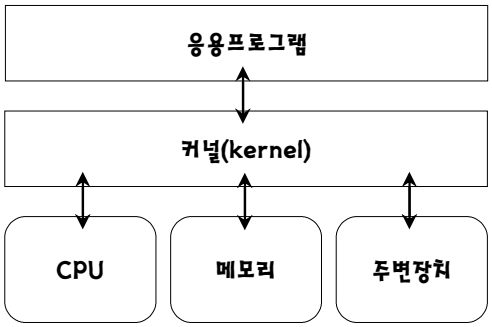
제11장 시스템 보안

1. 유닉스(unix) 개요

① unix는 **다중 사용자** 운영체제이다.
② unix는 시분할 시스템(time sharing system) 방식이다.
③ unix는 멀티태스킹 시스템이다. 즉, 여러 개의 프로세스가 동시에 수행될 수 있다.
④ 단말기를 통하여 컴퓨터와 사용자가 직접 대화하는 방식이다.**(인터랙티브 방식)**
⑤ unix는 계층적 파일 시스템을 제공한다. 관련된 파일을 그룹화시킬 수 있다.
⑥ unix는 **기계독립언어(C 언어)**로 작성되어 다른 컴퓨터에 쉽게 이식할 수 있다.
• 지금, unix는 약 90% 이상이 C로 기술되어 있다.
• unix를 처음 개발할 때는 PDP-7 어셈블리 언어를 사용하였다.

1. 커널(kernel)

커널은 운영체제의 핵심으로 **보안, 자원관리, 추상화**를 지원한다.



커널은 응용프로그램과 하드웨어를 연결한다.

① 커널은 컴퓨터 **자원**(CPU, 메모리, 디스크 등)을 **효율적으로 관리**하기 위한 것이다.
② 커널은 컴퓨터 하드웨어와 프로세스의 보안을 책임지고 관리한다.
→ 현재, 대부분의 커널은 다양한 통신 프로토콜을 지원한다.
③ 커널은 컴퓨터 하드웨어 장치에 대한 가장 기초 수준의 제어권을 제공한다.
④ 응용프로그램은 커널의 장치 드라이버를 사용하여 하드웨어를 간접 제어한다.
• 응용프로그램은 시스템 호출 방법으로 커널의 장치 드라이버에 접근하여 자료를 처리한다.
• 장치 드라이버는 특정 하드웨어를 제어하기 위한 커널의 일부분으로 동작하는 프로그램이다.
⑤ 커널은 부팅되면서 부트로더에 의해 메모리에 적재되어 컴퓨터 종료 시까지 **상주**한다.

2. 셸(shell)

① 유닉스에서 shell의 기능은 다음과 같다.

- 셸은 명령어 **해석기**(command interpreter)이다.(입력한 명령어를 읽고 해석 및 실행)
- 셸은 사용자와 운영체제(커널) 사이에서 인터페이스 기능을 담당한다.
- 운영체제의 커널을 둘러싼다고 shell이라 한다.
- 셸은 그 자체로 일종의 프로그래밍 언어이다.

② 유닉스에서 많이 사용되는 셸(shell)은 다음과 같다.

	기본 프롬프트 모양	
	관리자(슈퍼유저, root)	일반사용자
본 셸(Bourne shell, sh)	#	$
콘 셸(Korn shell, ksh)	#	$
C 셸(C shell, csh)	#	%

- 셸 종류는 위에 소개한 것 이외에 "TC shell, Z shell" 등 다수가 있다.
- unix는 한 시스템 내에서 사용자들은 각각 다른 셸을 사용할 수 있다.
- 이는 각 사용자에게 맞는 인터페이스를 제공하기 위한 것이다.

3. 슈퍼유저(superuser)

- 시스템 관리자를 슈퍼유저라고 한다.
- 슈퍼유저의 로그인명은 통상적으로 **root**를 사용한다.
- 슈퍼유저에게는 특수 프롬프트인 #을 제공한다.
- 슈퍼유저의 홈 디렉터리는 루트디렉터리(/)이다.
- 슈퍼유저가 사용하는 명령어 대부분은 디렉터리 /etc에 있다.
- 슈퍼유저는 파일 등에 설정된 접근권한에 구속되지 않는다.
- 슈퍼유저는 모든 파일에 대하여 읽기, 쓰기, 실행, 생성, 삭제할 수 있다.
- 일반사용자로 로그인 후에 **su** 명령어를 사용하여 슈퍼유저로 변경 가능하다.
- 명령어 **su**는 로그아웃하지 않고 다른 사용자 권한으로 셸을 실행하는데 사용한다.

〈실습〉

```
$ su root          ← 일반사용자 계정, su는 switch user 또는 substitute user 약어이다.
password: 1234     ← 슈퍼유저(root)의 비밀번호를 입력하면 슈퍼유저가 된다.
#                  ← 프롬프트 모양이 슈퍼유저가 사용하는 #으로 변경되었다.
```

4. 파일 속성

UNIX에서 디렉터리와 파일에 대한 정보 및 사용 권한은 다음과 같다.

```
- r w x r w x r w x
|   |     |     └──→ 기타 다른 사용자들의 권한
|   |     └────────→ 파일 소유자가 속한 그룹의 권한
|   └──────────────→ 파일 소유자의 권한
└──────────────────→ 코드 영역
```

① 코드 영역

- `-` : 일반 파일을 나타낸다.
- `d` : 디렉터리이면 d로 표시된다.
- `l` : 링크 파일을 의미한다.
- `b` : 블록 특수 파일(H/W 제어)
- `c` : 문자 특수 파일(H/W 제어)

② 사용자 접근권한

- `r` : 읽기 가능(read)
- `w` : 쓰기 가능(write)
- `x` : 실행 가능(execute)

• 유닉스에서 각 입출력장치(키보드, 프린터 등)는 하나의 특수 파일과 연관되어 있다.
• 즉, 특수 파일은 데이터는 없지만 물리적 장치를 파일명에 사상시키는 역할을 한다.

[예제] UNIX에서 어떤 파일 속성이 다음과 같은 경우

```
-rwxrw-r--
```

① 이 파일은 일반 파일이다.(디렉터리이면 첫 부분에 - 대신 d가 표시됨)
② 이 파일의 소유자는 읽기, 쓰기, 실행이 가능하다.
③ 이 파일의 소유자 그룹 구성원들은 읽기, 쓰기가 가능하다.(실행은 불가)
④ 일반 사용자들은 이 파일에 대해 읽기만 가능하다.

♤ ls 명령의 옵션 -l을 사용한 경우

`$ ls -l` → directory의 내용을 long 문법으로 출력한다.

```
-rwxrwxrwx   12    jyhong   hansung   10000000   Oct 25 17:30   /etc/han.c
    ↓        ↓       ↓        ↓          ↓           ↓             ↓
 접근권한   링크수   소유자    그룹     파일크기   작성(수정)일    파일명
```

5. inode(Index node)

- 먼저, 유닉스 계열에서는 거의 모든 것을 파일로 취급한다.
- 일반 파일뿐만 아니라 키보드, 마우스, 모니터, NIC 같은 기기들도 파일로 취급한다.
- i-node는 파일에 대한 정보를 디스크 상에 저장하고 있는 레코드이다.
- 유닉스의 모든 파일은 i-node로 관리된다. 하나의 파일은 하나의 i-node를 가진다.
- 유닉스에서 **파일** 또는 **디렉터리**가 생성되면 **i-node**가 생성된다.
- i-node는 내부적으로 구조체로 표현되며, 실제로 매우 복잡하다.
- i-node는 파일과 디렉터리에 대한 모든 정보를 가진다.

◈ inode에 포함된 정보

파일 모드	→ 파일의 종류, 접근, 수행 권한에 대한 정보를 저장(16bit 플래그)
연결 수	→ 파일에 대한 다양한 연결 수
소유자 식별자	→ 파일을 소유한 개인의 ID
그룹 식별자	→ 파일을 소유한 그룹의 ID
파일 크기	→ 파일의 바이트 수
파일 주소	→ 파일의 주소 정보(파일 위치)
파일 생성 시기	→ 파일이 처음 만들어진 시간
최종 접근 시간	→ 마지막으로 파일에 접근한 시간
최종 변경 시간	→ 마지막으로 파일을 변경한 시간
inode 변경 시간	→ 마지막으로 inode를 변경한 시간

- inode는 실제 파일명과 파일의 내용을 제외한 파일에 대한 모든 정보를 담고 있다.
- inode는 파일에 대한 정보를 저장하고 있지만, 파일명은 직접 갖지 않는다.

◈ 각 파일과 inode 연결

- 먼저, **파일명**은 **디렉터리 엔트리**(directory entry) 테이블에 저장되어 있다.
- 각 파일과 inode 연결은 디렉터리 엔트리(directory entry)라는 테이블 구조를 사용한다.
- 디렉터리 엔트리(directory entry) 테이블에는 파일명과 inode 번호 등이 저장되어 있다.
- 새로운 파일이 만들어지면, 디렉터리 엔트리에 파일명과 inode 번호가 등록된다.
- 디렉터리 엔트리는 단순하지만 연결된 inode는 복잡하다.

6. 유닉스 시스템의 디렉터리

◈ /
• 루트(root) 디렉터리

◈ /etc
• 시스템의 환경 설정 및 주요 설정 파일을 담고 있다.(시스템 관리를 위한 것)
• passwd, hosts 등 저장

◈ /bin
• 기본적으로 실행 가능한 파일을 담고 있다.
• 원시코드가 컴파일 되면 이진(binary) 형태로 번역된다.
• 실행 파일 종류에 따라 /bin, /sbin, user/bin 등으로 구분

◈ /home
• 각 사용자의 작업 디렉터리를 담고 있다.
• 각 사용자의 파일을 하위 디렉터리에 저장

◈ /dev
• device
• 하드웨어 디바이스 관련 파일을 담고 있다.

◈ /lib
• library
• /bin/과 /sbin/에 있는 바이너리에 필요한 라이브러리를 담고 있다.

◈ /sbin
• system-administrator's binary(시스템 관리자가 사용하는 실행 파일)
• 필수 시스템 바이너리(init, ip, mount 등)

◈ /var
• variable
• 수시로 변하기 쉬운 파일에 대한 저장소(로그, 임시 메일 등)

◈ /usr
• 모든 사용자가 사용할 수 있는 보조 계층구조로 중요도가 낮은 파일들을 저장한다.

기출문제 분석

1. 유닉스 운영체제에 대한 설명으로 옳지 않은 것은? [2018년 컴일 국가 9급]

① 계층적 파일시스템과 다중 사용자를 지원하는 운영체제이다.
② BSD 유닉스의 모든 코드는 어셈블리 언어로 작성되었다.
③ CPU 이용률을 높일 수 있는 다중 프로그래밍 기법을 사용한다.
④ 사용자 프로그램은 시스템 호출을 통해 커널 기능을 사용할 수 있다.

♤ **유닉스 운영체제**

- BSD 유닉스의 모든 코드는 어셈블리 언어로 작성되었다.**(×)**
  → 모든 유닉스 운영체제는 대부분 C로 개발되어 있다.(초기에는 어셈블리어로 개발)
  → 지금, unix는 약 90% 이상이 C로 기술되어 있다.
- BSD는 Berkeley Software Distribution(버클리 소프트웨어 배포판) 약어이다.

정답 : ②

2. UNIX 파일 시스템에서 i-node에 존재하는 정보로 옳지 않은 것은? [2017년 컴일 국회 9급]

① 파일 이름
② 파일 저장 위치
③ 파일 생성 일시
④ 파일 소유자
⑤ 파일 접근권한 비트

♤ **UNIX 파일 시스템에서 i-node**

- inode는 실제 파일명과 파일의 내용을 제외한 파일에 대한 모든 정보를 담고 있다.
- inode는 파일에 대한 정보를 저장하고 있지만, 파일명은 직접 갖지 않는다.

◉ **각 파일과 inode 연결**

- 먼저, **파일명**은 **디렉터리 엔트리**(directory entry) 테이블에 저장되어 있다.
- 각 파일과 inode 연결은 디렉터리 엔트리(directory entry)라는 테이블 구조를 사용한다.
- 디렉터리 엔트리(directory entry) 테이블에는 파일명과 inode 번호 등이 저장되어 있다.
- 새로운 파일이 만들어지면, 디렉터리 엔트리에 파일명과 inode 번호가 등록된다.
- 디렉터리 엔트리는 단순하지만 연결된 inode는 복잡하다.

정답 : ①

3. 유닉스 운영체제의 커널에 속하지 않는 것은? [2015년 컴일 지방 9급]

① 스케줄러 ② 파일 관리자
③ 메모리 관리자 ④ 윈도우 관리자

♤ **유닉스 운영체제의 커널**

• 윈도우 관리자는 윈도우 운영체제에서 사용되는 것이다.

정답 : ④

4. 유닉스 시스템의 디렉터리별 역할에 대한 설명을 바르게 연결한 것은? [2017년 정보보호 9급]

(가) 시스템의 환경 설정 및 주요 설정 파일을 담고 있다.
(나) 기본적으로 실행 가능한 파일을 담고 있다.
(다) 각 사용자의 작업 디렉터리를 담고 있다.

	(가)	(나)	(다)
①	/bin	/home	/etc
②	/home	/etc	/bin
③	/etc	/bin	/home
④	/bin	/etc	/home

♤ **유닉스 시스템의 디렉터리별 역할**

◈ **/etc**

• 시스템의 환경 설정 및 주요 설정 파일을 담고 있다.(시스템 관리를 위한 것)
• passwd, hosts 등 저장

◈ **/bin**

• 기본적으로 실행 가능한 파일을 담고 있다.
• 원시코드가 컴파일 되면 이진(binary) 형태로 번역된다.
• 실행 파일 종류에 따라 /bin, /sbin, user/bin 등으로 구분

◈ **/home**

• 각 사용자의 작업 디렉터리를 담고 있다.
• 각 사용자의 파일을 하위 디렉터리에 저장

정답 : ③

5. 리눅스 커널 보안 설정 방법으로 옳지 않은 것은? [2017년 서울 9급]

① 핑(ping) 요청을 응답하지 않게 설정한다.
② 싱크 어택(SYNC Attack) 공격을 막기 위해 백로그 큐를 줄인다.
③ IP 스푸핑된 패킷을 로그에 기록한다.
④ 연결 종료 시간을 줄인다.

♤ **리눅스 커널 보안 설정**

• 싱크 어택(SYNC Attack) 공격을 막기 위해 백로그 큐를 **줄인다.(×)**
→ 싱크 어택(SYNC Attack) 공격을 막기 위해서는 백로그 큐를 **늘려야** 한다.

◉ **리눅스 커널 보안 설정**

• 특정 계정만 su 명령을 사용할 수 있게 설정
• SSH로 root 계정 직접 로그인 차단
• ping 테스트 응답 차단
• SetUID, SetGID 점검
• tmp 디렉터리 보안
• iptable 설정 등

정답 : ②

6. 리눅스 및 유닉스 계열의 운영체제에서 디렉터리 탐색 공격을 위해 사용되는 etc 디렉터리에 있는 passwd 파일을 다운로드하는 경로명으로 가장 옳은 것은? [2022년 서울 7급]

① .../root/bin/passwd
② .../etc/passwd
③ .../etc/sbin/passwd
④ /home/passwd

♤ **리눅스 및 유닉스 디렉터리 : /etc**

• 시스템의 환경 설정 및 주요 설정 파일을 담고 있다.(시스템 관리를 위한 것)
• passwd, hosts 등 저장 (**/etc/passwd**)
• 유닉스에서 각 사용자들의 개인정보는 파일 "**/etc/passwd**"에 저장되어 있다.

/etc/passwd	root:**x**:0:1:Super-user:/root:/bin/csh ← 슈퍼유저 user001:**x**:120:100::/user/home/user001:/bin/csh ← 일반유저

정답 : ②

7. 유닉스/리눅스의 파일 접근제어에 대한 설명으로 옳지 않은 것은? [2020년 지방 9급]

① 접근권한 유형으로 읽기, 쓰기, 실행이 있다.
② 파일에 대한 접근권한은 소유자, 그룹, 다른 모든 사용자에 대해 각각 지정할 수 있다.
③ 파일 접근권한 변경은 파일에 대한 쓰기 권한이 있으면 가능하다.
④ SetUID가 설정된 파일은 실행시간 동안 그 파일의 소유자의 권한으로 실행된다.

♤ **유닉스/리눅스의 파일 접근제어**

• 파일 접근권한 변경은 파일에 대한 **쓰기 권한이 있으면 가능**하다.**(×)**
→ 변경은 파일 **소유자나 슈퍼유저**만 가능하다.

정답 : ③