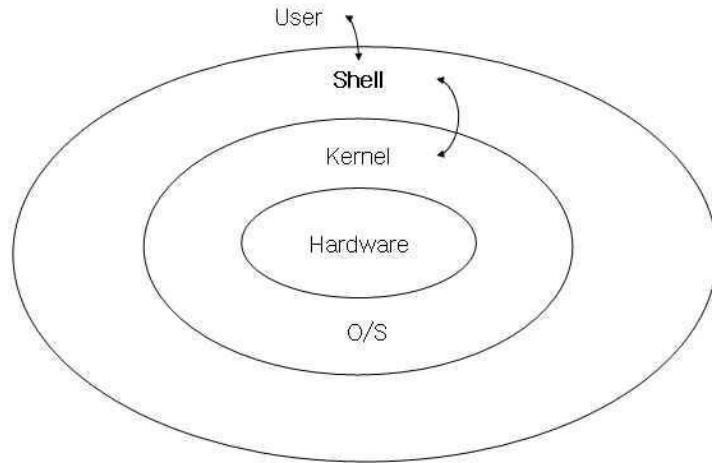


셸(Shell)의 개념 및 역할 및 종류

- 셸(Shell)은 리눅스의 핵심인 커널과 사용자를 연결해주는 인터페이스 역할을 합니다.



- 우리가 사용하는 프로그램은 일반적으로 바이너리 형태의 명령어 집합체이며, 컴퓨터는 기본적으로 바이너리 형태의 정보들을 읽어 들입니다.
- 따라서 관리자나 사용자가 컴퓨터에 일정한 명령을 내리기 위해서는 매개체로서의 특정한 인터페이스가 필요했고 이런 역할을 하도록 하는 것이 셸(Shell)입니다.

셸은 크게 세 가지의 역할을 합니다.

1. 대화식(Interactice) 사용

사용자의 요청을 기다려서 요청 즉시 결과 값을 출력해주는 대화형 구조를 가집니다.

2. 프로그래밍

복합적인 작업을 수행할 수 있도록 일련의 명령어들을 묶어서 사용할 수 있습니다. (스크립트)

3. 리눅스 세션(session)의 설정(Customization)

리눅스의 세션에 대한 변수들을 정의하여 사용자가 리눅스 환경을 자신이 원하는 상태로 설정할 수 있게 합니다.

- /etc/shells 파일을 열어 보면 현재 사용할 수 있는 셸들의 경로가 설정되어 있습니다.

```
# cat /etc/shells
```

```
/bin/sh
```

```
/bin/bash
```

```
/sbin/nologin
```

```
/bin/tchs
```

```
/bin/csh
```

```
/bin/ksh
```

bourne 셸 계열의 셸

1. sh (bourne shell)

- 가장 기본적인 셸로 유닉스의 초기부터 사용되어 온 셸
- 스크립트를 지원

2. ksh (korn shell)

- 본 셸을 확장한 셸
- 본 셸의 명령어를 모두 인식
- 명령어 히스토리(history) 기능과 알리어스(alias), 작업 제어 등의 기능이 추가됨
- 일반적으로 유닉스에서 가장 많이 사용되는 셸
- 명령행 편집기능을 제공

3. bash (Bourne Again Shell)

- 리눅스에서 가장 많이 사용하는 셸
- C 셸과 콘 셸의 장점을 결합하여 작성됨
- Bourne 셸 문법의 명령어 셋을 제공하여 Bourne Shell 과 호환되는 셸로 GNU 프로젝트에 의해 만들어지고 배포됨
- 명령행 편집기능을 제공

C 셸 계열의 셸

1. csh (C Shell)

- 명령행 편집기능을 제공하지 않음.
- C 언어 위주의 셸로 처음 작성되었을 때에는 본 셸이 가지고 있지 못한 기능들(작업제어, 명령어 히스토리 등)을 가지고 있었기 때문에 많이 사용

2. tcsh (TC Shell)

- csh 의 기능을 강화한 셸
- 확장 C Shell.
- 명령행 편집 기능을 제공