# 시스템 보안과 해킹

2009. 10

박 종 혁
(jhpark1@snut.ac.kr)
www.parkjonghyuk.net

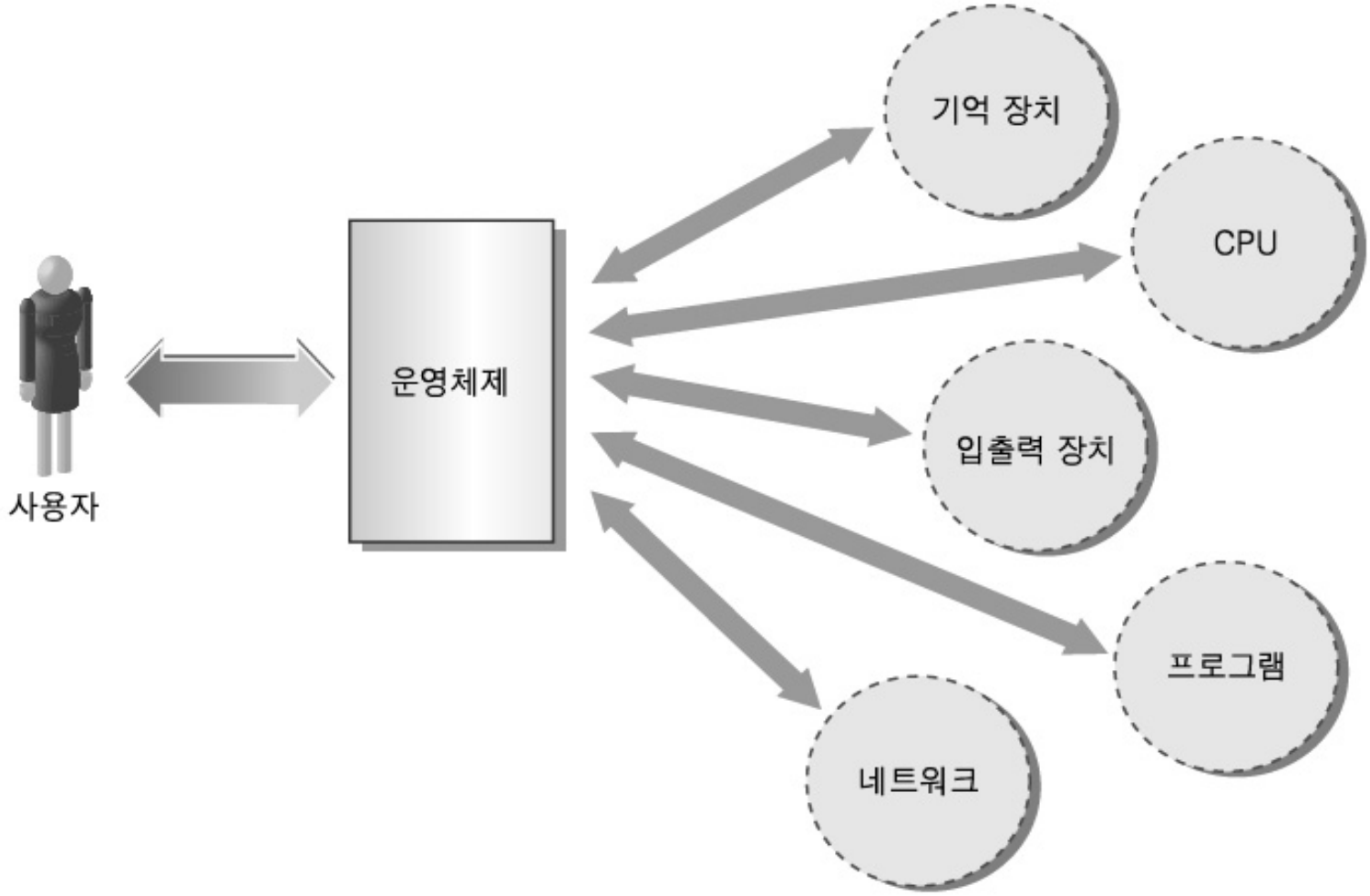
# 시스템 보안과 해킹

- 운영체제
- 시스템 해킹 공격
- 운영체제 설정과 취약점
- 로그와 침입탐지

# 운영체제

# 운영체제의 이해(1)

## ❒ 운영체제의 정의



# 운영체제의 이해(2)

## ❒ 운영체제의 기능

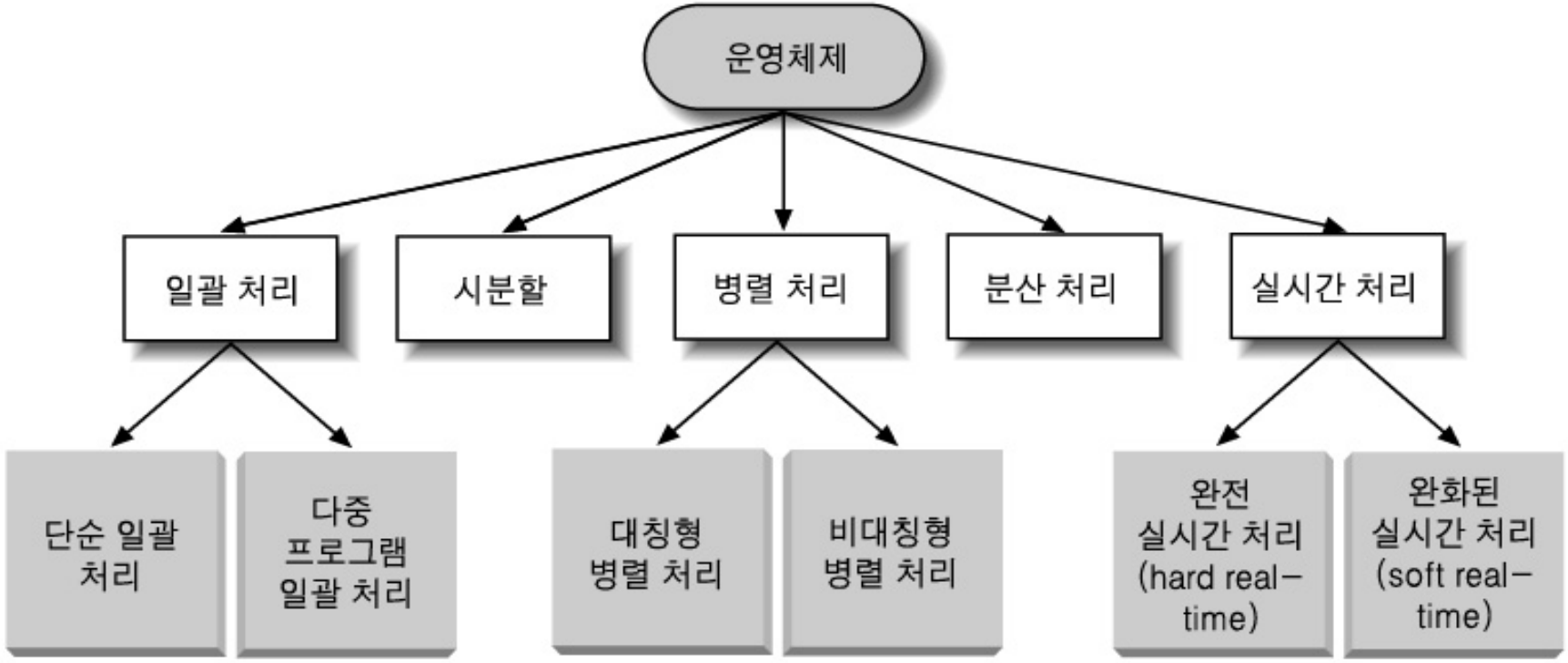
- ❍ 입출력 장치 관리
- ❍ 시스템 자원의 관리
- ❍ 에러 검출과 조치

## ❒ 커널

- ❍ 운영체제의 중심에 위치
- ❍ 인터럽트 처리
- ❍ 프로세스 관리
- ❍ 메모리 관리
- ❍ 파일 시스템 관리
- ❍ 프로그래밍 인터페이스 제공

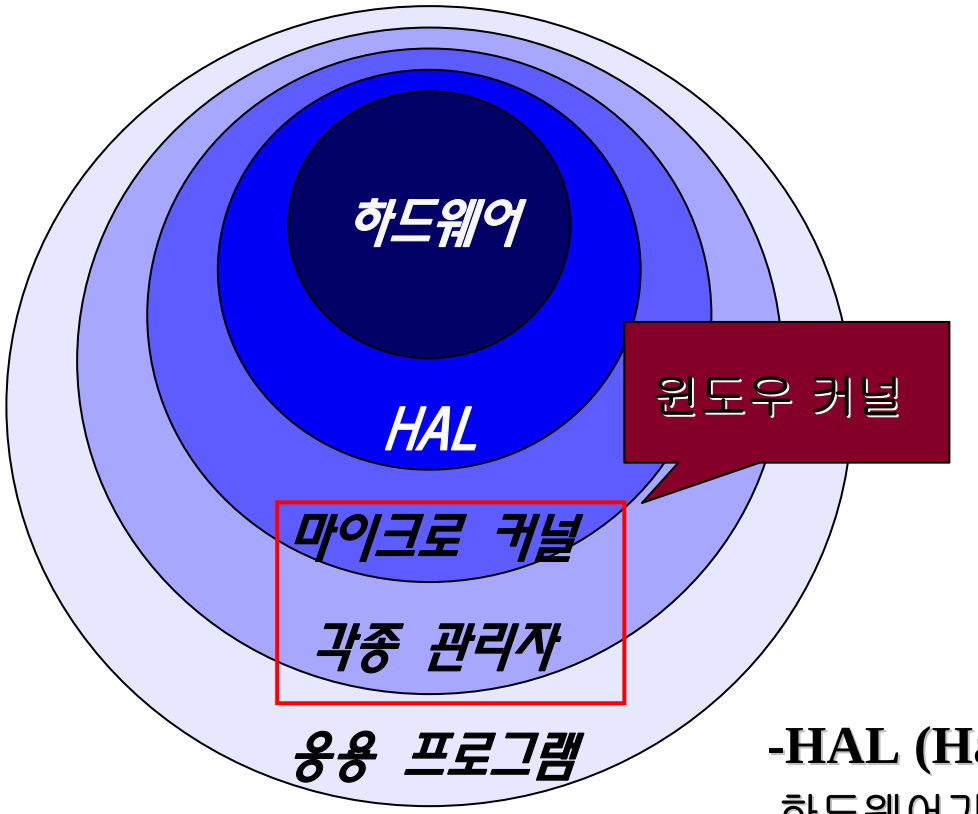
# 운영체제의 이해(3)

## ❒ 운영체제의 역사



# 윈도우 시스템

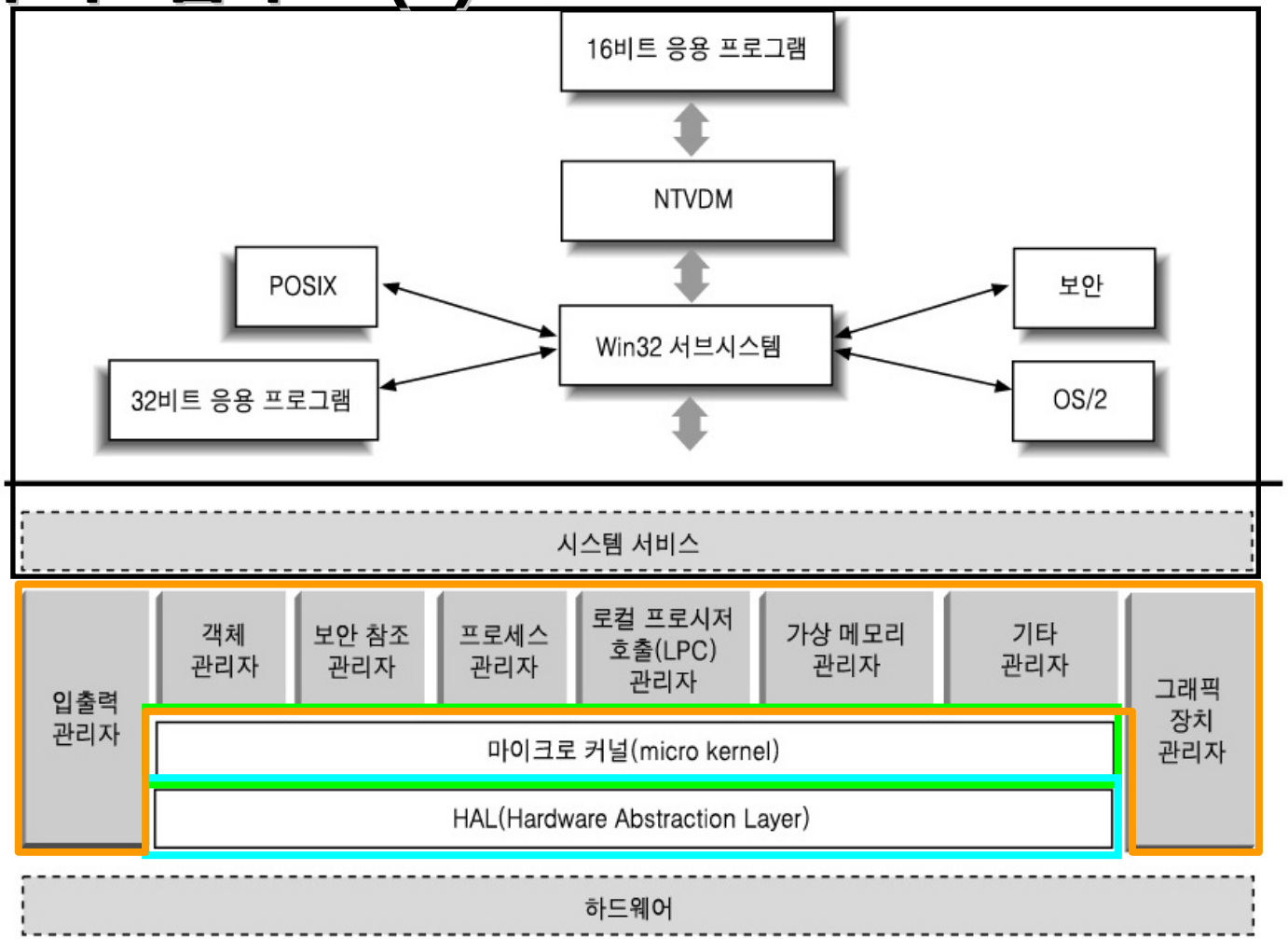
## ❒ 윈도우시스템 구조 (1)




**-HAL (Hardware Abstraction Layber) :**
하드웨어가 개발된 소프트웨어와 원활히 통신할 수 있도록 도와주는 번역자 역할

# 윈도우 시스템

## ❒ 윈도우시스템 구조 (2)



# 윈도우 파일시스템

## ❒ FAT(File Allocation Table)

- ❍ 파티션의 최대 크기: 2GB (4파티션 총 8GB까지 사용가능)

## ❒ FAT 32

- ❍ 윈도우 95 OSR(OEM Service Release) 2에서 처음 도입
- ❍ 파티션의 최대 크기: 4TB (윈도우에서 포멧시 32GB)

## ❒ NTFS

- ❍ 윈도우 NT 계열
- ❍ FAT와 HPFS (High Performance File System)의 장점을 채택
- ❍ 갑작스런 전원 차단 시에도 스스로 복구
- ❍ 파일과 폴더에 대해 사용 권한 설정 과 암호화 설정 가능

# 윈도우 부팅순서

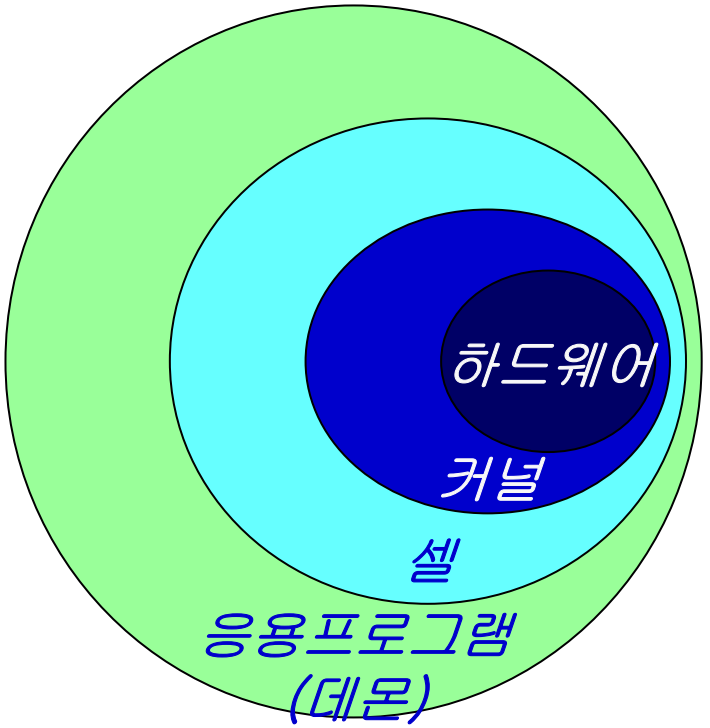
- POST (Power On Self Test)
  - BIOS에 의해 실행
- CMOS (Complementary Metal-Oxide Semiconductor)
  - -CMOS: 기본 부팅 매체 설정
  - -BIOS는 기본설정 사항을 디스템에 적용
- MBR (Master Boot Record)
  - -CMOS 정보를 읽어 부팅매체 확인 뒤, 부팅매체의 MBR정보를 읽음
  - -MBR에 기본적인 파일 시스템 정보가 있음
- NTLDR
  - -하드디스크의 부팅파티션에 있는 프로그램
  - -윈도우 2000이 부팅될 수 있도록 간단한 시스템 실행 (boot.ini)
- NTDETECT.com
  - -NTLDR로부터 제어권을 넘겨받아 시스템에 설치된 하드웨어 검사
  - → 문제가 없으며느 레지스트리의 하드웨어키 생성
- ntoskrnl.exe
  - HAL을 로드

# 유닉스 시스템(1)

❒ 유닉스 시스템 커널 : 모놀리식 커널 / 마이크로 커널

-하나의 거대한 프로그램 단위

-개별적인 모듈




❒ 유닉스 시스템 링 구조

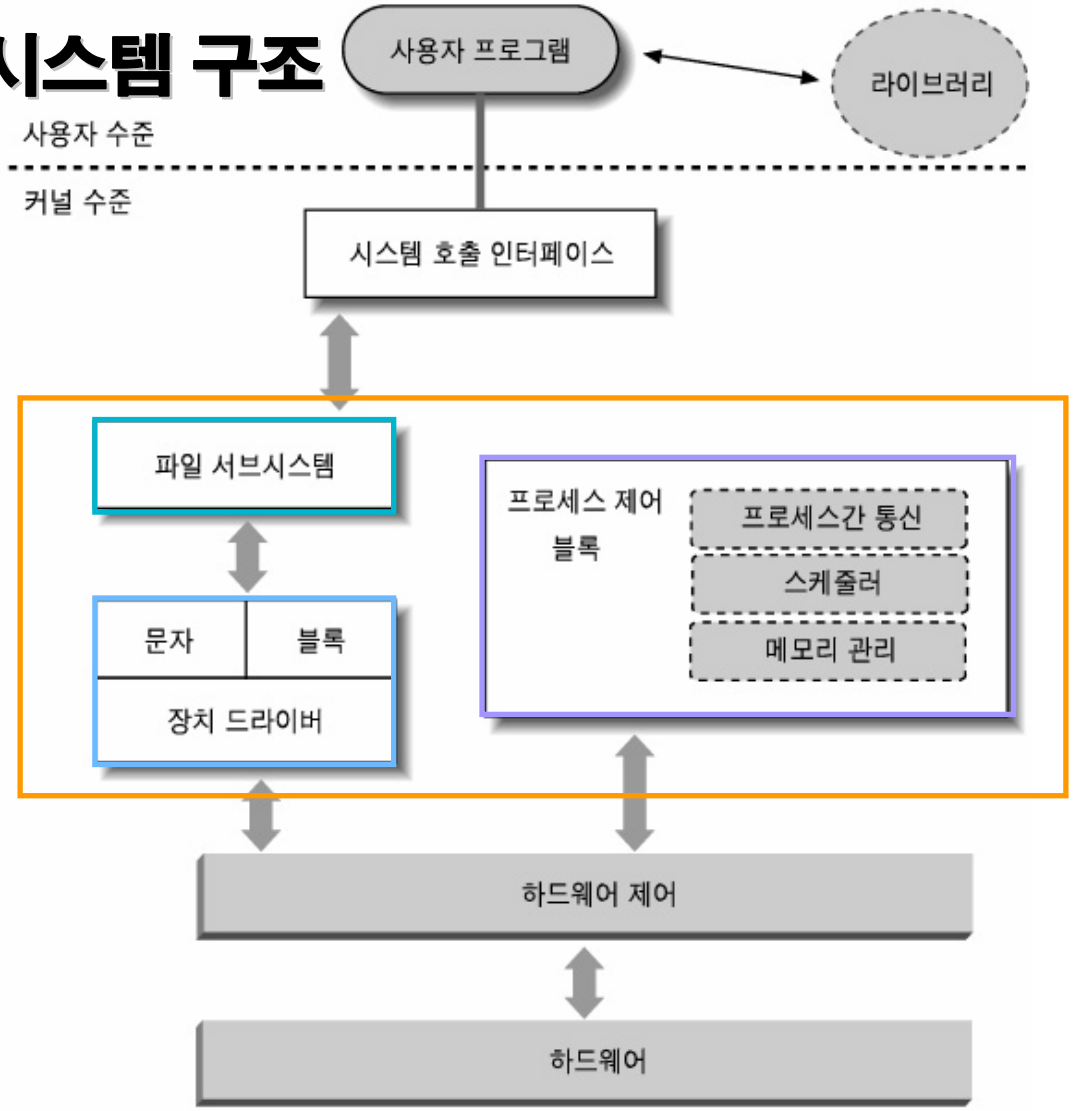
❒ 셸 (Shell)

❍ 응용 프로그램으로부터 명령을 받아 커널에 전송

❍ 사용자의 키보드 입력을 인식하고 해당 프로그램을 수행

# 유닉스 시스템(2)

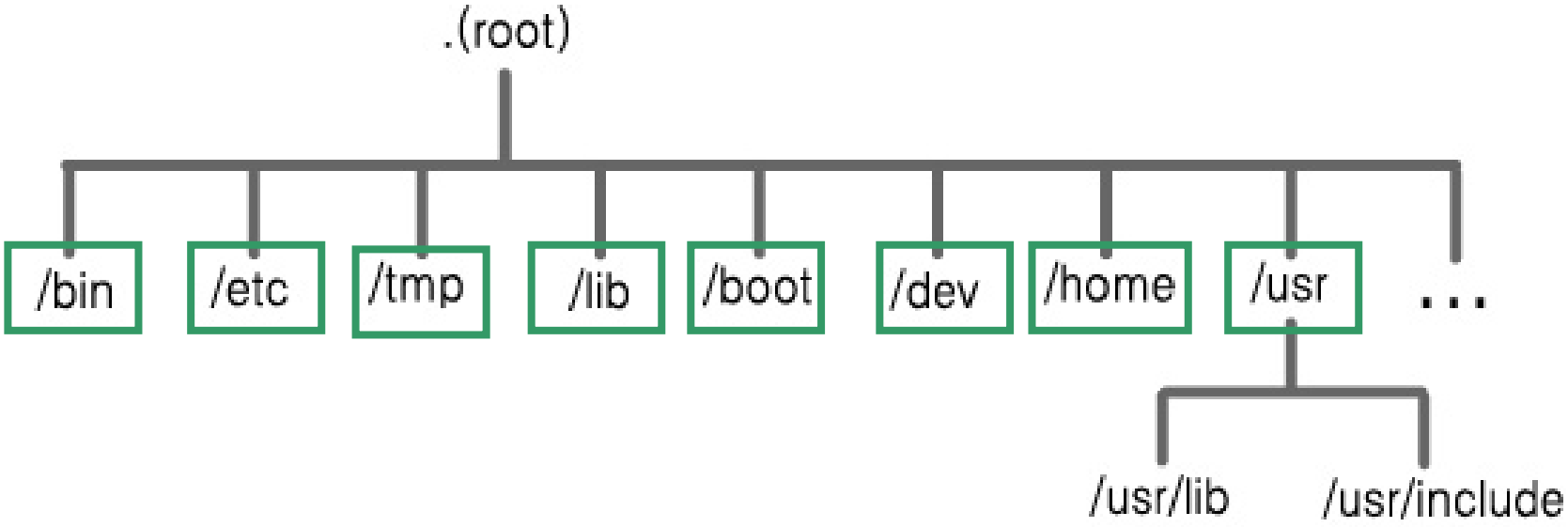
## ❒ 윈도우시스템 구조



# 리눅스/유닉스 파일시스템 (1)

❒ 파일종류: 일반 파일, 디렉토리 파일, 특수 파일, 파이프 파일

❒ 디렉토리 구조



# 리눅스/유닉스 파일시스템 (2)

| 디렉토리 | 내 용 |
|---|---|
| /bin | 기본적으로 실행 가능한 파일이 위치<br>echo, mv, copy, pwd, who 등 |
| /etc | 시스템의 환경 설정 및 주요 설정 파일이 위치<br>passwd, hosts, xined.conf 등 |
| /tmp | 프로그램 실행 및 설치 시 생성되는 임시 파일이 위치<br>※ tmp 디렉토리 밑에 파일을 저장할 경우, 재부팅 시<br>임의로 삭제될 수 있음 |
| /lib | 기본적인 프로그램의 모듈이 위치 |
| /boot | 커널을 위한 프로그램 파일로서 부팅할 때 읽혀 수행 |
| /dev | 프린터나 터미널 같은 물리적인 장치를 다루기 위한<br>특수 파일이 위치 |
| /home | 각 사용자의 작업 디렉토리 |
| /usr | 사용자가 직접 쓰는 파일이 위치 |
| /usr/lib | C나 FORTRAN의 라이브러리 디렉토리 |
| /usr/include | C 언어에 사용되는 헤더 파일들이 위치 |

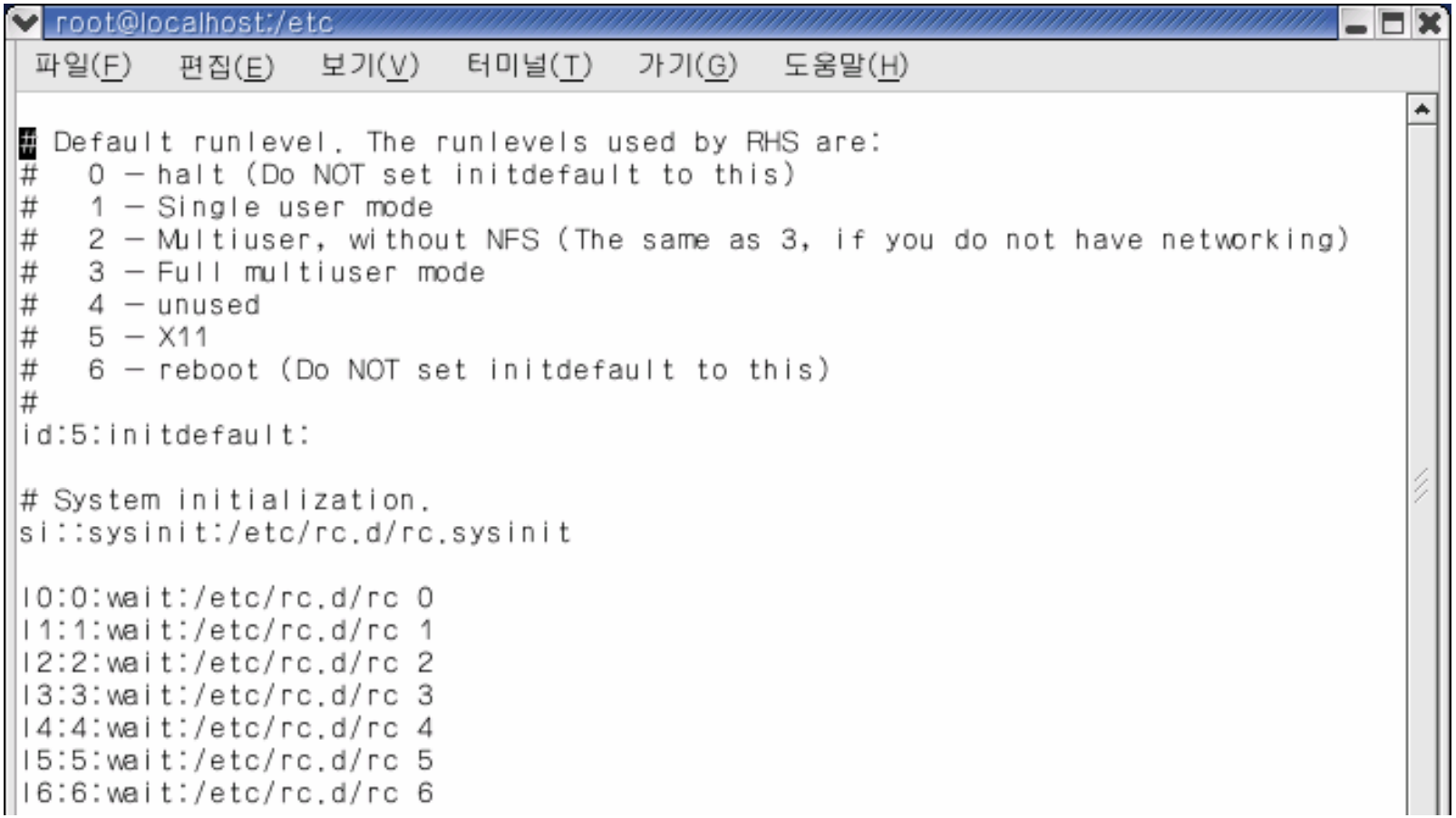
# 리눅스/유닉스 부팅순서

- POST (Power On Self Test)
- CMOS (Complementary Metal-Oxide Semiconductor)
- MBR (Master Boot Record)
- 부트 로더(Boot Loader)
  - -리눅스 부트로더: LILO / GRUB
  - -MBR에서 LILO로 부팅 제어권을 받으면 LILO는 입력받은 커널 이미지로 부팅
  - -부팅이미지는 /etc/lilo.conf파일을 설정 (선택된 이미지→메모리에 로드됨)
- 프로세스
  - -swapper (pid 0) 프로세스 실행
  - -swapper는 init (pid 1)프로세스 실행
  - -init 프로세스는 /etc/inittab파일을 읽음
- /sbin/update (버퍼 청소)

# 리눅스/유닉스 부팅레벨

**-- inittab 파일:**

**부팅시 실행 레벨을 제공할수 있도록 하는 파일**



# 리눅스/유닉스 부팅레벨

- 0 : 시스템을 종료할 때 사용
- 1 : 싱글 유저모드(Single User Mode), 관리자 권한의 셸, 기능 제약
- 2 : NFS(Network File System)를 지원하지 않는 다중 사용자 모드
- 3 : 일반적인 셸 기반의 인터페이스를 가진 다중 사용자 모드
- 4 : 기본적으로 사용되지 않으나 사용자가 정의해서 사용
- 5 : 기본적으로 실행레벨 3과 같으나, GUI 환경을 지원
- 6 : 재 부팅

# 시스템 해킹 공격

# 패스워드 (1)

## ❒ 패스워드의 중요성

- ❍ 패스워드 크래킹 : 다른 해킹과 달리 난이도가 쉬우면서도 강력한 공격

## ❒ 나쁜 패스워드

- ❍ 길이가 너무 짧거나 패스워드가 null 인 경우
- ❍ 사전에 나오는 단어나 이들의 조합, 키보드 자판의 나열
- ❍ 사용자 계정 정보에서 유추 가능한 단어

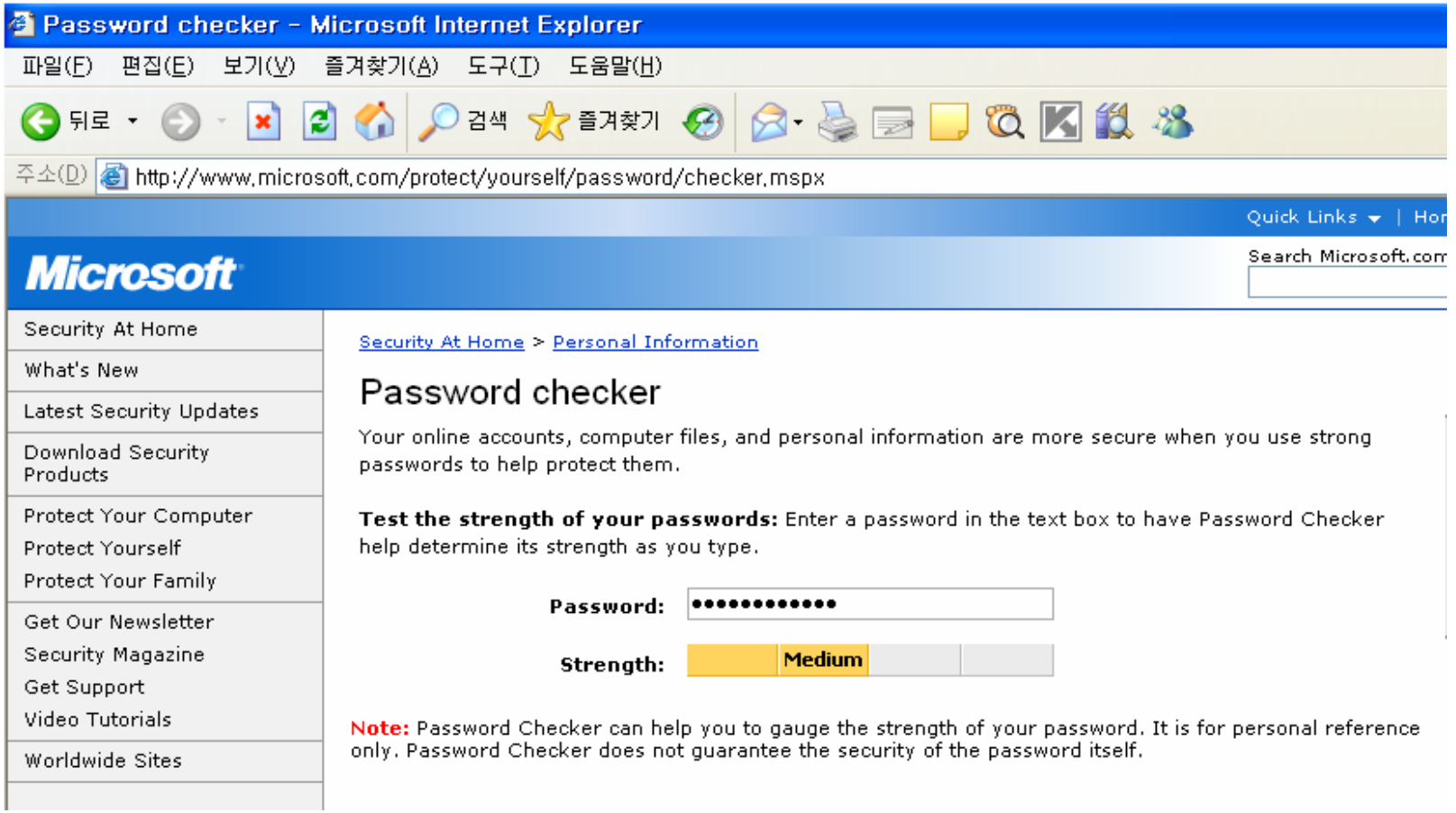
## ❒ 좋은 패스워드

- ❍ 기억하기 쉽지만 크랙하기 어려운 패스워드
- ❍ 영문과 하나 이상의 숫자 그리고 특수문자를 적절히 조합하여 사용
- ❍ 관리 : 패스워드를 주기적으로 교체 하며 적어두지 말아야 함

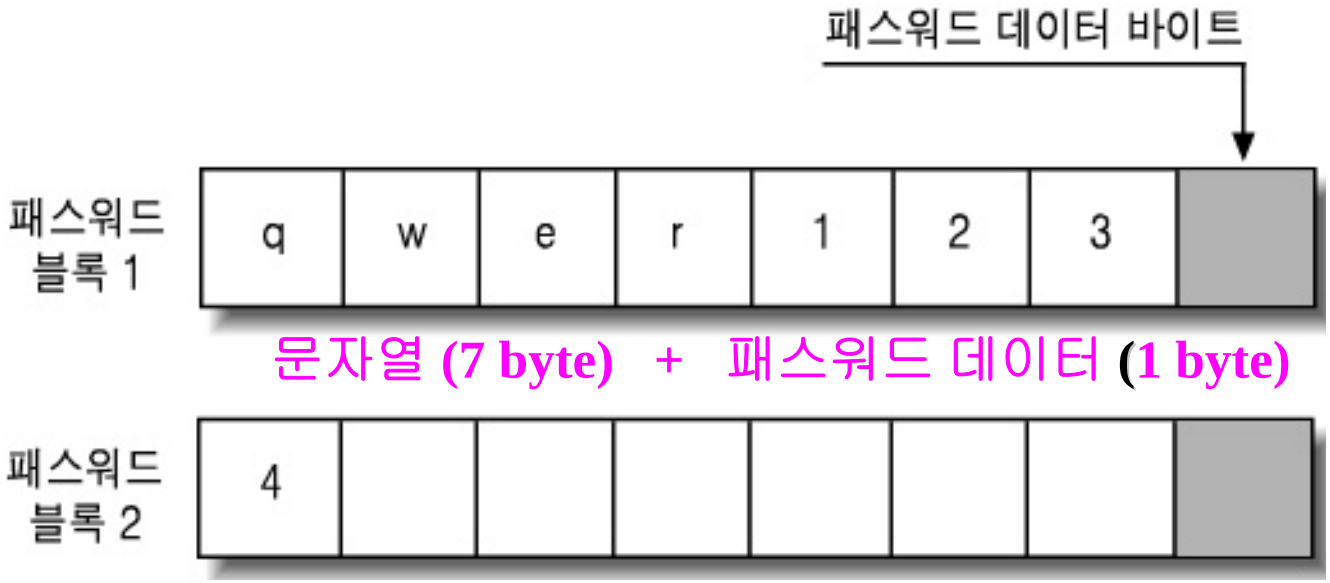
# 패스워드 (2)

## 패스워드 보안레벨 확인 사이트

http://www.microsoft.com/protect/yourself/password/checker.mspx



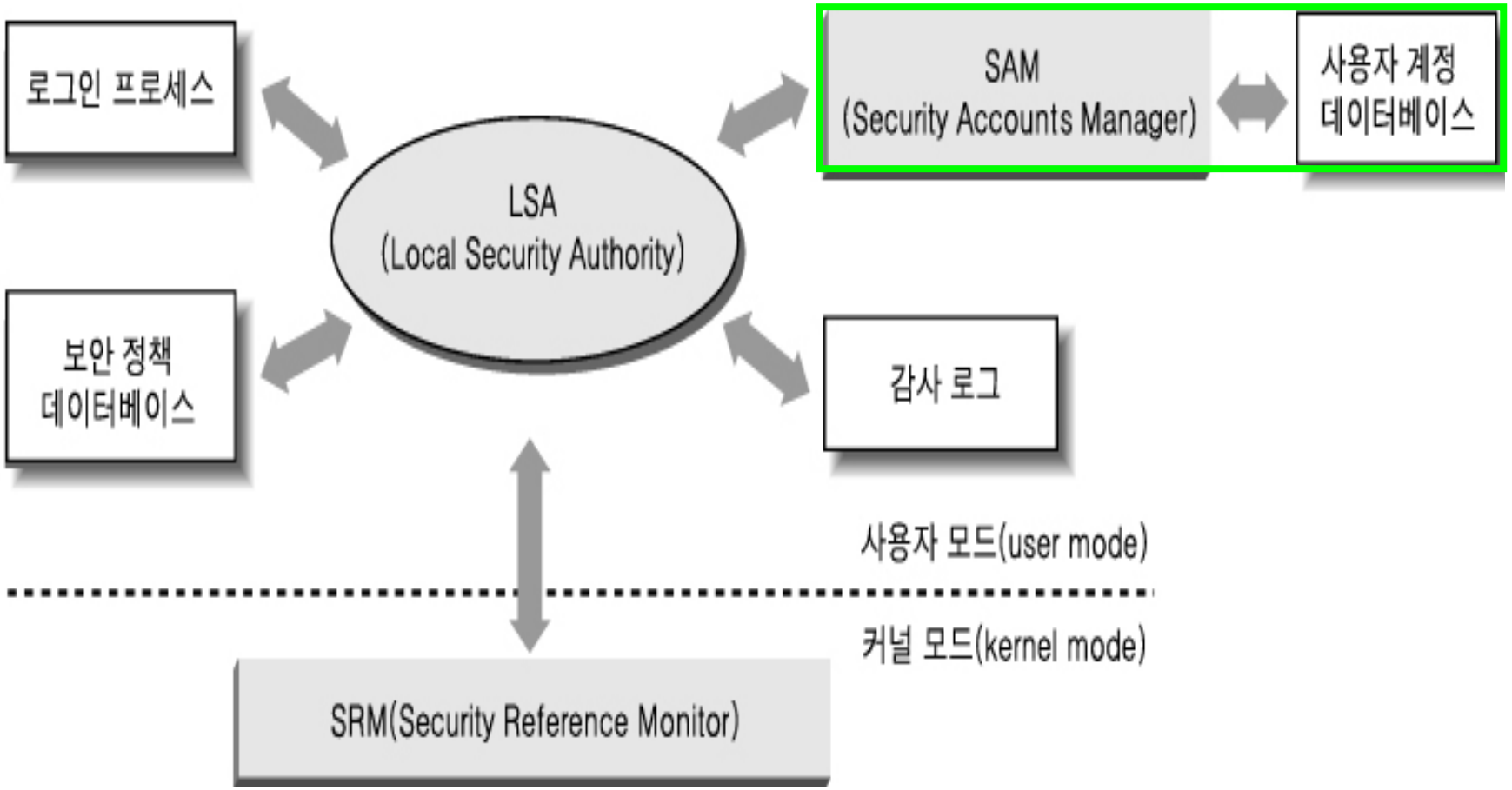
# 윈도우 패스워드 구조




## ❒ 윈도우 패스워드 블록구조 취약점

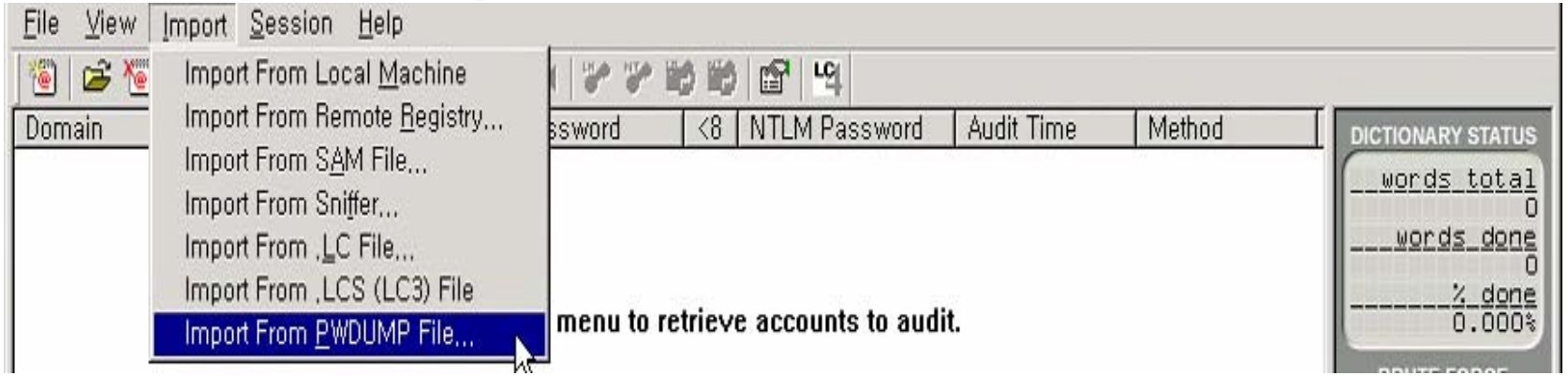
- ❍ qwer1234는 2개의 패스워드 블록 qwer123, 4로 나뉨
- ❍ 8개의 패스워드 크래킹은 7개의 패스워드 크래킹 2배의 노력
- ❍ 리눅스의 경우 1글자가 늘어남에 따라 패스워드 강도는 약 100배 정도 상승

# 윈도우 인증 구조

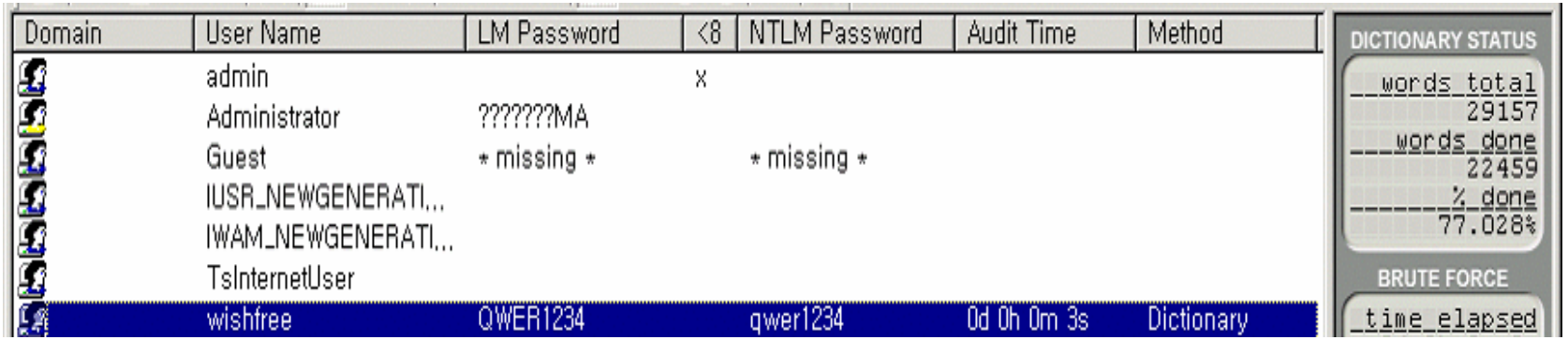

로그인 프로세스
SAM
(Security Accounts Manager)
사용자 계정
데이터베이스
LSA
(Local Security Authority)
보안 정책
데이터베이스
감사 로그
사용자 모드(user mode)
커널 모드(kernel mode)
SRM(Security Reference Monitor)

# 윈도우 패스워드 크랙

- SAM 파일의 덤프 획득
  - c:₩>pwdump3 ₩₩172.16.0.4 sam_file administrator
- LC4 에 pwdump 로드




- 패스워드 크랙



# 리눅스/유닉스 인증 구조

❒ /etc/passwd : 644 권한 → 600 권한으로 변경

❍ 일반 계정 권한도 읽기 가능(시스템에 존재하는 계정 확인 가능)

```
root : x : 0 : 0 : root : /root : /bin/bash
```

❒ /etc/shadow : 암호화된 패스워드가 저장

```
root : $1$pS/cfFID$pzmD5rjrl8qnXiM5xr/ : 12364 : 0 : 99999 : 7 :  :  :
```

# 리눅스 패스워드 크랙

## ❒ John 를 이용한 shadow 파일 크랙

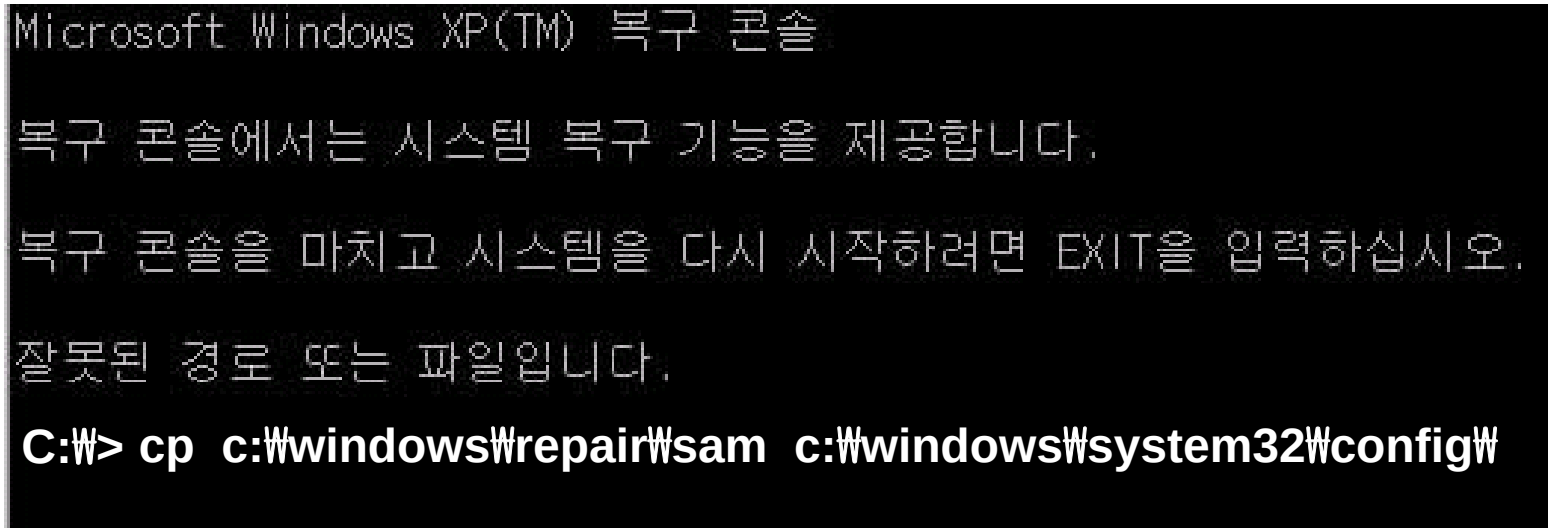
### ❍ john –wordfile:[패스워드사전파일] shadow

명령 프롬프트 (2)

```
C:₩john-16₩run>john -wordfile:word.txt shadow
Loaded 3 passwords with 3 different salts (FreeBSD MD5 [32/32])
qwer1234         (root)
qwer1234         (user1)
qwer1234         (wishfree)
guesses: 3  time: 0:00:00:08 100%  c/s: 3724  trying: qwer1234
```

# 윈도우 패스워드 복구(1)

- 1. 윈도우XP CD로 부팅 후 '복구<R>' 선택
- 2. 키보드 선택
- 3. 복구 콘솔모드에서 명령어 실행




- 4. 초기 상태로 됨

# 윈도우 패스워드 복구(2)

- 패스워드 복구용 CD를 이용한 부팅
- 파티션 마운트
  - 윈도우가 설치되어있는 /dev/sd1을 선택
  - 파티션 마운트 후 c:₩winnt₩system32₩config₩ 파일 열람 가능
  - ※ IDE : 파티션 마운트 필요 없음, SCSI하드 : 파티션 마운트 필요
- SAM 파일 편집(Edit user data and passwords 선택)
- SAM 파일 편집 후 추출
- 패스워드 변경
  - '*'를 입력하면 패스워드가 설정되지 않은 상태가 됨
- 변경 사항 저장 후 종료

# 리눅스 패스워드 복구

## ❒ 1. 부트로드에서 Single Mode 설정

```
grub edit> kernel /vmlinuz-2.4.20-8 ro root=LABEL=/ hdc=ide-scsi single
```

or

```
grub edit> kernel /vmlinuz-2.4.20-8 ro root=LABEL=/ 1
```

## ❒ 2. Single Mode 부팅후 패스워드 변경

```
Loading keymap:                                            [  OK  ]
Loading system font:                                       [  OK  ]
Telling INIT to go to single user mode.
INIT: Going single user
INIT: Sending processes the TERM signal
INIT: Sending processes the KILL signal
sh-2.05b#
sh-2.05b#
sh-2.05b# _
```

# 백도어

## ❒ 트로이 목마

- ❍ 정상적인 기능의 프로그램처럼 위장
- ❍ 악성코드가 들어있는 프로그램

## ❒ 백도어

- ❍ 정상적인 인증 과정 무시하고 OS나 프로그램에 접근하는 통로
- ❍ 외부 공격자가 시스템에 침입 후 이후 침입 때 복잡한 과정 없이 관리자 권한을 얻을 수 있도록 하는 뒷구멍
- ❍ 프로그래머가 유지보수 목적으로 응용프로그램에 넣을 수도 있음

# 백도어의 종류

- 로컬 백도어
- 원격 백도어
- 원격 GUI 백도어
- 패스워드 크래킹 백도어
- 시스템 설정 변경 백도어
- 트로이 목마 형태의 프로그램
- 거짓 업그레이드

# 백도어 탐지

- 1. 현재 동작중인 프로세스 확인
- 2. 열린 포트 확인
- 3. 주기적인 SetUID 파일 검사(유닉스)
- 4. 바이러스 및 백도어 탐지 툴 이용
- 5. 무결성 검사(MD5 해시 기법)
- 6. 로그 분석('Cyber Foresic‘)

# 윈도우 백도어 탐지 (1)

## ❒ 1. 프로세스 확인 : C:₩TEST₩Pstools>pslist

```
Name             Pid Pri Thd  Hnd     Mem    User Time    Kernel Time    Elapsed Time
Idle               0   0   1    0      16  0:00:00.000    0:57:27.126     1:13:15.720
System             8   8  45  221     288  0:00:00.000    0:00:32.456     1:13:15.720
```

## ❒ 2. 열린 포트 확인

### ❍ netstat –an

```
  TCP    192.168.68.3:1179      192.168.68.4:445       TIME_WAIT
  TCP    192.168.68.3:1180      192.168.68.4:445       TIME_WAIT
  TCP    192.168.68.3:1181      192.168.68.4:445       TIME_WAIT
```

### ❍ Fport로 포트별 응용 프로그램 확인

```
1104   inetinfo        ->  9722   TCP    C:₩WINNT₩System32₩inetsrv₩inetinfo.exe
1908   sndloader       ->  10176  TCP    C:₩WINNT₩System32₩sndloader.exe
1852   Patch           ->  12345  TCP    C:₩NetBus₩Patch.exe
```

### ❍ Promiscuous 모드 탐지 : promiscdetect

# 윈도우 백도어 탐지 (2)

- 3. 백도어 탐지 툴의 이용
- 4. 무결성 검사
  - SFC 의 이용(원본 CD필요) : sfc /scannow
  - md5sum 이용

```
C:\TEST2\Wishfree>md5sum *.* > test.md5
md5sum: .: Permission denied
md5sum: ..: Permission denied

C:\TEST2\Wishfree>type test.md5
30c3346fb767ce8b3c09037cf8319262 *Hwp.hwp
eb574b236133e60c989c6f472f07827b *md5sum.exe

C:\TEST2\Wishfree>md5sum -c test.md5
Hwp.hwp: FAILED
md5sum.exe: OK
Microsoft PowerPoint .ppt: FAILED
Microsoft Word.doc: OK
test.md5: FAILED
md5sum: WARNING: 3 of 5 computed checksums did NOT match
```

# 리눅스 백도어 탐지

- 1. 프로세스 확인 : ps -ef

```
root      1915  1912  0 19:25 ?        00:00:00 /sbin/pam_timestamp_check
root      1921     1  0 19:25 ?        00:00:00 /usr/libexec/notification
root      1935     1 10 21:00 ?        00:00:04 gnome-terminal
```

- 2. 열린 포트 확인 : netstat -an

```
tcp        0      0 0.0.0.0:32768           0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:32769         0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:513             0.0.0.0:*               LISTEN
```

- 3. SetUID 파일 검사 : find / -perm +4000
- 4. 백도어 탐지 툴의 이용 (Tripwire 등)
- 5. 무결성 검사

# 레이스 컨디션

- 레이스 컨디션
  - 관리자 권한으로 실행되는 프로그램 중간에 끼어들어 자신이 원하는 작업을 하는 것

- 하드 링크 : ln [원본파일] [하드링크파일]
  - 두 파일을 ln(link)으로 연결 → 링크 수 증가
  - 링크 파일을 수정 → 원본 파일도 수정,
  - 두 파일 중 하나를 삭제 → 링크 숫자만 하나 작아짐
  - 하드 링크는 동일한 파일시스템(UFS, EXT 등) 내에서만 생성

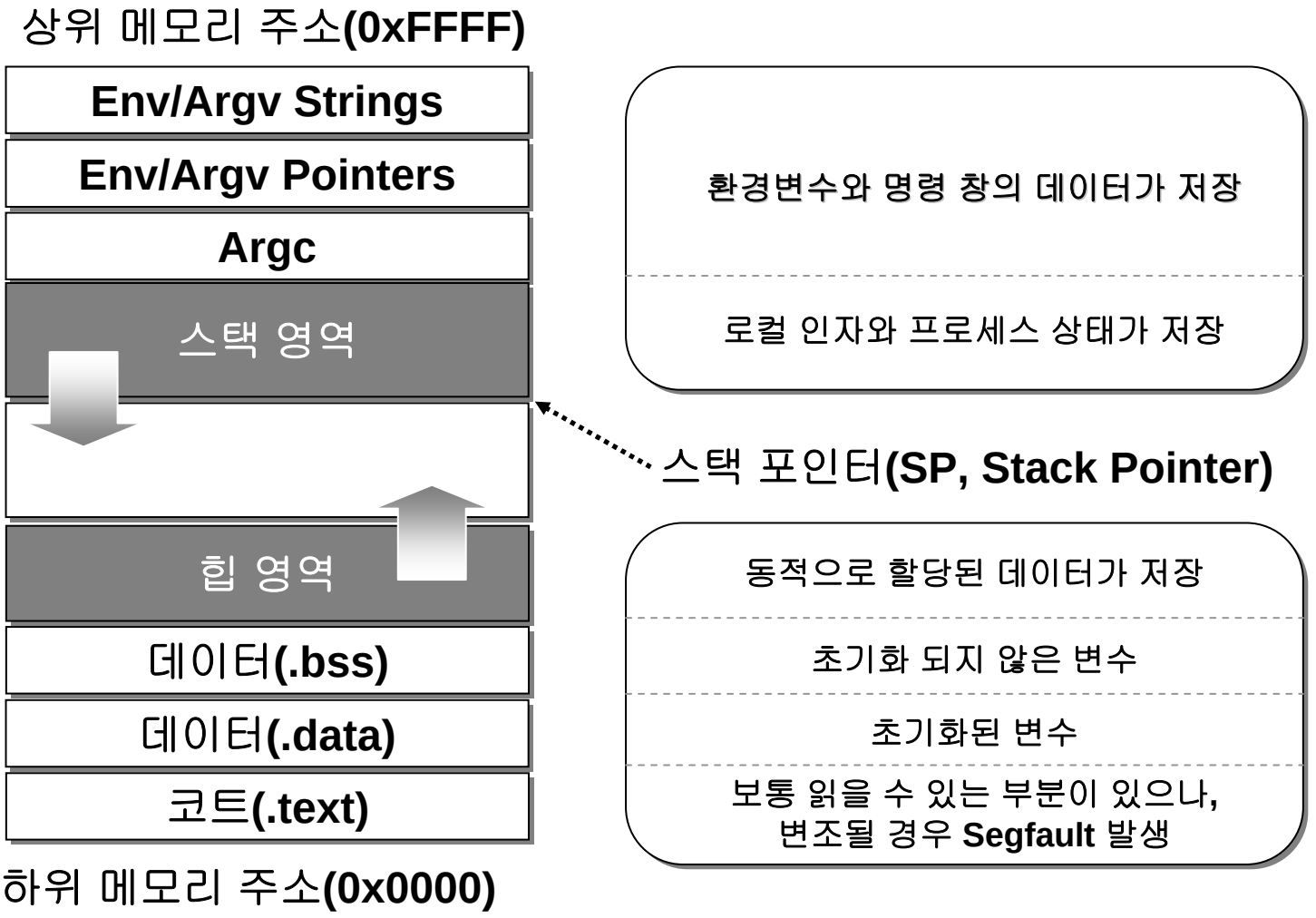
- 심볼릭 링크 : ln -s [원본파일] [심볼링크파일]
  - 윈도우의 단축아이콘이랑 비슷한 개념
  - 링크 수가 변하지 않음, 링크된 파일의 정보만 표시
  - 원본파일 삭제 시 심볼릭 파일에서는 아무런 데이터도 얻을 수 없음

# 레이스 컨디션 공격 순서(임시 파일)

- 1. 취약 프로그램이 생성하는 임시 파일의 이름을 확인
- 2. 생성될 임시 파일과 같은 이름의 파일을 생성
- 3. 생성한 임시 파일에 대한 심볼릭 링크 설정
- 4. 생성한 임시 파일 삭제
- 5. 취약 프로그램이 해당 임시 파일을 생성할 때 심볼릭 링크된 파일에 원하는 값을 삽입

# 버퍼 오버플로우(1)

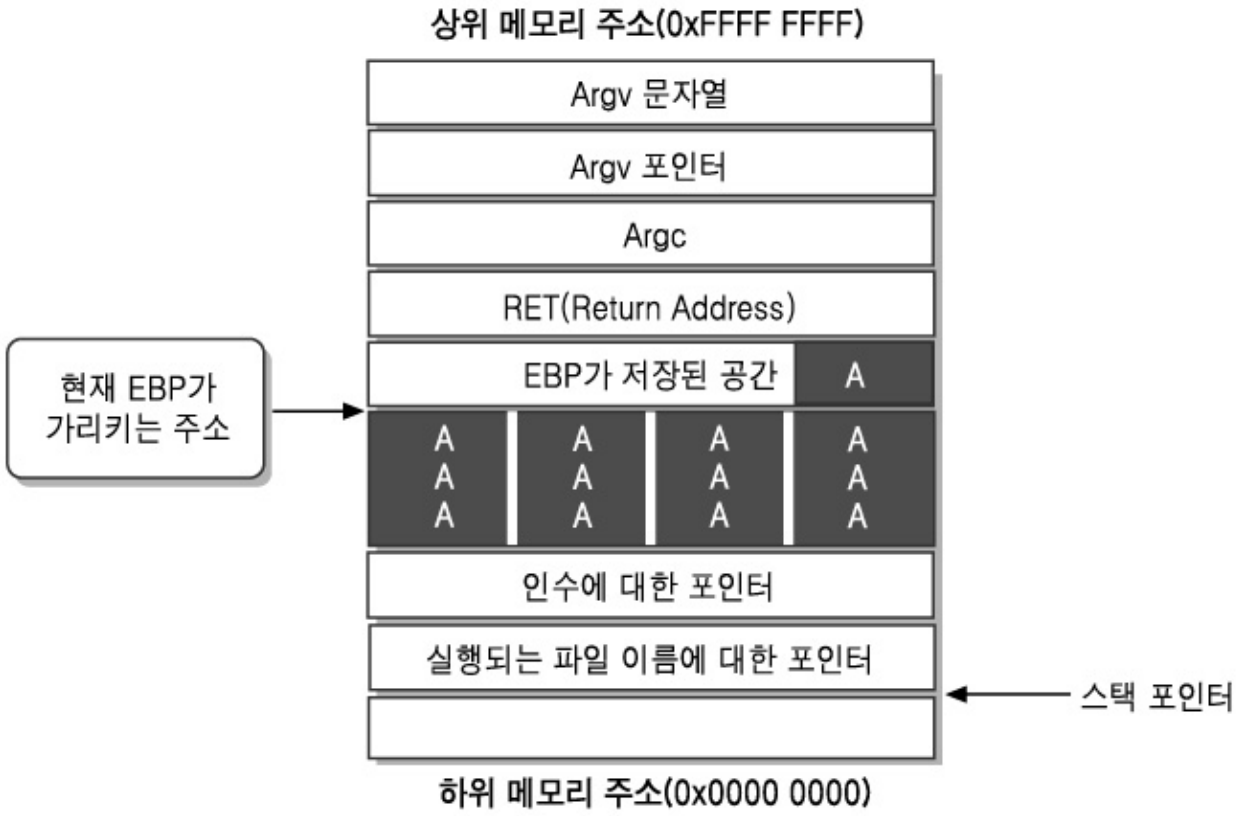
## ❒ 메모리 기본 구조



# 버퍼 오버플로우(2)

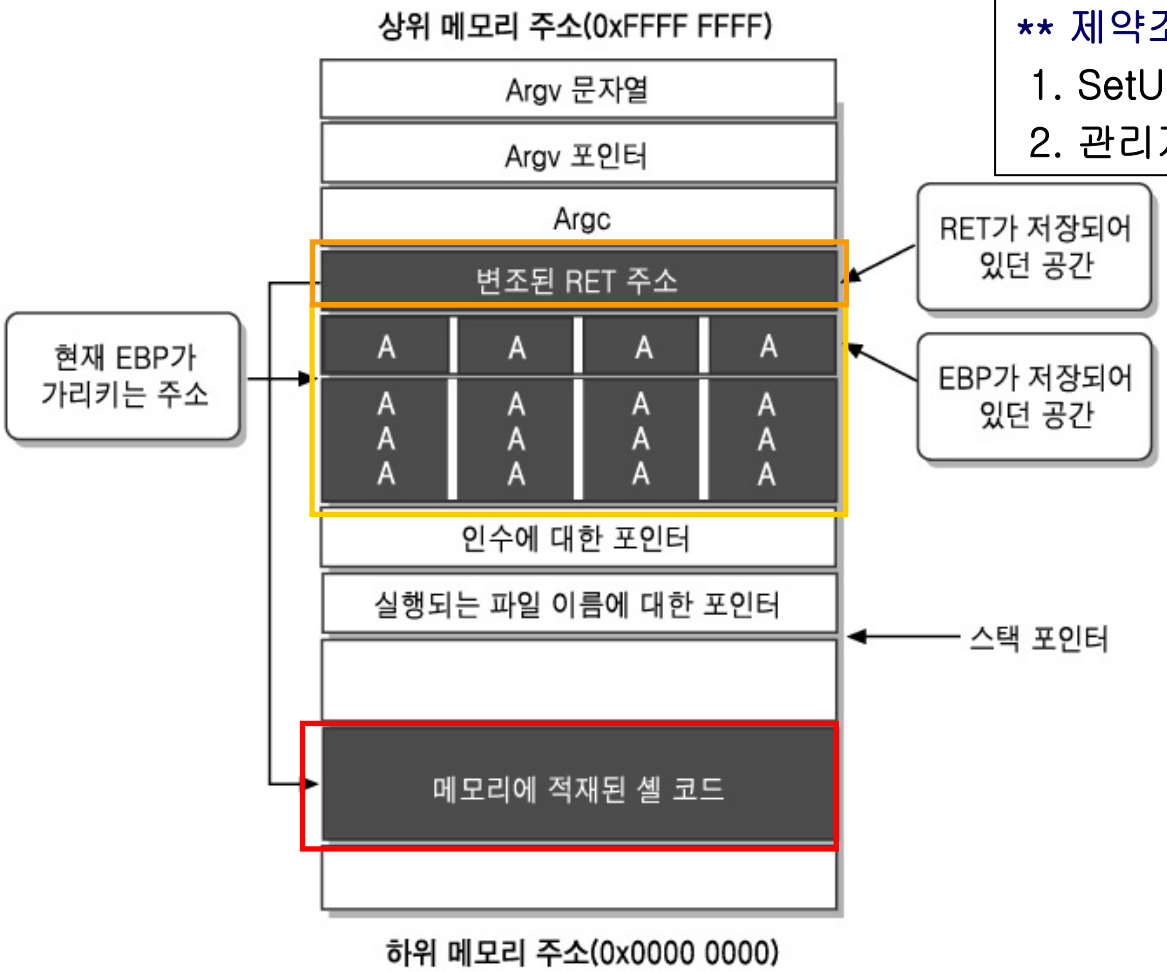
## ❒ 스택 버퍼 오버플로우 공격에 대한 이해

- ❍ 정상적인 경우에는 사용되지 않아야 할 주소 공간에 임의의 코드를 해커가 **덮어씌우는 것**



# 버퍼 오버플로우(3)

## ❒ 버퍼 오버플로우 공격 시 메모리 상태




** 제약조건 (공격 가능한 프로그램 내) **
1. SetUID 비트가 설정
2. 관리자 소유의 파일이 존재

# 버퍼 오버플로우 대응책

- 스택가드 (stack guard)
  - -Canary 바이트 이용:
    - .초기:스택에 각 값이 저장되는 사이에 특정한 값을 입력해 두어 그 값이 변조되는지 확인 후, 변조→ 프로세스 종료
    - .Randon Canary: 스택에 저장되는 Canary값이 실행될 때마다 변조
- Stack Shield
- 시스템 커널 업그레이드
- 보안패치
- 버퍼 오버플로우 취약점 → 시스템과 프로그램 개발 시 생성
- 시스템 관리자가 설정 등을 통해 시스템 보안수준을 높이는 것이 어려움

# 포맷 스트링(1)

| 올바른 함수 사용 | 잘못된 함수 사용 |
|---|---|
| [right.c]<br>#include <stdio.h><br>main(){<br>char *buffer = "wishfree";<br>printf("%s₩n", buffer);<br>}<br>포멧 스트링 | [wrong.c]<br>#include <stdio.h><br>main(){<br>char *buffer = "wishfree₩n";<br>printf(buffer);<br>}<br>실행 결과는 같지만<br>보안에 있어 치명적 함수<br>사용법 |

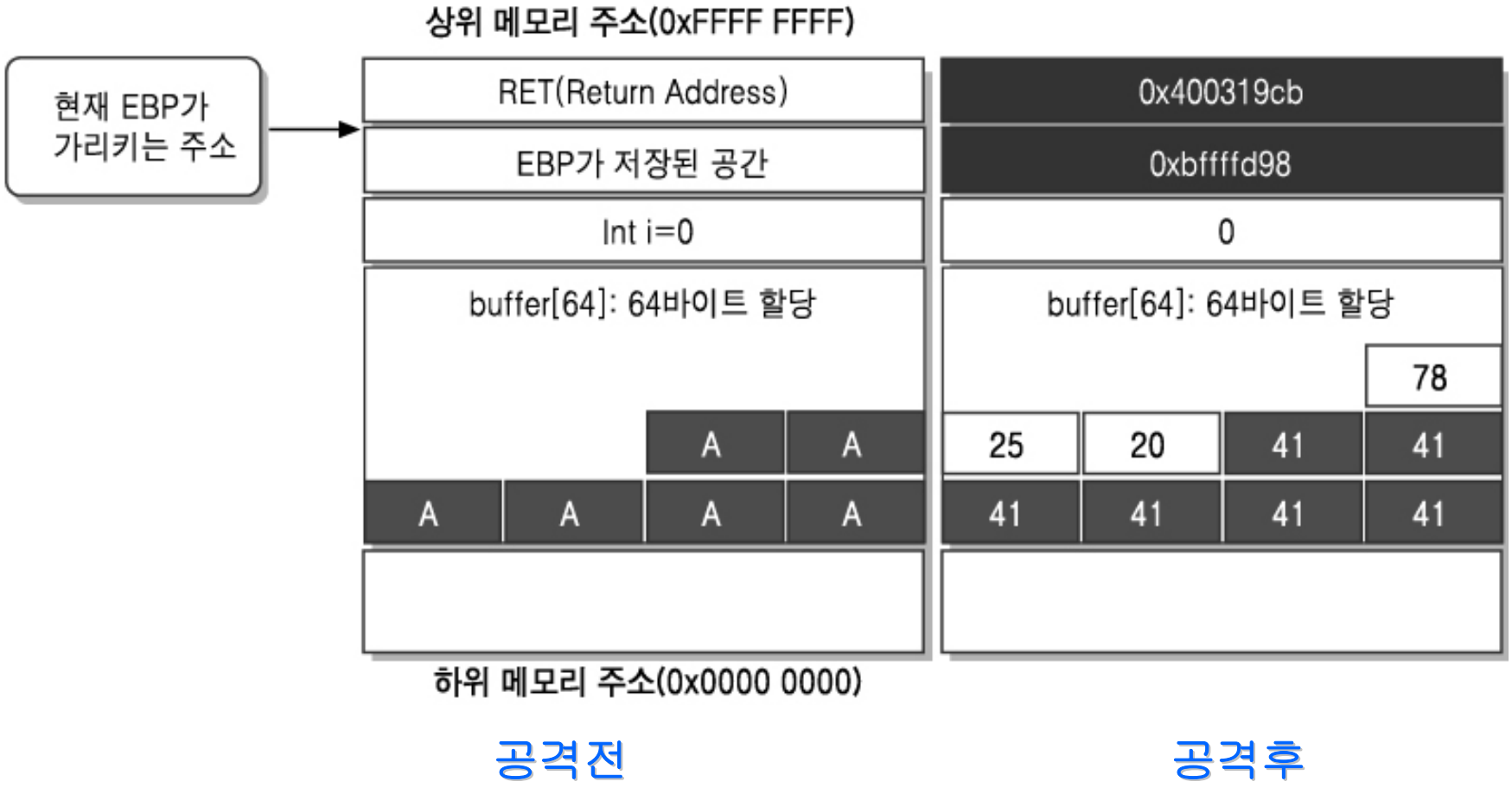
## ※버퍼 오버플루우 공격과 매우 유사

## ❒ 포맷 스트링의 문자의 종류

| 파라미터 | 변수 형식 | 파라미터 | 변수 형식 |
|---|---|---|---|
| %d | 정수형 10진수 상수 (integer) | %o | 양의 정수 (8 진수) |
| %f | 실수형 상수 (float) | %x | 양의 정수 (16 진수) |
| %lf | 실수형 상수 (double) | %s | 문자열 |
| %c | 문자 값 (char) | %n | * int (쓰인 총 바이트 수) |
| %s | 문자 스트링 ((const)(unsigned) char *) | %hn | %n의 반인 2바이트 단위 |
| %u | 양의 정수 (10 진수) | | |

# 포맷 스트링(2)

## ❒ 포맷 스트링 공격시 메모리



# 포맷 스트링(3)

## ❒ 포맷 스트링 대응책

- ❍ printf와 같이 포맷 스트링을 사용하는 함수를 정상적으로 사용

## ❒ 주요 취약 함수

| | | |
|---|---|---|
| 1 | printf("%s\n", buffer); | |
| 2 | int fprintf(fp, const char *fmt, ...) | |
| | fp**는 파일에 대한 포인터 값 파일로 출력** | fp = fopen("/dev/null", "w"); //**파일을 쓰기 모드로 개방**<br>fprintf(fp, "decimal=%d octal=%o", 123,123); |
| 3 | int sprintf (char *str, const char *fmt,...) | |
| | **결과를 문자열로 출력** | a= sprintf("decimal= $d octal= %o", 123,123);<br>Printf(" ", a); |
| 4 | snprintf (char *str, size_t count, const char *fmt,...) | |
| | **결과를 문자열로 출력, (복사될 문자열 크기 지정)** | |
| 5 | **유닉스** Sysem V**에서 사용하는 함수** : vfprintf, vprintf, vsprintf, vsnprintf | |

# 운영체제 보안 설정과 취약점

# 윈도우 시스템 보안설정 (1)

## ❒ 계정 관리

- ❍ 패스워드 관리 : 주기적으로 관리하는 시스템의 패스워드를 크랙
  → 취약한 패스워드를 가진 계정을 체크
- ❍ 계정의 생성은 문서화하여 기록하고 주기적으로 불필요한 계정을 삭제
- ❍ 계정 잠금 정책 : 임계 값 5회로 설정

## ❒ 로컬 보안 설정

- ❍ 네트워크에서 이 컴퓨터 액세스/거부
- ❍ 로컬 로그온 /로컬로 로그온 거부
- ❍ 시스템 종료/원격 시스템에서 강제로 시스템 종료
- ❍ 로그온 스크린에 마지막 사용자 이름 표시하지 않음
- ❍ 로그온 시도하는 사용자에 대한 메시지 제목 /텍스트
- ❍ 로그온 하지 않고 시스템 종료 허용
- ❍ 보안 감사를 로그 할 수 없는 경우 즉시 시스템 종료
- ❍ Administrator 계정 이름 바꾸기

# 윈도우 시스템 보안설정 (2)

- **FTP (포트 21)**
  - **IIS와 함께 설치되는 기본서비스 - 필요하지 않으면 중지**
  - **익명계정(anonymous)의 로그인 → 익명계정 로그인 금지 설정**
  - **필요 이상의 세션 수 제안 설정**
  - **FTP의 W3G(World Wide Web Consortium) 로그 정보**
    - c:/winnt/system32/logfiles/MSFTPSVC1
  - **FTP로그인 시 경고문 설정**
  - **IP에 대한 설정, FTP 기본 디렉터리 지정 및 FTP 계정 권한 설정**

- **SMTP (포트 25)**
  - **IIS와 함께 설치되는 기본서비스 - 필요하지 않으면 중지**
  - **telnet을 이용한 배너 그래빙 : SMTP의 버전과 운영체제를 추측**
    - %SYSTEMROOT%₩system32₩dllcache₩SMTPSVC.DLL 삭제
      → 배너 삭제 도구 실행
  - **로깅 설정**
    - 필수 로깅 대상 : 날짜, 시간, 클라이언트 IP, 사용자 이름, 서버 IP
    - W3C 로깅 파일 저장 위치 : c:/winnt/system32/logfiles/smtpsvc1
  - **특정 IP에 대한 접근제어**
  - **릴레이 제한 : 스팸을 전달서버에 대해 릴레이 제안을 하면 스팸 메일 감소**

# 윈도우 시스템 보안설정 (3)

- HTTP (포트 80)
  - 기본적으로 설치 되어있음(2000서버)
  - 보안을 위해 기본 웹 가상 디렉토리를 다른 임의의 디렉토리로 변경 필요
  - 서비스 포트와 연결제한, 로깅 설정
  - 익명 액세스 및 인증제어, IP 주소 및 도메인 이름 제한, 보안통신

# 윈도우 시스템 보안설정 (5)

## ❒ LOC-SRV, NetBIOS-SSN, SMB

### ❍ 윈도우 NetBIOS 관련 프로토콜

| 프로토콜 | 포트 | 서비스 |
|---|---|---|
| TCP | 135 | RPC/DCE Endpoint Mapper |
| UDP | 137 | NetBIOS **이름 해석 서비스** |
| UDP | 138 | NetBIOS **데이터그램 서비스** |
| TCP | 139 | NetBIOS **세션 서비스** (SMB/CIFS over NetBIOS) |
| TCP/UDP | 445 | Direct Host (SMB/CIFS over TCP) |

### ❍ 널 세션 막기 :
**HKEY_LOCAL_MACHINE₩System₩CurrentControlSet₩Control₩Lsa₩restrictanonymous 값=2**

### ❍ 관리자 공유 제거

### ❍ NetBIOS에 대한 취약점 제거 : Microsoft 네트워크 파일 및 프린터 공유 사용중지

# 윈도우 시스템 보안설정 (6)

## ❒ SNMP (포트 161)

- ❍ 네트워크 장비를 관리 감시하기 위한 목적의 프로토콜
- ❍ 원격지 시스템 정보 유출
- ❍ 서비스 관리자에서 SNMP 서비스 중지

## ❒ MS-Term-Serv (포트 3389)

- ❍ 터미널 서비스 포트 변경
  - 터미널 포트에 대한 비인가 접속을 막기 함
  - 레지스트리 키 값 :
    'HKEY_LOCAL_MACHINE₩System₩CurrentControlSet ₩Control₩TerminalServer₩ Wds₩Rdpwd₩Tds₩Tcp₩포트Number'

## ❒ 패치

- ❍ 마이크로 소프트 사이트에서 해당 패치를 검색하여 업데이트

# 유닉스 시스템 보안설정 (1)

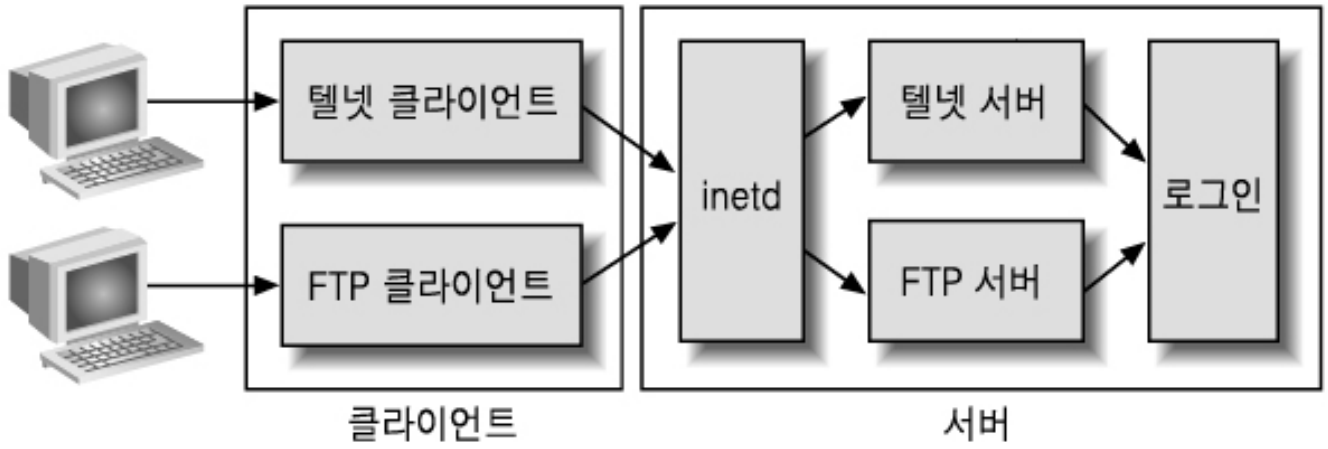
## ❒ 계정 관리

- ❍ 취약한 패스워드의 점검
- ❍ 사용하지 않는 계정의 제거
- ❍ 중복된 root 계정의 존재 여부 : grep ':0:' /etc/passwd
- ❍ 계정에서 셸 제거 : /etc/passwd

```
ntp:x:38:38::/etc/ntp:/sbin/nologin
gdm:x:42:42::/var/gdm:/sbin/nologin
postgres:x:26:26:PostgreSQL Server:/var/lib/pgsql:bin/bash
wishfree:x:500:500:yang:/home/wish:/bin/fales
```

# 유닉스 시스템 보안설정 (2)

## ❒ Inetd 데몬




❍ **텔넷 / FTP 클라이언트 → 서버에 접속 요구 → Inetd 데몬이 확인**

❍ **텔넷 연결 시도일 경우:**

- 데몬 설정 로드(/etc/inetd.conf)
- 설정된 포트 번호에 대해 클라이언트 요청(/etc/services)
- → 텔넷데몬 실행

# 유닉스 시스템 보안설정 (3)

- FTP (port 21)
  - 원격 버퍼 오버플로우 공격, 포맷 스트링 공격 취약
  - anonymous 계정 생성 금지, root 계정의 로그인 금지
  - /etc/ftpusers 를 이용한 접근제어(등록된 계정은 ftp 로그인 불가)
- SSH (port 22)
  - SSH는 암호화된 Telnet 서비스
  - SSH1은 접속과 인증을 위해서 RSA(Rivest-Shamir-Adleman)를 이용
  - 통신의 암호화를 위해서는 Blowfish, DES, 3DES, RC4 및 IDEA 등을 사용
- Telnet (port 23)
  - 암호화 되지 않은 통신 → 암호화된 세션 생성
  - 네트워크를 이용한 다양한 공격에 매우 취약
  - 관리자 원격 로그인 제한 - /etc/default/login
  - 로그인 시 경고문 출력 - /etc/issue

# 유닉스 시스템 보안설정 (4)

- SMTP (port 25)
  - 메일을 보내기 위한 프로토콜
  - expn, vrfy 명령 금지 설정 : vi /etc/mail/sendmail.cf

```
# privacy flags
O PrivacyOptions=authwarnings, noexpn, novrfy
```

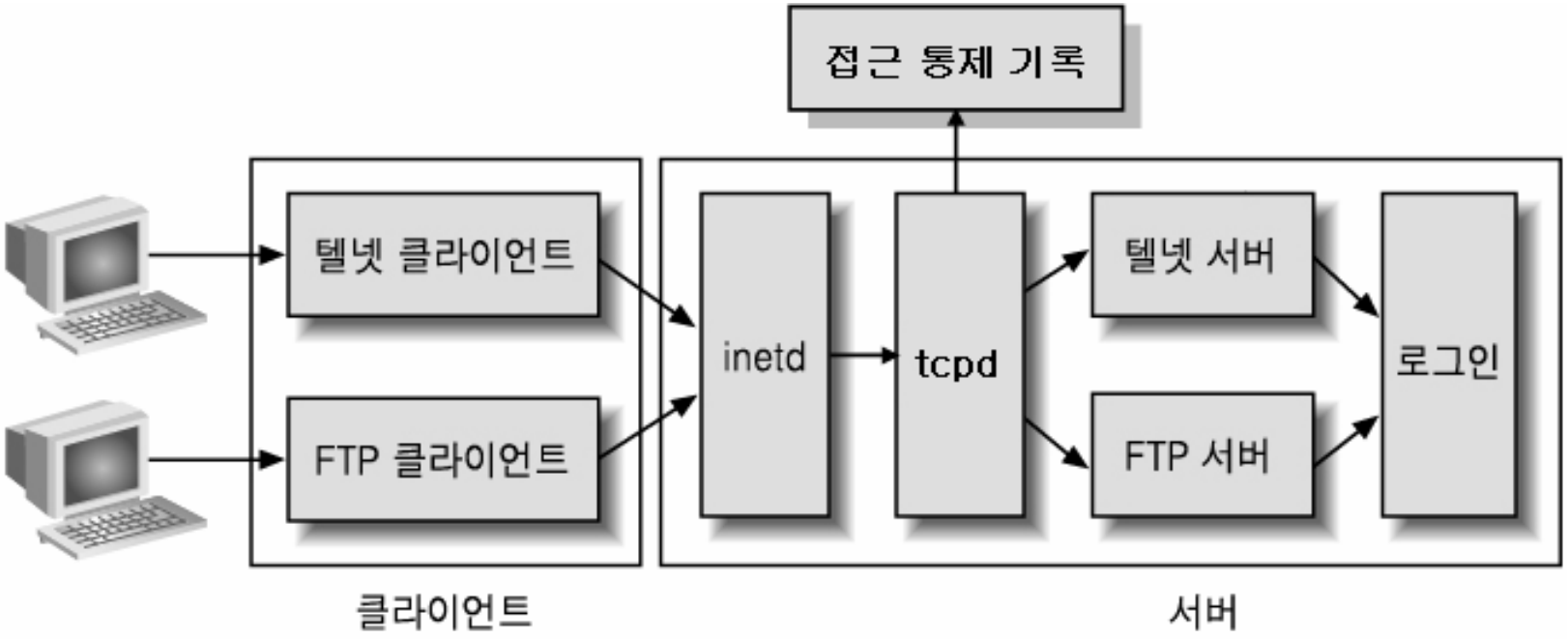
- TFTP (port 69)
  - 간략화된 ftp, ftp와 달리 인증과정이 없음
  - 운영하지 않는 것이 최상(운영 시 –s (secure) 옵션을 주어 실행)

- Finger (port 79)
  - 서버에 현재 로그인 중인 사용자 계정에 대한 정보 확인
  - 현재는 거의 사용하지 않으므로 중지

- SNMP (port 161)
  - 시스템에 대한 많은 정보를 유출할 수 있음
  - SNMP의 중지 : /etc/rc3.d에서 제거

# 유닉스 시스템 보안설정 (5)

## ❒ TCPWrapper를 이용한 접근제어




### ❍ TCPWrapper는 모든 프로토콜에 대한 접근 제어는 불가능

- inetd 데몬이 관리하지 않는 standalone 데몬은 통제 불가
- TCP 프로토콜 외에 일부 프로토콜에 대해서만 통제 가능

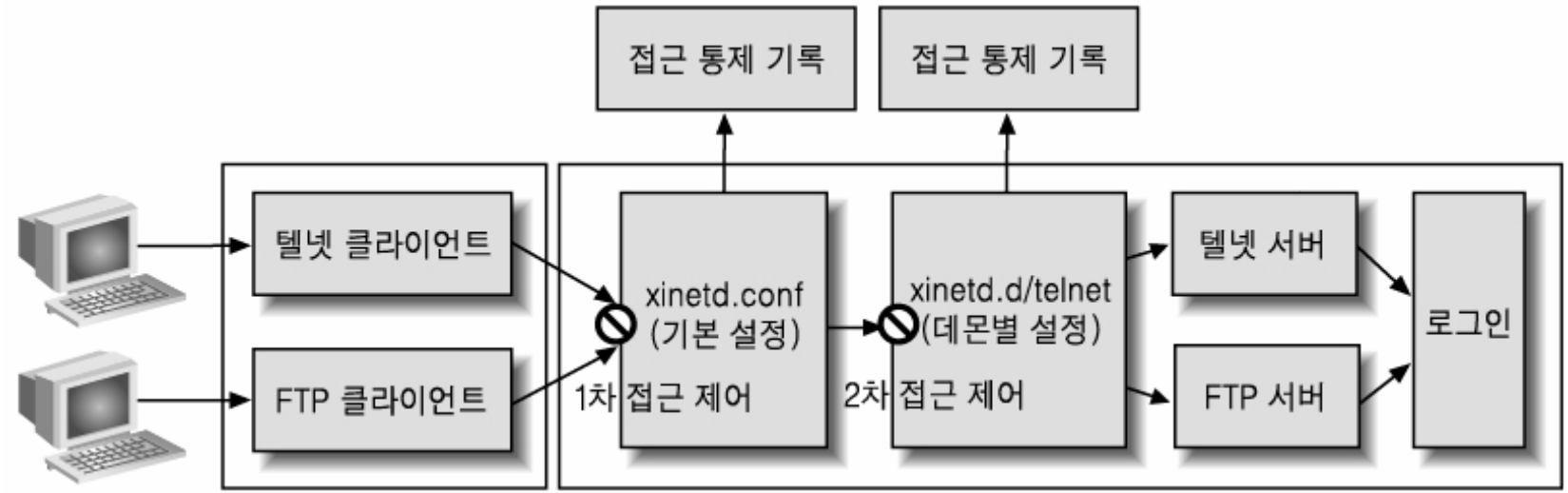
# 유닉스 시스템 보안설정 (6)

## ❒ 접근제어의 설정

- ❍ /etc/hosts.allow(허용할 시스템과 네트워크를 정의)
- ❍ /etc/hosts.deny(차단할 시스템과 네트워크를 정의)
- ❍ 접근제어 규칙 : 데몬 목록 : 클라이언트 목록 [ :셸 명령]
- ❍ ** host.allow 파일에 사용할수 있는 규칙 예 **
  - ALL : 192.168.68.3, 192.168.68.4 EXCEPT 192.168.68.5
    모든 데몬에 대해 접근 허락 접근 금지
  - in.ftpd : 192.168.68.0/255.255.255.0
    192.168.68.*** 네트워크에 해당하는 시스템이 FTP에 접속 허락
  - in.telentd : .co.kr
    도메인 이름이 co.kr로 끝나는 시스템으로부터의 telnet 접근 허락

# 유닉스 시스템 보안설정 (7)

## ❒ Xinetd를 이용한 접근제어




- ❍ inetd와 매우 유사
- ❍ 접근제어와 로그를 양쪽에서 설정 가능
- ❍ 접근제어의 설정의 예(192.168.68.3에 대한 접근만 허용)
  - vi /etc/xinetd.conf

```
only_from               = 192.168.68.3
instances               = 60
log_type                = SYSLOG authpriv
```

# 유닉스 시스템 보안설정 (8)

- 파일 및 디렉토리에 대한 보안 설정
  - /etc 접근 권한 : 시스템에 대한 설정파일
    - 기본: 755(d rwx r-x r-x) → 권장: 751 (d rwx r-x --x)
  - /bin 접근 권한 : ls, cd 등의 명령에 해당하는 실행 파일이 존재
    - 기본: 777(d rwx rwx rwx) → 권장: 771 (d rwx rwx --x)
  - /usr/bin 접근 권한 : bin디렉토리와 유사
    - 기본: 755(d rwx r-x r-x) → 권장: 751 (d rwx r-x --x)
  - /sbin 접근 권한 : bin디렉토리와 유사
    - 기본: 777(d rwx rwx rwx) → 권장: 771 (d rwx rwx --x)
  - /tmp 접근 권한 : 임의의 사용자가 모두 사용
    - 기본: 1777 (d rwx rwx rwt) → 권장: 중요파일에 스티키 비트 설정
  - /var/tmp 접근 권한 : 템프와 동일
  - /var/log 접근 권한 : 여러가지 로그 정보
    - 기본 755(d rwx r-x r-x) → 권장: 751 (d rwx r-x --x)

# 유닉스 시스템 보안설정 (9)

- **/etc/ssh 접근 권한 : 연결시 필요한 공개키 등과 같은 정보**
  - 기본 755(d rwx r-x r-x) → 권장: 750 이상은 부여하지 않음
- **/var/log/messages 접근 권한 : xinetd 데몬의 로그 저장 최소권한 부여**
  - 640 (- rw- r-- ---), 관리자만 접근
- **/etc/crontab 접근 권한 : corn 데몬에 의한 자동 실행 설정**
  - 600 (- rw- --- ---)
- **/etc/syslog.conf 접근 권한 : 시스템 로그에 의한 설정**
  - 640 (- rw- r-- ---) → 권장: 640 이상은 부여하지 않음
- **/var/log/wtmp 접근 권한 : 시스템 접속에 대한 로깅**
  - 권장: 640 이상은 부여하지 않음
- **/var/log/lastlog 접근 권한 : 시스템에 로그인한 사용자에 대한 로그**
  - 권장: 640 이상은 부여하지 않음
- **/etc/ftpusers 접근권한 : ftp로그인 불가능하도록 설정하는 접근제어 파일**
  - 기본: 644 → 권장: 600 이상은 부여하지 않음

# 유닉스 시스템 보안설정 (10)

- /etc/passwd 접근 권한 : 계정에 대한 기본적인 정보 제공 파일
  - 기본 400(d r-- r-- r--) → 권장: 444 (- r-- r-- ---)
- /etc/shadow 접근 권한 : 암호화된 패스워드가 있는 파일
  - 기본 400(d r-- --- ---) → 권장: 600 (- rw- --- ---)
- /etc/hosts.equiv 접근 권한 : 시스템에 대한 신뢰를 설정할 수 있는 파일
  - 권장: 600 이상은 부여하지 않음
- /etc/hosts.allow 접근 권한 : TCPWrapper에서 접근권한 설정 파일
  - 권장: 600 이상은 부여하지 않음
- /etc/hosts.deny 접근 권한 : TCPWrapper에서 접근권한 설정 파일
  - 권장: 600 이상은 부여하지 않음
- /etc/lilo.conf 접근 권한 : 리눅스 부팅 이미지에 대한 설정 변경 파일
  - 권장: 600 이상은 부여하지 않음
- /etc/securetty 접근 권한 : 텔넷 등에 대한 접근 제어를 설정하는 파일
  - 권장: 600 이상은 부여하지 않음

# 유닉스 시스템 보안설정 (11)

- /etc/rc.d/init.d 접근 권한 : 시스템을 부팅할 때의 설정 파일
  - 기본 750((- rwx r-x r-x -) → 권장: 750 (- rwx r-x ---)
- /etc/init.d 접근 권한 : /etc/rc.d/init.d 와 같다
  - 기본 750((- rwx r-x r-x -) → 권장: 750 (- rwx r-x ---)
- /etc/inetd.conf 접근 권한 : 시스템이 제공하는 서비스에 대한 정보
  - 기본 777(d rwx rwx rwx) → 권장: 600 (- rw- --- ---)
- /etc/cron.allow 접근 권한 : cron 데몬에 대한 접근 제어 파일
  - 400 (- r-- --- ---)
- /etc/cron.deny 접근 권한 : /etc/cron.allow 와 같다
  - 400 (- r-- --- ---)
- /etc/hosts 접근 권한 : 시스템에 접근하는 호스트에의 IP와 이름 확인
  - 644 (- rw- r-- r--) 이상은 부여하지 않음

# 로그 분석 과 침입 대응

# 시스템 로그

- Authentication : 계정과 정상적인 패스워드를 입력 과정
- Authorization : 올바른 패스워드를 입력해서 시스템에 로그인이 허락된 사용자라고 판명되어 로그인 되는 과정
- Accounting : 시스템에 로그인한 후 이에 대한 기록을 남기는 과정
- Audit Trail (감사추적) : Accounting하여 남긴 로그 정보를 통한 추적
- 윈도우와 유닉스 로그 저장의 차이
  - 윈도우는 중앙 집중화된 로그(이벤트) : 관리 편리, 삭제 당할 위험 큼
  - 유닉스는 로그가 분산되어 있음 : 분산되어 있어 모든 로그 삭제가 힘듬

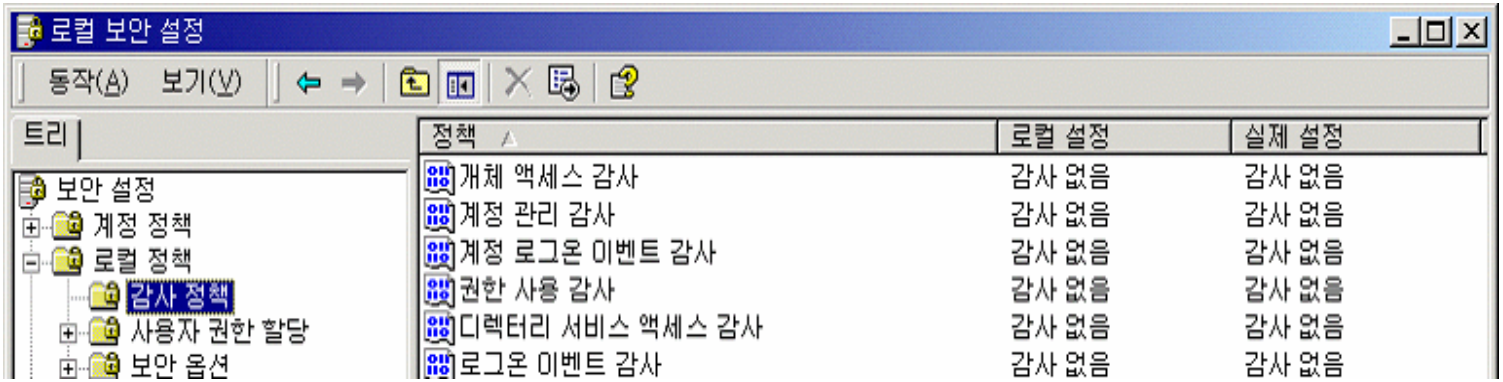
# 윈도우 로그 분석과 설정

## ❒ 세션 관리

- ❍ 세션 정보 : net session
- ❍ 세션 끊기 : net session /delete
- ❍ 세션을 알아보는 다른 툴 : nbtstat –c, psloggedon
- ❍ 명령 프롬프트 히스토리 : doskey /history

## ❒ 로그 정책 설정

- ❍ 윈도우 감사 정책에 대한 정책 설정




- ❍ 원격 시스템 감사 정책 확인 툴 : Auditpol

# 윈도우의 감사 로그(1)

| 이벤트 ID | | 내 용 |
|---|---|---|
| 개체 액세스 감사 | 560 | 개체에 접근 허가 |
| | 562 | 개체에 대한 핸들 닫힘 |
| | 563 | 삭제할 목적으로 개체에 접근 |
| | 564 | 보호된 개체의 삭제 |
| 계정 관리 감사 | 624 | 사용자 계정 만듦 |
| | 625 | 사용자 계정 유형 바꿈 |
| | 626 | 사용할 수 있는 사용자 계정 |
| | 627 | 암호 변경 시도 |
| | 628 | 사용자 계정 암호 설정 |
| | 629 | 사용하지 않는 사용자 계정 |
| | 630 | 삭제된 사용자 계정 |
| | 636 | 보안 사용 로컬 그룹 구성원 추가됨 |
| | 637 | 보안 사용 로컬 그룹 구성원 제거됨 |
| | 642 | 변경된 사용자 계정 |

| 이벤트 ID | | 내 용 |
|---|---|---|
| 계정 관리 감사 | 643 | 변경된 도메인 정책 |
| | 644 | 사용자 계정 잠김 |
| 권한 사용 감사 | 576 | 권한의 할당 |
| | 577 | 권한이 있는 서비스 호출 |
| | 578 | 권한이 있는 개체 작동 |
| 계정 로그온 감사 | 680 | 로그온 성공 정보 |
| | 681 | 로그온 실패 정보 |
| 시스템 이벤트 감사 | 512 | 윈도우 시동 |
| | 513 | 윈도우 종료 |
| | 514 | LSA(Local Security Authority) 인증 패키지 로드 |
| | 515 | 신뢰할 수 있는 로그인 포로세스가 LSA로 등록 |
| | 516 | 저장 공간의 부족으로 인해 일부 보안 이벤트 메시지 소실 |
| | 517 | 보안 로그 삭제 |

# 윈도우의 감사 로그(2)

| 이벤트 ID | | 내 용 | 이벤트 ID | | 내 용 |
|---|---|---|---|---|---|
| 로그온 이벤트 감 사 | 528 | 성공적인 로그인 | 로그온 이벤트 감 사 | 540 | 로그인 성공 |
| | 529 | 알 수 없는 계정이나 잘못된 암호를 이용한 로그인 시도 | | 682 | 연결이 끊기 터미널 서비스 세션에 사용자 재연결 |
| | 530 | 로그인 시 허용 시간 이내에 로그인 실패 | | 683 | 사용자가 로그오프하지 않고 터미널 서비스 세션 연결 끊음 |
| | 531 | 사용이 금지된 계정을 이용한 로그인 시도 | 정책 변경 감사 | 608 | 사용자 권한 할당 |
| | 532 | 사용 기간이 만료된 계정을 이용한 로그인 시도 | | 609 | 사용자 권한 제거 |
| | 533 | 로그인이 허용되지 않는 계정을 이용한 로그인 시도 | | 610 | 다른 도메인과의 신뢰 관계 형성 |
| | 534 | 허용되지 않은 로그인 유형을 통한 로그인 시도 | | 611 | 다른 도메인과의 신뢰 관계 제거 |
| | 535 | 암호 사용기간의 만료 | | 612 | 감사 정책 변경 |
| | 536 | Net Logon 서비스 비활성화 상태 | 프로 세스 추적 감사 | 592 | 새 프로세스 생성 |
| | 537 | 위의 사항에 해당되지 않으나 로그인 실패인 경우 | | 593 | 프로세스 종료 |
| | 538 | 로그오프 | | 594 | 개체에 대한 힌트의 중복 |
| | 539 | 로그인하려는 계정이 잠겨 있음. 패스워드 크래킹 공격시 가능 | | 595 | 개체에 대한 간접적인 접근 |

# 리눅스/유닉스 로그 분석

- 유닉스 로그는 중앙 집중식 로그 관리 syslog + 각 데몬별 로그
- 로그를 모두 파악, 관리하기 어려움
- 해커를 추적하기에는 윈도우 시스템보다 쉬움
- 주요 시스템의 로그 디렉토리 위치

| 경 로 | 적용 시스템 |
|---|---|
| /usr/adm | **초기 유닉스**, BSD **계열** : HP-UX 9.X, SunOS 4.x |
| /var/adm | **최근 유닉스**, SVR **계열** : SUN Solaris, HP-UX 10.x **이후**, IBM AIX |
| /var/log | **일부** BSD **계열** : BSD, FreeBSD, SUN Solaris, Linux |
| /var/run | **일부 리눅스** |

# 리눅스/유닉스 로그(1)

## ❒ UTMP

- ❍ 가장 기본적인 로깅을 제공하는 데몬(/etc/lib/utmpd)
- ❍ 현재 시스템에 로그인한 사용자의 상태를 출력
- ❍ 로그 출력명령어 : w, who, users, whodo, finger

## ❒ WTMP

- ❍ utmp 데몬과 비슷한 역할
- ❍ 사용자들의 로그인, 로그아웃, 시스템의 재부팅에 대한 정보를 저장
- ❍ 선별적인 로그 출력 : last 명령

## ❒ SU(Switch User) 로그

- ❍ su 명령을 이용한 권한 변경에 대한 로그(su 명령 로그)

# 리눅스/유닉스 로그(2)

- PACCT(Process Account)
  - 시스템에 로그인한 모든 사용자가 수행한 프로그램에 대한 정보를 저장

- .sh_history or .bash_history
  - 각 계정마다 실행했던 명령에 대한 기록을 [셸의 종류]_history 형식으로
    각 계정의 홈 디렉토리에 저장

- lastlog(/var/adm/lastlog)
  - AIX의 경우에는 /etc/security 디렉토리에 저장
  - 선별적인 로그 출력 : last 명령

# 리눅스/유닉스 로그(2)

- syslog
  - 시스템의 로그에 대한 대부분의 정보를 수집하여 로깅
  - /etc/syslog.conf 파일에서 지정한 사항에 대해 로깅을 실시
- loginlog
  - 실패한 로그인 시도에 대한 로깅
- FTP 파일 전송 로그(xferlog)
  - 파일을 전송한 날짜와 시간, 접근시스템의 IP, 전송한 파일에 대한 로깅
  - /etc/inetd.conf에서 in.ftpd데몬 –l 옵션 설정
- HTTPD log(Access_log, Error_log)
  - /usr/local/apache/logs 아래 Error_log, Access_log, 등과 같은 로그 파일로 남음

# 리눅스/유닉스 로그 파일의 위치

| 로그파일 | 리눅스 (레드햇) | 솔라리스 | HP-UX(10.x 이상) | IBM-AIX |
|---|---|---|---|---|
| utmp, wtmp | /var/run(utmp)<br>/var/log(wtmp) | /var/adm | /var/adm | /var/adm |
| utmpx, wtmpx | **존재하지 않음** | /var/adm | **존재하지 않음** | **존재하지 않음** |
| Btmp | /var/log | **존재하지 않음** | /var/adm | **존재하지 않음** |
| Syslog | **존재하지 않음** | /var/log | /var/adm/syslog/syslog.log | /var/adm |
| Sercure | /var/log | **존재하지 않음** | **존재하지 않음** | **존재하지 않음** |
| sulog | **존재하지 않음** | /var/adm | /var/adm | /var/adm |
| pacct | /var/log | /var/adm | /var/adm | /var/adm |
| authlog | **존재하지 않음** | /var/log | **존재하지 않음** | **존재하지 않음** |
| messages | /var/log | /var/adm | /var/adm | /var/adm |
| loginlog | **존재하지 않음** | /var/adm | **존재하지 않음** | **존재하지 않음** |
| lastlog | /var/log | /var/adm | /var/adm | /etc/security |
| access_log | /var/log/httpd | /var/log/httpd | /usr/local/etc/httpd/logs | /usr/local/etc/httpd/logs |
| error_log | /var/log/httpd | /var/log/httpd | /usr/local/etc/httpd/logs | /usr/local/etc/httpd/logs |
| shutdownlog | **존재하지 않음** | **존재하지 않음** | /etc/shutdownlog | **존재하지 않음** |
| failedlogin | **존재하지 않음** | **존재하지 않음** | **존재하지 않음** | /etc/security |

# 해킹 침입사고 발생시 대응절차 (1/2)

❒ 1.로그 확인

❒ 2. 현재 해커가 침투한 경우의 대응 – ① 메모리 덤프

- ❍ 메모리를 덤프(강력한 법적 효력) → 재기록 불가능 미디어(CD-ROM)

❒ 3. 현재 해커가 침투한 경우의 대응 - ② 해커의 접속 차단

- ❍ 디스크의 이미지 저장(재기록 불가능 미디어)
- ❍ 법적 증거물 제출용과 침입 추적(Forensic)용으로 2개의 이미지 저장

❒ 4. 현재 해커가 침투한 경우의 대응 - ③ 운영자에게 통보

- ❍ 상위 관리자에게 전화와 같은 즉각적인 통신 수단을 이용하여 통보

# 해킹 침입사고 발생시 대응절차 (2/2)

## ❒ 5. 현재 해커가 침투한 경우의 대응 - ④ 침입자 추적

- ❍ 침입 추적(Forensic)용 이미지를 이용해 똑같은 다른 시스템을 CD-ROM으로 만든 후 이를 이용해서 침입자 추적
- ❍ 침입 추적 과정에서 공격자가 삭제한 파일을 복구
- ❍ 이를 증거로서 제출할 수 있도록 하는 과정이 필요
- ❍ 시스템 분석과 동시에 시스템을 복구
- ❍ 추적 시 확인사항
  - 시스템에 설치된 응용 프로그램의 변조 여부
  - 시스템 설정 파일의 변조 여부
  - 데이터의 삭제 및 변조 여부
  - 운영중인 다른 시스템에 대한 침투 여부 확인

## ❒ 6. 침입 후 대응 및 시스템 복구

- ❍ 백업장치를 이용해 해킹으로 손상된 부분을 복구
- ❍ 법적 절차에 따라 해커를 처벌

# 윈도우 시스템에서의 흔적 제거

## ❒ 1. 세션의 생성 및 로깅 정책의 파악

- ❍ net use ₩₩192.168.68.4 "qwer1234" /u:administrator
- ❍ auditpol ₩₩192.168.68.4

## ❒ 2. 원격지 시스템의 로깅 중지

- ❍ 보안감사 정책 중지 : auditpol ₩₩192.168.68.4 /disable

## ❒ 3. 원격지 시스템의 로그 삭제

- ❍ clearlogs ₩₩192.168.68.4 -sec
- ❍ 로컬 시스템의 로그 삭제 : logkiller

# 윈도우 시스템에서의 침입 추적

- 현재 시스템에 로그인한 사용자 확인 : ntlast -i
- 로그인 시도시 실패 목록 : ntlast –f4
- 특정 계정에 대한 로그인 로그 확인
  : ntlast –i –v –u wishfree
- 프로세스 덤프 획득 : userdump [프로세스 ID] (저장이름)
- Ghost를 이용한 디스크 이미지 복사
- Final Data를 이용한 삭제된 파일 복구

# 유닉스 시스템에서의 메모리 덤프

- 솔라리스에 make 유틸리티 설치
  - pkgadd –d make-3.80-sol9-intel-local
- 2. PATH 지정
  - PATH=$PATH:/usr/local/bin:/opt/sfw/bin
- 3. memdump 설치
  - gzip –d memdump-1.0.tar.gz
  - tar xvf memdump-1.0.tar
  - cd memdump-1.0
  - make
- 4. 메모리 덤프
  - ./memdump > memorydump

# 참고문헌


- 양대일, "정보보안 개론과 실습 시스템 해킹과 보안", 한빛 미디어, 2006.7
- 양대일, "정보보안 개론과 실습 네트워크 해킹과 보안", 한빛 미디어, 2005.8
- 최용락 외 3명, "컴퓨터 통신보안", 그린, 2006.1
- 홍승필, "유비쿼터스 컴퓨팅 보안", 한티미디어, 2006,9
- 박창섭, "암호이론과 보안", 대영사, 2006,2

# 질의응답

감사합니다