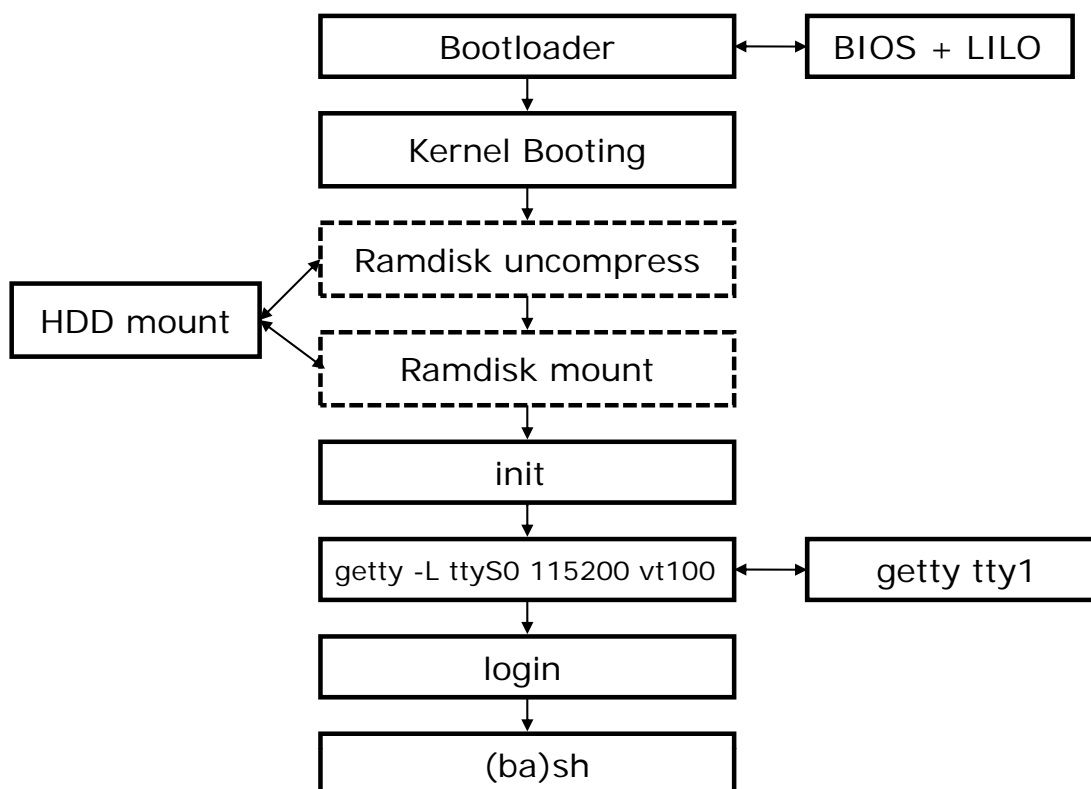


Chapter 2. 유닉스 시스템 시작

- 2.1 유닉스 시스템 사용과 종료
- 2.2 셸(shell)

BOOT 절차





Linux Boot process

- ❖ **BIOS**가 시스템 이상여부 테스트
 - 부트로더에서 수행
- ❖ 부팅할 드라이브 선택
 - 커널의 플래쉬 위치 선택
 - 선택된 드라이브의 **MBR**읽어 드림
 - **MBR**의 파티션 테이블을 읽어 부팅할 파티션을 선택
- ❖ 커널 로드
- ❖ 커널을 램상으로 복사
- ❖ 커널 압축 해제/ 재배치
- ❖ 장착된 하드웨어 검사, 장치 드라이버 설정
- ❖ 파일 시스템 검사
- ❖ 파일 시스템 마운트
- ❖ **Init** 실행

3



첫번째 과정

- ❖ 모든 **PC** 시스템들은 롬(정확히는 **BIOS**)내의 코드를 실행시키는 것으로 부팅을 시작된다. 이 코드는 부트 드라이브의 섹터 **0**, 실린더 **0** 부분을 읽는다.

- ❖ **BIOS** 는 읽어들이는 섹터의 내용을 실행한다.

4



두 번째 과정

- ❖ 대부분의 부트 가능한 디스크들은 섹터 0, 실린더 0 영역에 다음 내용 중 한 가지를 담고 있다.
 - LILO 등과 같은 부트로더(**boot loader**)의 코드.
 - 리눅스 커널 등과 같은 운영체제 커널의 시작 부분의 코드.

5



세 번째 과정

- 부트로더는 커널을 찾아 메모리에 로드한 후 실행시키는 방식으로 부트를 시작한다
- 커널은 하드웨어 장치 초기화, 첫번째 리눅스 프로세스의 실행 환경 구성
- 기본적인 디바이스들과 그 내부 데이터 구조를 초기화 시킨다.

6



네 번째 과정

- 루트 파일시스템을 `ReadOnly`로 마운트 한다
- 루트 파일시스템이란 단순히 "/" 에 마운트되는 파일시스템을 말한다. 커널은 어디에서 루트 파일시스템을 찾아야 하는지를 알아야만 한다
- 만일 커널이 그 위치에서 로드 가능한 이미지를 찾지 못한다면 시스템은 멈춰버리게 된다.

7



다섯 번째 과정

- ❖ 일반적인 하드디스크(`/dev/hda1`)를 루트파일 시스템으로 잡아서 부팅을 한다
 - 루트 파일 시스템을 램디스크로하여 로드하기도 한다
 - 일반적으로 임베디드 시스템은 램디스크를 루트 파일 시스템으로 한다

8



여섯 번째 과정

- ❖ 커널에서 **init**을 실행하는 과정
 - `mount_block_root (init/do_mounts.c)`
 - `mount_root() (init/do_mounts.c)`
 - `prepare_namespace() (init/do_mounts.c)`
 - `Init() (init/main.c)`
 - `/sbin/init`

9



일곱 번째 과정(**init**) (1)

- ❖ 일단 시스템이 루트 파일시스템을 로드하는데 성공하면, 다음으로 루트 파일시스템에 있는 `init` 프로그램을 찾아 실행을 시도한다.
 - `/bin` 이나 `/sbin`에 존재
 - `init` 는 그 설정파일인 `/etc/inittab` 에 있는대로 수행해 나간다.

10

일곱 번째 과정 (inittab) (2)

```
id:3:initdefault:
# System initialization.
si::sysinit:/etc/rc.d/rc.sysinit
l0:0:wait:/etc/rc.d/rc 0
l1:1:wait:/etc/rc.d/rc 1
l2:2:wait:/etc/rc.d/rc 2
l3:3:wait:/etc/rc.d/rc 3
l4:4:wait:/etc/rc.d/rc 4
l5:5:wait:/etc/rc.d/rc 5
l6:6:wait:/etc/rc.d/rc 6
ca::ctrlaltdel:/sbin/shutdown -t3 -r now
pf::powerfail:/sbin/shutdown -f -h +2 "Power Failure; System Shutting Down"
pr:12345:powerokwait:/sbin/shutdown -c "Power Restored; Shutdown Cancelled"
1:2345:respawn:/sbin/mingetty tty1
2:2345:respawn:/sbin/mingetty tty2
3:2345:respawn:/sbin/mingetty tty3
4:2345:respawn:/sbin/mingetty tty4
5:2345:respawn:/sbin/mingetty tty5
6:2345:respawn:/sbin/mingetty tty6
```

11

Boot Script - 2

- ❖ /etc/init.d
 - All services

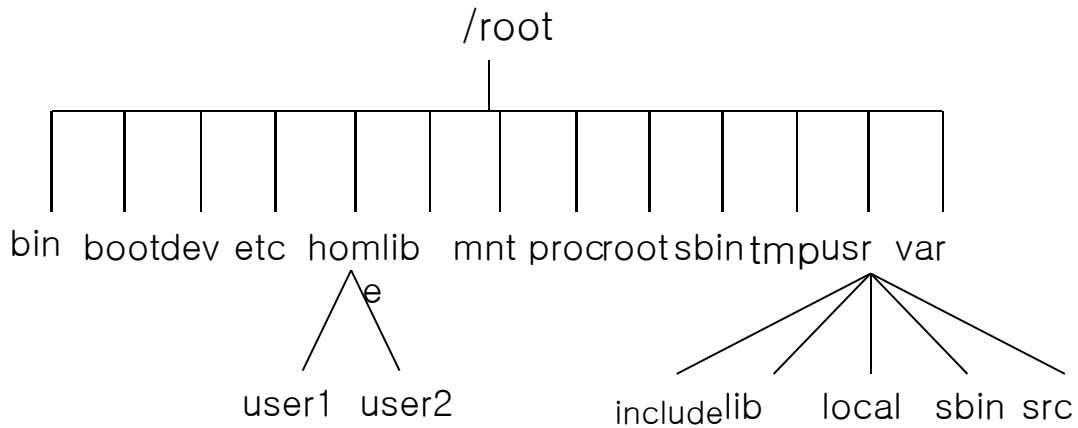
```
[root@bedguy rc3.d]# ls -l
lrwxrwxrwx 1 root root 16 4월 4 2000 S05syslog -> ../init.d/syslog
lrwxrwxrwx 1 root root 14 4월 4 2000 S50inet -> ../init.d/inet
[root@bedguy rc3.d]#
```

- ❖ /etc/rc.sysinit
 - PATH setting
 - HOSTNAME setting
 - /proc filesystem mount
 - Time setting
 - module load
- ❖ /etc/rc
 - runlevel script exec
 - runlevel 3 -> /etc/rc.d/rc3.d/

12

리눅스 디렉토리

❖ 리눅스 디렉토리 구조



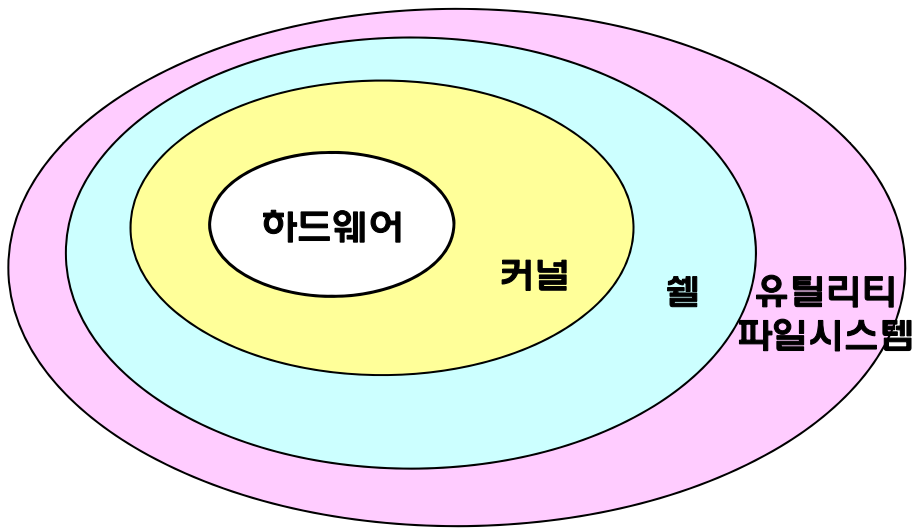
13

/ 디렉토리

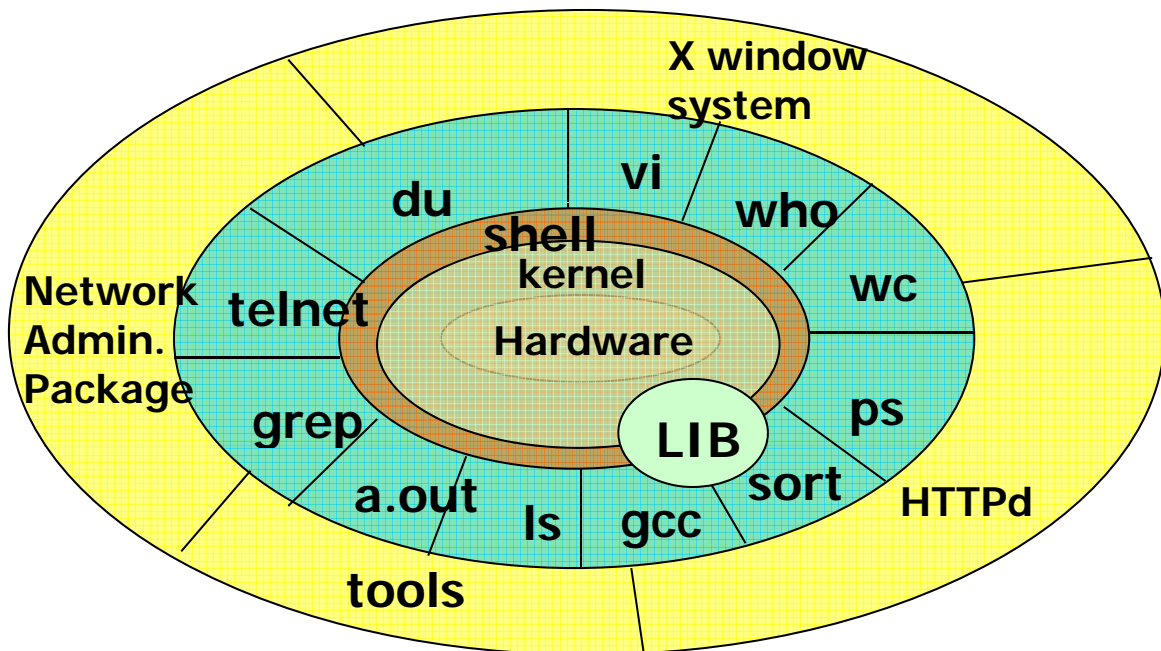
- 리눅스의 최상위 디렉토리(**Root Dir**) 이다.
- 파티션 설정시 반드시 존재하여야 한다.
- 모든 디렉토리는 최상위 디렉토리를 기준으로 위치된다.

14

- 유닉스는 크게 커널, 셸, 유틸리티와 파일시스템으로 구분



UNIX OS





로그인 전 준비사항

- ❖ 유닉스 시스템을 이용하기 위해서는 사용자 ID(계정)를 가지고 있어야 함
- ❖ 자신의 로그인 명(계정)이 없으면 시스템 관리자에게 문의하여 발급 받아야 함
- ❖ 다른 사용자로부터의 자신의 정보들을 보호하기 위해 비밀번호(password)를 설정해야 함
 - 일반적으로 최초 접속시 사용자가 직접 설정함

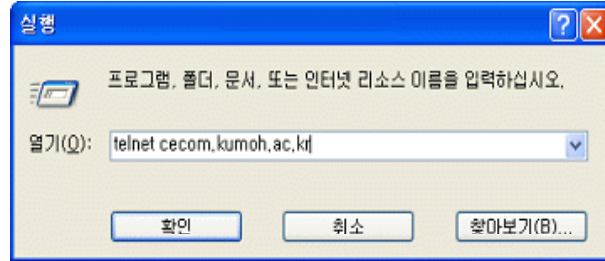


로그인 방법

- ❖ 텔넷(telnet)
 - 사용자가 유닉스 시스템에 연결하고자 하는 경우 텔넷(telnet)을 이용하여 연결
 - 텔넷은 네트워크 상에 존재하는 유닉스 운영체제를 가진 컴퓨터들에 접근하여 마치 자신의 컴퓨터처럼 사용할 수 있는 서비스
 - 텔넷을 이용하여 유닉스 시스템과 연결하기 위해 자신의 로그인명과 암호를 넣게 되면, 유닉스 시스템에 설정되어 있는 사용자의 홈 디렉터리에 연결

로그인 방법

- ❖ 1. 윈도우 환경에서 텔넷을 이용하기 위해 [시작-실행(R)] 을 선택하여 'telnet 호스트명 or IP'를 입력한 후 접속



- ❖ 2. login 프롬프트에 본인의 계정을 입력하고 password란 에 본인이 설정한 비밀번호를 입력함
 - 최초 접속시 자신의 비밀번호를 설정하게 됨
 - 유닉스에서는 대소문자를 구분하기 때문에 계정과 비밀번호 입력 시 주의해야 함

로그인 방법

- ❖ 로그인명과 비밀번호를 잘못 입력하여 수정해야 할 경우 '백 스페이스' 키가 아닌 'delete' 키를 이용
 - 잘못 입력했을 시 'login incorrect'라는 메시지가 나타나며 로그인 처음 과정부터 다시 시작해야 함

SunOS 5.8

그림 2-2. 유닉스 시스템 최초 접속 과정

```
login: redfox ← 발급 받은 로그인명(계정)
Choose a new password.
New Password: ← 최초의 비밀번호 설정
Re-enter new Password: ← 비밀번호 확인
telnet: password successfully changed for redfox ← 비밀번호 설정 완료메시지
Sun Microsystems Inc. SunOS 5.8 Generic Patch October 2001
%
```

SunOS 5.8

그림 2-3. 로그인명과 비밀번호를 잘못 입력한 예

```
login: redfox^H^H^H
Password:
Login incorrect ← 에러메시지
login: shcho ← 로그인 과정 처음부터 다시시작
Password:
```

비밀번호 변경 (passwd)(1/2)

- ❖ 시스템 관리자에 의해 사용자 계정이 부여된 후 사용자가 처음으로 시스템에 로그인하면 새로운 암호를 물어 본다.
- ❖ 이를 입력하면 다시 한번 입력하도록 요구하고 사용자가 처음 입력했던 암호와 정확하게 일치하게 입력하면 시스템과의 연결을 완료하고 프롬프트를 보여준다.
- ❖ 추후, 비밀번호를 변경하기 위해서는 'passwd' 명령을 이용
 - 암호를 선택할 때 다른 사람이 추측하여 알아낼 수 없는 것으로 하는 것이 좋음.

```
% passwd ← 사용자 비밀번호 변경
passwd: Changing password for redfox
Enter existing login password: ← 기존 비밀번호 입력
New Password: ← 새로 설정할 비밀번호 입력
Re-enter new Password: ← 새로 설정할 비밀번호 다시 입력
passwd: password successfully changed for redfox
%
```

그림 2-4. 비밀번호 변경 방법

셸 프롬프트

- ❖ 기본적으로 로그인할 때마다 어떤 셸이 자동적으로 시작하도록 정해져 있음
- ❖ 각 사용자와 관련된 로그인 정보는 암호파일 (/etc/passwd)에 저장되어 있음
 - 수록된 엔트리 안에 해당 사용자가 로그인할 때 구동되어야 할 셸 이름(/bin/csh)이 들어 있음
- ❖ 셸 프롬프트 모양
 - C 셸 : '%'
 - Bourne, Korn 셸 : '\$'
 - 관리자가 어떻게 설정했느냐에 따라 다를 수 있음
 - 또한 사용자 본인이 프롬프트 모양을 설정할 수 있음

셸 프롬프트

```
SunOS 5.8
```

```
login: redfox
```

```
Password:
```

```
Last login: Thu Sep 15 11:42:00 from cecom
```

```
Sun Microsystems Inc. SunOS 5.8 Generic Patch October 2001
```

```
%
```

그림 2-5. 기본적인 프롬프트 모양

```
cecom%
```

그림 2-6. 관리자의 프롬프트 설정내용에 따른 모양

```
% ksh ← Korn 셸 구동
```

```
$ ← Korn 셸 프롬프트 모양
```

```
$ exit ← Korn 셸을 종료하고 로그인 셸로 되돌아 감
```

```
%
```

그림 2-7. Korn 셸 구동시 셸 프롬프트 모양

셸 프롬프트 설정

- ❖ C 셸의 경우, ‘%’ 기호만 나타나 있고 현재의 위치를 알 수 없음. 다음과 같이 명령 줄에서 프롬프트를 설정하면 항상 자신의 위치를 파악 가능
- ❖ 매번 로그인할 때마다 본인이 설정해 놓은 프롬프트 모양을 적용하고 싶으면 ‘.cshrc’(C 셸), ‘.profile’(Bourne 셸) 파일에 넣어두면 됨(5장, 부록 B 참고)

```
cecom% set prompt="[$cwd]% "  
[/home/redfox]%
```

그림 2-8. 프롬프트 상에서 프롬프트 모양 설정

사용자 확인(identity)

- ❖ 유닉스 시스템은 시스템 관리자가 각 사용자에게 부여한 사용자번호(userid)와 그룹번호(groupid)에 의해 사용자를 확인
- ❖ 일반적으로 시스템이 사용자명과 userid, 그룹명과 groupid를 자동으로 바꾸어주기 때문에 사용자는 이를 알 필요는 없음
- ❖ 사용자는 *id*와 *groups* 명령을 이용하여 자신의 userid와 groupid를 볼 수 있음
- ❖ 기본 그룹은 패스워드 파일에 있는 사용자명과 연관된 그룹명이 되며, 이것 역시 시스템 관리자가 부여

```
cecom% id
uid=225(redfox) gid=100(student)
cecom% groups
student sys ← student, sys 그룹에 속해있음을 보여줌
```

그림 2-9. 사용자의 userid, groupid 확인

터미널 제어(stty)

- ❖ 제어 키
 - 명령어 줄, 혹은 편집기에서 특수한 기능을 수행할 때 사용
 - 사용자는 Control 키와 함께 다른 키를 치면 된다. (^Key)
 - 예를 들어 Control-S를 치면 ^S 모양으로 나타난다.
 - 제어 키에서 보이는 대소문자는 같은 의미를 가짐
 - 예를 들어, ^S와 ^s는 같다. 이 예에서 ^S는 stop 시그널이고, 이것은 터미널에 입력을 멈추도록 하는 시그널을 보낸다.
- ❖ Control-U는 보통 사용자 터미널에 "line-kill" 시그널을 보낸다. 이것은 사용자가 타이핑할 때 전체 입력 줄을 지운다.
- ❖ vi 편집기(유닉스 시스템의 대표적인 편집기)를 이용하여 텍스트 파일을 작성할 때 제어키를 사용하고자 하는 경우
 - Control-V(^V)를 먼저 치고 그 뒤에 Control 키를 누르면서 원하는 문자를 타이핑
 - 예를 들어, ^H(Control-Backspace)를 입력하고 싶다면 ^V^H와 같이 타이핑

터미널 제어(stty)

- ❖ 터미널의 제어 속성을 설정하거나 조사하는데 'stty' 이용
- ❖ 사용중인 터미널에서 삭제(erase)기능이 어떤 것으로 설정되었는지 확인 후 사용자가 이용하기 편한 키로 변경

```

cecom% stty -a ← 현재 터미널에 설정된 값 확인
speed 38400 baud;
행 = 26; 열 = 80; ypixels = 0; xpixels = 0;
csdata ?
eucw 1:0:0:0, scrw 1:0:0:0
intr = ^c; quit = ^W; erase = ^?; kill = ^u;
eof = ^d; eol = <undef>; eol2 = <undef>; swtch = <undef>;
start = ^q; stop = ^s; susp = ^z; dsusp = ^y;
rprnt = ^r; flush = ^o; werase = ^w; lnext = ^v;
<이하 생략...>
    
```

그림 2-10. 터미널에 설정된 메타문자 값 확인

터미널 제어에 사용되는 옵션 및 메타문자

옵션	의미
(none)	터미널 설정상태를 보여줌
a	모든 옵션을 보여줌
erase	한 문자를 지움
kill	현재 작업 중인 줄을 모두 지움
lnext	다음 문자를 특수문자로 처리하지 않음
susp	나중에 깨우도록 하고 프로세스를 임시 정지
intr	코어 덤프 없이 전면 작업 종료
stop	터미널 출력을 정지/다시 시작함
eof	입력의 끝
-echo	타이핑한 문자들을 다시 보여주지 않음
echo	타이핑한 문자들을 다시 보여줌
-raw	메타문자의 특별한 의미를 사용할 수 있도록 함
raw	메타문자의 특별한 의미를 사용할 수 없도록 함
-tostop	후면 작업이 터미널에 출력을 보낼 수 있도록 함
tostop	후면 작업이 터미널에 출력을 보내는 것을 중단함

터미널 제어(stty)

- ❖ 기존에 설정된 erase 값인 '^?'(delete키)에서 ^H(백스페이스 키)로 변경하기 위해서는 다음과 같이 설정

```
cecom% stty erase ^H ← erase 값을 ^H 값으로 설정
cecom% stty -a ← 변경된 erase 값을 확인
speed 38400 baud;
행 = 26; 열 = 80; ypixels = 0; xpixels = 0;
csdata ?
eucw 1:0:0:0, scrw 1:0:0:0
intr = ^c; quit = ^W; erase = ^h; kill = ^u;
eof = ^d; eol = <undef>; eol2 = <undef>; swtch = <undef>;
start = ^q; stop = ^s; susp = ^z; dsusp = ^y;
rprnt = ^r; flush = ^o; werase = ^w; lnext = ^v;
<이하 생략...>
```

그림 2-11. 터미널에 설정된 기존의 메타문자 값 변경

last 명령

- ❖ 마지막으로 로그인한 사용자명과 터미널을 보기 위해 사용. 즉, 로그인한 정보를 살펴볼 때 사용
- ❖ 해당 사용자가 로그인 했을 때의 정보. 즉, 터미널(pts/2), IP주소, 로그인시간, 로그아웃 시간, 접속한 시간 등을 확인할 수 있음
 - 또한, 사용자가 이전에 각각 몇분 몇초 동안 접속하여 작업을 했는지 확인할 수 있음

```
cecom% last redfox
redfox pts/7 IP 주소 Mon Oct 3 14:19 아직 로그인 되어있습니다
redfox pts/2 218.54.204.12 Sat Oct 1 01:52 - 03:52 (02:00)
<중략 ...>
redfox pts/2 IP 주소 Thu Sep 15 10:42 - 10:42 (00:00)
redfox pts/3 IP 주소 Thu Sep 15 10:30 - 10:42 (00:11)

wtmpt가 Mon Feb 21 17:41을(를) 시작합니다
```

그림 2-13. redfox 계정에 대한 last 명령 수행 결과

유닉스 시스템 연결 종료 : logout하기

❖ 원격 시스템과의 연결을 끊는 방법들

- ^D
 - 데이터 스트림(stream)의 끝을 의미. 또한 사용자와의 연결을 끊음(log off). 이 명령으로 로그아웃 되는 것을 막을 필요가 있음
 - C 셸 경우, '.cshrc' 파일 내 'set ignoreeof' 로 설정하여 logout 되는 것을 막을 수 있음
- ^C
 - 프로세스 종료(인터럽트), 실제 ^C로 logout 되지 않음을 확인할 수 있다.(이유는?)
- *logout*
 - 시스템 종료
- *exit*
 - 셸(프로세스) 종료
- 일반적으로 로그인 셸을 종료시키는 방법은 *logout*, *exit*를 이용함

유닉스 시스템 연결 종료 : logout하기

SunOS 5.8

login: *redfox*

Password:

Sun Microsystems Inc. SunOS 5.8 Generic Patch October 2001

cecom% ^D

cecom% 로그아웃

호스트에 대한 연결을 잃었습니다.

그림 2-14. C 셸의 .cshrc 파일 내 ignoreeof가 설정되지 않은 경우

```
cecom% cat > .cshrc
```

```
set ignoreeof
```

```
^d
```

```
cecom% source .cshrc
```

```
cecom% ^D
```

```
로그아웃하려면 "logout"을 사용하십시오.
```

```
cecom%
```

그림 2-15. C 셸의 .cshrc 파일 내 ignoreeof가 설정된 경우

유닉스 시스템 연결 종료 : logout하기

```
cecom% ^C
cecom% ^C
cecom%
```

그림 2-16. Control-C 사용 예

```
cecom% logout
```

호스트에 대한 연결을 잃었습니다.

그림 2-17. logout 사용 예

```
cecom% exit
cecom% 로그아웃
```

호스트에 대한 연결을 잃었습니다.

그림 2-18. exit 사용 예

셸(shell)

- ❖ 셸은 유닉스 운영체제와 사용자 사이에서 중간적인 교량 역할을 하는 프로그램
 - 즉, 명령어 해석기(command interpreter)로서 동작하는 프로그램
 - DOS에서의 command.com과 유사
- ❖ 또한, 사용자가 프로그램 및 명령어를 수행하고, 출력을 파일에 저장, 하나 이상의 프로그램이 수행되도록 함
- ❖ 유닉스에서 많이 사용되는 세가지 셸(4장 참고)
 - Bourne 셸
 - Korn 셸
 - C 셸