

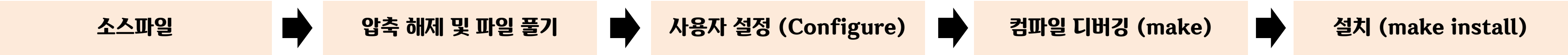
copyright 김성우

# LINUX

## 패키지와 커널

리눅스의 프로그램 설치

- 리눅스에 사용되는 프로그램은 공개 소프트웨어라는 특징과 유닉스의 영향으로 C언어로 된 소스(Source) 파일을 압축하여 배포됨
- 대부분 C언어로 작성된 프로그램의 소스코드를 계층적인 디렉터리 구조이고, 컴파일을 위한 Makefile과 같은 관련 문서를 포함함



make와 cmake(cross platform make system)

- 패키지의 소스 파일을 받아서 직접 설정 및 컴파일 하는 방식
- Makefile이라는 파일을 읽음으로서 소스 코드에서 실행 가능한 프로그램과 라이브러리를 자동으로 빌드하는 자동화 도구
- apt로 설치될 때 적용되지 않는 특별한 옵션, 사용자 설정, 소스코드의 수정 등이 필요할 경우 사용
- 일반 사용자보다 프로그래머 혹은 고급 시스템 사용자에게 적합
- cmake: configure와 make작업을 통합하는 프로그램으로, 운영체제에 맞는 make파일의 생성을 목적으로 함.

순서	명령어	설명
1	./configure --help	configure로 지원되는 옵션 정보 출력
2	./configure --prefix=/usr/local/apache	설치되는 디렉터리를 설정
3	make	configure에 의해 변경된 내용을 반영하고 컴파일 작업을 실행 후, 최종적으로 실행 파일 생성
4	make clean	configure 작업으로 생성된 다양한 파일 제거
5	make install	컴파일된 실행 파일을 실행하여 프로그램 설치

패키지 설치

- 레드햇계열 : rpm, yum
- 데비안계열 : dpkg apt-get
- 수세(SUSE)계열 : YaST zypper

dpkg

- dpkg(Debian Package): Windows의 setup.exe와 같이 바로 실행할 수 있는 설치 파일.
- \*.deb 확장자를가지며 패키지(Package)라 부름
- 단점: 의존성  
ex) 우분투의 Firefox는 X 윈도우에 의존적이므로 X 윈도우가 설치 되어있지 않으면 설치가 불가능.

apt(apt-get)

- dpkg가 가지는 의존성 문제를 해결해주며 특정 패키지를 설치하고자 할 때, 의존성이 있는 다른 패키지를 자동으로 먼저 설치해줌.
- 우분투가 제공하는 deb 파일 저장소에서 deb파일 뿐만 아니라, 의존성이 있는 다른 deb파일까지 인터넷을 통해 알아서 다운로드 후 자동 설치해줌
- 단점: 인터넷에 연결되어있어야함.

우분투 전체 패키지 파일:

<http://kr.archive.ubuntu.com/ubuntu/pool/>

deb 파일 저장소 URL정보:

</etc/apt/sources.list>

## deb파일과 dpkg 명령어

- **deb 파일이름 형식**

패키지명\_버전-개정번호\_아키텍처.deb

- **설치/최신버전 업그레이드**

#dpkg -i [패키지파일명.deb] (--install)

- **삭제**

#dpkg -r [패키지명] (--remove)

#dpkg -P [패키지명] (--purge) (기존에 설치된 패키지 및 설정파일 전부 삭제)

- **패키지 조회**

#dpkg -S [패키지명] (해당 패키지명으로 시스템에 설치된 파일 이름을 찾음)

#dpkg -l [패키지명] (해당 패키지명과 일치하는 패키지 리스트 정보)

#dpkg -L [패키지명] (해당 패키지명으로 설치된 파일 목록 정보)

- **deb 파일 내용 조회**

#dpkg --info [패키지명.deb]

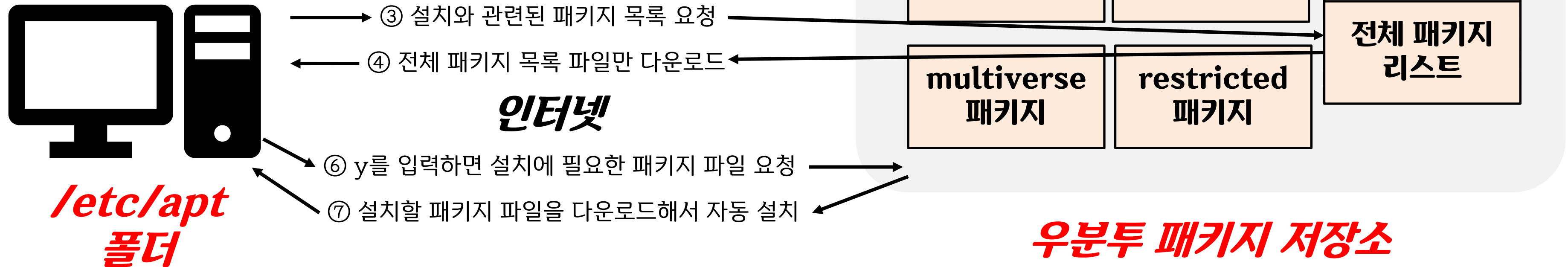
### apt (Advanced Packaging Tool)

- **설치**  
#apt -y install [패키지명]  
(-y는 yes로, 사용자에게 설치 시 yes/no를 묻는 과정을 생략함-모두 yes)
- **패키지 목록 업데이트**  
#apt update (/etc/apt/sources.list 파일의 내용이 수정되었을 경우)  
#apt upgrade (받은 패키지의 업그레이드를 해야할 때. apt update 우선 적용 후 사용)
- **삭제**  
#apt remove [패키지명]  
#apt purge [패키지명] (환경설정까지 전부 삭제)
- **내려받은 파일 제거**  
#apt clean (패키지를 설치할 때 내려받기 한 파일 제거)

### apt 명령어

- #apt-cache: 패키지를 설치하기 전에 패키지에 대한 정보나 의존성 문제를 미리 확인해볼 수 있는 명령어  
패키지 정보 보기: apt-cache show 패키지명  
패키지 의존성 확인: apt-cache depends 패키지명  
패키지 역의존성 확인: apt-cache rdepends 패키지명
- main: 우분투에서 공식적으로 지원하는 무료 소프트웨어
- universe: 우분투에서 공식적으로 지원하지 않는 무료 소프트웨어
- restricted: 우분투에서 공식적으로 지원하는 유료 소프트웨어
- multiverse: 우분투에서 공식적으로 지원하지 않는 유료 소프트웨어

- ① apt install 입력
- ② /etc/apt/sources.list 파일에서 URL 주소 확인
- ⑤ 설치할 패키지와 관련된 패키지의 이름을 화면에 출력



Kernel Compile

- 커널에 포함될 것과 모듈로 나누는 작업으로, 최근 하드웨어와의 호환 혹은 효율적 사용을 위해 최신버전으로 업데이트 하는 것
- 모듈: 별도로 보관했다가 필요할 때마다 호출하여 사용되는 코드들
- /boot/vmlinuz-커널버전파일명: 부팅 시 메모리에 로딩되며 부팅 시 꼭 필요한 부분만 포함
- /lib/modules/커널버전/: 부팅 시에는 꼭 필요하지 않거나, 언제 사용할지 모르는 유동적인 정보들
- 주로 하드웨어 지원 문제를 해결하기 위함
- make, gcc, g++ 등 의존 패키지 필수

최신 커널 파일:  
<https://www.kernel.org>

시스템 커널 버전 확인:  
`cat /proc/version`  
버전명: `#uname -r`  
모든정보: `#uname -a`  
커널명: `#uname -s`  
CPU종류: `#uname -p`

배포형태	설명
Prepatch	개발 단계인 커널 버전으로, 소스명에 rc가 붙음
MainLine	리눅스 토발즈에 의해 관리되는 주 버전으로 2~3개월마다 새로운 커널 공개
Stable	메인라인 버전에 버그가 패치된 안정성이 입증된 커널
Longterm	EOL(End of Life, 서비스 종료일)까지 커널 메인테이너에 의해 관리되는 커널로, 그 기간 동안에 발생하는 버그들은 패치가 이뤄짐



Kernel Compile

- make: 커널 이미지와 모듈들이 컴파일되며, 환경에 따라 몇십분~몇시간 소요( -jcpu개수-1 로 설정하여 스레딩 작업 가능)
- make modules\_install: 새로 컴파일된 모듈들이 /lib/modules/커널버전/kernel 디렉터리로 설치됨  
새 커널로 부팅 후 의존성 오류가 날 때의 조치: #depmod -a
- make install: 컴파일된 커널 이미지를 /boot 디렉터리로 옮기며 부팅 이미지 생성
- makefile: 어느 파일들을 어떤 방법으로 컴파일링하여 최종적으로 어떤 실행파일을 만들 것인지를 알려주는 설정 파일

명령어	실행 환경	특징
make oldconfig	콘솔, X Window	텍스트로 각 커널 옵션 일일이 선택
make menuconfig	콘솔, X Window	텍스트 환경으로 각 커널 메뉴 선택
make xconfig	X Window	QT 기반의 그래픽 커널 메뉴 설정 환경
make gconfig	X Window	GTK+2 기반의 그래픽 커널 메뉴 설정 환경
make defconfig	콘솔, X Window	모든 옵션을 기본값으로 적용하여 설정
make allmodconfig	콘솔, X Window	모든 옵션을 가능한 모듈로 선택하여 설정
make allyesconfig	콘솔, X Window	모든 옵션을 yes로 설정, 비바람직한 설정
make allnoconfig	콘솔, X Window	모든 옵션을 최소 상태로 설정, 비바람직한 설정



# 패키지와 커널

# 커널

## Kernel Compile

커널 설정 옵션 카테고리	설명
Code maturity level options	개발중의 코드 및 드라이버 선택 유무
General setup	커널에 대한 일반적인 설정
Loadable module support	모듈 적재에 관한 설정
Block layer	블록 장치에 활성화 및 해제에 관한 설정
Processor type and features	CPU 및 Memory 기능에 관한 설정
Power management options	절전 기능에 관한 설정
Bus options	메인보드의 PCI, EISA, PCMCIA 지원 설정
Executable file formats	커널 지원 가능한 바이너리 형태 설정
Networking	네트워크 기능 지원에 관한 설정
Device Drivers	HDD, SCSI, 네트워크, 멀티미디어장치 드라이버 설정
File systems	커널 지원 파일 시스템 설정
Instrumentation Support	커널 프로필 지원 설정
Kernel hacking	커널 해킹 설정
Security options	커널 보안 옵션 설정
Cryptographic options	비밀번호화 및 인증에 관련된 알고리즘 모듈 설정
Library routines	외부 모듈의 라이브러리 함수 사용 여부

01. 다음 중 리눅스 시스템에 설치되어 있는 패키지 목록을 확인하는 명령으로 알맞은 것은?

- ① dpkg -i                      ② dpkg -l
- ③ dpkg -l                    ④ dpkg -L

02. 다음 중 온라인 패키지 관리 도구로 가장 거리가 먼 것은?

- ① YaST                        ② yum
- ③ apt-get                    ④ zypper

03. 다음 중 SUSE 리눅스에서 사용하는 패키지 관리 도구로 가장 알맞은 것은?

- ① rpm                        ② yum
- ③ dpkg                      ④ zypper

04. 아파치 웹 서버를 소스 설치하는 과정에서 configure를 진행했으나 다시 configure 하기 위해 관련 파일들을 제거하려고 한다. 다음 ( 괄호 ) 안에 들어갈 내용으로 알맞은 것은?

- ① clean                      ② delete
- ③ remove                    ④ reconfigure

```
[root@www httpd-2.4.46] # make ( 괄호 )
```

05. 다음 중 소스 파일로 프로그램을 설치하는 단계로 알맞은 것은?

- ① make → configure → make install
- ② make → make install → configure
- ③ configure → make → make install
- ④ configure → make install → make

06. 다음 중 소스 설치 과정 중에서 configure 작업 후에 관련 정보가 저장되는 파일명으로 알맞은 것은?

- ① install                      ② .config
- ③ .configure                  ④ Makefile

07. apt-get에 대한 설명으로 틀린 것은?

- ① 데비안 계열 리눅스 배포판에서 사용되는 유틸리티이다.
- ② /etc/apt/sources.list 파일을 참고하여 패키지설치 관련 정보를 관리한다.
- ③ remove 명령어는 /var/cache/apt/archive 에 생성된 파일을 전부 삭제한다.
- ④ APT(Advanced Packaging Tool) 라이브러리를 이용한 명령행 기반의 도구이다.

08. 다음 ( 괄호 ) 안에 들어갈 내용으로 알맞은 것은? 2

- ① make                        ② cmake
- ③ configure                  ④ dnf

( 괄호 )는 소스 컴파일 시 사용되는 Make의 대체프로그램으로 멀티플랫폼을 지원하기 위한 목적으로 등장한 오픈 소스 프로젝트이다. 이 방법을 사용하는 대표적인 프로그램에는 mysql 이 있다.

09. 다음 중 온라인 패키지 관리 도구로 가장 거리가 먼 것은?

- ① YaST                        ② yum
- ③ apt-get                    ④ zypper

10. 다음 중 리눅스 시스템에 설치되어 있는 패키지 목록을 확인하는 명령으로 알맞은 것은?

- ① dpkg -i                    ② dpkg -l
- ③ dpkg -l                    ④ dpkg -L

수고하셨습니다