



1. 리눅스 개발환경 만들기 ...

박종원 교수
(리눅스프로그래밍)

1. 클라이언트 서버 환경

- 1) 개요
- 2) 서버 운영체제
- 3) 유닉스와 리눅스 비교
- 4) 데스크탑 형식/ 서버 형식
- 5) 실제 서버/ 가상서버

2. 개발환경 구축

- 1) 서버 사이드 설치
 - 운영체제 (리눅스 설치)
 - 네트워크 설정 및 터미널 연결
 - 웹서버 데몬, 데이터베이스 설치 (나중에)
 - 서버의 데이터베이스와 연결 (나중에)
- 2) 클라이언트 사이드 설치 (추후 때가 되면 함)
 - NPP를 이용한 서버측 실행 (실습화면으로)
 - 이클립스를 이용한 서버측 배포 (나중에)

처음 리눅스 수업때는 리눅스 실습이 가능한 수준의
서버측만 설치하고
추후 다시 개발환경 전체를 설치하도록 실습한다

● 1. 강의 들어가기

✓ 학습내용 소개

- 리눅스는 기업이나 공공기관, 학교 등 **정보시스템 서비스를 제공하는 서버의 운영체제**로 많이 사용되는 소프트웨어이다.
- 리눅스 운영체제를 개인용 PC의 윈도우 운영체제를 대신하는 운영체제로 사용할 수도 있다.
- 또한 스마트폰에서 많이 사용되는 안드로이드 운영체제에 기본 바탕은 리눅스로 구성되어 있다.
- 본 과목에서는 **서버형식의 리눅스 운영체제를 사용하는 방법 및 이를 활용하여 셸 프로그램을 다루는 방법**을 배우고 실습한다.
- 이를 위하여 본 단원에서는 운영체제, 서버 등 관련용어를 정리하고, 리눅스 운영체제를 설치하고 이 시스템에 접속하여 간단한 활용을 하는 부분은 기본이며, 이를 활용한 개발환경을 구축한다.

✓ 학습목표 제시

- 유닉스와 리눅스라는 OS를 정의할 수 있다.
- 일반적인 리눅스의 종류인 데스크 탑 형식과 본 과목에서 학습하는 서버형식에 대하여 설명할 수 있다.
- 실습학습 중 어렵거나 혼자서 수행하기 어려울 때 게시판 등 커뮤니티 활용 및 웹사이트 검색 등 도움되는 정보와 문제해결방법을 고안할 수 있다.
- 클라이언트/서버환경의 개발자 환경을 스스로 구축할 수 있다.

● 2. 생각해볼 문제 및 용어

✓ 학습전 생각해볼 문제

- “유닉스”와 “리눅스”라는 단어를 웹 검색(네이버, 구글 등).
- 일반적으로 “서버”라고 지칭하는 시스템의 의미를 조사.
- “우분투(ubuntu)”라는 리눅스 종류를 웹 검색을 통하여 알아보고, 우분투 웹사이트에 가서 정보를 검색.

✓ 용어 (강의 정리 시 필기 할 것)

- 클라이언트와 서버(Server)
- 운영체제(OS)
- GUI (Graphical User Interface) TUI(Text User Interface)
- 네트워크 기본 이론을 조사하고 모르는 용어 정리(IP, 라우팅, DNS, NAT/고정IP, 공인IP/사설IP, 포트포워딩)
- Ssh, putty, 포트
- 웹서버 (Apache), WAS (Tomcat),

3. 이해하기

1. 클라이언트 서버 환경

1) 개요

- 일상적 정보처리 시스템은 클라이언트/서버 환경으로 구성되어 있음(vs Standalone)
- 클라이언트는 사용자에게 보여지는 부분
- 서버는 클라이언트가 수행할 수 있는 서비스를 지원하는 부분

Client



YouTube



Server



통신
- 인터넷

※ 강좌목록 ※ 입학완료취소는 각 컴퓨터 교학처에 문의하시기 바랍니다.

과목코드	과목명	학과	직종	학년	주/야	반	교직원번호	교수명	입력완료여부	N
115372	취업준비교육	데이터융합SW과	응용소프트웨어	1	주간	A	20150693	홍필두	N	
115372	취업준비교육	데이터융합SW과	응용소프트웨어	1	주간	B	20150693	홍필두	N	
118497	융합시스템 구축	데이터융합SW과	응용소프트웨어	1	주간	A	20150693	홍필두	N	
118497	융합시스템 구축	데이터융합SW과	응용소프트웨어	1	주간	B	20150693	홍필두	N	

● 3. 이해하기

1. 클라이언트 서버 환경

2) 서버의 운영체제

- 클라이언트는 화면에 보이는 것, 기기의 다양한 기능 사용, 사용자 중심
- 서버는 데이터처리, 통신, 업무 중심, 대용량, 가용성, 확장성, 안정성
- 서버의 운영체제는? Windows보다는 linux/Unix가 유리하며 90%이상을 차지
- 앞으로 교안에 나오는 그림 뜯어 붙인 내용은 교재 내용임.

1) 운영체제(Operating System, OS)

① 운영체제 : 컴퓨터의 하드웨어와 소프트웨어를 제어하여, 사용자가 컴퓨터를 쓸 수 있게 만들어주는 프로그램. <그림

1-1>

② 운영체제의 기능.

- 하드웨어와 응용프로그램간의 인터페이스 역할
- CPU, 주기억장치, 입출력장치 등의 컴퓨터 자원 관리
- 인간과 컴퓨터간의 상호작용을 제공함과 동시에 컴퓨터의 동작을 구동(Booting)
- 작업의 순서를 정하며 입출력 연산을 제어 (CPU스케줄링, 타임셰어링) task를 ms단위로 나누어 실행
- 프로그램의 실행을 제어
- 데이터와 파일의 저장을 관리

여전에는 프로그래머만 컴퓨터는 사용했는데
운영체제를 통해 일반인들도 쉽게 사용하게 됨

운영체제 → API → 소프트웨어



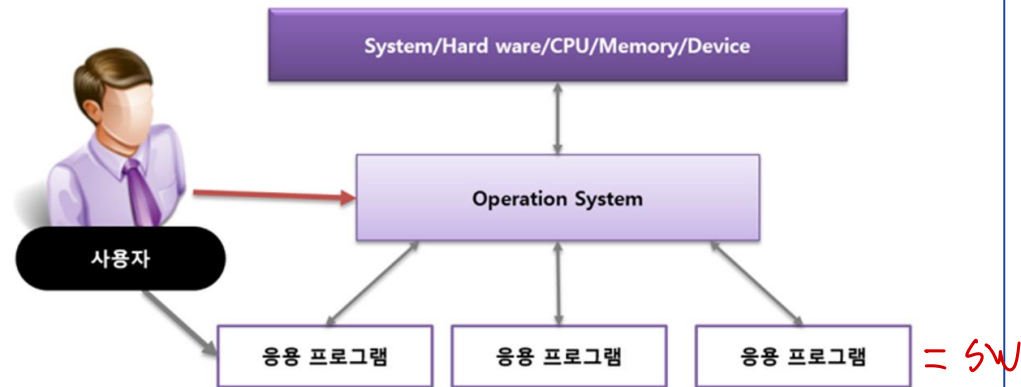
운영체제가 없으면 소프트웨어 개발자가
하드웨어를 다뤄야 함

● 3. 이해하기

1. 클라이언트 서버 환경

2) 서버의 운영체제

- 클라이언트는 화면에 보이는 것, 기기의 다양한 기능 사용, 사용자 중심
- 서버는 데이터처리, 통신, 업무 중심, 대용량, 가용성, 확장성, 안정성
- 서버의 운영체제는? Windows보다는 linux/Unix가 유리하며 90%이상을 차지



<그림 1-1> 운영체제의 개념도

보충학습

운영체제는 하드웨어와 소프트웨어의 중간에 위치

사용자가 컴퓨터 시스템을 사용하기 위하여 운영체제는 사용자의 명령을 하드웨어에 전달하고, 응용프로그램이 하드웨어 자원 등을 사용하는 경우, 하드웨어 제어 운영을 중간에서 조정하고 관리하는 역할을 수행하고 있다. 운영체제는 사용자와 응용프로그램이 복잡한 컴퓨터 하드웨어 부분을 이해하지 않아도 쉽게 사용되거나 운영될 수 있도록 해주는 시스템 프로그램이다

● 3. 이해하기

1. 클라이언트 서버 환경

3) 유닉스와 리눅스 비교

유닉스에서 리눅스로 간여하고있다!

- 유닉스 : 안정성, 대용량, 기존의 강자, 코어 업무(금융-계정계, 국가, 국방 ...)
- 리눅스 : 오픈소스, 최근에 서버의 OS로 신흥강자, SMB는 당연, 기존 업무도 많이 변환(U2L)

① 유닉스

- 중 · 대형서버시스템에서 가장 많이 사용되는 OS, 고성능, 고 가용성의 운영 체제
- 예: IBM(AIX) , HP(HP-UX), SUN(Solaris)

② 리눅스

원격개발자, git도 개발 (협업관계) → 흔

- 1991년 리누스 토발즈(Linus Torvalds)가 중 · 대형 기종에서만 작동하던

유닉스 운영체제를 PC에서도 작동할 수 있게 만든 운영 체제

③ 리눅스의 기능

- 프로그램 소스 코드를 무료로 공개하여 사용자는 원하는 대로 특정 기능을 추가함.
- 어느 플랫폼에도 포팅이 가능함.
- 현재는 개인용PC, 기업의 중대형 컴퓨터, 임베디드 기기, 모바일 (Android.iOS)기기에서도 리눅스가 사용됨.

유닉스와 리눅스, 맥OS
posix (포지식) 표준을 따른다.
(윈도우 OS 제외)

● 3. 이해하기

1. 클라이언트 서버 환경

gui : 데스크탑 버전

tui : 서버 버전

4) 데스크탑 형식과 서버형식

- 리눅스 : 서버의 OS 역할 뿐만 아니라, 현재 윈도우(MacOS)의 PC OS의 역할도 담당할뿐..
- 즉 text기반 OS에서 Windows같은 UI기반의 데스크탑 형식에 대한 노력
- 안드로이드의 탄생 -> 리눅스 데스크탑이 진화한 모바일 버전
- 많은 리눅스 메이커가 존재(명령어들은 대부분 비슷)
- 시장강자 Ubuntu (70~80%), 또다른 강자 CentOS (나머지)
- Ubuntu : 시장에 강자다 보니 참고할 문서, 구글링이 쉽다. 오픈소스계열 SMB의 강자
- CentOS : 돈주고 사는 기존 기업용 리눅스인 red Hat과 호환, 오라클 사용 가능, 큰 기업 고려
- Ubuntu는 데스크탑, 모바일, 서버, 클라우드등의 다양한 버전을 제공
- 데스크탑과 달리 서버형식은 딱히 화면을 보여주는 윈도우UI가 필요하지 않다.
- 경량화, 고성능화를 위하여 굳이 낭비할 필요는 없다(근데 관리자도 UI편한게 더 좋은데..)

● 3. 이해하기

1. 클라이언트 서버 환경

4) 데스크탑 형식과 서버형식

1) 데스크 탑(Desktop) 형식 - 잘 사용하지 않음

① 일반 사용자 PC의 Windows 운영체제와 유사한 형식임

② 윈도우의 상용성의 대응하여 GNU정신의 입각하여 태동함(무료 윈도우).

프리스프레이징 정신

③ 서버의 용도로 사용할 수 있지만, 실제 서버 환경을 운영하는데 윈도우 형태의 GUI가 불필요한 서버 리소스를 사용함.

④ 다양한 GUI를 선호하는 경우 Desktop형식을 설치하여 서버로 사용할 수 도 있음



<그림 1-3> 데스크 탑 형식의 리눅스 GUI

● 3. 이해하기

1. 클라이언트 서버 환경

최신자료는 오른쪽 영역에 되어 있다.

4) 데스크탑 형식과 서버형식

2) 서버(Server) 형식

- ① 불필요한 윈도우 GUI를 없애고, 서버용으로 적합하게 구성된 리눅스 형식임
- ② 본 교재에서는 리눅스 명령어 실습, 리눅스 셸 프로그래밍을 익히기에 적절한 형식인 서버 형식으로 실습을 진행함

```
login as: kopotc
kopotc@192.168.56.101's password:
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.13.0-74-generic i686)

* Documentation:  https://help.ubuntu.com/

System information as of Mon Jan 18 09:47:26 KST 2016

System load:  0.0              Processes:            90
Usage of /:   28.7% of 6.75GB   Users logged in:     0
Memory usage: 5%              IP address for eth0: 10.0.2.15
Swap usage:   0%              IP address for eth1: 192.168.56.101

Graph this data and manage this system at:
https://landscape.canonical.com/

Last login: Wed Jan 13 16:51:05 2016 from 192.168.56.1
kopotc@kopotc:~$
```

<그림 1-4> 서버 형식의 리눅스

● 3. 이해하기

1. 클라이언트 서버 환경

50x : 서버 도큐

40x : 클라이언트 도큐

5) 실제 서버, 가상 서버

- 실제서버는 대용량, 고가용성, 다중사용자, 너 혼자 개발하다 안되면 컷다 컷다 불가..
- 가상서버 (Virtual Machine/ Cloud기술)기술이 나오기 전에는 작은 PC/서버에 리눅스를 설치하여 개발용으로 사용
- 현재는 PC의 성능이나 용량이 커서 해당 자원을 나누어 사용 (개인 가상머신, 버추얼 클라우드), 하지만 이 가상서버의 용량이 보통 개발자 1인용도이며, 범용 서비스를 하기 위해서는 용량확장을 고민해야 함
- 또는 퍼블릭 클라우드 (AWS, GCP, Azure)에서 작은 용량의 개발서버를 인터넷공간에 두어 개발..-> 서비스 개시때 대용량으로 증설된 클라우드 서버 활용
- 우리 학교에도 고가의 서버가 서버실에 있지만, 하나의 서버를 여러 사람이 사용하기는 번거롭고, 굳이 그러한 큰 용량이 필요하지도 않으며(이마트 덤프트럭 몰고 가야 운전사만 불편), 좀더 실력을 갈고 닦으면 한번 만져보게 해줌.. ///실습하다 뭐 눌렀더니 확 다 날라갔어요.. 포맷되었나 봐요/// 이러면 대책없기에 각자 PC에서 가상서버로 실습함.

3. 이해하기

1. 클라이언트 서버 환경

5) 실제 서버, 가상 서버

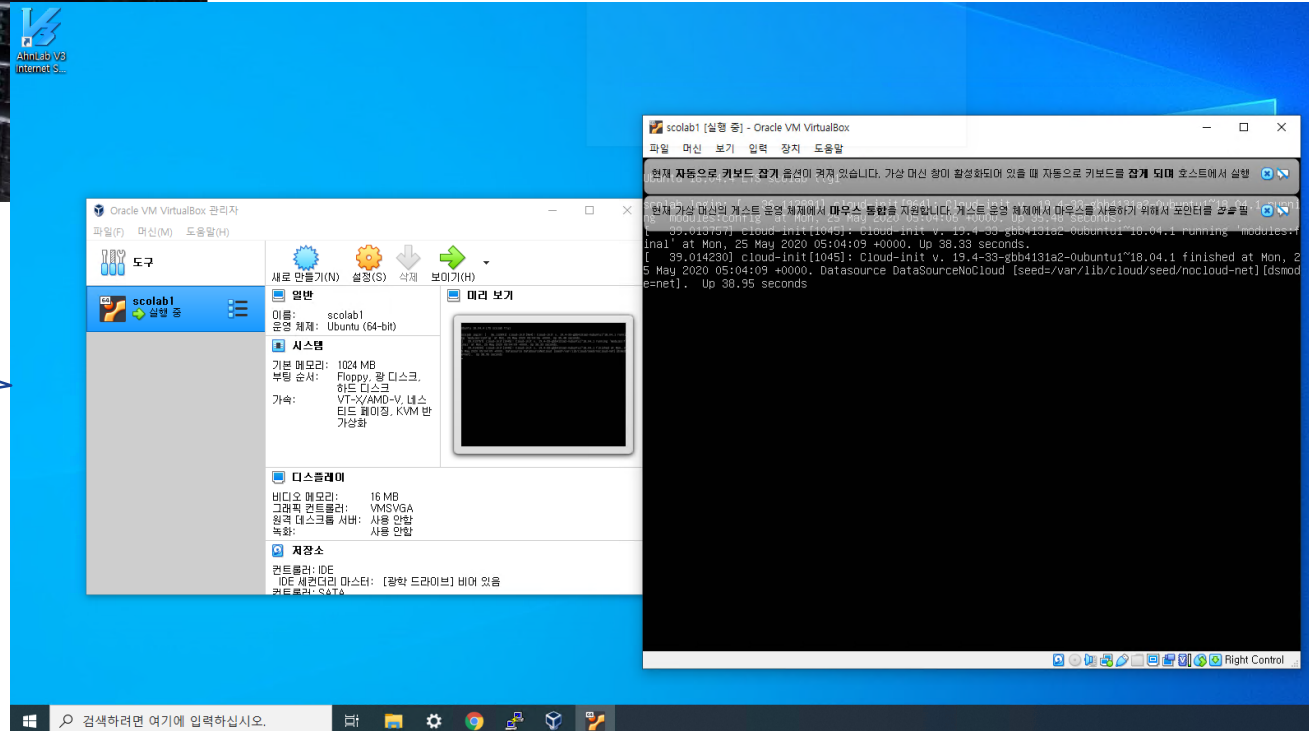


실제 서버는 서버실에서 곱게
모셔놓고 에어컨에 항온항습기에
UPS등 각종 대우를 받고

자 우리 KB은행, 네이버에 지금
얼마나 많은 사람이 정보서비스를
요청할까?

개발 테스트를 위하여 서버(작은
용량의)를 나의 개발PC에 일부를
할당하여 리눅스서버를 지원하는
Virtual Machine 프로그램으로 운영

실제 PC가 대용량
서버급..(CPU병렬연결
128,256,512개)에서는 이렇게
분할하여 대용량서버로 가능
여러분 실습PC는 워크스테이션
급으로 제온CPU가 8개 있음



● 3. 이해하기

2. 개발환경 구축

1) 서버 측 설치

