

1주차 3차시 리눅스/유닉스에 대한 이해

【학습목표】

1. 유닉스/리눅스의 구조 및 부팅 순서에 대해 설명할 수 있다.

학습내용1 : 유닉스/리눅스의 역사

1. 유닉스의 역사

유닉스의 시초

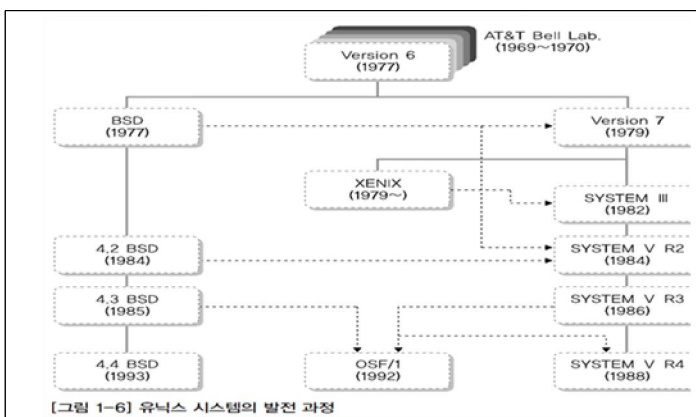
멀티스(Multics)라는 시분할 운영체제로 1963년부터 3년동안 MIT, GE(General Electric), 벨 연구소가 공동으로 개발
멀티스는 미 국방성(ARPA)의 지원으로 추진된 프로젝트로 GE가 하드웨어를 만들었고 PL/1(Programming Language 1) 언어로 제작

데니스 리치(Dennis Ritchie)는 파견지에서 돌아온 뒤 켄 톰슨(Ken Thomson)과 함께 PDP-7을 만들기 시작하며 멀티스의 여러 개념 구현에 노력

PDP-7에는 파일 시스템 구성의 개념, 사용자가 명령을 내려 바로 실행하는 명령어 인터프리터(Command Interpreter)의 개념, 각 명령이 새로운 프로세스를 형성해서 실행하도록 하는 개념이 모두 포함

AT&T 연구소로부터 유닉스 소스 코드를 400달러에 구입, 빌 조이(Bill Joy)와 척 핼리(Chuck Haley)는 BSD(Berkeley Software Distribution)라는 이름을 붙여 당시50달러 가격으로 판매

2. 유닉스 발전 과정



3. 리눅스의 역사



리눅스 정식 버전은 1991년 10월에 발표 ‘Copy Left’ 원칙에 의해 배포, 유닉스의 표준 인터페이스 POSIX, POSIX.1(핵심부, 운영체제 인터페이스), POSIX.2(셸과 툴, 사용자 명령 등), POSIX.3(표준 규격의 적합성 검증법), POSIX.4(실시간 POSIX 확장), POSIX.5(에이다 언어), POSIX.6(보안 기능) 등이 있음

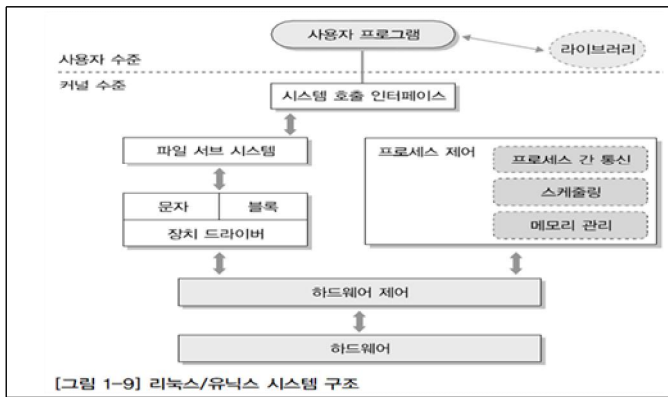
학습내용2 : 유닉스/리눅스의 구조

1. 리눅스/유닉스 링 구조

- * 하드웨어
- * 커널
- * 셸
- * 응용 프로그램



2. 리눅스/유닉스 구조



프로세스 제어

전체 프로세스 간 통신, 스케줄링, 메모리 관리를 구현

장치 드라이버

장치 드라이버하드웨어와 소프트웨어를 연결해주는 인터페이스를 제공

파일 서브 시스템

저장 공간에 유닉스의 파일을 저장하고 읽는 역할

3. 셸

- ① 응용 프로그램에서 명령을 받아 커널에 전송
- ② 사용자의 키보드 입력 인식
- ③ 해당 프로그램 수행

4. 셸의 기능

- ① 자체의 내장 명령어 제공
- ② 입력/출력/오류의 방향 변경(Redirection)
- ③ 와일드카드(Wildcard)
- ④ 파이프라인
- ⑤ 조건부/무조건부 명령 열 작성
- ⑥ 서브 셸 생성
- ⑦ 백그라운드 처리(Background Processing)
- ⑧ 셸 스크립트(프로그램) 작성

5. 리눅스/유닉스의 파일 시스템

일반파일

텍스트 파일, 이미지 파일, 실행 파일 등

디렉터리

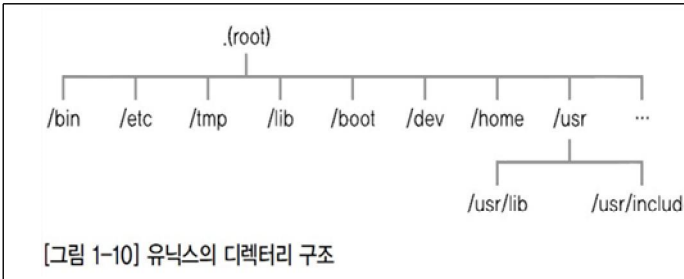
- 유닉스에서는 디렉터리도 파일에 해당
- 디렉터리가 담고 있는 여러 파일과 하위 디렉터리 정보가 담겨 있음

특수 파일

- 프린터, 터미널 같은 물리적인 장치에 접근하기 위한 파일
- /dev 에 존재

파이프 파일

- | 문자를 의미
- 2개의 명령을 연결 시 사용
- 임시 파일이 생성 되었다가 명령 수행을 마치면 사라짐



6. 유닉스 시스템 디렉터리별 역할

디렉터리	내용
/bin	<ul style="list-style-type: none"> · 기본적으로 실행 가능한 파일을 담고 있음 · echo, mv, copy, pwd, who 등
/etc	<ul style="list-style-type: none"> · 시스템의 환경 설정 및 주요 설정 파일을 담고 있음 · passwd, hosts, xined.conf 등
/tmp	<ul style="list-style-type: none"> · 프로그램 실행 및 설치 시 생성되는 임시 파일을 담고 있음 · 이 디렉터리에 파일을 저장하면 재부팅 시 이리로 삭제될 수 있음
/lib	<ul style="list-style-type: none"> · 기본 프로그램의 모듈을 담고 있음
/boot	<ul style="list-style-type: none"> · 커널을 위한 프로그램 파일을 담고 있으며, 부팅할 때 읽혀 수행됨

디렉터리	내용
/dev	<ul style="list-style-type: none"> · 프린터나 터미널 같은 물리적인 장치를 다루기 위한 특수 파일을 담고 있음
/home	<ul style="list-style-type: none"> · 각 사용자의 작업 디렉터리를 담고 있음 · 각 계정으로 로그인할 때 이 디렉터리 밑에 자신의 작업 디렉터리가 시작 디렉터리가 됨
/usr	<ul style="list-style-type: none"> · 사용자가 직접 쓰는 파일을 담고 있음 · 다른 디렉터리에 있는 파일이 똑같이 위치하고 있는 것이 많은데, 이는 링크되어 있는 것임
/usr/lib	<ul style="list-style-type: none"> · C언어나 포트란의 라이브러리를 담고 있음
/usr/include	<ul style="list-style-type: none"> · C언어에 사용되는 헤더 파일을 담고 있음

학습내용3 : 유닉스/리눅스의 부팅순서

① 1단계. POST(Power On Self Test) 수행

② 2단계. 기본 부팅 관련 설정사항 로드

③ 3단계. MBR(Master Boot Record) 로드

④ 4단계. 부트 로더(Boot Loader) 실행

부트 로더를 통해 부팅하고자 하는 운영체제의 커널 이미지를 메모리에 로드

LILO

- Linux Loader

- /etc/lilo.conf

GRUB

- Grand Unified Bootloader

- /etc/grub.conf(=/boot/grub/grub.conf)

⑤ 5단계. 실행 레벨에 따른 서비스 실행

부트 로더는 스와퍼(Swapper)라는 pid 0번 프로세스 실행

스와퍼는 pid 1번 init(/sbin/init) 프로세스 실행

init 프로세스는 /etc/inittab 파일 읽어서 부팅할 기본 모드 선택

시스템 초기화 스크립트 실행

/etc/rc.d/rc.sysinit

실행 레벨에 따른 데몬 실행

/etc/rc.d 아래에 존재

1. inittab 파일

*부팅할 기본 모드를 선택하여 그에 따른 환경을 제공하는 분기점

Default runlevel. The runlevels used are:

0 - halt (Do NOT set initdefault to this)

1 - Single user mode

2 - Multiuser, without NFS (The same as 3, if you do not have networking)

3 - Full multiuser mode

4 - unused

5 - X11

6 - reboot (Do NOT set initdefault to this)

2. inittab 실행 레벨

실행 레벨	내용
실행 레벨 0	시스템을 종료할 때 사용
실행 레벨 1	단일 사용자 모드(Single User Mode) 기본적으로 관리자 권한의 셸을 지원 대부분 데몬이 실행되지 않으므로 기능은 제한되어 있음
실행 레벨 2	NFS(Network File System)를 지원하지 않는 다중 사용자 모드
실행 레벨 3	일반 셸 기반의 인터페이스를 가진 다중 사용자 모드

실행 레벨	내용
실행 레벨 4	사용자가 임의로 정의하여 사용 기본적으로 사용되지 않음
실행 레벨 5	기본 실행 레벨 기본은 실행 레벨 3과 같으나 GUI 환경을 지원
실행 레벨 6	재부팅

【학습정리】

1. 리눅스/유닉스 시스템은 '하드웨어 > 커널 > 셸 > 응용 프로그램'의 링 구조를 이룬다.
2. 리눅스/유닉스의 파일 시스템의 종류는 일반 파일, 디렉터리, 특수 파일, 파이프 파일이 있다.