만 출력

- tail <-옵션> [파일명]

➤ 옵션 : -c (지정 숫자의 바이트 만큼 출력), -n (지정 줄 수 만큼 출력, 기본 10줄),
-f (새롭게 추가되는 내용을 실시간 출력), -v (파일명 표시), -q (파일명 표시하지 않음)

```
[LinuxMaster] #tail -3 /etc/httpd/conf/httpd.conf
```

```
#
```

```
# Load config files in the "/etc/httpd/conf.d" directory, if any.
```

```
IncludeOptional conf.d/*.conf
```

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ more : 화면 단위로 파일의 내용을 출력

- more <-옵션> [파일명]

➤ 옵션 : -숫자 (페이지 당 표시할 줄 수)

```
[LinuxMaster]#more -4 /etc/httpd/conf/httpd.conf
# This is the main Apache HTTP server configuration file. It contains the
# configuration directives that give the server its instructions.
# See <URL:http://httpd.apache.org/docs/2.4/> for detailed information.
# In particular, see
--More--(1%)
```

➤ 보기 작업 : 다음 페이지 (Space), 다음 줄 (Enter), ½ 페이지 (Ctrl + D), 이전 페이지 (Ctrl + B), 종료 (Q)
문자열 패턴 검색 (/패턴)

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

- ✓ **less** : 텍스트 파일을 한 화면씩 표시
 - 특징 : more와 다르게 커서를 이동할 수 있음

- ✓ **wc** : 파일의 라인 수, 단어 수, 알파벳 수 표시
 - **wc <-옵션> [파일명]**
 - 옵션 : -l (라인수 표시), -w (단어 수 표시), -c (알파벳 수 표시)

```
[LinuxMaster] #cat hi.txt
```

```
hihi
```

```
[LinuxMaster] #wc -w hi.txt
```

```
1 hi.txt
```

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ grep : 특정한 문자열 패턴, 정규식을 이용하여 단어를 검색

- grep <-옵션> [문자열] [파일명]

- 옵션 : -r (모든 디렉터리, 파일), -E (정규표현식), -i (대소문자 무시), -v (검색 대상이 포함되지 않은 라인만 표시)
-n (라인수 표시), -l (문자열이 포함된 파일명 표시)

```
[LinuxMaster]#grep chris /etc/passwd
chris:x:1001:1001::CHRIS:/bin/bash
[LinuxMaster]#cat /etc/passwd | grep chris
chris:x:1001:1001::CHRIS:/bin/bash
```

- 예) grep '^[AB]' sample.txt (A 혹은 B로 시작하는 모든 라인),
grep 'fine\$' sample.txt (fine으로 끝나는 모든 라인)

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ sort : 정렬하여 표시하는 명령어

- sort <-옵션> [파일명]

➤ 옵션 : -b (앞 공백 무시), -o (저장 파일명), -r (역순), -f (대소문자 구분하지 않음), -t (필드 구분자 지정)
-u (중복 행 제거), -m (정렬 파일을 병합), -n (숫자만 비교)

```
[LinuxMaster] #sort -b -u /etc/httpd/conf/httpd.conf | grep '^[a-zA-Z]'
```

```
AddDefaultCharset UTF-8
```

```
DocumentRoot "/var/www/html"
```

```
EnableSendfile on
```

```
ErrorLog "logs/error_log"
```

```
Group apache
```

```
Include conf.modules.d/*.conf
```

```
...
```

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ cut : 구분자를 이용하여 특정 필드를 추출하는 명령어

- cut <-옵션> [파일명]

➤ 옵션 : -b (바이트 수를 기준), -c (문자열, 콤마와 하이픈으로 범위 지정), -f (필드를 기준), -d (구분 문자, 기본 값은 탭)
-z (라인의 구분자로 개행이 아닌 NUL을 이용)

```
[LinuxMaster] #cut -c 1-5 /etc/httpd/conf/httpd.conf | grep '^[a-zA-Z]'
```

Serve

Liste

Inclu

User

Group

...

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

✓ split : 여러개의 파일로 분리하여 저장

- split <-옵션> [파일명]

➤ 옵션 : -b (byte 단위로 분할), -l (라인 수 단위로 분할)

```
[LinuxMaster]#split -l 30 /etc/httpd/conf/httpd.conf
```

```
[LinuxMaster]#ls
```

```
xaa xab xac xad xae xaf xag xah xai xaj xak xal xam
```

02. 리눅스 기본 명령어

디렉터리 및 파일 관련 명령어

- ✓ diff : 파일을 라인 단위로 비교하여 표시
 - diff <-옵션> [파일A|디렉터리A] [파일B|디렉터리B]
 - 옵션 : -b (연속 공백 무시), -i (대소문자 구분 없음), -S (동일한 파일인지 확인), -r (디렉터리 차이점 출력)
- ✓ cmp : 파일을 바이트(문자) 단위로 비교하여 표시
 - cmp <-옵션> [파일A] [파일B]
 - 옵션 : -b (다른 바이트 출력), -l (다른 문자 위치, 문자 출력), -S (비교 결과만 표시)
- ✓ comm : 파일을 라인 단위로 비교하여 표시
 - comm <-옵션> [파일A] [파일B]
 - 옵션 : -1 (파일 A에만 있는 내용 제외), -2 (파일 B에만 있는 내용 제외), -3 (공통으로 존재하는 내용 제외)

02. 리눅스 기본 명령어

리다이렉션과 정규 표현식

✓ 리다이렉션(redirection)의 개념과 활용

- 표준 입력 (stdin, standard input, 키보드), 표준 출력 (stdout, standard output, 모니터), 표준 에러 (stderr, standard error, 모니터)의 방향을 재지정하는 방법

➤ 방법 : > (출력 재지정), >> (추가모드로 출력 재지정), < (입력 재지정)

```
[LinuxMaster]#ls -al > out.txt
```

```
[LinuxMaster]#tail -3 out.txt
```

```
-rw-r--r--  1 root root 979 1월 8 19:25 xak
```

```
-rw-r--r--  1 root root 906 1월 8 19:25 xal
```

```
-rw-r--r--  1 root root  96 1월 8 19:25 xam
```

02. 리눅스 기본 명령어

리다이렉션과 정규 표현식

✓ 파이프(pipe)의 개념과 활용

- 두개 이상의 명령을 상호 연결하여, 한 명령의 출력을 다른 명령의 입력으로 지정하여 실행하는 방법

➤ 방법 : 명령 A | 명령 B | ...

```
LinuxMaster]#ls -al | grep txt
-rw-r--r--  1 root root  0 1월 8 18:23 empty.txt
-rw-r--r--  1 root root  0 1월 8 18:14 hello.txt
-rw-r--r--  1 root root  5 1월 8 18:27 hi.txt
-rw-r--r--  1 root root 1006 1월 8 19:45 out.txt
```

02. 리눅스 기본 명령어

리다이렉션과 정규 표현식

✓ 정규 표현식의 개념과 예

- 특정한 규칙을 갖는 문자열의 집합을 표현하는 데 사용하는 형식 언어
- 정규 표현식의 예
 - ^ (첫 글자), \$ (마지막 글자), . (한 글자), * (0회 이상 반복), [] (2개 이상대체 목록, -로 범위 지정), [^] (대체하지 못할 글자 목록, -로 범위 지정), + (1회 이상 반복), ? (0 혹은 1개만 존재), () (부분 정규식의 시작, 끝), | (구분된 단어 중 최소 1개 이상), {m, n} (m개 이상, n개 이하)
 - 예) a..b (a와 b사이에 2개 문자), ab* (b가 0번 이상), [ab]c (ac, bc), [^ab]c (ac, bc는 안됨), ab+ (b가 1개 이상), ab? (b가 없거나 1개만 존재), a(bcd)+ (bcd가 1번 이상 반복), (a|b)c (a 혹은 b가 1개 이상, ac, bc, abc), a{1,2}b (a가 1~2개, ab, aab)

02. 리눅스 기본 명령어

리다이렉션과 정규 표현식

✓ 정규 표현식의 개념과 예 (계속)

- 정규 표현식 정리

기호	의미
^	라인의 첫 글자
\$	라인의 끝 글자
.	한 글자
*	* 기호 바로 이전의 글자나 정규 표현식이 0회 이상 반복

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

02. 리눅스 기본 명령어

리다이렉션과 정규 표현식

✓ 정규 표현식의 개념과 예 (계속)

- 정규 표현식 정리

기호	의미
[]	<ul style="list-style-type: none">대체 글자 목록을 []에 나열“-”로 문자의 범위 지정
[^]	<ul style="list-style-type: none">대체 못할 글자 목록을 [^]에 나열“-”로 문자의 범위 지정
+	+ 기호 바로 이전 글자나 정규 표현식이 1회 이상 반복

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

02. 리눅스 기본 명령어

리다이렉션과 정규 표현식

✓ 정규 표현식의 개념과 예 (계속)

- 정규 표현식 정리

기호	의미
?	? 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

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

02. 리눅스 기본 명령어

기타 명령어 - 네트워크 관련 명령어

✓ ping - 상대 호스트와 IP 통신이 가능한지 확인하는 명령어

- ping <옵션> [IP주소 | 도메인명]

- 옵션 : -c (요청 수, 기본 무제한), -i (통신을 위한 신호를 전달하는 시간 간격, 기본 1초)
-s (전송할 패킷 크기, 바이트)

```
[LinuxMaster] #ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=118 time=1.16 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=118 time=1.15 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=118 time=1.14 ms
...
```

02. 리눅스 기본 명령어

기타 명령어 - 네트워크 관련 명령어

✓ traceroute - 상대 호스트까지의 경로 표시

- traceroute [IP주소 | 도메인명]

```
[LinuxMaster] #traceroute 8.8.8.8
```

```
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 60 byte packets
```

```
1 v118-27-118-2.2078.static.cnode.io (118.27.118.2) 0.650 ms 0.609 ms 1.549 ms
```

```
2 * * *
```

```
3 150.95.5.117 (150.95.5.117) 0.913 ms 0.899 ms 0.880 ms
```

```
4 150.95.5.102 (150.95.5.102) 1.050 ms 1.041 ms 1.044 ms
```

```
5 150.95.5.77 (150.95.5.77) 1.172 ms 1.148 ms 1.109 ms
```

```
...
```


02. 리눅스 기본 명령어

기타 명령어 - 네트워크 관련 명령어

✓ nslookup - 도메인 정보를 조회하는 명령어

- nslookup <옵션> [도메인명]

- 옵션 : -type=레코드타입 (레코드 타입 별 지정)

- 레코드 타입 : NS (네임서버), MX (메일서버), A (IPv4 주소), AAA (IPv6 주소), CNAME (별칭), SOA (도메인 정보), PTR (IP 주소에 대한 도메인명)

```
[LinuxMaster]#nslookup -type=mx google.com
```

```
Server:      157.7.180.133
```

```
Address:     157.7.180.133#53
```

```
Non-authoritative answer:
```

```
google.com   mail exchanger = 10 smtp.google.com.
```

```
...
```

02. 리눅스 기본 명령어

기타 명령어 - 네트워크 관련 명령어

✓ dig - 도메인 정보를 조회하는 명령어 (domain information groper)

- dig <서버명> [도메인명] <타입>

- 서버명 : 네임서버 지정, 기본값은 /etc/resolv 에 등록된 네임서버 이용

- 타입 : a (IPv4), any (모든 정보), mx (메일서버), ns (네임서버), soa (SOA 정보)

```
[LinuxMaster] #dig google.com
```

```
...
```

```
:: ANSWER SECTION:
```

```
google.com.      300  IN  A   142.250.76.142
```

```
...
```

02. 리눅스 기본 명령어

기타 명령어 - 네트워크 관련 명령어

✓ host - 도메인/호스트 정보를 조회하는 명령어

- host <옵션> [도메인명 | IP 주소] <DNS 서버>

➤ 옵션: -d (디버깅 모드), -l zone (zone 아래의 모든 정보 출력), -r (반복처리 안함), -t (타입 지정)

```
[LinuxMaster]#host google.com
google.com has address 142.250.76.142
google.com has IPv6 address 2404:6800:400a:80e::200e
google.com mail is handled by 10 smtp.google.com.
```

02. 리눅스 기본 명령어

기타 명령어 - 네트워크 관련 명령어

✓ hostname - 시스템(호스트) 이름 확인 및 변경

- hostname <옵션> [파일명]

➤ 옵션: -a (별칭 표시), -d (도메인명 표시), -F (파일에서 호스트명 설정), -v (세부 정보 출력)

```
[LinuxMaster]#hostname DevSmile.com
```

```
[LinuxMaster]#hostname
```

```
DevSmile.com
```

※ 참고 : /etc/sysconfig/network 에 HOSTNAME을 설정하면 리부팅 시에도 호스트이름이 유지됨.

02. 리눅스 기본 명령어

기타 명령어 - 시스템 운영

- ✓ shutdown - 시스템을 종료 혹은 재 부팅하는 명령어
 - shutdown <옵션> <시간> <메시지>
 - 옵션: -r (리부팅), -h (종료), -c (명령 취소), -k (경고 메시지만 출력), -f (리부팅시 fsck 명령 생략), -n (init 호출없이 종료), -t (지장 시간에 재시동)
 - 사용 예) shutdown -h now (즉시 종료), shutdown -r 10:00 (10시에 재부팅)
- ✓ init - 런레벨을 이용한 shutdown 수행
 - init [런레벨]
 - 런레벨 : 0 ~ 6

02. 리눅스 기본 명령어

기타 명령어 - 시스템 운영

- ✓ reboot - 시스템 리부팅 명령어
 - reboot <옵션>
 - 옵션: -f (강제 리부팅)

- ✓ halt - 시스템 종료 명령어
 - halt <옵션>
 - 옵션 : -f (강제 종료)

02. 리눅스 기본 명령어

기타 명령어 - 유틸리티

✓ cal - 달력 출력

- cal <옵션> <날짜>

➤ 옵션: -i (1월 1일 이후 날짜 수), -y (올해 달력 표시)

```
[LinuxMaster]#cal
  1월 2023
일 월 화 수 목 금 토
 1  2  3  4  5  6  7
 8  9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31
```

02. 리눅스 기본 명령어

기타 명령어 - 유틸리티

- ✓ date - 시스템 날짜 표시 혹은 변경
 - date <옵션> <날짜>
 - 옵션: -s (설정, 시간순서는 hh:mm:ss)
- ✓ clear - 화면 지우기
- ✓ tty - 접속에 사용한 장치 정보 표시

```
[LinuxMaster] #tty  
/dev/pts/0
```


02. 리눅스 기본 명령어

기타 명령어 - 유틸리티

- ✓ time - 명령 수행에 소요된 시간 표시
 - 출력결과 : real (총 수행 시간), user (사용자 모드 수행 시간), sys (시스템 호출 수행 시간)

- ✓ wall - 로그인한 사용자에게 메시지 전송
 - wall [메시지]

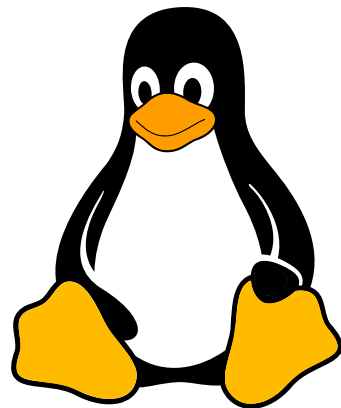
- ✓ write - 특정 사용자에게 메시지 전송
 - write [계정명] <터미널 이름> → 메시지 입력 후 CTRL+D

- ✓ mesg - write로 전송되는 메시지 수신 여부 확인 및 제어
 - mesg [y|n]

리눅스마스터 (2급)

- 파일시스템, 프로세스 -

김윤수
(kchris000@gmail.com)



03. 파일 시스템과 프로세스

파일 시스템과 프로세스

- ✓ 권한 및 그룹 설정
- ✓ 파일 시스템의 관리
- ✓ 셸(Shell)과 환경 설정
- ✓ 프로세스의 개념과 관련 명령어

03. 파일 시스템과 프로세스

권한 및 그룹 설정

✓ 리눅스의 파일 권한 관리

- 모든 파일(디렉토리)에 접근권한(permission)과 소유권을 설정하여 관리
- 권한 : 사용자, 그룹이 해당 파일을 어떻게 사용할 수 있는지 설정

```
[LinuxMaster]#ls -al
```

합계 24

```
drwxr-xr-x  3 root root 4096 1월 10 20:53 .
```

```
dr-xr-x---.11 root root 4096 1월 10 20:48 ..
```

```
-rw-r--r--  1 chris chris  12 1월 10 20:53 chrisMsg.txt
```

```
-rwxr-xr-x  1 root root   60 1월 10 20:51 list.sh
```

```
-rw-r--r--  1 root root   12 1월 10 20:49 msg.txt
```

```
drwxr-xr-x  2 root root 4096 1월 10 20:49 mydirectory
```

➤ 파일유형+접근권한, 물리적 파일 연결개수 (하드링크), 소유자, 소유 그룹, 크기, 마지막 변경 날짜+시간, 파일명

03. 파일 시스템과 프로세스

권한 및 그룹 설정

✓ chown - 파일(디렉터리)의 사용자(그룹) 소유권 변경

- chown <옵션> [소유자] [:그룹명] [파일명]

➤ 옵션: -R (하위 디렉터리 포함)

```
[LinuxMaster]#ls -al
...
-rw-r--r-- 1 root root  6 1월 10 21:18 sample.txt
[LinuxMaster]#chown chris:chris sample.txt
[LinuxMaster]#ls -al
...
-rw-r--r-- 1 chris chris  6 1월 10 21:18 sample.txt
```

03. 파일 시스템과 프로세스

권한 및 그룹 설정

- ✓ chgrp - 파일(디렉터리)의 그룹 소유권 변경
 - chgrp <옵션> [그룹명] [파일명]
 - 옵션: -R (하위 디렉터리 포함)

- ✓ chmod - 파일(디렉터리)의 접근 권한 변경
 - chmod <옵션> [권한] [파일명]
 - 옵션: -R (하위 디렉터리 포함)
 - chmod 명령의 다양한 사용 방식
 - 예) chmod +x file, chmod -x file, chmod u+x file, chmod g=rx file, chmod o-x file, chmod a+r file, chmod 766 file

03. 파일 시스템과 프로세스

권한 및 그룹 설정

- ✓ umask - 파일/디렉터리 생성시 기본으로 적용할 접근권한 지정
 - 주의 : 파일은 '666 - umask' 값, 디렉터리는 '777 - umask' 값으로 설정.
 - umask <옵션> [설정값]
 - 옵션: -S (문자로 표시)

```
[LinuxMaster] #umask
0022
[LinuxMaster] #umask 002
[LinuxMaster] #touch new.txt
[LinuxMaster] #ls -al
...
-rw-r--r--  1 root root  12 1월 10 20:49 msg.txt
-rw-rw-r--  1 root root   0 1월 10 22:38 new.txt
```

03. 파일 시스템과 프로세스

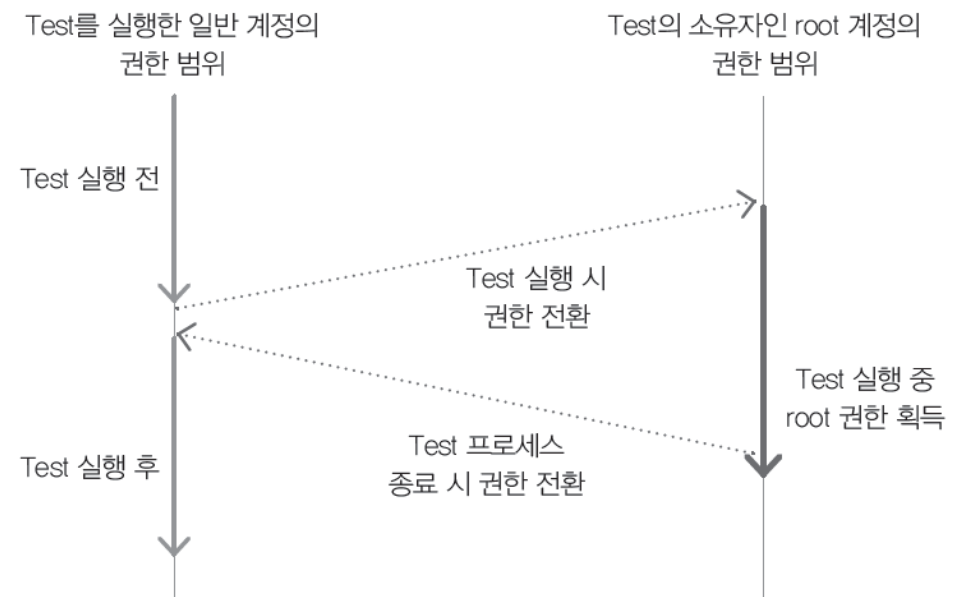
권한 및 그룹 설정

✓ 리눅스의 파일의 특수 권한

- 대상 파일을 사용하는 프로세스가 실행되는 동안 설정된 접근권한으로 전환하여 사용하는 특징 → 유효사용자/유효그룹이 변경

- SetUID : 소유자 권한으로 실행 (4000)
- SetGID : 소유 그룹 권한으로 실행 (2000)
- s (실행파일), S (일반파일)

```
[LinuxMaster] #chmod u+s test.sh
[LinuxMaster] #ls -al test.sh
-rwsr-xr-x 1 root root 60 1월 10 23:11 test.sh
[LinuxMaster] #chmod 2644 test.sh
[LinuxMaster] #ls -al test.sh
-rw-r-Sr-- 1 root root 60 1월 10 23:11 test.sh
```



〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

03. 파일 시스템과 프로세스

권한 및 그룹 설정

✓ 리눅스의 파일의 특수 권한

- (공유) 폴더에 대하여 소유자 혹은 root 만 파일을 수정하거나 삭제할 수 있는 특징

➤ Sticky bit (1000)

➤ t 로 표시

```
[LinuxMaster] #mkdir tmp
```

```
[LinuxMaster] #chmod +t tmp
```

```
[LinuxMaster] #ls -al
```

```
...
```

```
drwxrwxr-t 2 root root 4096 1월 10 23:16 tmp
```

03. 파일 시스템과 프로세스

권한 및 그룹 설정

✓ 디스크 쿼터 (Disk Quota)

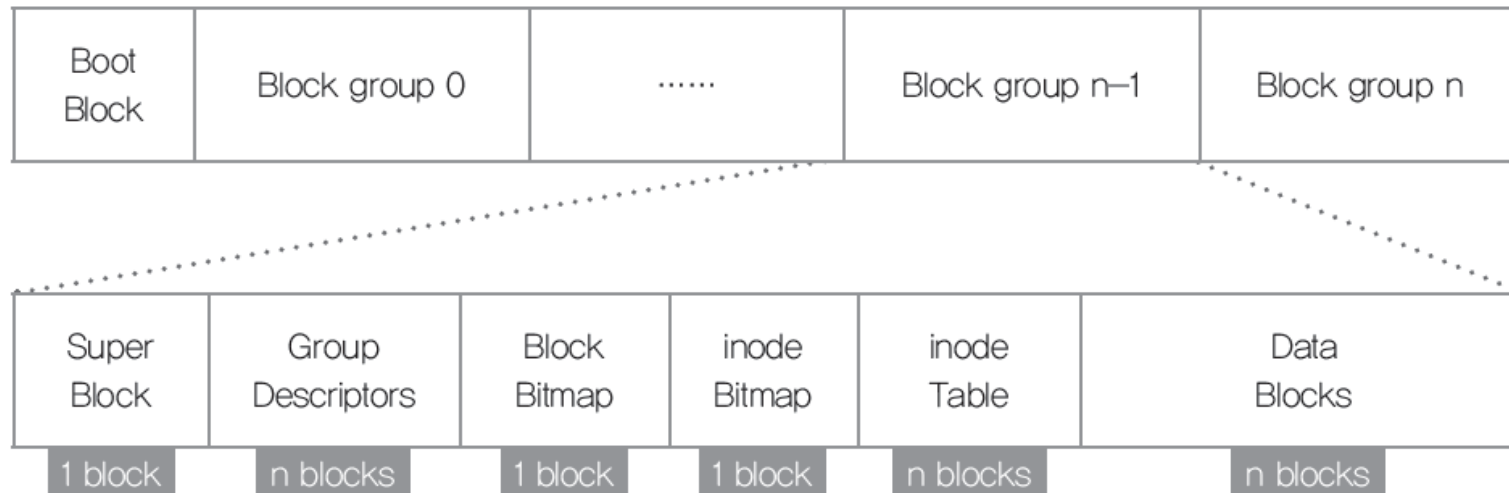
- 사용자(그룹)이 파일 시스템에서 생성(사용)할 수 있는 파일의 용량(개수)
 - inode 개수 제한, 할당된 디스크 블록 수 제한 → 사용자(그룹)별, 파일 시스템별 동작
- 디스크 쿼터 설정 파일 및 관련 명령어
 - 설정 파일 (/etc/fstab), 사용자 쿼터 파일 (aquota.user), 그룹 쿼터 파일 (aquota.group)
 - 마운팅, 재마운팅 : `mount -o remount /QUOTA`
 - `quotaoff` (비활성화), `quotaon`(활성화), `quotacheck`(파일시스템의 디스크 사용 상태 점검)
 - `setquota` (명령어를 이용한 사용자/그룹 쿼터 설정), `edquota` (편집기를 이용한 사용자/그룹 쿼터 설정)

03. 파일 시스템과 프로세스

파일 시스템의 관리

✓ 파일 시스템의 개요

- 스토리지 장치(디스크) 상에 파일을 관리할 수 있도록 만들어 놓은 구조 혹은 관리하는 방식 → 서버 파일 접근, 가상 파일까지 확대
- 리눅스의 파일 시스템 구조



〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

- Super Block (파일 시스템 관련 정보, 블록크기 및 개수, inode 개수), Group Descriptor Table (Block 그룹 관리)
- Block Bitmap (블록의 사용 상태), inode (파일 이름을 제외한 정보, 고유번호, 파일 형태, 크기, 위치, 소유자 등)

03. 파일 시스템과 프로세스

파일 시스템의 관리

✓ 리눅스 파일 시스템 종류와 특징

파일 시스템	설명
ext(ext1)	<ul style="list-style-type: none">• 리눅스 초기에 사용되던 파일 시스템이며 호환성이 없음• ext2의 원형• 2GByte의 데이터와 파일명을 255자까지 지정 가능• 파일 접근에 대한 타임 스탬프, 아이노드 수정 지원 불가
ext2	<ul style="list-style-type: none">• ext 파일 시스템의 다음 버전• 고용량 디스크 사용을 염두하고 설계된 파일 시스템• 쉽게 호환되며 업그레이드도 쉽게 설계되어 있음

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

03. 파일 시스템과 프로세스

파일 시스템의 관리

✓ 리눅스 파일 시스템 종류와 특징 (계속)

파일 시스템	설명
ext3	<ul style="list-style-type: none">• ext2의 확장판• 리눅스의 대표적인 저널링(journaling)을 지원하도록 확장된 파일 시스템• ACL(Access Control List)을 통한 접근 제어 지원
ext4	<ul style="list-style-type: none">• ext2 및 ext3와 호환성이 있는 확장 버전• 파일에 디스크 할당 시 물리적으로 연속적인 블록을 할당• 64비트 기억 공간 제한을 없앴• 16 TeraByte의 파일을 지원

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

03. 파일 시스템과 프로세스

파일 시스템의 관리

✓ 리눅스 파일 시스템 종류와 특징 (저널링 파일 시스템)

파일 시스템	설명
JFS	<ul style="list-style-type: none">• Journaling File System의 약자• IBM사의 독자적인 저널링 파일 시스템• GPL로 공개하여 현재 리눅스용으로 개발
xfs	<ul style="list-style-type: none">• eXtended File System• 고성능 저널링 시스템• 64비트 주소를 지원하며 확장성이 있는 자료 구조와 알고리즘 사용• 파일 수에 관계없이 예상치 못한 상황으로부터 신속한 복구와 재시작 가능• 데이터 읽기/쓰기 트랜잭션으로 성능 저하를 최소화• 64비트 파일 시스템으로 큰 용량의 파일도 다룰 수 있음• 높은 확장성과 처리량을 가짐

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

03. 파일 시스템과 프로세스

파일 시스템의 관리

✓ 리눅스 파일 시스템 종류와 특징 (저널링 파일 시스템 - 계속)

파일 시스템	설명
ReiserFS	<ul style="list-style-type: none">독일의 한스 라이저(Hans Reiser)가 개발한 파일 시스템리눅스용 저널링 파일 시스템 중에서 가장 안정적이라는 평가를 받음모든 파일 객체들을 B트리에 저장, 간결한 색인화 된 디렉터리 지원

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

03. 파일 시스템과 프로세스

파일 시스템의 관리

✓ 리눅스 파일 시스템 종류와 특징 (네트워크 파일 시스템)

파일 시스템	설명
SMB	<ul style="list-style-type: none">· 삼바(samba) 파일 시스템을 마운트 지정· Server Message Block(서버 메시지 블록)· 윈도우 계열 OS 환경에서 사용되는 파일/프린터 공유 프로토콜· 리눅스, 유닉스 계열 OS와 윈도우 OS와의 자료 및 하드웨어 공유· 윈도우에서 표준 클라이언트 환경으로 처음부터 구성되었고 NFS 유닉스 표준 응용부터 운용상 쉽다는 장점이 있음
CIFS	<ul style="list-style-type: none">· Common Internet File System· SMB를 확장한 파일 시스템· SMB를 기초로 응용하여 라우터를 뛰어넘어 연결할 수 있는 프로토콜

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

03. 파일 시스템과 프로세스

파일 시스템의 관리

✓ 리눅스 파일 시스템 종류와 특징 (네트워크 파일 시스템 - 계속)

파일 시스템	설명
NFS	<ul style="list-style-type: none">• Network File System(네트워크 파일 시스템)• 썬마이크로시스템(SUN)이 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

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

03. 파일 시스템과 프로세스

파일 시스템의 관리

- ✓ 리눅스 파일 시스템 종류와 특징 (지원 가능한 기타 파일 시스템)

파일 시스템	설명
FAT	<ul style="list-style-type: none">• Windows NT가 지원하는 파일 시스템 중 가장 간단한 시스템• FAT로 포맷된 디스크는 클러스터 단위로 할당• 클러스터 크기는 볼륨 크기에 따라 결정• 읽기 전용, 숨김, 시스템 및 보관 파일 특성만 지원• 삼바(samba) 파일 시스템을 마운트 지정
VFAT (Virtual FAT)	<ul style="list-style-type: none">• FAT 파일 시스템이 확장된 것으로 FAT와 호환• FAT보다 제한이 적음• 파일 이름도 최고 255자까지 만들 수 있음• 공백이나 여러 개의 구두점도 포함• 대소문자는 지정한 대로 보존되기는 하나 구별하지는 않음

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

03. 파일 시스템과 프로세스

파일 시스템의 관리

- ✓ 리눅스 파일 시스템 종류와 특징 (지원 가능한 기타 파일 시스템 - 계속)

파일 시스템	설명
FAT32	<ul style="list-style-type: none">• SMB를 확장한 파일 시스템• Windows 95, 98 및 ME와 같은 버전과 다중 부팅을 구성• 32GB보다 큰 파티션을 만들 수 없고 파티션에 4GB를 초과하는 파일을 저장할 수 없음
NTFS	<ul style="list-style-type: none">• 윈도우에서 사용하는 파일 시스템• 안정성이 뛰어나고 대용량 파일도 저장• 안정성, 자세한 사용자 제한, 보안성 등이 FAT32보다 뛰어남• 파일 크기 및 볼륨은 이론상으로 최대 16EB(ExaByte=10의18승byte)이나 실질적으로는 2TB로 한계가 있음

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

03. 파일 시스템과 프로세스

파일 시스템의 관리

✓ 리눅스 파일 시스템 종류와 특징 (지원 가능한 기타 파일 시스템 - 계속)

파일 시스템	설명
ISO 9660	<ul style="list-style-type: none">• CD-ROM의 표준 파일 시스템• 1988년에 제정된 표준
UDF	<ul style="list-style-type: none">• Universal Disk Format의 약자로 최신 파일 시스템 형식• OSTA(Optical Storage Technology Association)에 의해 개발• 광학 매체용 파일 시스템 표준• ISO 9660 파일 시스템을 대체하기 위한 것으로 대부분 DVD에서 사용• DVD 멀티미디어 디스크들은 MPEG 오디오 및 비디오 스트림을 담기 위해 사용
HPFS	<ul style="list-style-type: none">• OS/2 운영체제를 위해 만들어진 파일 시스템• 1988년 발표, 마이크로소프트와 IBM이 공동 개발

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

03. 파일 시스템과 프로세스

파일 시스템의 관리

- ✓ mount - (스토리지) 장치를 연결하여 디렉토리 처럼 사용
 - 설정 파일 : /etc/fstab (부팅 시 참조하여 마운트), /etc/mtab (마운트된 블록 시스템 정보)
 - mount <옵션> [장치명] [디렉터리명]
 - 옵션: -a (/etc/fstab의 내용 모두 마운트), -n (/etc/mtab 설정정보 기록 안함), -f (마운트 테스트), -t (파일 시스템), -o (remount, ro, rw, sync, noatime 등 추가 설정)
- ✓ unmount - 마운트를 해제하는 명령어
 - unmount <옵션> [장치명] [디렉터리명]
 - 옵션: -a (/etc/fstab의 내용 모두 마운트 gowp), -n (/etc/mtab 설정정보 기록 안함), -t (파일 시스템)

03. 파일 시스템과 프로세스

파일 시스템의 관리

- ✓ eject - (미디어) 장치를 해제하고 장치 제거
 - eject <옵션> [장치명]
 - 옵션: -r (CD-ROM 장치), -f (Floppy 장치)

- ✓ mkfs - 파일 시스템 생성
 - mkfs <옵션> [장치명]
 - 옵션: -V (자세한 정보 표시), -t (파일 시스템), -c (Bad Block 검사), -l (지정한 파일명에서 Bad Block 목록 읽기),
-v (진행 상태 표시)

03. 파일 시스템과 프로세스

파일 시스템의 관리

- ✓ mk2fs - ext2, ext3, ext4 리눅스 파일 시스템 생성
 - mk2fs <옵션> [장치명]
 - 옵션: -t (파일타입), -b (블록 크기, 바이트), -i (inode 당 바이트 수), -j (저널링 파일 시스템 ext3), -R (RAID4)

- ✓ fsck - 파일시스템 점검 및 복구
 - /lost+found 디렉터리에 손상된 파일 관리 → 정상 복구 된 후 삭제
 - fsck <옵션> [장치명]
 - 옵션: -A (/etc/fstab의 모든 시스템 점검), -a (자동 복구), -r (확인 후 복구), -t (파일 시스템)

03. 파일 시스템과 프로세스

파일 시스템의 관리

✓ e2fsck - ext2, ext3, ext4 리눅스 파일 시스템 점검 및 복구

- e2fsck <옵션> [장치명]

- 옵션: -p (자동 복구), -n (모든 확인에 no로 응답), -y (모든 확인에 yes로 응답), -c (Bad Block 점검), -f (문제가 없는 시스템까지 강제 점검)

✓ du - 디렉터리 별 디스크 사용량 확인

- du <옵션> [파일/디렉터리]

- 옵션: -n (용량단위로 표시),
-a (모든 파일 대상),
-m (MB단위로 표시),
-k (KB단위로 표시), -s (총 합계만 표시),
-c (사용정보 표시 후 합계 표시)

```
[LinuxMaster]#du master/  
4   master/tmp  
8   master/mydirectory  
32  master/
```


03. 파일 시스템과 프로세스

파일 시스템의 관리

- ✓ df - 마운트 된 저장장치의 용량 표시 → 파티션 단위로 사용량 확인, 기본 1024byte 블록단위 출력
 - df <옵션> <파일명>
 - 옵션: -n (용량 단위로 표시), -T (파일시스템 유형 정보, 파티션 정보), -t (표시 파일 시스템 유형), -a (모든 파일 시스템), -k (--block-size=1K), -i (inode 사용률)

```
[LinuxMaster]#df
```

Filesystem	1K-blocks	Used	Available	Use%	Mounted on
devtmpfs	496808	0	496808	0%	/dev
tmpfs	507252	0	507252	0%	/dev/shm
tmpfs	507252	51312	455940	11%	/run
tmpfs	507252	0	507252	0%	/sys/fs/cgroup
/dev/vda2	103077796	5635476	93022332	6%	/
tmpfs	101452	0	101452	0%	/run/user/0

03. 파일 시스템과 프로세스

파일 시스템의 관리

- ✓ /etc/fstab - 파일 시스템 정보 저장 및 관리하는 설정 파일, 부팅 시 마운트 정보 포함

```
[LinuxMaster] #cat /etc/fstab
```

```
#
```

```
# /etc/fstab
```

```
# Created by anaconda on Wed Nov 25 03:32:03 2020
```

```
#
```

```
# Accessible filesystems, by reference, are maintained under '/dev/disk'
```

```
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
```

```
#
```

```
UUID=e29905e8-a03a-448f-836c-bafc868a02a3 / ext4 defaults 1 1
```

```
/var/spool/swap/swapfile none swap defaults 0 0
```

03. 파일 시스템과 프로세스

셸(Shell)과 환경 설정

✓ 리눅스의 셸(Shell)의 개요

- 명령어 해석기 : 사용자의 명령어 ↔ 커널
 - 로그인 시 사용자별 셸 지정
 - 리다이렉션, 파이프, 셸 스크립트, 포그라운드/백그라운드 프로세스 실행
- 셸의 분류
 - 본셸 (Bourne Shell) : \$ 프롬프트 → korn shell, bash shell (리눅스 표준셸, POSIX호환), z shell (확장 기능 제공)
 - C 셸 : % 프롬프트 → tc shell (BSD 계열에서 많이 사용, 명령어 편집 기능)

03. 파일 시스템과 프로세스

셸(Shell)과 환경 설정

✓ 리눅스의 셸(Shell)의 개요 (계속)

- 사용할 수 있는 셸의 목록과 설정

- /etc/shells

- /etc/passwd 의 마지막 필드

```
[LinuxMaster] #cat /etc/shells
```

```
/bin/sh
```

```
/bin/bash
```

```
/usr/bin/sh
```

```
/usr/bin/bash
```

```
...
```

```
[LinuxMaster] #echo $SHELL
```

```
/bin/bash
```

03. 파일 시스템과 프로세스

셸(Shell)과 환경 설정

✓ 환경 변수와 셸 변수의 특징

환경 변수(전역 변수)	셸 변수(지역 변수)
전체 셸에서 사용 가능한 전역 변수	현재 로그인 셸에서만 사용 가능한 지역 변수
서브 셸에 기능 상속 가능	서브 셸에 기능 상속 불가능
환경 변수 확인 명령 : env	셸 변수 확인 명령 : set

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

- 주요 환경 변수

- PATH, HOME, HOSTNAME, USER, DISPLAY, PS1, PWN, SHELL, TERM, TMOUT, LANG, PRINT, MAIL

- 환경 변수의 설정과 해제

- 설정 : export 변수명=값 → \$변수명으로 참조

- 해제 : unset 변수명

03. 파일 시스템과 프로세스

셸(Shell)과 환경 설정

✓ 환경 설정 파일 - 셸 시작 시 참조하여 자동 실행

- 주요 환경 설정 파일 (bash shell)
 - 공통 설정 : /etc/profile, /etc/bashrc
 - 사용자 설정 : ~/.bash_profile (로그인 시), ~/.bash_history (명령어 재사용), ~/.bashrc (alias와 자동실행), ~/.bash_logout (로그아웃 직전)
- /etc/profile.d : 특정 응용 프로그램 시작 시 자동 실행할 스크립트 경로 저장

03. 파일 시스템과 프로세스

셸(Shell)과 환경 설정

✓ History 기능 - .bash_history에 저장된 명령어 재 사용

- 사용 방법

- history : 명령어 목록 출력
- 상/하 화살표 → 이전/이후 명령어
- !! (마지막 명령어), !n (n번째 명령어), !-n (현재 이전 n번째 명령어), !string (string으로 시작하는 최근 명령어)
!\$ 혹은 !!\$ (마지막 명령의 인자), !* (마지막 명령의 모든 인자), !?string? (string을 포함하는 가장 최근 명령어)

- History 관련 환경 변수

- HISTSIZE, HISTFILE, HISTFILESIZE, HISTTIMEFORMAT, HISTCONTROL

03. 파일 시스템과 프로세스

프로세스의 개념과 관련 명령어

✓ 프로세스의 개요

- 프로그램이 실행된 상태 → CPU, 메모리 등 자원을 할당받아 실행된 상태
 - PID (Process ID)로 관리, init (최초 실행 프로세스, PID는 1) → CentOS7 부터 systemd가 최초 실행되면 PID가 1이 됨
- 프로세스의 유형

포그라운드 프로세스 (Foreground)

- 사용자와 상호작용하는 프로세스
- 터미널에 직접 연결되어 입출력을 주고받는 프로세스
- 명령 입력 후 수행 종료까지 기다려야 하는 프로세스
- 화면에서 실행되는 것이 보이는 프로세스
- ㉠ 응용 프로그램, 명령어 등

백그라운드 프로세스 (Background)

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

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

03. 파일 시스템과 프로세스

프로세스의 개념과 관련 명령어

✓ 프로세스 실행 방식 및 관련 용어

- `fork()` : 기본 프로세스를 복제하여 실행 → 부모 프로세스와 자식 프로세스의 관계
 - `exec()` : 새로운 프로세스로 변경되어 실행 → 하나의 프로세스만 남게 됨
 - 데몬(daemon) : (부팅 시 자동으로 실행되는) 백그라운드 프로세스
 - 일반적으로 서비스에 대한 요청을 담당
 - 데몬 실행 방법 : Stand Alone (메모리에 단독 상주, `httpd` 등),
inetd 타입 데몬 (커널 2.4부터 `xinetd` 슈퍼 데몬에 의해 실행, `FTP` 등)
- ※ CentOS7 부터는 `systemd`에 의해 관리
- 시그널(Signal) : 프로세스에게 전달할 수 있는 이벤트(신호) → `kill` 명령 사용, `kill -l`로 가능한 시그널 목록 확인
 - 대표적인 시그널 : `SIGINT` (2), `SIGKILL` (9), `SIGTERM` (15), `SIGSTOP` (19), `SIGSTP` (20, `CTRL+Z`, 대기 전환)

03. 파일 시스템과 프로세스

프로세스의 개념과 관련 명령어

✓ ps - 실행 중인 프로세스의 목록/상태 정보 표시 → CPU 사용도가 낮은 순

- ps <옵션>

➤ 옵션: a (실행 중인 모든 프로세스), e (모든 프로세스), u (사용자 이름, 시작 시간), x (터미널 정보, 모든 프로세스), l (상세 정보 표시)

```
[LinuxMaster]#ps axl | grep http
4      0   958      1  20    0 548284 18088 poll_s Ss   ?           13:28 /usr/sbin/httpd -DFOREGROUND
0      0  7147 15610  20    0 116976  1020 -      R+   pts/0       0:00 grep --color=auto http
5     48 11883   958  20    0 551228 15916 inet_c S    ?           0:00 /usr/sbin/httpd -DFOREGROUND
5     48 11884   958  20    0 551352 16348 inet_c S    ?           0:00 /usr/sbin/httpd -DFOREGROUND
5     48 11885   958  20    0 551356 15496 inet_c S    ?           0:00 /usr/sbin/httpd -DFOREGROUND
5     48 11886   958  20    0 551484 15564 inet_c S    ?           0:00 /usr/sbin/httpd -DFOREGROUND
5     48 11887   958  20    0 551220 19424 inet_c S    ?           0:00 /usr/sbin/httpd -DFOREGROUND
5     48 12104   958  20    0 551228 15580 inet_c S    ?           0:00 /usr/sbin/httpd -DFOREGROUND
5     48 15541   958  20    0 551344 16100 inet_c S    ?           0:00 /usr/sbin/httpd -DFOREGROUND
5     48 15542   958  20    0 551084 15808 inet_c S    ?           0:00 /usr/sbin/httpd -DFOREGROUND
```

03. 파일 시스템과 프로세스

프로세스의 개념과 관련 명령어

✓ pstree - 실행 중인 프로세스의 목록을 tree 형식으로 표시

- pstree <옵션>

➤ 옵션: -p (PID 표시), -n (PID 정렬), -a (명령 인자 포함), -h (현재 프로세스와 부모 프로세스 강조)

```
[LinuxMaster] #pstree -n
systemd└─systemd-journal
        ├─lvmetad
        ├─systemd-udev
        ├─auditd───{auditd}
        ├─abrt
        ├─lsmd
        └─abrt-watch-log
...
```

03. 파일 시스템과 프로세스

프로세스의 개념과 관련 명령어

- ✓ jobs - 작업 (프로세스)의 동작 상태 표시
 - jobs <옵션> <작업번호>
 - 옵션: -l (프로세스 그룹 ID 표시), -n (프로세스 그룹 중 대표 PID 표시), -p (PID를 한 라인씩 표시)
- ✓ fg - 백그라운드에서 포그라운드로 작업 전환
 - fg <작업번호> → 작업 번호가 없을 경우 현재 백그라운드 작업을 대상으로 함

03. 파일 시스템과 프로세스

프로세스의 개념과 관련 명령어

✓ 프로세스 관리 명령의 사용 예

```
[LinuxMaster]#ping 8.8.8.8 > NULL &  
[1] 8480  
[LinuxMaster]#jobs  
[1]+  Running                ping 8.8.8.8 > NULL &  
[LinuxMaster]#fg  
ping 8.8.8.8 > NULL  
^Z  
[1]+  Stopped                ping 8.8.8.8 > NULL  
[LinuxMaster]#fg  
ping 8.8.8.8 > NULL  
^C[LinuxMaster]#
```

03. 파일 시스템과 프로세스

프로세스의 개념과 관련 명령어

- ✓ kill - 프로세스에 시그널(이벤트) 전달
 - kill <옵션> <PID|작업번호>
 - 옵션: -l (시그널 종류), -s (시그널 번호 혹은 시그널 이름 지정), -1 (HUP, 재시작), -9 (SIGKILL, 강제종료)
- ✓ killall - 동일한 데몬의 여러 프로세스를 한번에 종료
 - killall <옵션> [프로세스명]
 - -g (그룹지정), -i (확인 후 종료), -l (시그널 목록), -v (상세 정보)

03. 파일 시스템과 프로세스

프로세스의 개념과 관련 명령어

✓ kill 명령의 사용 예

```
[LinuxMaster]#ping 8.8.8.8 > NULL &  
[1] 8898  
[LinuxMaster]#ps -al  
F S  UID  PID  PPID  C PRI  NI ADDR SZ  WCHAN  TTY      TIME CMD  
4 S   0  8898 15610 0  80   0 - 32139 poll_s pts/0    00:00:00 ping  
0 R   0  8907 15610 0  80   0 - 38332 -   pts/0    00:00:00 ps  
[LinuxMaster]#kill 8898  
[LinuxMaster]#ps -al  
F S  UID  PID  PPID  C PRI  NI ADDR SZ  WCHAN  TTY      TIME CMD  
0 R   0  8932 15610 0  80   0 - 38324 -   pts/0    00:00:00 ps  
[1]+  종료됨          ping 8.8.8.8 > NULL
```

03. 파일 시스템과 프로세스

프로세스의 개념과 관련 명령어

- ✓ nice - 우선순위 변경(설정)하여 프로세스 실행 → NI 값 이용
 - nice <옵션> [프로세스명]
 - 옵션: -n 조정값, -조정값, --adjustment 조정값 → 기본 조정값은 10
 - NI 값 범위 : -20 ~ 19
 - 일반 사용자는 NI값 증가만 가능
- ✓ renice - 실행 중인 프로세스의 우선순위 변경
 - renice <옵션> <NI값> [PID]
 - -p (PID 지정), -u (사용자 지정), -g (GID 지정)

03. 파일 시스템과 프로세스

프로세스의 개념과 관련 명령어

✓ top - 시스템 운영 상태 혹은 프로세스 상태 확인

- top <옵션> [PID]

➤ 옵션: -d (실시간 화면 출력 시간 지정, 기본 5초), -p (PID 지정)

```
top - 02:35:07 up 126 days, 18 min, 1 user, load average: 0.00, 0.01, 0.05
```

```
Tasks: 99 total, 1 running, 98 sleeping, 0 stopped, 0 zombie
```

```
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
```

```
MiB Mem: 990.7 total, 160.8 free, 221.8 used, 608.2 buff/cache
```

```
MiB Swap: 2048.0 total, 2004.0 free, 44.0 used. 559.4 avail Mem
```

```
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
```

```
9 root 20 0 0 0 0 S 0.3 0.0 105:16.18 rcu_sched
```

```
...
```

03. 파일 시스템과 프로세스

프로세스의 개념과 관련 명령어

- ✓ nohup - 프로세스 중단없이 백그라운드로 작업 실행 → 로그아웃 되어도 계속 실행
 - 표준 출력 내용을 nohup.out에 기록

```
[LinuxMaster]#nohup ping 8.8.8.8 &
```

```
[1] 11121
```

```
[LinuxMaster]#nohup: ignoring input and appending output to 'nohup.out'
```

```
^C
```

03. 파일 시스템과 프로세스

프로세스의 개념과 관련 명령어

✓ cron - crond 데몬에 의해 반복적인 작업 실행

- 설정 파일 : /etc/crontab
- 관련 명령 : crontab <옵션> <파일명>

➤ 옵션:

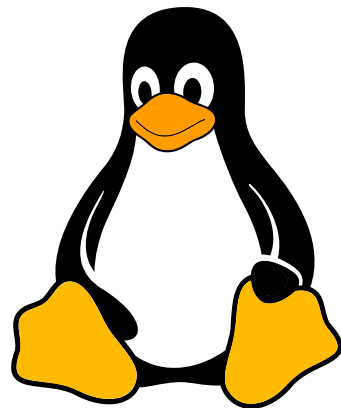
- | (설정 내용 출력),
- e (crontab 작성, 수정),
- r (crontab 내용 삭제),
- u (지정한 사용자의 스케줄 수정)

```
[LinuxMaster]#cat /etc/crontab
...
# 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
```

리눅스마스터 (2급)

- 소프트웨어와 장치의 활용 -

김윤수
(kchris000@gmail.com)



04. 소프트웨어와 장치의 활용

소프트웨어와 장치의 활용

- ✓ 에디터 활용
- ✓ 프로그램 설치
- ✓ 장치 설정과 활용
- ✓ X 윈도우

04. 소프트웨어와 장치의 활용

에디터 활용

- ✓ pico 에디터 - 워싱턴 대학의 Aboil Kasar,
유닉스 기반,
메뉴 선택 방식,
자유 소프트웨어 라이선스가 아님,
복제 버전인 nano

- 관련 명령 : pico <옵션> [파일명]

➤ 옵션:

-w (긴 라인이 깨지는 경우 사용)

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

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

04. 소프트웨어와 장치의 활용

에디터 활용

- ✓ emacs 에디터 - 리처드 스톨만, 매크로 기능, 제임스 고슬링이 LISP 언어로 기능 추가 개발, 소스 코드 작성 모드 (통합 환경)

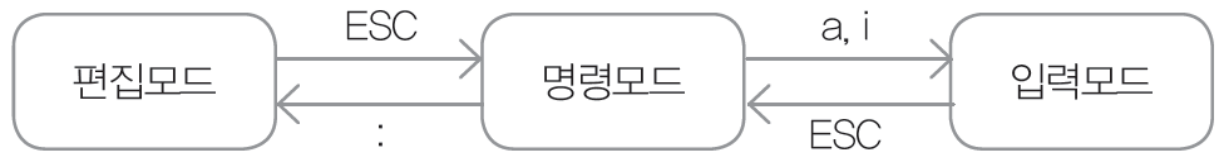
메뉴	기능
[Ctrl] + [X], [Ctrl] + [S]	파일 저장
[Ctrl] + [X], [Ctrl] + [C]	편집 종료
마크 설정 후 [Ctrl] + [W]	잘라내기 (마크 설정 방법 : 설정을 시작하고 싶은 부분으로 커서를 이동시킨 후 [Ctrl] + [Space Bar]로 마크 설정)
[Ctrl] + [K]	커서 뒤에 있는 한 줄이 모두 지워짐 (지워진 문자는 클립보드에 저장)
[Ctrl] + [A]	커서를 줄의 맨 앞으로 이동
[Ctrl] + [E]	커서를 줄의 맨 뒤로 이동
[Ctrl] + [N]	커서를 한 줄 아래로 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

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

04. 소프트웨어와 장치의 활용

에디터 활용

- ✓ vi 에디터 - 명령모드, 입력모드, 편집모드, 리눅스 배포판에 기본 포함



〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

vi <파일명> : 기본 명령어를 입력 받을 수 있는 명령어 상태
i : 입력 명령, 입력 상태로 전환
dd : 커서가 있는 라인 한줄을 삭제
esc: 명령어 상태로 전환
: 명령줄 상태로 전환 (여러개의 명령어 입력)
w : 명령줄 상태에서 변경사항 저장
q : 명령줄 상태에서 vi 편집기 종료
q! : 수정된 사항 반영없이 강제 종료 (편집한 내용 취소)

- ✓ vim 에디터 - vi 편집기와 호환, 색상 사용한 가시성, ex모드에서 히스토리 기능, 확장 정규 표현식, 다중 되돌리기, 문법 검사

04. 소프트웨어와 장치의 활용

에디터 활용

✓ nano 에디터 - 단축키 기반의 메뉴

```
GNU nano 2.3.1      File: hell.txt      Modified
Hello ^^~

^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is   ^V Next Page  ^U UnCut Text ^T To Spell
```

```
GNU nano 2.7.4      File: ./test.c

#include <wiringPi.h>
int main (void)
{
    wiringPiSetup () ;
    pinMode (0, OUTPUT) ;
    for (;;)
    {
        digitalWrite (0, HIGH) ; delay (500) ;
        digitalWrite (0, LOW) ; delay (500) ;
    }
    return 0 ;
}

^G Get Help  ^O Write Out  ^W Where Is   ^K Cut Text
^X Exit      ^R Read File  ^\ Replace    ^U Uncut Text
```

✓ gedit 에디터 - GNOME (GUI) 데스크톱 환경, GTK+와 GNOME 라이브러리를 이용하여 개발, UTF-8과 호환, CLI에서 사용 불가함

04. 소프트웨어와 장치의 활용

프로그램 설치

✓ 리눅스 소프트웨어의 설치 - 배포판의 패키지 설치, 소스코드 컴파일

- 패키지 관리 프로그램

데비안 계열	배포 업체	Debian, Ubuntu, Xandros, Linspire
	패키지 툴	dpkg, apt-get, aptitude
레드햇 계열	배포 업체	Fedora, CentOS, RHEL, openSUSE, Mandriva
	패키지 툴	rpm, yum

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

04. 소프트웨어와 장치의 활용

프로그램 설치

- ✓ RPM - Redhat Package Manager, 패키지 확장명 (.rpm)
 - 프로그램 패키지 파일 형식의 예 : sendmail-8.12.3-5.fc11.i586.rpm
 - 패키지명-버전(주버전.부버전.패치번호)-릴리즈번호.페도라버전.아키텍처
 - 커널 패키지 파일 형식의 예 : kernel-3.5.0-327.el7.x86_x64.rpm
 - 패키지명-버전(주버전.부버전.패치번호)-릴리즈번호.CentOS버전.아키텍처
 - 패키지 관리 명령 : rpm <옵션> [패키지명]
 - -i (설치), -h (#마크표시), -U (업그레이드), -e (제거), -q (설치여부확인), -v (진행 메시지 표시),
-vv (메시지 상세 표시), --nodeps (의존성 무시), --oldpackage (다운그레이드), --replacepkgs (재설치),
--replacefiles (덮어쓰더라도 강제 설치), --force (기존 패키지 설치여부와 관계없이 강제 설치)
 - 패키지 변조 검사 옵션 : -V (Verify, 검증), -a (검증 시 모든 패키지 검사)
 - 검사 결과 코드 : 5 (MD5 체크섬 변경), S (크기 변경), L (링크 변경), T (수정일 변경), D (장치파일 변경), U (소유자 변경)

04. 소프트웨어와 장치의 활용

프로그램 설치

- ✓ YUM - Yellowdog Updater Modified, 레포지토리에 접속하여 패키지 관리, RPM 의존성 문제 해결, 페도라22 이후 DNF로 전환
 - 패키지 관리 명령 : yum <옵션> [패키지명]
 - install (설치, -y 옵션으로 자동 설치), update (설치 갱신), localinstall (rpm 이용), remove (제거), groupremove (패키지 그룹 제거), info (패키지 요약 정보),
 - 기타 : yum list (전체 패키지 정보 출력), yum grouplist (패키지 그룹 정보 출력),
yum check update (업데이트 가능 목록), yum search 문자열 (문자열이 포함된 패키지),
yum check-update (업데이트 필요한 패키지), yum history (작업 이력)

04. 소프트웨어와 장치의 활용

프로그램 설치

- ✓ dpkg - 데비안의 low level 패키지 관리 도구, deb 패키지 관리, 의존성 문제가 발생할 수 있음
 - deb 패키지 형식 : 패키지이름_버전-릴리즈버전-리버전_아키텍처.deb
 - 패키지 관리 명령 : dpkg <옵션> [패키지명.deb]
 - -i (설치), -r (설치된 패키지 삭제), -P (패키지/설정 모두 삭제), -s (패키지 정보 표시), -L (설치 목록 확인)

04. 소프트웨어와 장치의 활용

프로그램 설치

✓ apt-get (Advanced Package tool Get)

- 데비안 계열 리눅스의 패키지 관리 도구
- /etc/apt/source.list 참고 (의존성 충돌 해결)

➤ 패키지 유형(소스코드, 바이너리), 레포지토리 주소(URL), 우분투 버전 정보, 카테고리 분류

```
# See http://help.ubuntu.com/community/UpgradeNotes for how to upgrade to  
deb http://archive.ubuntu.com/ubuntu/ focal main restricted  
# deb-src http://archive.ubuntu.com/ubuntu/ focal main restricted
```

- 패키지 관리 명령 : apt-get <옵션> [명령어] [패키지명]

➤ 명령어 : install (설치), dist-upgrade (의존성 검사 및 설치), update (/etc/apt/source.list 정보 갱신)
upgrade (업그레이드), remove (제거)

➤ 옵션 : -y (자동 설치), -u (업그레이드 패키지 목록 표시), -V (자세한 버전)

04. 소프트웨어와 장치의 활용

프로그램 설치

- ✓ 소스파일을 이용한 설치 - 환경설정 (configure) → 컴파일 (make) → 설치 (make install)
 - 파일 아카이브(archive) : 여러 개의 파일을 하나로 묶어 놓은 것 → 백업, 배포에 활용
 - tar 명령을 이용한 압축 : tar <옵션> [파일명]
 - 옵션 : c (새로 만들기), x (압축 해제), d (차이 비교), u (업데이트), f (아카이브 파일명 지정), v (진행 정보 표시)
z (gzip으로 처리), j (bzip2로 처리), J (xz 로 처리), P (절대경로 유지), --exclude (지정한 디렉터리 제외)
 - 파일 압축과 압축 해제
 - 관련 명령어 : gzip (.gz), gunzip, gzcac, bzip2 (.bz2), bunzip2, bzcat
compress (.Z, 가장 낮은 압축률), uncompress,
xz (.xz, 가장 높은 압축률), unxz
- ※ -d 옵션으로 파일 압축 해제, (일반적으로) -v 옵션으로 진행과정 표시

04. 소프트웨어와 장치의 활용

프로그램 설치

✓ 소스파일을 이용한 설치 (계속) - 환경설정 (configure) → 컴파일 (make) → 설치 (make install)

〈1단계〉 환경설정	./configure	<ul style="list-style-type: none"> · 프로그램 설치 과정에서 필요로 하는 환경파일 makefile 생성 <ul style="list-style-type: none"> - 시스템 파일 위치 지정 또는 설치 파일의 위치 지정 - configure 뒤에 옵션을 넣음
〈2단계〉 컴파일	make	<ul style="list-style-type: none"> · makefile을 기반으로 소스 파일을 컴파일 · make가 끝나면 소스파일은 실행파일(setup)로 전환
〈3단계〉 파일 설치	make install	<ul style="list-style-type: none"> · 컴파일 된 실행파일을 지정된 속성으로 지정된 디렉터리에 설치

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

※ CMake (Cross Platform Make) : 운영체제 별 Makefile 생성, 의존관계 분석, 다양한 개발 환경 지원, 타임스탬프(변화 확인)

04. 소프트웨어와 장치의 활용

장치 설정과 활용

✓ 프린터 (Printer) 인쇄 시스템 - LPRng, CUPS

- LPRng (Line Printer next generation) : 리눅스 초기, 라인 프린터 데몬 프로토콜 → 스푼링, 네트워크 프린트
 - 설정 파일 : /etc/printcap
- CUPS (Common Unix Printing System) : 애플, HTTP 기반의 IPP (Internet Printing Protocol, 631 포트), 인증
 - 데몬 이름 : cupsd
 - 설정 파일의 디렉터리 : /etc/cups
 - /etc/cups/cupsd.conf (데몬 환경설정), /etc/cups/printer.conf (프린터 큐 환경설정, lpamin 명령), /etc/cups/classes.conf (데몬의 클래스 설정파일)

04. 소프트웨어와 장치의 활용

장치 설정과 활용

✓ 프린트 설정 - X Windows의 프린트 설정 도구, 로컬 접속, 네트워크 프린터 설정

- 네트워크 프린터 설정 방법

AppSocket/HP jetDirect	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	<ul style="list-style-type: none">윈도우 시스템에 연결된 프린터 설정 시 사용삼바 기반의 SMB(Server Message Block) 프로토콜 사용
인터넷 프린터 프로토콜(https)	https 프로토콜 기반의 프린터 설정 시 사용
인터넷 프린터 프로토콜(IPP)	IPP 프로토콜 기반의 프린터 설정 시 사용

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

04. 소프트웨어와 장치의 활용

장치 설정과 활용

✓ 사운드 카드 시스템 - OSS, ALSA

- OSS (Open Sound System) : 표준 장치 시스템 콜 기반 (POSIX read, write, ioctl 등) → ALSA 로 대체
- ALSA (Advanced Linux Sound Architecture) : 사운드 장치 드라이버 제공을 위한 리눅스 커널 요소, GPL/LGPL
 - 환경 설정 파일 : /etc/asound.state
 - 사운드 카드 자동 구성, 다수의 장치 관리
 - OSS의 지원으로 미디어 합성, 다중 채널 하드웨어 믹싱, 전이중 통신 등의 기능 제공

04. 소프트웨어와 장치의 활용

장치 설정과 활용

✓ 스캐너 시스템 - SANE, XSANE

- SANE (Scanner Access Now Easy) : 스캐너, 비디오 캠 등 이미지 관련 H/W를 제어하는 API, GPL, 멀티 OS 지원
 - SCSI 스캐너 : /dev/sg0, /dev/scanner
 - USB 스캐너 : /dev/usb/scanner, /dev/usbscanner
- XSANE (X based interface for the SANE) : X-Windows 기반 스캔/이미지 수정 프로그램, SANE 이용, GPL, 멀티 OS 지원

04. 소프트웨어와 장치의 활용

장치 설정과 활용

- ✓ 프린터 명령어 - BSD (lpr, lpq, lprm, lpc), System V (lp, lpstat, cancel)
 - BSD 계열 프린트 명령어
 - lpr <옵션> [파일명] : -# (인쇄 매수 지정), -m (작업 완료 후 메일), -P (프린터 지정), -r (출력 후 지정파일 삭제)
 - lpq <옵션> : -a (모든 프린터 작업 정보 표시), -l (자세히 표시), -P (프린터 지정)
 - lprm <옵션> : - (모든 작업 취소), -U (사용자 작업), -P (프린터 지정), -h (서버 지정)
 - lpc : 프린터 상태 확인, 큐 활성화/비활성화, 큐의 작업 순서 변경 ○
 - System V 계열 프린트 명령어
 - lp <옵션> [파일명] : -n (인쇄매수 지정), -d (프린터 지정)
 - lpstat <옵션> : -p (인쇄 가능 여부), -t (상태 정보), -a (Accept된 상태 정보),
 - cancel [ReqID] : -a (모든 작업 취소)

04. 소프트웨어와 장치의 활용

장치 설정과 활용

- ✓ 사운드 카드 명령어 - alsactl, alsamixer, cdparanoia
 - alsactl <옵션> [명령]
 - 옵션 : -d (디버그 모드), -f (환경설정파일 지정)
 - 명령 : store (환경 설정 파일에 사운드 카드 정보 저장), restore (사운드 카드 정보 로딩), init (초기화)
 - alsamixer : ncurses 라이브러리 기반 오디오 프로그램
 - cdparanoia <옵션> : 오디오 CD 음악 추출
 - -w (wave 파일), -a (Applied AIFF-C 형식), -B (모든 트랙을 cdda2wav 형식으로 추출)

04. 소프트웨어와 장치의 활용

장치 설정과 활용

- ✓ 스캐너 명령어 - sane-fine-scanner, scanimage, scanadf, xcam
 - sane-fine-scanner <옵션> [장치명]
 - -q (스캐너 장치만 대상), -v (자세한 정보), -p (시리얼 포트의 스캐너만 대상)
 - scanimage <옵션>
 - -d (SANE 장치 파일명), --format (이미지 형식, png/tiff), -L (사용 가능한 스캐너 목록)
 - scanadf <옵션> : 자동 문서 공급장치에서 여러 개 스캔
 - -d (SANE 장치 파일명), -L (사용 가능한 스캐너 목록)
 - xcam : GUI, 평판 스캐너 혹은 카메라로 부터 이미지 스캔

04. 소프트웨어와 장치의 활용

X 윈도우

- ✓ X-Window의 개요 - GUI, 플랫폼 독립적, X11, X, X Windows System이라고도 불림, X.org (오픈 그룹)
 - X 프로토콜 기반 클라이언트/서버 시스템
 - 서버 : 클라이언트의 디스플레이 접근 제어, 클라이언트 간 자원 공유, 메시지 전달, 클라이언트와 입출력 장치의 중계
 - 클라이언트 : 어플리케이션을 이용하여 X 서버의 기능 이용
 - 오픈 데스크톱 환경 : KDE, GNOME, XFCE
 - 구성요소
 - X 프로토콜 (XLib, Xtoolkit 인터페이스 사용), Xlib (저수준 클라이언트 라이브러리), XCB (향상된 쓰레드, 확장성)
 - Xtoolkit (고급 GUI 개발, Widget과 Intrinsic 지원),
 - XFree86 (X86 계열의 X 서버, 무료), XF86Config (XFree86 설정파일)

04. 소프트웨어와 장치의 활용

X 윈도우

- ✓ X-Window 설정 - /etc/inittab
 - /etc/inittab : init 프로세스가 참조
 - 사용 환경 초기화 : 파일시스템 점검, 서비스 프로세스 관리, 가상 콘솔 관리, 런 레벨 관리

- ✓ X-Window 실행 - startx
 - 명령 형식 : startx -- [값]
 - 시스템 환경 초기화 및 xinit을 실행하는 스크립트

- ✓ DISPLAY 환경 변수 - X 윈도우의 Display 위치 지정
 - export DISPLAY=IP주소:Display번호,스크린번호

04. 소프트웨어와 장치의 활용

X 윈도우

- ✓ 윈도우 매니저 - 창의 배치와 표시를 담당하는 프로그램, XLib와 XCB 라이브러 사용, fwm, twm, mw, windowmaker, AfterStep
 - 대표적인 윈도우 매니저
 - Enlightenment : GNOME의 기본 윈도우 매니저
 - kwm : KDE v1.x의 기본 윈도우 매니저
- ✓ 데스크톱 환경 - 사용자에게 제공되는 인터페이스 스타일, 데스크톱 관리자, 패키지 형태의 프로그램, 위젯, drag&drop, IPC
 - 대표적인 데스크톱 환경
 - KDE : Kool Desktop Environment, QT툴 기반, UNIX/Linux, Solaris, FreeBSD 등, Kwin 윈도우
 - GNOME : GNU Network Object Model Environment, 공개 자유 소프트웨어, GTK+, BSD 등에 포함
윈도우 관리자 선택 사용, 세션 매니저 (환경 유지),
 - LXDE (Raspbian, GTK2, 가벼운 모델), XFCE (GTK+2, Xfwm 윈도우 매니저, 가벼운 모델)

04. 소프트웨어와 장치의 활용

X 윈도우

- ✓ 디스플레이 매니저 - X Windows system에서 동작하는 프로그램, XDM, GDM, KDM, dtlogin 등, X server 접속 및 세션 시작 담당
 - 대표적인 디스플레이 매니저
 - kdm : KDE의 디스플레이 매니저
 - gdm : GNOME의 디스플레이 매니저, GTK, GNU GPL

04. 소프트웨어와 장치의 활용

X 윈도우

✓ X 윈도우의 활용- xhost, xauth

- xhost - X 서버에 접속할 수 있는 클라이언트 지정 혹은 해제, 호스트 단위로 제어
 - 명령 형식 : xhost [+|-] [IP|도메인명]
 - 사용예 : xhost + (모든 클라이언트 접속 허용), xhost + IP주소 (해당 IP로 부터 접속 허용)
- xauth - MMC 방식의 인증을 사용하기 위한 유틸리티 (MIT-MAGIC-COOKIE), X 윈도우 실행 시 생성되는 키 값으로 인증
 - .Xauthority 파일에 매직 쿠키 추가, 삭제, 항목 출력
 - ~/.Xauthority에 RW 권한 필요
 - 명령 형식 : xauth <옵션> → list (모든 쿠키값 리스트),

04. 소프트웨어와 장치의 활용

X 윈도우

✓ X 윈도우 프로그램

응용 프로그램		설명
오피스	LibreOffice	오피스 프로그램 패키지
	gedit	텍스트 편집 프로그램
	kwrite	KDE 기반의 텍스트 편집기
그래픽	GIMP	이미지 편집 프로그램
	ImageMagick	이미지를 생성 및 편집을 지원하는 프로그램
	eog	GNOM의 이미지 뷰어 프로그램
	kolourpaint	Ubuntu 이미지 편집 프로그램
	gThumb	GNOME 데스크톱 이미지 뷰어 프로그램
	gwenview	KDE의 기본 이미지 뷰어

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

04. 소프트웨어와 장치의 활용

X 윈도우

✓ X 윈도우 프로그램

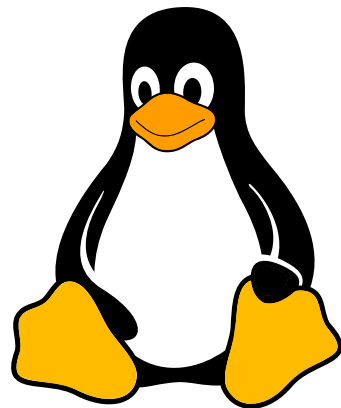
응용 프로그램		설명
오피스	LibreOffice	오피스 프로그램 패키지
멀티미디어	Totem	GNOME 기반의 사운드 및 비디오 재생 프로그램
	RHYTHMBOX	통합형 음악 관리 프로그램
	CHEESE	GNOME 기반의 카메라 동영상 프로그램
개발	ECLIPSE	통합 개발 환경으로 자바를 비롯한 다양한 언어를 지원
기타	Dolphin	KDE용 파일 관리자
	KSnapshot	스크린샷 프로그램

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

리눅스마스터 (2급)

- 인터넷과 리눅스의 활용 -

김윤수
(kchris000@gmail.com)



05. 인터넷과 리눅스 활용

인터넷과 리눅스 활용

- ✓ 네트워크와 인터넷 서비스
- ✓ 리눅스 기술 동향과 활용

05. 인터넷과 리눅스 활용

네트워크와 인터넷 서비스

✓ 네트워크의 개요 - LAN, MAN, WAN, SAN

- 네트워크(Network)의 개념 : 전송 매체(링크)로 연결되어 데이터를 상호 교환하는 시스템(노드)들의 그룹
 - LAN(Local Area Network) : 사무실, 학교 등 지리적으로 한정된 범위의 근거리 통신망 (약 10km 이내의 범위)
 - Ethernet (CSMA/CD), Token Ring(Token Passing), FDDI (이중 링, Token Passing)
 - MAN(Metropolitan Area Network) : 도시 지역을 잇는 통신, LAN을 연결한 백본라인(Backbone line)
 - 광섬유, 동축케이블, DBQB(Distributed Queue Dual Bus, Dual Bus, 회선교환/패킷 교환)
DBQB는 MAN에서 사용하는 QPSX(Queued Packet Synchronous Exchange) 접속제어에 사용
 - WAN(Wide Area Network) : 국가, 대륙 등 넓은 지역의 네트워크
 - 거리제한 없음, 다양한 경로 (상대적으로 느린 속도, 높은 에러율), 전용선, 회선교환망, 패킷 교환망
- ※ SAN(Storage Area Network) : 스토리지 구성을 위한 고속 전용 네트워크, Fiber Channel
 - 파일 I/O 요청을 블록 I/O로 전환 후 SAN 스토리지로 전달

05. 인터넷과 리눅스 활용

네트워크와 인터넷 서비스


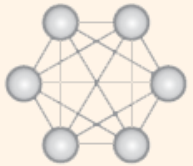
✓ 네트워크의 개요 교환 방식 - 회선 교환 네트워크

- 회선 교환 네트워크 (Circuit Switching Network) : 송수신 단말장치 사이에서 연결 경로를 미리 설정 후 데이터 전송
- 메시지 교환 네트워크 (Message Switching Network) : 메시지 헤더에 목적지 주소 표시하여 전송, 사전 경로 설정 없음
 - 교환기가 전송자의 메시지를 받은 후 수신자 확인 및 전달
- 패킷 교환 네트워크 (Packet Switching Network) : 메시지를 일정 크기의 패킷으로 분할하여 전송
 - 교환기는 패킷의 목적지 주소를 참고하여 전송 경로 선택, 가상회선과 데이터그램 방식

05. 인터넷과 리눅스 활용

네트워크와 인터넷 서비스

- ✓ LAN 토폴로지(Topology) - 호스트 및 장비들의 물리적인 배치 형태, 성형, 망형, 버스형, 링형, 트리형 등

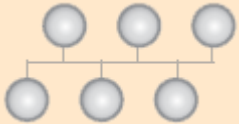
토폴로지	특징	연결 형태
성형 (star)	<ul style="list-style-type: none"> 중앙 컴퓨터에 여러 대의 컴퓨터가 허브 또는 스위치와 같은 장비로 연결 중앙 집중식 형태로 네트워크 확장이 용이 고속의 대규모 네트워크에 적합 관리하는 중앙 컴퓨터 고장 시 전체 네트워크 사용이 불가능 	
망형 (full mesh)	<ul style="list-style-type: none"> 모든 노드가 서로 일대일로 연결된 형태 대량의 데이터를 송수신할 경우 적합 장애 발생 시 다른 시스템에 영향이 적고 우회할 수 있는 경로가 존재하여 가장 신뢰성이 높은 방식 회선 구축 비용이 많이 듭 	

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

05. 인터넷과 리눅스 활용

네트워크와 인터넷 서비스

- ✓ LAN 토폴로지(Topology) - 호스트 및 장비들의 물리적인 배치 형태, 성형, 망형, 버스형, 링형, 트리형 등

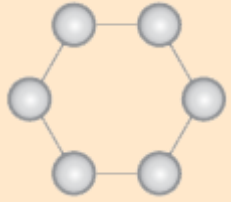
토폴로지	특징	연결 형태
버스형 (bus)	<ul style="list-style-type: none">· 하나의 통신회선에 여러 컴퓨터를 연결해서 전송· 연결된 컴퓨터 수에 따라 네트워크 성능이 변동· 단말기 추가 및 제거가 용이하며 설치 비용이 저렴· 노드 수 증가 시 트래픽 증가로 병목현상 발생, 네트워크 성능 저하 초래· 문제가 발생한 노드의 위치를 파악하기 어려움	

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

05. 인터넷과 리눅스 활용

네트워크와 인터넷 서비스

- ✓ LAN 토폴로지(Topology) - 호스트 및 장비들의 물리적인 배치 형태, 성형, 망형, 버스형, 링형, 트리형 등

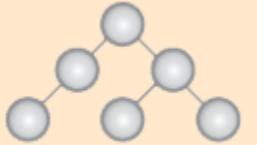
토폴로지	특징	연결 형태
링형(ring)	<ul style="list-style-type: none"> · 각 노드가 좌우의 인접한 노드와 연결되어 원형을 이룬 형태 · 앞의 컴퓨터로부터 수신한 내용을 다음 컴퓨터로 재전송하는 방법 · 토큰패싱(Token passing)이라는 방법을 통해 데이터 전송 · 고속 네트워크로 자주 네트워크 환경이 바뀌지 않는 경우 구성 · 분산 제어와 검사 및 회복이 가능 · 네트워크의 전송상 충돌이 없고 노드 숫자가 증가하더라도 망 성능의 저하가 적음 · 논리적인 순환형 토폴로지이므로 하나의 노드장애가 전체 토폴로지에 영향 · 노드의 추가 및 삭제가 용이하지 않음 	

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

05. 인터넷과 리눅스 활용

네트워크와 인터넷 서비스

- ✓ LAN 토폴로지(Topology) - 호스트 및 장비들의 물리적인 배치 형태, 성형, 망형, 버스형, 링형, 트리형 등

토폴로지	특징	연결 형태
트리형(tree)	<ul style="list-style-type: none">· 버스형과 성형 토폴로지의 확장 형태· 백본(backbone)과 같은 공통 배선에 적절한 분기 장치(허브, 스위치)를 사용하여 링크를 덧붙여 나갈 수 있는 구조· 트래픽 양 증가 시 병목 현상의 가능성 증대	

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

05. 인터넷과 리눅스 활용

네트워크와 인터넷 서비스

- ✓ 네트워크 장비 - LAN 카드, 케이블, 브릿지, 허브, 스위치, 리피터, 게이트웨이, 라우터
 - LAN 카드 : 컴퓨터내에 설치, 전기신호를 이용한 데이터 송/수신, MAC 주소
 - 케이블(Cable) : 두개 이상의 전선 혹은 광섬유, TP(Twisted Pair) 케이블, 동축케이블, 광섬유케이블 등
 - 100 BASE FX : 속도 채널 케이블타입
 - 브릿지(Bridge) : 수신 프레임을 버퍼에 저장, 목적지 포트로 전달, 필터링, Segment 구성
 - 허브(Hub) : 노드에 신호 전달, 네트워크 확장, 상호 연결, 신호 증폭
 - 스위치(Switch) : 맥 주소 테이블을 기반으로 프레임 전송, 스위칭 허브 (전용매체교환 기술, 병목 제거, 포트별 속도 보장)
 - 리피터(Repeater) : 신호 재생, 증폭, 물리거리 확장
 - 게이트웨이(Getway) : 서로 다른 네트워크 연결, 서로 다른 데이터 포맷 중계
 - 라우터(Router) : OSI 모델의 물리/데이터/네트워크 계층, 네트워크간 통신, 목적지까지 경로설정 및 전달

05. 인터넷과 리눅스 활용

네트워크와 인터넷 서비스

- ✓ OSI 7 모델과 TCP/IP 모델 - 응용, 표현, 세션, 전송, 네트워크, 데이터링크, 물리 vs. 응용, 전송, 인터넷, 네트워크

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(encoding), 압축(Compression), 암호화(Encryption)	
5	세션 계층	종단 간 애플리케이션들의 연결 설정, 유지, 해제	
4	전송 계층	종단 간 연결(End-to-End Connection), 응용 계층 사이에 논리적인 통로 제공(Virtual Circuit)	전송 계층
3	네트워크 계층	<ul style="list-style-type: none"> 논리적인 주소를 사용 경로 관리, 최적 경로 결정 	인터넷 층

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

05. 인터넷과 리눅스 활용

네트워크와 인터넷 서비스

- ✓ OSI 7 모델과 TCP/IP 모델 - 응용, 표현, 세션, 전송, 네트워크, 데이터링크, 물리 vs. 응용, 전송, 인터넷, 네트워크

OSI 7 모델		기능	TCP/IP 모델
2	데이터링크 계층	<ul style="list-style-type: none">· 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	

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

05. 인터넷과 리눅스 활용

네트워크와 인터넷 서비스

✓ 계층별 프로토콜 - 프로토콜의 구성요소는 형식, 의미, 순서

- 프로토콜(Protocol) : 컴퓨터 네트워크에서 데이터 교환 방식을 정의한 규약, 주고 받는 데이터의 형식 및 수행 절차.
- 프로토콜의 구성요소

형식(Syntax : 문법, 구문)	데이터 포맷(형식), 부호화 및 신호 레벨 등
의미(Semantic)	<ul style="list-style-type: none">• 특정 패턴을 어떻게 해석하고, 어떤 동작을 할 것인가 결정• 전송의 조정 및 오류 처리를 위한 제어 정보 등
순서(Timing)	속도 일치 및 순서 제어 등

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

- 응용 계층 프로토콜 : SMTP, POP, Telnet, SSH, FTP, HTTP, SNMP, TFTP, DHCP
- 전송 계층 프로토콜 : TCP(Transmission Control Protocol), UDP(User Datagram Protocol)
- 인터넷/네트워크 계층 프로토콜 : IP, ICMP, IGMP, ARP, RARP

05. 인터넷과 리눅스 활용

네트워크와 인터넷 서비스

✓ IP 주소와 도메인

- IP (IPv4) 주소 체계 : 4개의 옥텟(octet, 8비트 모임), 32비트 체계, E 클래스 (240~255, IP 부족위해 예약)
→ 네트워크 ID + 호스트 ID, 서브네팅으로 관리
- 서브네팅(subnetting) : 네트워크를 여러 개의 네트워크, 브로드캐스트 도메인으로 분할, IP 주소 부족 현상 해소 방안
→ 서브넷 마스크
- 특수 네트워크 주소
 - Network 주소 : 호스트 비트가 모두 0
 - Direct Broadcast 주소 : 호스트 비트가 모두 1
 - Limited Broadcast 주소 : 255.255.255.255, DHCP 클라이언트가 DHCP 서버를 찾을 때 사용
 - Lookback 주소 : 127.0.0.0 ~ 127.255.255.255, 내부 시험 테스트 용
 - 0.0.0.0 : 자신의 IP를 모를 경우 (부팅 시)

05. 인터넷과 리눅스 활용

네트워크와 인터넷 서비스

✓ IP 주소와 도메인

- IPv6 주소의 특징 : 128 비트, :로 구분된 16진수 8자리

구분	IPv4	IPv6
IP 주소 비트 수	32비트	128비트
IP 주소 방식	지정 주소 방식 (일반주소, 브로드캐스트주소)	자동 설정 주소 방식 (유니캐스트주소, 멀티캐스트주소, 애니캐스트주소)
IP 헤더 길이	20바이트(기본헤더) ~ 60바이트(옵션필드 사용 시)	40바이트(기본헤더)와 확장필드 이용
기능	IP 헤더를 이용한 기본 기능	확장 필드를 이용한 다양한 기능 - 암호 및 인증 기능 - QoS 관련 기능 - 보안 기능

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

05. 인터넷과 리눅스 활용

네트워크와 인터넷 서비스

✓ 인터넷 서비스의 종류

- WWW (World Wide Web : HTTP 를 기반으로 한 서비스, 분산 클라이언트-서버 모델)
 - HTTP : Request (GET, POST), Response (응답 코드), Header + Body(Data)
 - 표준 웹 프로토콜 : HTTP, XML, SOAP, USDL, UDDI), 상호 운영성
 - 다양한 웹 브라우저와 웹 서버
- FTP 서비스 : TCP/IP 기반, 파일 복사/전송
 - 통신 모드 : 액티브 모드, 패시브 모드
 - 계정 정책 : 사용자 계정, 익명 계정
- DNS 서비스 : 호스트(도메인)이름 ↔ IP 주소, 분산된 트리 구조, DNX 레코드
 - DNS 레코드 : A, AAA, MX, TXT, SOA, PTR, CNAME

05. 인터넷과 리눅스 활용

네트워크와 인터넷 서비스

✓ 인터넷 서비스의 종류

- Telnet과 SSH 서비스 : 원격 로그인, 실행 제어
 - Telnet : Byte 스트림 형식으로 전송
 - SSH : RSA 등 암호화 기법, 압축 기술
- NFS(Network File System) : 파일 공유 위한 클라이언트/서버, pormap 데몬을 이용한 RPC 연결 필요
 - NFS 관련 데몬 : nfsd, rpc.mouted, rpc.rockd, rpc.rquotad 등
- RPC(Remote Procedure Call) : 서비스와 포트 연결
 - 정적 포트 : /etc/services
 - 동적 포트 : rpcbind 사용 (※ SUN은 sunrpc)

05. 인터넷과 리눅스 활용

네트워크와 인터넷 서비스

✓ 네트워크 인터페이스 설정 : 자동, 수동(컴파일 된 모듈 적재), 설정 파일들

- 네트워크 설정 파일

- `/etc/sysconfig/network` : 기본 설정 정보,
- `/etc/sysconfig/network-scripts/ifcfg-ethX` : 지정된 네트워크 인터페이스의 환경 설정 정보
- `/etc/resolv.conf` : 도메인 명, 네임서버
- `/etc/hosts` : IP 주소와 도메인 정보 설정 (먼저 참고)

- IP 주소 설정 방법

- 설정 파일 : `/etc/sysconfig/network`, `/etc/sysconfig/network-scripts/ifcfg-ethX`
- 명령어 : `ifconfig`
- 유틸리티 : `netconfig`, `system-config-network`, `redhat-config-network` 등

05. 인터넷과 리눅스 활용

네트워크와 인터넷 서비스

✓ 네트워크 관련 명령어

TCP/IP 주소 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

〈출처 : 이기적 리눅스마스터 2급, 영진닷컴〉

05. 인터넷과 리눅스 활용

리눅스 기술동향과 활용

- ✓ 리눅스 동향X - POSIX 표준, 인터넷 표준 프로토콜, 초기에는 주로 서버 → 서버, 데스크탑, 개발, 임베디드 → 클라우드
- ✓ 리눅스 관련 기술 및 서버 분야
 - 클러스터링 : 여러 개의 시스템 연결, 하나의 큰 컴퓨팅 환경 → 고계산용, 부하 분산, 고 가용성 (HA, High Available)
 - 임베디드 시스템 : H/W와 S/W가 통합된 특정 목적을 수행하는 컴퓨팅 시스템, 실시간 처리, 높은 신뢰성, 소형/경량/저전력
 - 서버 가상화 : 하나의 물리적 호스트에서 여러 운영체제(게스트 OS) 실행, 가상화 기술을 기반으로 한 IT 인프라, 가용성
 - 클라우드 컴퓨팅 : 가상화 기술을 기반으로 IT 자원 및 서비스 제공, 인터넷(네트워크) 기반, On-Demand
 - IaaS (Infrastructure as a Service), PaaS (Platform as a Service), SaaS (Software as a Service)
 - Public Cloud, Private Cloud, Hybrid Cloud
 - ※ 오픈스택 (Open Stack), 클라우드 스택, 유칼립투스
 - 빅데이터 : 크기 (Volume), 속도 (Velocity), 다양성 (variety)

05. 인터넷과 리눅스 활용

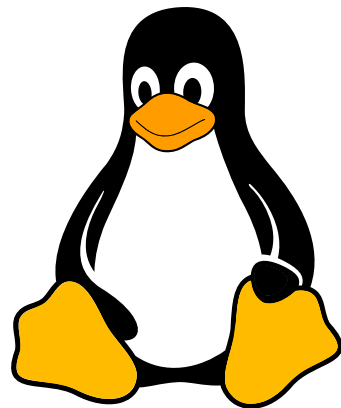
리눅스 기술동향과 활용

- ✓ 임베디드 시스템(Embedded System) : 안드로이드, iOS, Maemo, Moblin, MeeGo, Limo, Tizen 등
- ✓ 스마트 TV : 인터넷과 서비스가 통합된 TV, Tizen, WebOS
- ✓ IVI (In Vehicle Infortainment) : 정보(information) + 오락(Entertainment), GENIVI 표준 플랫폼 기반, 운전자 편의성/안전성
→ AUTOSAR (상호운영)

리눅스마스터 (2급)

- 기출문제 동향과 풀이 -

김윤수
(kchris000@gmail.com)



07. 기출 문제 동향과 풀이

기출문제 동향과 풀이

- ✓ 기출문제 동향
- ✓ 주제별 문제 풀이

07. 기출 문제 동향과 풀이

기출 문제 동향

✓ 리눅스마스터(2급) 출제기준

1차 시험 (50문항)	2차 시험 (48문항) - 리눅스 운영 및 관리	2차 시험 (32문항) - 리눅스 활용
<ul style="list-style-type: none">● 리눅스의 이해 (10문항)● 리눅스 설치 (10문항)● 기본 명령어 (30문항)	<ul style="list-style-type: none">● 파일 시스템 관련 명령어● Shell● 프로세스 관리● 에디터● 소프트웨어 설치, 장치 설정	<ul style="list-style-type: none">● X윈도● 인터넷 활용● 응용 분야

- 공통 : 60점 이상 합격
- 1차 시험 : 60분
- 2차 시험 : 100분, 과목당 40% 미만 과락

07. 기출 문제 동향과 풀이

기출 문제 동향

✓ 기출 문제는 어디에 있는가?

- 리눅스마스터2급 시행처 : <https://www.ihd.or.kr>

KAIT
정보통신기술자격검정

홈 | 로그인 | 회원가입

자격안내

시험접수

합격발표

자격증 발급

고객지원

마이페이지



합격발표

공정하고 신뢰하는 KAIT
정보통신기술자격검정은
ICT자격분야의 가치를
전해드립니다.

성적확인 >

문제공개 >

이의신청 >

문제공개

홈 > 합격발표 > 문제공개

문제공개



KAIT에서 시행하는 정보통신기술자격검정시험에 많은 관심과 성원을 보내주시는 수험자 여러분과 교육 기관관계자 여러분께 감사의 말씀을 드립니다.
급속한 정보통신관련기술 등의 변화로 인해 KAIT 자격검정홈페이지에 공개된 기출문제 중 다수가 현재의 기술 상황에 부합하지 않고 있습니다. KAIT에서는 시험시행 6개월 내의 기출문제만 공개하오니 이점 널리 양해 부탁드립니다.

알려드립니다.

본 시험문제는 한국정보통신진흥협회(KAIF)의 소유로 무단 이용을 일체 금합니다.
온 오프라인 교육기관 (학교, 학원), 출판사 등에서 이용을 원하는 경우 사전에 KAIT로 사용허가 신청바랍니다.



시험문제는 PDF 형태의 전자문서로서 제공됩니다.
좌측 버튼을 클릭하여 Acrobat reader를 다운로드 받아 설치하시기 바랍니다.

전체 ▾

번호	제목	작성일	파일	조회수
1	제2204회 리눅스마스터 2급 2차 가답안 공개	2022-12-13		2165
2	제2202회 리눅스마스터 1급 2차 모범답안(답안추가) 공개	2022-11-21		1473
3	제2203회 리눅스마스터 2급 2차 가답안 공개	2022-09-05		3569

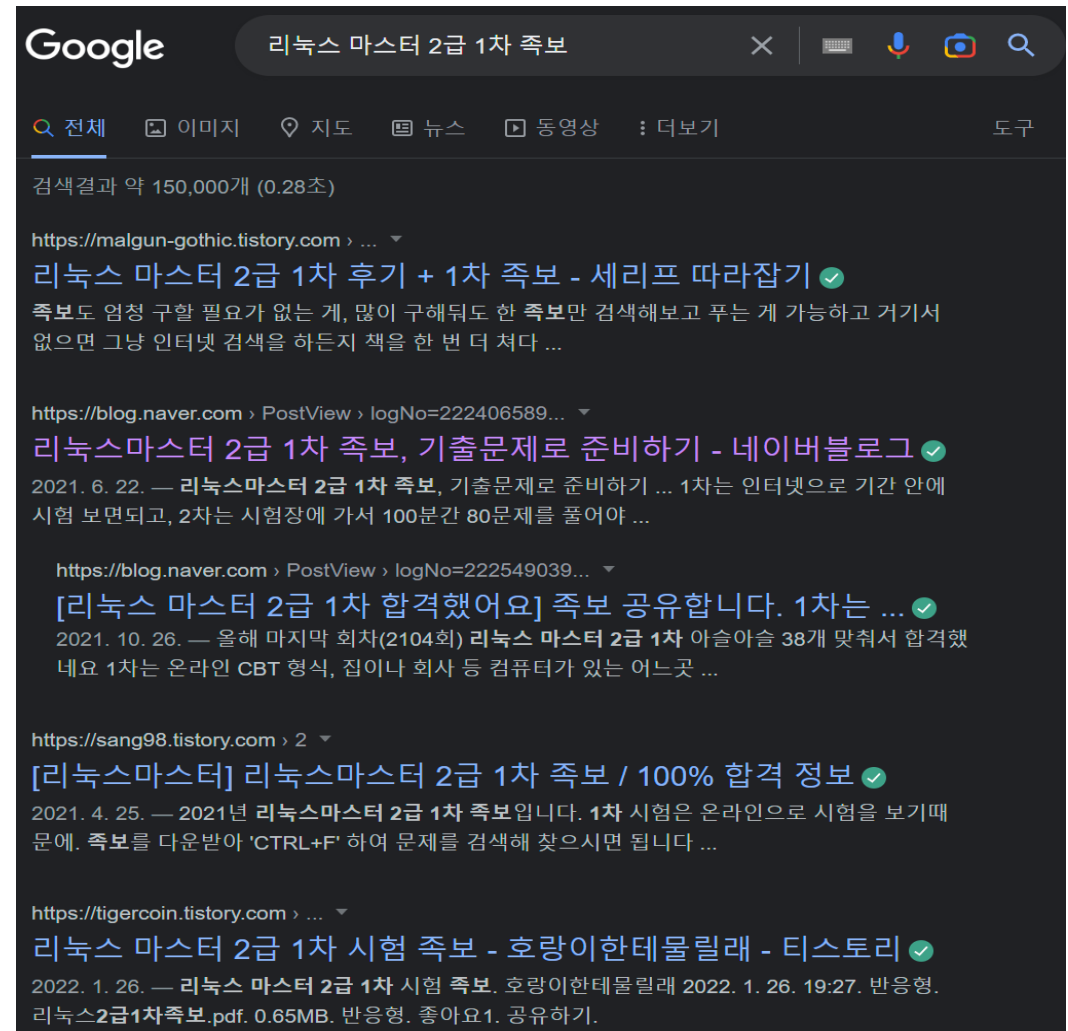
〈출처 : <https://www.ihd.or.kr>, 웹사이트 화면 캡처〉

07. 기출 문제 동향과 풀이

기출 문제 동향

✓ 기출 문제는 어디에 있는가?

- 1차 문제 : 구글 검색
 - 리눅스 마스터 2급 1차 족보
 - 리눅스 마스터 2급 1차 요약본



Google 리눅스 마스터 2급 1차 족보

전체 이미지 지도 뉴스 동영상 더보기 도구

검색결과 약 150,000개 (0.28초)

<https://malgun-gothic.tistory.com> > ...
리눅스 마스터 2급 1차 후기 + 1차 족보 - 세리프 따라잡기 ✓
 족보도 엄청 구할 필요가 없는 게, 많이 구해뒀던 한 족보만 검색해보고 푸는 게 가능하고 거기서 없으면 그냥 인터넷 검색을 하든지 책을 한 번 더 쳐다 ...

<https://blog.naver.com> > PostView > logNo=222406589...
리눅스마스터 2급 1차 족보, 기출문제로 준비하기 - 네이버블로그 ✓
 2021. 6. 22. — 리눅스마스터 2급 1차 족보, 기출문제로 준비하기 ... 1차는 인터넷으로 기간 안에 시험 보면되고, 2차는 시험장에 가서 100분간 80문제를 풀어야 ...

<https://blog.naver.com> > PostView > logNo=222549039...
[리눅스 마스터 2급 1차 합격했어요] 족보 공유합니다. 1차는 ... ✓
 2021. 10. 26. — 올해 마지막 회차(2104회) 리눅스 마스터 2급 1차 아슬아슬 38개 맞춰서 합격했네요 1차는 온라인 CBT 형식, 집이나 회사 등 컴퓨터가 있는 어느곳 ...

<https://sang98.tistory.com> > 2
[리눅스마스터] 리눅스마스터 2급 1차 족보 / 100% 합격 정보 ✓
 2021. 4. 25. — 2021년 리눅스마스터 2급 1차 족보입니다. 1차 시험은 온라인으로 시험을 보기때문에. 족보를 다운받아 'CTRL+F' 하여 문제를 검색해 찾으시면 됩니다 ...

<https://tigercoin.tistory.com> > ...
리눅스 마스터 2급 1차 시험 족보 - 호랑이한테물릴래 - 티스토리 ✓
 2022. 1. 26. — 리눅스 마스터 2급 1차 시험 족보. 호랑이한테물릴래 2022. 1. 26. 19:27. 반응형. 리눅스2급1차족보.pdf. 0.65MB. 반응형. 좋아요1. 공유하기.

07. 기출 문제 동향과 풀이

기출 문제 동향

✓ 기출 문제는 어디에 있는가?

• 2차 문제 : 구글 검색

➤ 리눅스 마스터 2급 CBT 기출문제

➤ 리눅스 마스터 2급 기출문제

1과목 : 리눅스 운영 및 관리

다음은 /etc/passwd 파일의 내용을 출력하는 과정이다. (괄호) 안에 들어갈 명령어로 알맞은 것은?

(괄호) /etc/passwd

① lp ② lpc
③ lpstat ④ lprm

다음 중 System V 계열에 속하는 프린트 관련 명령어로 틀린 것은?

① lp ② lpc
③ lpstat ④ cancel

다음 설명에 해당하는 LVM 관련 용어로 알맞은 것은?

사용자가 필요한 만큼 할당하여 만들어지는 공간으로 물리적 디스크에서 분할하여 생성하는 파티션과 같은 개념이다.

① 볼륨 그룹(VG) ② 논리적 볼륨(LV)
③ 물리적 볼륨(PV) ④ 물리적 확장(PE)

① rpm ② yum
③ dpkg ④ YaST

10. 다음 중 yum을 이용해서 nmap 패키지를 제거하는 명령어로 알맞은 것은?

① yum delete nmap ② yum clean nmap
③ yum remove nmap ④ yum destory nmap

11. 다음 중 아파치 웹 서버 소스 파일을 내려받은 후 압축을 해제하는 과정이다. (괄호) 안에 들어갈 내용으로 알맞은 것은?

tar (괄호) httpd-2.4.53.tar.bz2







① jxvf ② Jxvf
③ zxvf ④ Zxvf

12. 다음 중 소스 파일을 이용한 설치 단계로 가장 알맞은 것은?

① make clean → make → make install
② make → make clean → make install
③ configure → make → make install
④ configure → make clean → make install

Google

리눅스마스터 2급 CBT

전체

이미지

동영상

뉴스

지도

더보기

도구

검색결과 약 149,000개 (0.29초)

https://www.comcbt.com > ...

리눅스마스터 2급 필기 기출문제 전자문제집 CBT - 최강 자격증 ...

전자문제집 **CBT** 최신 기출문제, 복원중, 해설집, 자주 하는 질문 모음 · 리눅스마스터 2021년 시험부터 기출문제 비공개로 전환 및 기출문제 복원 관련.

이 페이지를 여러 번 방문했습니다. 최근 방문 날짜: 23. 1. 14

https://www.comcbt.com > ...

리눅스마스터 2급 필기 기출문제 전자문제집 CBT - 최강 자격증 ...

전자문제집 **CBT** 최신 기출문제, 복원중, 해설집, 자주 하는 질문 모음 · 리눅스마스터 2021년 시험부터 기출문제 비공개로 전환 및 기출문제 복원 관련 · 리눅스마스터 2급 2 ...

https://www.kinz.kr > subject

리눅스마스터 2급 객관식 필기 기출문제 CBT 목록 - 킨즈 ✓

리눅스마스터 2급 57 ... 리눅스마스터 2급(2022. 9. 3.) 리눅스마스터 2급(2022. 6. 11.) 리눅스마스터 2급(2021. 12. 11.) 리눅스마스터 2급(2021. 9. 11.) 리눅스마스터 2 ...

이 페이지를 2번 방문했습니다. 최근 방문 날짜: 23. 1. 14

https://q.fran.kr > 시험 > 리눅스마스터 1급

리눅스마스터 1급 기출문제 문제은행 CBT ✓

리눅스마스터 1급 기출 :: 문제은행, 기출문제 모의고사, 해설 포함. 전 회차 **CBT**, IBT, 국내 최대 문제 보유. ... 영문명: Linux Master Level-1

이 페이지를 4번 방문했습니다. 최근 방문 날짜: 22. 9. 29

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) bash 환경에서 아래 명령의 실행 결과에 대한 설명으로 가장 적절한 것은?

```
#export PATH=$PATH:/var
```

- ① 기존의 경로(PATH)의 내용을 제거하고 새로운 경로를 추가
- ② 기존의 경로(PATH)에 /var라는 새로운 경로를 추가
- ③ 잘못된 명령, export \$PATH=PATH:/var로 실행해야 한다.
- ④ 잘못된 명령, export %PATH=%PATH:/var로 실행해야 한다.

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) 다음 중 다수의 디스크나 파티션을 하나로 묶어서 구성하는 기술로 가장 적절한 것은 무엇인가?

- ① NTFS, LVM
- ② ext4, RAID
- ③ ext3, SAN
- ④ LVM, RAID

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) 현재 디렉토리의 위치가 /var/chris이다. 다음 중 그 결과가 다른 명령어는 무엇인가?

- ① cd ..
- ② cd /var
- ③ cd ./..
- ④ cd var

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) 리눅스 시스템의 파티션 분할에 대한 설명으로 가장 거리가 먼 것은 무엇인가?

- ① 최소한 /와 스왑(Swap) 등 2개의 파티션이 필요
- ② 서버로 사용하는 경우에는 4개 이상의 파티션 분할을 권장
- ③ /home, /var 등의 파티션을 분할하여 사용하도록 권장
- ④ 설치 시 분할한 파티션을 운영 시 필요에 따라 변경하여 사용

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) 패스워드 설정에 대한 설명으로 틀린 것은?

- ① passwd chris 와 같은 형식의 명령어 사용
- ② New password 프롬프트에서 암호를 입력하면 화면에 내용이 표시
- ③ 동일한 암호를 재 입력하여 확인
- ④ 사용자가 원할 경우 변경 가능

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) 시스템의 네트워크 정보를 확인하는 명령어와 가장 거리가 먼 것은 무엇인가?

- ① netstat
- ② cat /etc/crontab
- ③ route
- ④ ifconfig

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) 리눅스 시스템에서 sjcu.ac.kr까지 연결되는 경로를 확인할 때 사용할 수 있는 명령어는 무엇인가?

- ① traceroute sjcu.ac.kr
- ② ping sjcu.ac.kr
- ③ ssh sjcu.ac.kr
- ④ telnet sjcu.ac.kr

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) 리눅스 시스템에서 sjcu.ac.kr까지 연결되는 경로를 확인할 때 사용할 수 있는 명령어는 무엇인가?

- ① traceroute sjcu.ac.kr
- ② ping sjcu.ac.kr
- ③ ssh sjcu.ac.kr
- ④ telnet sjcu.ac.kr

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) ls 명령의 옵션에 대한 설명으로 잘못된 것은 무엇인가?

- ① -a : 해당 디렉터리의 모든 파일을 표시
- ② -l : 사용자의 물음에 답하는 형식으로 동작
- ③ -R : 하위 디렉토리의 포함하여 표시
- ④ -F : 파일의 특성에 따라 끝에 특수문자 추가

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) shutdown 명령어의 옵션에 대한 설명으로 가장 거리가 먼 것은 무엇인가?

- ① -c : 예약된 shutdown 명령 취소
- ② -h : shutdown 명령이 완료 후 시스템 정지
- ③ -f : shutdown 전에 실행 중인 프로세스에게 kill 시그널 전송
- ④ -r : shutdown 명령이 완료 후 시스템을 재부팅

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) chris 사용자가 현재 AI이라는 그룹에 속해 있다. 다음 중 chris 사용자의 그룹을 Computer로 변경할 때 알맞은 것은?

- ① `usermod -n Computer chris`
- ② `usermod -g Computer chris`
- ③ `usermod -n Computer AI`
- ④ `usermod -g Computer AI`

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) 파일에서 특정 패턴의 텍스트를 갖는 줄(line)을 찾아서 출력해주는 명령으로 가장 적절한 것은 무엇인가?

- ① file
- ② grep
- ③ head
- ④ find

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) 하드디스크 일부를 메모리처럼 사용하는 기술로 가장 적절한 것은 무엇인가?

- ① Makefile
- ② 스왑
- ③ 파이프
- ④ 리다이렉션

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) 리처드 스톨만(Richard Stallman)에 의해 설립되었으며 컴퓨터 프로그램의 복제와 배포, 개작의 자유를 확대하기 위한 목적을 가진 단체는 무엇인가?

- ① FSF
- ② ANSI
- ③ ISO
- ④ IEEE

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) 다음 명령의 실행 결과에 대한 설명으로 가장 적절한 것은 무엇인가?

```
[LinuxMaster]echo $PWD
```

```
/root
```

- ① 현재 작업 디렉토리의 위치를 확인
- ② PWD 환경 변수는 홈 디렉토리 정보 저장하고 있다
- ③ 시스템 관리자(root)만 실행할 수 있는 명령이다
- ④ echo %PWD와 동일한 결과를 갖는다

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) 다음 중 콘솔 혹은 텍스트 모드의 실행레벨(Run Level)로 가장 적절한 것은 무엇인가?

- ① 0
- ② 3
- ③ 5
- ④ 6

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) 다음은 hello 파일이 어떤 종류의 파일인지 확인해 보기 위한 명령이다. () 안에 포함될 가장 적절한 명령은 무엇인가?

[LinuxMaster]#() hello

- ① file
- ② type
- ③ what
- ④ check

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) 두 개의 파일명을 매개 변수로 사용하여 차이점을 확인하는 명령으로 가장 적절한 것은 무엇인가?

- ① find
- ② check
- ③ diff
- ④ pass

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) 다음 중 파일 혹은 디렉토리와 관련된 명령으로 가장 적절하지 않은 것은 무엇인가?

- ① cp
- ② mv
- ③ rm
- ④ ps

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) 다음은 네트워크를 관리하는 명령어이다. 가장 적절한 설명은 무엇인가?

```
[LinuxMaster]# ifconfig eth0 down
```

- ① eth0 네트워크를 재설정한다.
- ② eth0 네트워크를 비활성화 한다.
- ③ eth0 네트워크를 시스템에서 완전히 제거한다.
- ④ eth0 네트워크와 down 네트워크의 IP 주소를 표시한다.

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) /etc/passwd 파일의 내용 중 일부를 발췌한 내용이다. 이에 대한 설명으로 가장 거리가 먼 것은 무엇인가?

```
chris:x:500:500::/home/chris:/bin/bash
```

- ① chris 계정의 비밀번호는 암호화 되어 있으며 /etc/secure 파일에 저장된다.
- ② chris 계정의 UID와 GUI는 각각 500 이다.
- ③ chris 계정의 홈 디렉터리는 /hom/chris 이다.
- ④ chris가 사용하는 shell은 bash이다.

07. 기출 문제 동향과 풀이

주제별 문제 풀이

1차 문제

문) 리눅스에서 PATH(실행파일 탐색경로) 환경 변수를 설정하는 파일로 가장 적절한 것은 무엇인가?

- ① /etc/profile
- ② /etc/passwd
- ③ /etc/shadow
- ④ etc/hosts

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) 다음은 /etc/hosts 파일의 내용을 프린트로 출력하는 명령어이다. (괄호) 안에 들어갈 명령어로 가장 적절한 것은 무엇인가?

(괄호) /etc/hosts

- ① lp ② lpc
- ③ lpstat ④ lprm

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) RAID 구성 방법 중 한 개의 디스크라도 오류가 발생하면 데이터 복구가 불가능한 모드는 무엇인가?

- ① RAID-0 ② RAID-1
- ③ RAID-5 ④ RAID-6

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) yum을 이용해서 httpd 패키지를 제거하는 명령으로 가장 적절한 것은 무엇인가?

- ① yum delete httpd ② yum clean httpd
- ③ yum remove httpd ④ yum destory httpd

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) 소스 파일을 이용하여 프로그램을 설치 단계로 가장 적절한 것은 무엇인가?

- ① make clean → make → make install
- ② make → make clean → make distclean
- ③ configure → make → make install
- ④ configure → make install → make distclean

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) vi 편집기에서 사용할 수 있는 명령 중 변경된 내용을 저장하지 않고 종료하는 명령은 무엇인가?

- ① :w!
- ② :q!
- ③ :x!
- ④ :bye

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) emacs 편집기를 개발한 사람으로 가장 적절한 이름은 무엇인가?

- ① 리누스 토발즈 ② 리처드 스톨먼
- ③ 빌 게이츠 ④ 스티브 잡스

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) 백그라운드로 수행 중인 프로세스를 확인하는 명령어로 가장 적절한 것은 무엇인가?

- ① list ② find
- ③ jobs ④ fg

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) CentOS 7 버전에서 모든 프로세스의 시작이 되는 프로세스로 가장 적절한 것은 무엇인가?

- ① init ② inetd
- ③ xinetd ④ systemd

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) 사용자가 터미널을 닫아도 계속해서 백그라운드로 프로세스를 유지하고자 할 때 사용할 수 있는 명령어로 가장 적절한 것은 무엇인가?

- ① bg ② exec
- ③ jobs ④ nohup

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) 사용 중인 셸 변수를 제거하는 명령어로 가장 적절한 것은 무엇인가?

- ① env ② set
- ③ unset ④ export

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) chris 사용자의 디스크 사용량을 확인하는 명령으로 괄호() 안에 가장 적절한 명령어는 무엇인가?

```
[LinuxMasger]# ( 괄호 ) -sh ~/chris
```

- ① quota ② mount
- ③ df ④ du

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) 다음 명령 실행 후 sample.txt 파일의 퍼미션 값으로 가장 적절한 것은 무엇인가?

```
[LinuxMasger]# umask 022  
[LinuxMasger]# touch sample.txt
```

- ① ----r--r-- ② -rwxr-xr-x
- ③ -rw-r--r-- ④ -rw-rw-r--

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) 다음 빅데이터 관련 기술에 대한 설명으로 가장 적절한 용어는 무엇인가?

대용량의 데이터를 처리할 수 있도록 클러스터에서 동작하며 분산 응용 프로그램 지원한다. 오픈소스 소프트웨어(프리웨어) 들의 모음으로 구성되며 Java 언어를 이용하여 프로그램을 개발 할 수 있다.

- ① Docker ② R
- ③ NoSQL ④ Hadoop

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) e-mail 서비스와 관련된 프로토콜로 가장 거리가 먼 것은 무엇인가?

- ① SNMP ② SMTP
- ③ IMAP ④ POP3

주제별 문제 풀이

문) 다음 중 CentOS 7에서 네트워크 카드(장치)를 장착했을 때 ifconfig 명령으로 확인할 수 있는 장치명으로 가장 적절한 것은 무엇인가?

- ① lo
 - ② eth0
 - ③ enp0s3
 - ④ eth1

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) SSH 서버에 접속 시 사용하는 ssh 명령어의 옵션 중 포트 번호를 지정하는 옵션은 무엇인가?

- ① -l ② -n
- ③ -p ④ -x

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) C 클래스 네트워크 대역에서 서브넷 마스크값이 255.255.255.192일 경우 사용할 수 있는 서브넷의 개수는 몇 개인가?

- ① 1 ② 4
- ③ 8 ④ 16

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) IP에 대한 MAC 주소를 확인할 수 있는 명령어로 가장 적절한 것은 무엇인가?

- ① ip
- ② ifconfig
- ③ arp
- ④ nslookup

07. 기출 문제 동향과 풀이

주제별 문제 풀이

2차 문제

문) X 서버에 접근할 수 있는 클라이언트를 허가하는 명령어로 () 안에 포함될 가장 적절한 명령어는 무엇인가?

[LinuxMaster]# () 10.0.2.5

- ① xhost ② xauth
- ③ xset ④ startx