



패키지 개요

2 패키지 설치

3 스냅 패키지 설치

1. 때키지 개요

❖우분투 패키지의 특징

- 바이너리 파일로 구성되어 있는 컴파일이 필요 없다.
- 패키지의 파일이 관련 디렉터리에 바로 설치된다.
- 패키지를 삭제할 때 관련된 파일을 일괄적으로 삭제할 수 있다.
- 기존에 설치한 패키지를 삭제하지 않고 바로 업그레이드 할 수 있다.
- 패키지의 설치 상태를 검증할 수 있다.
- 패키지에 대한 정보를 제공한다.
- 해당 패키지가 의존하는 패키지가 무엇인지 알려준다. 따라서 의존성이 있는 패키지를 미리 설치 할 수도 있고, apt-get 명령을 사용하면 의존성이 있는 패키지를 자동으로 설치된다.



❖우분투 패키지의 카테고리

- main : 우분투에 의해 공식적으로 지원되며 자유롭게 배포할 수 있다.
- restricted : 우분투에 의해 지원되지만 완전한 자유 라이선스 소프트웨어는 아니다.
- universe : 리눅스에서 사용할 수 있는 대부분의 소프트웨어로 자유 소프트 웨어일 수도 있고 아닐 수도 있지만 기술적 지원이 보장되지는 않는다.
- multiverse : 자유 소프트웨어가 아닌 소프트웨어가 포함되어 있으며, 개인 이 직접 라이선스를 확인해야 한다.

❖우분투 패키지의 이름 구성

패키지명

```
user1@Server1:~$ ls -l /media/user1/Ubuntu\ 22.04.3\ LTS\ amd64/ubuntu/pool/main/b/b43-fwcutter/합계 28
-r--r--- 1 user1 user1 27890 3월 24 2022 b43-fwcutter_019-7build2_amd64.deb

b43-fwcutter 019-7build2 amd64.deb

아키텍쳐
```

❖우분투 패키지 저장소

- 우분투는 패키지와 패키지에 대한 정보를 저장하고 있는 서버인 패키지 저장소라는 개념을 사용한다.
- 패키지 저장소에는 패키지의 기능 추가나 보안 패치 등 지속적인 업그레이드를 집중적으로 관리하고 사용자는 저장소에 접속하여 최신 패키지를 내려 받아 설치 할 수 있다.

```
user1@Server1:~$ cat /etc/apt/sources.list
#deb cdrom:[Ubuntu 22.04.3 LTS _Jammy Jellyfish_ - Release amd64 (20230807.2)]/ jammy main restricted

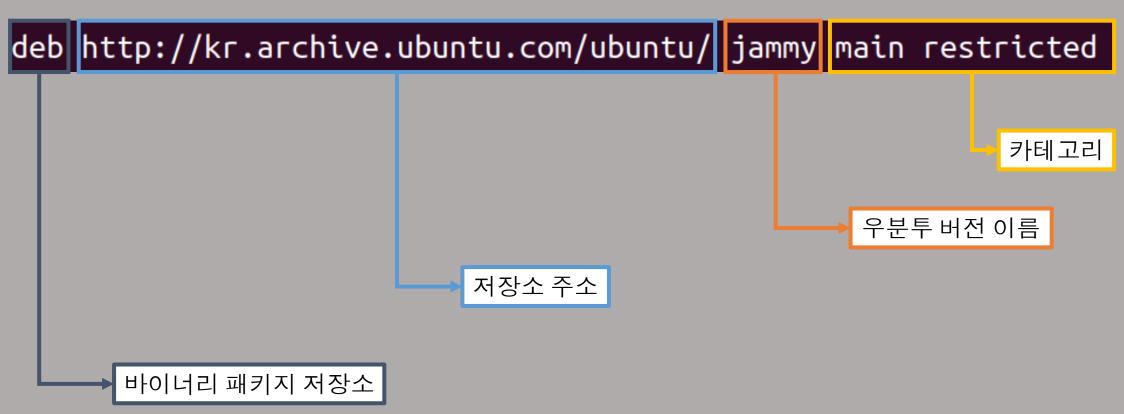
# See http://help.ubuntu.com/community/UpgradeNotes for how to upgrade to

# newer versions of the distribution.

deb http://kr.archive.ubuntu.com/ubuntu/ jammy main restricted

# deb-src http://kr.archive.ubuntu.com/ubuntu/ jammy main restricted
```





2. 우분투 패키지 설치

❖APT 명령으로 패키지 관리하기

■ APT는 'Advanced package tool'의 약자로 보다 편리하게 패키지를 관리 할 수 있는 명령이다. 대표적으로 apt-get, apt-cache가 있다.

apt-cache				
기능	APT 캐시에 질의하여 여러 가지 정보를 검색한다.			
형식	apt-cache [옵션] [명령]			
옵션	-f : 검색 결과를 패키지에 대한 전체 기록을 출력한다. -h : 간단한 도움말을 출력한다.			
서브 명령	stats: 캐시에 대한 통계 정보를 출력한다. dump: 현재 설치된 패키지를 업그레이드 한다. search [키워드]: 캐시에서 키워드를 검색한다. showpkg [패키지명]: 패키지에 대한 의존성 정보를 검색하여 출력한다. show [패키지명]: 패키지에 대한 간단한 정보를 출력한다. pkgnames: 사용 가능한 모든 패키지의 이름을 출력한다.			

■ APT 캐시 통계 정보 보기

user1@Server1:~\$ apt-cache stats 전체 패키지 이름 : 119728 (3,352 k) 전체 패키지 구조: 114983 (5,059 k) 일반 패키지: 74885 순수 가상 패키지: 2748 단일 가상 패키지: 22953 혼합 가상 패키지: 2701 빠짐: 11696 개별 버전 전체: 88060 (7,749 k) 개별 설명 전체: 171109 (4,107 k) 전체 의존성: 570485/155336 (13.9 M) 전체 버전/파일 관계: 9279 (223 k) 전체 설명/파일 관계: 40037 (961 k) 전체 제공 매핑: 44330 (1,064 k) 전체 패턴 문자열: 224732 (5,360 k) 전체 빈 용량: 98.5 k

전세 인 용당: 98.5 k 차지하는 전체 용량: 43.4 M

Total buckets in PkgHashTable: 196613

Unused: 113993

전체 패키지 이름 : 패키지 이름의 전체 수 일반 패키지 : 일반적으로 사용하는 패키지의 수 순수 가상 패키지 : 패키지는 없지만 기능을 가진 패키지 수 단일 가상 패키지 : 한 패키지가 특정한 가상 패키지의 기능 을 제공하는 수 혼합 가상 패키지 : 특정 가상 패키지를 제공하거나 가상 패 키지의 이름을 패키지 이름으로 사용하는 경우의 수 빠짐 : 의존성은 있지만 어떠한 패키지도 제공하는 않는 수 개별 버전 전체 : 캐시에 있는 패키지 버전의 수 전체 의존성 : 캐시에 있는 모든 패키지의 의존성 관계의 수

■ 사용 가능한 패키지 이름 확인

```
user1@Server1:~$ apt-cache pkgnames | more
lib32stdc++-11-dev-s390x-cross
bazel-bootstrap
ruby-jmespath
goverlay
libgiza0
librust-quickcheck-macros-dev
r-cran-egg
r-bioc-snpstats
ruby-ami
libghc-parseargs-dev
golang-github-munnerz-goautoneg-dev
dpic
libnova-0.16-0
libvisp-blob-dev
linux-objects-nvidia-525-server-5.19.0-1027-lowlatency
python3-trufont
python-django-celery-beat-doc
```

■ 패키지 이름 검색하기

```
user1@Server1:~$ apt-cache search vsftpd
vsftpd - lightweight, efficient FTP server written for security
vsftpd-dbg - lightweight, efficient FTP server written for security (debug)
ccze - robust, modular log coloriser
ftpd - File Transfer Protocol (FTP) server
resource-agents-extra - Cluster Resource Agents
yasat - simple stupid audit tool
```

■ 패키지 정보 검색하기

```
user1@Server1:~$ apt-cache show vsftpd
Package: vsftpd
Architecture: amd64
Version: 3.0.5-Oubuntu1
Priority: extra
Section: net
Origin: Ubuntu
Maintainer: Ubuntu Developers <ubuntu-devel-discuss@lists.ubuntu.com>
```



■ 패키지 의존성 검색하기

❖apt-get 명령

	apt-get			
기능	패키지를 관리한다.			
형식	apt-get [옵션] [명령]			
옵션	-d:패키지를 설치하지 않고 다운로드만 한다 -f:의존성이 깨진 패키지를 수정하려고 시도한다. -h:간단한 도움말을 출력한다.			
명령	update: 패키지 저장소에서 새로운 패키지 정보를 가져온다. upgrade: 현재 설치 된 패키지를 업그레이드 한다. install [패키지명]: 패키지를 설치한다. remove [패키지명]: 패키지를 삭제한다. download [패키지명]: 패키지를 현재 디렉터리로 내려받는다. autoclean: 불완전하게 내려 받거나 오래된 패키지를 삭제한다. clean: 저장된 모든 패키지를 삭제하여 디스크 공간을 확보한다. check: 의존성이 깨진 패키지를 확인한다.			

■ 패키지 정보 업데이트하기

```
user1@Server1:~$ sudo apt-get update
[sudo] user1 암호:
기존:1 http://kr.archive.ubuntu.com/ubuntu jammy InRelease
받기:2 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
받기:3 http://kr.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
받기:4 http://kr.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
내려받기 338 k바이트, 소요시간 2초 (154 k바이트/초)
패키지 목록을 읽는 중입니다... 완료
```

■ 패키지 업그레이드 하기

user1@Server1:~\$ sudo apt-get upgrade 패키지 목록을 읽는 중입니다... 완료 의존성 트리를 만드는 중입니다... 완료 상태 정보를 읽는 중입니다... 완료 업그레이드를 계산하는 중입니다... 완료 다음 패키지를 과거 버전으로 유지합니다:

■ 특정 패키지 설치하기

```
user1@Server1:~$ sudo apt-get install xterm
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다... 완료
상태 정보를 읽는 중입니다... 완료
다음의 추가 패키지가 설치될 것입니다 :
```

■ 업그레이드 없이 패키지 설치하기

user1@Server1:~\$ sudo apt-get install xterm --no-upgrade

■ 패키지 설치 없이 업그레이드만 하기

user1@Server1:~\$ sudo apt-get install xterm --only-upgrade

■ 패키지 삭제하기

```
user1@Server1:~$ sudo apt-get remove xterm
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다... 완료
상태 정보를 읽는 중입니다... 완료
다음 패키지가 자동으로 설치되었지만 더 이상 필요하지 않습니다:
libutempter0
```

■ 설정 파일까지 모두 삭제하기

```
user1@Server1:~$ sudo apt-get purge xterm
user1@Server1:~$ sudo apt-get remove --purge xterm
```

■ 패키지 자동 정리 및 삭제하기

```
user1@Server1:~$ sudo apt-get autoremove
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다... 완료
상태 정보를 읽는 중입니다... 완료
다음 패키지를 지울 것입니다:
libutempter0
0개 업그레이드, 0개 새로 설치, 1개 제거 및 16개 업그레이드 안 함.
```

■ 디스크 공간 정리하기

user1@Server1:~\$ sudo apt-get clean

■ 설치 하지 않고 다운로드 하기

```
user1@Server1:~$ sudo apt-get download xterm
받기:1 http://kr.archive.ubuntu.com/ubuntu jammy/universe amd64 xterm amd64 372-1ubuntu1 [857 kB]
내려받기 857 k바이트, 소요시간 3초 (337 k바이트/초)
```



■ 패키지의 소스 관련 명령 ▶패키지의 소스 코드만 다운받아서 사용할 때 사용하는 명령이다.

resource			
패키지만 다운	sudo apt-get –download-only source [패키지명]		
패키지 압축 풀기	sudo apt-get source [패키지명]		
컴파일	sudo apt-get –compile source [패키지명]		

Quiz

- ❖아래의 내용을 실습 해 본다.
 - python3을 삭제 한다.
 - python3을 설치 한다.
 - 설치한 패키지의 정보를 확인한다.

❖dpkg 명령으로 패키지 관리하기

■ 일반적으로 패키지를 설치할 때는 APT 명령을 사용하지만 보다 세부적인 기능을 사용하려면 dpkg 명령이 사용 된다.

	dpkg			
기능	데비안의 패키지 관리 명령			
형식	dpkg [옵션] [파일명 또는 패키지명]			
옵션	-1:설치된 패키지의 목록을 출력한다1[패키지명]: 패키지의 설치 상태를 출력한다s[패키지명]: 패키지의 상세 정보를 출력한다S[경로명]: 경로명이 포함된 패키지를 검색한다L[패키지명]: 패키지에서 설치된 파일의 목록을 출력한다c[.deb 파일]: 지정한 .deb 파일의 내용을 출력한다i[.deb 파일]: 해당 파일을 설치 한다r[패키지명]: 해당 패키지를 삭제한다P[패키지명]: 해당 패키지와 설정 정보를 모두 삭제한다x[.deb 파일][디렉터리]: 해당 파일을 지정한 디렉터리에 저장한다.			

■ 패키지 목록 보기

■ 특정 패키지 정보 보기

■ 패키지 상세 정보 보기

```
user1@Server1:~$ dpkg -s zip
Package: zip
Status: install ok installed
Priority: optional
Section: utils
Installed-Size: 531
```

Maintainer: Ubuntu Developers <ubuntu-devel-discuss@lists.ubuntu.com>

■ 특정 파일이 포함된 패키지 검색하기

```
user1@Server1:~$ dpkg -S /bin/ls
coreutils: /bin/ls
```

■ 패키지가 설치된 파일 목록 검색하기

```
user1@Server1:~$ dpkg -L zip
/.
/usr
/usr/bin
/usr/bin/zip
```

■ 패키지의 파일 목록 검색 하기

```
      user1@Server1:~$ dpkg -c xterm_372-1ubuntu1_amd64.deb

      drwxr-xr-x root/root
      0 2022-03-12 07:13 ./etc/

      drwxr-xr-x root/root
      0 2022-03-12 07:13 ./etc/X11/
```

■ 패키지 설치하기

```
user1@Server1:~$ sudo dpkg -i xterm_372-1ubuntu1_amd64.deb
(데이터베이스 읽는중 ...현재 152577개의 파일과 디렉터리가 설치되어 있습니다.)
Preparing to unpack xterm_372-1ubuntu1_amd64.deb ...
Unpacking xterm (372-1ubuntu1) over (372-1ubuntu1) ...
```

■ 패키지 삭제하기

```
user1@Server1:~$ sudo dpkg -r xterm
(데이터베이스 읽는중 ...현재 152577개의 파일과 디렉터리가 설치되어 있습니다.)
xterm (372-1ubuntu1)를 제거합니다...
user1@Server1:~$ sudo dpkg -P xterm
(데이터베이스 읽는중 ...현재 152536개의 파일과 디렉터리가 설치되어 있습니다.)
Purging configuration files for xterm (372-1ubuntu1) ...
```

■ deb 패키지 풀기

```
user1@Server1:~$ mkdir xterm
user1@Server1:~$ sudo dpkg -x xterm_372-1ubuntu1_amd64.deb xterm
user1@Server1:~$ ls -l xterm
합계 8
drwxr-xr-x 3 root root 4096 3월 12 2022 etc
drwxr-xr-x 4 root root 4096 3월 12 2022 usr
```

Quiz

- ❖아래와 같이 실습 해 본다.
 - gnome-chess 패키지가 설치 되어 있는지 확인한다.
 - 설치 되어 있지 않다면 apt 명령으로 패키지를 다운로드 받는다.
 - 패키지를 구성하는 파일의 내용을 확인한다.
 - 패키지를 설치한다.
 - 의존성이 있는 패키지가 있으면 apt 명령으로 설치 한다.
 - 다시 gnome-chess 패키지를 설치한다.
 - 설치된 파일을 확인 한다.

3. 스냅 패키지 설치

- ❖스냅은 우분투가 새로 도입한 패키지 형식으로 샌드박스 형태의 패키지다. 패키지를 만들 때 프로그램이 사용하는 모든 라이브러 리를 패키지 안에 포함한다.
- ❖샌드박스란 원래 외부에서 받은 파일을 그냥 실행하는 것이 아니라 보호 영역에서 실행해보는 것으로 외부 파일이 내부 시스템에 악영향을 주는 것을 방지하는 보안 기술이다.

- ❖스냅의 장점
 - 개발자가 다른 패키지나 라이브러리와의 의존성을 신경 쓰지 않는다.
 - 기존 시스템과 격리되어 실행하는 형식으로 보안이 강화된다.

❖스냅 사용하기

snap				
기능	스냅 패키지를 설치하고 설정하고 삭제한다.			
형식	snap [옵션] [명령]			
명령	disable : 스냅 서비스와 실행 파일의 사용을 중지한다. download [스냅명] : 지정한 스냅 패키지를 다운로드 한다. enable : 스냅 서비스와 실행 파일의 사용을 시작한다. find [스냅명] : 지정한 스냅을 검색한다. info [스냅명] : 지정한 스냅의 상세 정보를 출력한다. install [스냅명] : 지정한 스냅을 설치한다. list : 설치한 스냅의 목록을 출력한다. remove [스냅명] : 지정한 스냅을 삭제한다.			

■ 스냅 목록 출력하기

<pre>user1@Server1:~\$ snap list</pre>					
이름	버전	개	정 추적	발행인	노트
bare	1.0	5	latest/stable	canonical√ base	
соге20	20230801	2015	latest/stable	canonical√ base	- 1
соге22	20230725	858	latest/stable	canonical√ base	
firefox	116.0.3-2	3026	latest/stable/	mozilla√ -	
gnome-3-38-2004	0+git.efb213a	143	latest/stable/	canonical√ -	
gnome-42-2204	0+git.ff35a85	126	latest/stable/	canonical√ -	
gtk-common-themes	0.1-81-g442e511	1535	latest/stable/	canonical√ -	
snap-store	41.3-71-g709398e	959	latest/stable/	canonical√ -	
snapd	2.59.5	19457	latest/stable	canonical√ snapd	
snapd-desktop-integration	0.9	83	latest/stable/	canonical√ -	

■ 스냅 찾기

user1@Server1:~\$ snap find hello-world						
이름	버전	발행인	노트 요약			
hello-world	6.4	canonical√ -	The 'hello-world' of snaps			
hello-world-lc	0.1	lolachangmwc2017 -	Just testing			
hello-world-jbennett	0.1	jamiebennett -	Just testing			
hello-world-jbennett2	0.1	jamiebennett -	Just testing			

■ 스냅 설치하기

```
user1@Server1:~$ sudo snap install hello-world
"hello-world"에 대한 필수 구성 요소를 사용할 수 있는지 확인
"hello-world"에 대한 필수 구성 요소를 사용할 수 있는지 확인
"hello-world"에 대한 필수 구성 요소를 사용할 수 있는지 확인
"hello-world"에 대한 필수 구성 요소를 사용할 수 있는지 확인
```

■ 스냅의 상세 정보 확인하기

user1@Server1:~\$ snap info hello-world

name: hello-world

summary: The 'hello-world' of snaps

publisher: Canonical√

store-url: https://snapcraft.io/hello-world

contact: snaps@canonical.com

license: unset

■ 스냅 삭제하기

user1@Server1:~\$ sudo snap remove hello-world hello-world 제거됨

Quiz

- ❖아래의 내용을 실습 해 본다.
 - 스냅을 이용해서 game을 검색한다.
 - 검색된 스냅 중 game-2048 스냅을 설치한다.
 - game-2048 스냅의 상세 정보를 확인한다.

