



Unit 12

Compress & Archiving



<http://cafe.daum.net/bsscolaris>

단원 목표

- 압축 관련 명령어
 - gzip/gunzip 명령어
 - bzip2/bunzip2 명령어
- 압축 + 아카이빙 관련 명령어
 - tar 명령어
 - zip/unzip 명령어

<http://cafe.daum.net/bsscolaris>

용어(terms)

- * 압축(compress)
- * 아카이빙(achiving)

■ 압축(compress)이란? 용량을 줄이는 기술

파일에 저장되어 있는 정보를 압축하여 보다 작은 기억 공간에 동일한 정보를 저장하는 기술입니다. 일반적으로 정보에 포함되어 있는 중복된 내용을 삭제하거나 보다 적은 길이의 코드를 사용하여 정보를 표현하는 방법을 사용하여 저장에 필요한 공간의 크기를 줄입니다. 이러한 과정을 '압축한다'고 하며, 압축된 정보를 사용하기 위해서 다시 원래의 상태로 복원하는 과정을 '압축을 푼다'고 합니다.([참고] altools.co.kr 자주 묻는 질문 게시판)

■ 아카이빙(Archive)이란? 데이터 보관 기술(여러개의 파일을 한개의 파일로 묶어 주는 기능)

디지털 아카이빙(digital archiving)이란 지속적으로 보존할 가치를 가진 디지털 객체를 장기간 관리하여 이후의 이용을 보장하는 활동을 말한다. 기록 관리 분야에서는 아카이빙이라는 용어를 공적인 기록의 보존을 처리하는 과정이라는 의미로 사용하는 데 비하여 컴퓨터 분야에서는 컴퓨터 파일의 일시적인 백업 컬렉션을 의미하기도 한다. 그러한 혼동을 피하기 위하여 아카이빙 대신 '디지털 보존(digital preservation)'이라는 용어를 사용하기도 한다. 디지털 아카이빙의 핵심 개념은 장기 보존과 지속적 가치이다.([참고] wikipedia 디지털 아카이빙)

압축 관련 명령어 (gzip/gunzip 명령어)

• gzip/gunzip 명령어

```
# gzip file1
# gunzip -c file1.gz (# zcat file1.gz)
# gunzip file1.gz      (# gzip -d file1.gz)
```

<http://cafe.daum.net/bcsolaris>

1 gzip / gunzip CMD

NAME

gzip, gunzip, zcat - compress or expand files

SYNOPSIS

```
gzip [ -acdfhILnNrtvV19 ] [-S suffix] [ name ... ]
gunzip [ -acfhlLnNrtvV ] [-S suffix] [ name ... ]
zcat [ -fhLV ] [ name ... ]
```

DESCRIPTION

The gzip command reduces the size of the named files using Lempel-Ziv coding (LZ77). Whenever possible, each file is replaced by one with the extension .gz, while keeping the same ownership modes, access and modification times. (The default extension is z for MSDOS, OS/2 FAT, Windows NT FAT and Atari.) If no files are specified, or if a file name is "-", the standard input is compressed to the standard output. The gzip command will only attempt to compress regular files. In particular, it will ignore symbolic links.

If the compressed file name is too long for its file system, gzip truncates it. The gzip command attempts to truncate only the parts of the file name longer than 3 characters. (A part is delimited by dots.) If the name consists of small parts only, the longest parts are truncated. For example, if file names are limited to 14 characters, gzip.msdo.exe is compressed to gzi.msdo.exe.gz. Names are not truncated on systems which do not have a limit on file name length.

By default, gzip keeps the original file name and timestamp in the compressed file. These are used when decompressing the file with the -N option. This is useful when the compressed file name was truncated or when the timestamp was not preserved after a file transfer.

OPTIONS

-d --decompress --uncompress
Decompress.

-f --force

Force compression or decompression even if the file has multiple links or the corresponding file already exists, or if the compressed data is read from or written to a terminal. If the input data is not in a format recognized by gzip, and if the option --stdout is also given, copy the input data without change to the standard output: let zcat behave as cat. If -f is not given, and when not running in the background, gzip prompts to verify whether an existing file should be overwritten.

```

-r --recursive
    Travel the directory structure recursively. If any of the file names specified on
    the command line are directories, gzip will descend into the directory and compress
    all the files it finds there (or decompress them in the case of gunzip ).

-S .suf --suffix .suf
    When compressing, use suffix .suf instead of .gz. Any non-empty suffix can be
    given, but suffixes other than .z and .gz should be avoided to avoid confusion when
    files are transferred to other systems.

    When decompressing, add .suf to the beginning of the list of suffixes to try, when
    deriving an output file name from an input file name.

-v --verbose
    Verbose. Display the name and percentage reduction for each file compressed or de-
    compressed.

```

(1). gzip/gunzip CMD

gzip 명령어는 파일을 압축할 때 사용한다. gzip 명령어 다음에 파일의 이름을 쓰면 파일이 압축되며, 파일에 확장자(.gz)가 붙게 된다. gzip 명령어는 Lempel-Ziv coding (LZ77)을 사용한다. 압축되기 전의 파일의 속성 정보는 압축이 된 이후에도 유지된다. gzip 명령어로 압축된 파일의 내용은 zcat 명령어나 gunzip 명령어의 -c 옵션을 사용하여 볼 수 있다. 압축을 해제 할 때는 gzip 명령어 다음에 압축된 파일을 쓰면 된다. 그럼 압축이 해제되며, 파일의 확장자(.gz)가 없어지게 된다.

[명령어의 형식]

(압축) # gzip file1

file1.gz

(확인) # gunzip -c file1.gz (# zcat file1.gz) (# gzip -dc file1.gz)

(해제) # gunzip file1.gz (# gzip -d file1.gz)

[명령어 옵션]

| 옵 션 | 설 명 |
|-----|---|
| -S | 기본 생성 파일인 .gz 형태가 아닌 지정한 파일로 생성한다. |
| -d | 압축된 파일의 압축을 해제한다. 이는 uncompress명령과도 같다. |
| -f | 이미 이전에 압축파일이 존재하더라도, 무시하고 압축파일을 생성한다. |
| -r | 지정한 것이 디렉토리라면 하위 디렉토리와 파일까지 모두 압축한다. |
| -v | 압축 통계를 보여준다. |

[실습] gzip/gunzip 압축 실습

■ 사용 시스템

- server1

■ 실습 시나리오

- gzip 명령어를 사용하여 파일 압축
- zcat으로 압축파일 내용 확인
- 파일의 압축 해제
- 여러개의 파일을 하나로 압축할 수 있을까?

[EX1] gzip 명령어를 사용하여 파일 압축

① 실습을 위한 파일 생성

```
[root@server1 ~]# cd /test
[root@server1 /test]# rm -rf /test/*
[root@server1 /test]# cp /etc/services file1
```

```
[root@server1 /test]# cp file1 file2
[root@server1 /test]# cp file1 file3
[root@server1 /test]# cp file1 file4
[root@server1 /test]# ls
```

```
file1 file2 file3 file4
```

② 압축하기

```
[root@server1 /test]# gzip file1    (# gzip -v file1)
[root@server1 /test]# ls -l
```

```
-rw-r--r-- 1 root root 140K Feb 22 13:56 file1.gz
-rw-r--r-- 1 root root 677K Feb 22 13:56 file2
-rw-r--r-- 1 root root 677K Feb 22 13:56 file3
-rw-r--r-- 1 root root 677K Feb 22 13:56 file4
```

③ 디렉토리 안의 파일들 압축하기

```
[root@server1 /test]# cp -L -r /usr/share/doc dir1
• -R, -r, --recursive      copy directories recursively
• -L, --dereference        always follow symbolic links in SOURCE
[root@server1 /test]# du -sh dir1          /* -s : sum, -h : human */
```

```
85M    dir1
```

- -s, --summarize display only a total for each argument
- -h, --human-readable print sizes in human readable format (e.g., 1K 234M 2G)

```
[root@server1 /test]# gzip -r dir1      /* -r : recursive */
[root@server1 /test]# du -sh dir1
```

```
37M    dir1
```

```
[root@server1 /test]# tree dir1 (# find dir1)
```

```
..... (중략) .....
|-- zenity
|   |-- AUTHORS.gz
|   |-- NEWS.gz
|   |-- README.gz
|   |-- THANKS.gz
|-- zip
|   |-- CHANGES.gz
|   |-- README.CR.gz
|   |-- README.gz
|   |-- TODO.gz
|   |-- WHATSNEW.gz
|   |-- WHERE.gz
|   |-- algorithm.txt.gz
|-- zlib
|   |-- ChangeLog.gz
|   |-- FAQ.gz
```

```
[root@server1 /test]# gzip -d -r dir1    (# gunzip -r dir1)
```

```
[root@server1 /test]# du -sh dir1
```

```
89M    dir1
```

[EX2] zcat으로 압축파일 내용 확인

```
[root@server1 /test]# mkdir -p /test ; cd /test ; rm -rf /test/*
```

```
[root@server1 /test]# cp /etc/services file1
```

```
[root@server1 /test]# gzip file1
```

```
[root@server1 /test]# cat file1.gz
```

```
..... (중략) .....
>L7aZ:~f5B}aVn~C'w~p~d~Wr~N~X~QzE~I~t~?W)~
~5sZ~vx%$~EDT MC:|~d~l~B~$~q~lu~"~`W~N~i~5~(+wq~M~A~>~3)
~6~W~k~}H~*~W~>JR~E~j~i0~e~n~E~'~
OE~
```

```
[root@server1 /test]# gunzip -c file1.gz    (# zcat file1.gz)
```

```
* -c, --stdout          write on standard output, keep original files unchanged
```

```
..... (중략) .....
axio-disc      35100/tcp      # Axiomatic discovery protocol
axio-disc      35100/udp      # Axiomatic discovery protocol
pmwebapi       44323/tcp      # Performance Co-Pilot client HTTP API
cloudcheck-ping 45514/udp      # ASSIA CloudCheck WiFi Management keepalive
cloudcheck     45514/tcp      # ASSIA CloudCheck WiFi Management System
spremotetablen 46998/tcp      # Capture handwritten signatures
```

(결론) cat 명령어는 text 파일의 내용을 확인할 때 사용하는 명령어이기 때문에 압축된 파일의 내용을 확인할 때 사용하면 정상적으로 내용이 보이지 않는다. 따라서, 압축을 임시적으로 해제하고 보여 주는 zcat/gunzip -c 명령어를 사용해야 한다.

[EX3] 파일의 압축 해제

```
[root@server1 /test]# gzip -d file1.gz    /* -d : decompress */
```

```
* -d, --decompress    decompress
```

```
[root@server1 /test]# ls -l
```

```
-rw-r--r-- 1 root root 677K Feb 22 15:29 file1
```

[EX4] 여러개의 파일을 하나로 압축할 수 있을까?

- 결론적으로 하나의 파일로 압축을 하기 위해서는 "아카이빙 + 압축" 기능을 사용해야 한다. 단순히 gzip 명령어만 사용하는 경우는 여러개의 파일에 대한 압축만 진행한다.

```
[root@server1 /test]# cd /test ; rm -rf /test/*
```

```
[root@server1 /test]# cp /etc/services file1
```

```
[root@server1 /test]# cp file1 file2
```

```
[root@server1 /test]# cp file1 file3
```

```
[root@server1 /test]# ls
```

```
file1 file2 file3
```

```
[root@server1 /test]# gzip file1 file2 file3    (# gzip file*)
```

```
[root@server1 /test]# ls
```

```
file1.gz file2.gz file3.gz
```

```
[root@server1 /test]# gunzip *.gz    (# gzip -d *.gz)
```

```
[root@server1 /test]# ls
```

```
file1 file2 file3
```

(결론) gzip 명령어를 사용하여 압축하는 방법은 한개씩 개별적으로 압축을 하기 때문에, 파일을 한꺼번에 압축하기 위해서는 한개의 파일로 아카이빙(ex: tar CMD 사용)하고, 그 파일을 압축해야 한다.

```
# tar cvf file.tar file1 file2 file3
```

```
# gzip file.tar
```

압축 관련 명령어 (bzip2/bunzip2 명령어)

• bzip2/bunzip2 명령어

```
# bzip2 file1
# bzip2 -c file1.bz2 (# bzcatt file1.bz2)
# bunzip2 file1.bz2 (# bunzip2 -d file1.bz2)
```

<http://cafe.daum.net/bsscolaris>

2 bzip2 / bunzip2 CMD**이름**

bzip2, bunzip2 - 블럭 정렬 파일 압축기, v0.9.5
 bzcatt - 파일의 압축을 풀어 stdout으로 보냄
 bzip2recover - 손상된 bzip2 파일로부터 자료를 복구

사용법

```
bzip2 [ -cdfkqstvzVL123456789 ] [ filenames ... ]
bunzip2 [ -fkvsVL ] [ filenames ... ]
bzcatt [ -s ] [ filenames ... ]
bzip2recover filename
```

설명

bzip2 는 Burrows-Wheeler 블럭 정렬 텍스트 압축 알고리즘과 Huffman 코딩을 이용하여 파일을 압축한다. 이 압축은 전통적인 LZ77/LZ78 기반의 압축기에 의해 이루어지는 압축보다 일반적으로 상당히 좋다. 그리고 통계적인 압축기중 PPM 계열의 성능에 근접한다.

명령행 옵션은 GNU gzip 의 그것과 매우 유사하나 동일하지는 않다. bzip2 명령행 플래그 다음에 파일 이름의 목록이 올 것으로 예상된다. 각각의 파일은 그것들의 압축된 상태로 대체되는데, 파일 이름은 "원래이름.bz2" 이 된다. 각각의 압축된 파일은 같은 수정 날짜, 퍼미션, 그리고 가능하면 원본에 대응하는 소유자를 갖는다. 그래서 이러한 속성이 압축을 풀때 정확하게 복구될 수 있다. 파일명 처리는 파일명, 허가권, 소유자 혹은 날짜 개념이 없는 파일 시스템이나 파일명의 길이에 심각한 제한이 있는 MS-DOS 같은 경우는 원래의 속성을 보존할 수 있는 방법이 없으므로 단순하다고 할 수 있다.

bzip2 과 bunzip2 는 기본적으로 존재하는 파일을 덮어쓰지 않는다. 이렇게 하려면, -f 플래그를 사용하라.

만약 파일 이름이 지정되지 않으면, bzip2 는 표준 입력을 압축하여 표준 출력으로 보낸다. 이 경우에, bzip2 는 압축된 출력을 터미널로 보낸다. 그래서 이 출력은 모두 이해할 수 없으며 무의미하다.

bunzip2 (혹은 bzip2 -d) 은 모든 지정된 파일의 압축을 푼다. bzip2 에 의해 생성되지 않은 파일이 발견되면 무시되고, 경고를 출력할 것이다. bzip2 는 압축이 풀린 파일의 이름을 압축된 파일로부터 다음과 같이 추측할 것이다.

| | | |
|---------------|---------|------------------|
| filename.bz2 | becomes | filename |
| filename.bz | becomes | filename |
| filename.tbz2 | becomes | filename.tar |
| filename.tbz | becomes | filename.tar |
| anyothername | becomes | anyothername.out |

옵션

- c --stdout
압축이나 해제를 해서 표준 출력으로 보낸다.
- d --decompress
강제 압축해제 bzip2, bunzip2 그리고 bzip2 는 실제로 같은 프로그램이다. 그리고 어떤 행동이 취해질지는 사용된 이름에 의해 결정된다. 이 플래그는 그 방법을 무시하고 강제로 bzip2 이 압축해제를 하도록 한다.
- q --quiet
불필요한 경고 메시지를 무시한다. I/O 에러와 다른 치명적인 사건을 포함한 메시지는 무시되지 않을 것이다.
- v --verbose
장황한 모드 -- 처리된 각 파일의 압축 비율을 출력한다. -v 를 몇 붙이면 장황한 정도를 증가시켜서, 진단을 목적으로 하는 흥미가 주된 정보를 잔뜩 출력하게 된다.
- 1 to -9
압축할 때 블록 크기를 100 k, 200 k .. 900 k 로 설정한다. 압축해제시에는 아무런 영향이 없다. 아래의 메모리 관리를 참조하라.

높은 성능의 압축률을 자랑하는 유틸리티(Utility)이다. Burrows-Wheeler 블록 정렬 테스트 압축 알고리즘과 호프만 코딩을 사용하여 파일을 압축한다. bzip2는 tar 아카이브 파일을 압축하게 되면 기존 아카이브 파일이 제거되고, bz2 확장자가 생성된다.

압축된 파일은 압축전의 날짜, 퍼미션, 소유자 등의 속성을 그대로 갖는다. 그러므로 압축된 파일을 풀었을 때에도 원본 파일의 속성을 그대로 간직한다.

[명령어 형식]

(압축) # bzip2 file1

file1.bz2

(확인) # bunzip2 -c file1.bz2

(# bzip2 -d file1.bz2) (# bzip2 -dc file1.bz2)

(해제) # bunzip2 file1.bz2

(# bzip2 -d file1.bz2)

[명령어 옵션]

| 옵 션 | 설 명 |
|-------|-------------------------------------|
| -c | 파일을 압축 또는 풀어서 표준 출력으로 보낸다. |
| -d | 압축을 푼다. |
| -z | 파일을 압축한다. |
| -t | 지정된 파일의 무결성을 검사하지만, 압축을 풀지 않는다. |
| -f | 압축, 해제할 때 같은 이름의 파일이 있을 경우 덮어쓰기 한다. |
| -k | 파일을 압축이나 해제할 때 원본파일을 지우지 않는다. |
| -q | 경고 메시지가 나오지 않는다. |
| -v | bzip2 작업사항을 자세히 볼 수 있다. |
| -1~-9 | 압축할 때의 블록 크기를 정해준다. |

[실습] bzip2/bunzip2 압축 실습

■ 사용 시스템

- server1

■ 실습 시나리오

- bzip2 명령어를 사용한 파일 압축
- bzip2로 압축파일 내용 확인
- 파일 압축 해제

[EX1] bzip2 명령어를 사용한 파일 압축

① 실습 준비

```
[root@server1 /test]# cd /test
[root@server1 /test]# rm -rf /test/*
[root@server1 /test]# cp /etc/services file1
[root@server1 /test]# cp file1 file2
[root@server1 /test]# cp file1 file3
```

② 파일 압축하기

```
[root@server1 /test]# bzip2 file1
[root@server1 /test]# ls -l
```

```
-rw-r--r-- 1 root root 127K Feb 22 17:08 file1.bz2
-rw-r--r-- 1 root root 677K Feb 22 17:08 file2
-rw-r--r-- 1 root root 677K Feb 22 17:08 file3
```

```
# vi ~/.bashrc
alias bzip2='bzip2 -v'
alias bunzip2='bunzip2 -v'
# . ~/.bashrc
# alias bzip2
```

③ gzip versus bzip2 압축률 비교

```
[root@server1 /test]# gzip file2
[root@server1 /test]# ls -l file1* file2*
```

```
-rw-r--r-- 1 root root 127K Feb 22 17:08 file1.bz2
-rw-r--r-- 1 root root 140K Feb 22 17:08 file2.gz
```

(결론) bzip2 명령어를 사용하여 압축한 것이 더 압축률이 좋은 것으로 판단이 된다.

[EX2] bzip2로 압축파일 내용 확인

```
[root@server1 /test]# cat file1.bz2
```

```
..... (중략) .....
5HJ...:j...].+w...B...Y...20...$N...4fB...CvXB...Y...p...z]...Z...,,...4...BL...#...f'...U
[...Kr...[W...H...Bb..."+...WDLu}W^...b...N...7C...%...6l...Ut)...ja...f...q...6...Cf...
T...g...E...πy5x...K...W$d...HUIu...Sä...y...V...I0...d...0...c...U...Ä...U...S...&...
J...r...)|...*R...ZmI...E...2...I...<...d...h...$...d...V...B...IL...M...
U...<q...|...ÉQ...v...G...Z{'ü...e:
n<...)-...h...Y3}T...Gb...J...I'3Ga^...<...{...7...K"...M>-03-...r...
6...+...
I...j...I...SR...T...+T...Rë...
{...oJ]...X...g...a>...B...h...w...h...h...f...d...0...F...I...{*...4%/...Wi?s...uUh...
[%Y...5'...^...+... (cwF...N...F...?...?...;...H...
```

```
[root@server1 /test]# bzip2 file1.bz2      (# bunzip2 -c file1.bz2)
```

```
..... (중략) .....
pmwebapi      44323/tcp      # Performance Co-Pilot client HTTP API
cloudcheck-ping 45514/udp      # ASSIA CloudCheck WiFi Management keepalive
cloudcheck     45514/tcp      # ASSIA CloudCheck WiFi Management System
spremotetablet 46998/tcp      # Capture handwritten signatures
```

(결론) *.bz2 파일은 bzip2 명령어를 통해 압축된 파일이며, 압축된 파일은 bzip2/bunzip2 -c 명령어를 사용하여 안의 내용을 정상적으로 볼수 있다.

[EX3] 파일 압축 해제

① 파일 압축 해제

```
[root@server1 /test]# bzip2 -d file1.bz2  (# bunzip2 file1.bz2)
[root@server1 /test]# ls -l
```

```
-rw-r--r-- 1 root root 677K Feb 22 17:08 file1
-rw-r--r-- 1 root root 140K Feb 22 17:08 file2.gz
-rw-r--r-- 1 root root 677K Feb 22 17:08 file3
```

[참고] xz CMD

(설치) # yum -y install xz

(압축) # xz file1

file1.xz

(확인) # unxz -c file1.xz

(해제) # unxz file1.xz

[참고] 압축률 비교(gzip CMD & bzip2 CMD & xz CMD)

* 압축률 : gzip < bzip2 < xz

압축 + 아카이빙 (tar 명령어)

• tar 명령어

```
# tar cvf file.tar file1 file2 file3
# tar tvf file.tar
# tar xvf file.tar

# tar cvzf file.tar.gz file1 file2 file3
# tar tvzf file.tar.gz
# tar xvzf file.tar.gz

# tar cvjf file.tar.bz2 file1 file2 file3
# tar tvjf file.tar.bz2
# tar xvzf file.tar.bz2
```

<http://cafe.daum.net/bsscolaris>

3 tar CMD

NAME

tar - (Tape ARchive) The GNU version of the tar archiving utility

SYNOPSIS

tar <operation> [options]

Operations:

```
[ - ]A --catenate --concatenate
[ - ]c --create
[ - ]d --diff --compare
[ - ]r --append
[ - ]t --list
[ - ]u --update
[ - ]x --extract --get
--delete
```

Common Options:

```
-C, --directory DIR
-f, --file F
-j, --bzip2
-p, --preserve-permissions
-v, --verbose
-z, --gzip
```

FUNCTION LETTERS

```
-c, --create
    create a new archive

-t, --list
    list the contents of an archive

-x, --extract, --get
    extract files from an archive

-v, --verbose
    verbosely list files processed
```

tar(tape archive) 명령어는 여러개의 파일이나 디렉토리를 한 개의 묶음파일로 만들 때 사용한다. 또는 Tape 디바이스에 백업할 때 사용한다. tar 명령어는 c, x, v, f 옵션들을 사용할 수 있다. 한 개의 아카이빙 할 때는 c 옵션을 사용하고 f 옵션으로 만들어질 파일의 이름을 지정하면 된다. 아카이빙 파일의 내용을 볼 때는 t 옵션을 사용하고 f 옵션으로 아카이빙 파일을 지정한다. 아카이빙 파일을 해제 할 때는 x 옵션을 사용하고 f 옵션으로 아카이빙 파일을 지정하면 된다.

[명령어 형식] tar 명령어 사용시 옵션에 -기호 생략가능

```
# tar cvf <묶음파일명> <파일명1> <파일명2> ... /* c: create, v: verbose, f: file or tape */
# tar tvf <묶음파일명> /* t: content */
# tar xvf <묶음파일명> /* x: extract */
```

■ (tar CMD)

```
# tar cvf file.tar file1 file2 file3
# tar tvf file.tar
# tar xvf file.tar
```

■ (tar CMD + gzip CMD)

```
# tar cvzf file.tar.gz file1 file2 file3 (# tar cvzf file.tgz file1 file2 file3)
# tar tvzf file.tar.gz
# tar xvzf file.tar.gz (# tar xvf file.tar.gz)
```

■ (tar CMD + bzip2 CMD)

```
# tar cvjf file.tar.bz2 file1 file2 file3 (# tar cvjf file.tbz file1 file2 file3)
# tar tvjf file.tar.bz2
# tar xvjf file.tar.bz2 (# tar xvf file.tar.bz2)
```

■ (tar CMD + xz CMD)

```
# tar cvJf file.tar.xz file1 file2 file3 (# tar cvjf file.txz file1 file2 file3)
# tar tvJf file.tar.xz
# tar xvJf file.tar.xz (# tar xvf file.tar.txz)
```

[명령어 옵션]

| 옵 션 | 설 명 |
|--------|--|
| -c | 여러 개의 파일을 하나의 파일로 묶을 때 |
| -v | 생성과정을 보여 줌 |
| -x | 파일의 압축을 해제하고 풀어 낼 때 |
| -r | 파일 및 디렉토리 추가 |
| -u | tar 파일과 새로운 디렉토리 내의 파일과 비교하여 최근 파일을 아카이브에 추가 |
| -t | tar 파일의 내용을 확인할 때 |
| -z | gzip과 관련하여 압축이나 해제를 한꺼번에 할 때 |
| -j, -l | bzip2를 이용해서 압축한다. |

[참고] cpio CMD

```
# man cpio
cpio - copy files to and from archives

(생성) # cd /test ; find . | cpio -o > /backup/LOG.cpio (cpio -ocv)
(확인) # cpio -t < /backup/LOG.cpio (cpio -itv)
(해제) # cd /test ; cpio -i < /backup/LOG.cpio (cpio -iv)

[EX] cpio 간단한 실습
# cd /test
# ls
-> file1, file2, file3
# find . -name "file[123]" | cpio -o > file.cpio
# cpio -t < file.cpio
# rm -f file?
# cpio -i < file.cpio
```

[실습] tar 명령어 실습

■ 실습 시스템

- server1

■ 실습 시나리오

- tar 명령어 실습
- 여러개의 파일 압축하기
- tar을 이용한 묶음과 압축(gzip)
- tar을 이용한 묶음과 압축(bzip2)

[EX1] tar 명령어 실습

(실습 준비)

```
[root@server1 /test]# cd /test ; rm -f /test/*
```

```
[root@server1 /test]# cp /etc/passwd file1
```

```
[root@server1 /test]# cp file1 file2
```

```
[root@server1 /test]# cp file1 file3
```

```
[root@server1 /test]# ls
```

```
file1 file2 file3
```

(tar 명령어 실습)

```
[root@server1 /test]# tar cvf file.tar file1 file2 file3  (# tar cvf file.tar file*)
```

```
file1
file2
file3
```

```
[root@server1 /test]# ls -l file*
```

```
-rw-r--r-- 1 root root 20480 Aug 19 13:46 file.tar
-rw-r--r-- 1 root root 3082 Aug 19 13:45 file1
-rw-r--r-- 1 root root 3082 Aug 19 13:46 file2
-rw-r--r-- 1 root root 3082 Aug 19 13:46 file3
```

```
[root@server1 /test]# rm -f file1 file2 file3  (# rm -f file?)
```

```
[root@server1 /test]# ls
```

```
file.tar
```

```
[root@server1 /test]# tar tvf file.tar
```

```
-rw-r--r-- root/root      3082 2010-08-19 13:45:54 file1
-rw-r--r-- root/root      3082 2010-08-19 13:46:02 file2
-rw-r--r-- root/root      3082 2010-08-19 13:46:04 file3
```

```
[root@server1 /test]# tar xvf file.tar
```

```
file1
file2
file3
```

```
[root@server1 /test]# ls file*
```

```
file.tar file1 file2 file3
```

[EX2] 여러개의 파일 압축하기

```
[root@server1 /test]# rm -rf file.tar
[root@server1 /test]#
```

```
[root@server1 /test]# tar cvf file.tar file1 file2 file3
```

```
file1
file2
file3
```

```
[root@server1 /test]# gzip file.tar
[root@server1 /test]# ls
```

```
file.tar.gz file1 file2 file3
```

```
[root@server1 /test]# gzip -d file.tar.gz      (# gunzip file.tar.gz)
[root@server1 /test]# ls
```

```
file.tar file1 file2 file3
```

```
[root@server1 /test]# rm -f file?
[root@server1 /test]# tar xvf file.tar
```

```
file1
file2
file3
```

[참고] 명령어 비교(EX: file.tar.gz)

```
# tar cvf file.tar f1 f2 f3
# gzip file.tar
file.tar.gz
```

```
# tar cvzf file.tar.gz f1 f2 f3
file.tar.gz
```

```
# gzip -d file.tar.gz
file.tar
# tar xvf file.tar
f1 f2 f3
```

```
# tar xvzf file.tar.gz
f1 f2 f3
```

[EX3] tar을 이용한 묶음과 압축(gzip)

```
[root@server1 /test]# rm -f file.tar
[root@server1 /test]#
```

```
[root@server1 /test]# ls
```

```
file1 file2 file3
```

[참고] 명령어 비교

```
# tar cvf file.tar file1 file2 file3
```

```
# gzip file.tar
```

or

```
# tar cvzf file.tar.gz file1 file2 file3
```

```
[root@server1 /test]# tar cvzf file.tar.gz file1 file2 file3
```

```
[root@server1 /test]# file file.tar.gz
```

```
file.tar.gz: gzip compressed data, from Unix, last modified: Thu Mar 18 12:02:09 2010
```

```
[root@server1 /test]# tar tvzf file.tar.gz
```

-> 내용 확인

```
[root@server1 /test]# rm -f file?
```

```
[root@server1 /test]# ls
```

```
file.tar.gz
```

```
[root@server1 /test]# tar xvzf file.tar.gz    (# tar xvf file.tar.gz)
```

```
file1
file2
file3
```

```
[root@server1 /test]# ls
```

```
file.tar.gz file1 file2 file3
```

[EX4] tar을 이용한 묶음과 압축(bzip2)

```
[root@server1 /test]# rm -f file.tar.gz
[root@server1 /test]# ls
```

```
file1 file2 file3
```

[참고] 명령어 비교

```
# tar cvf file.tar file1 file2 file3
```

```
# bzip2 file.tar
```

or

```
# tar cvjf file.tar file1 file2 file3
```

```
[root@server1 /test]# tar cvjf file.tar.bz2 file1 file2 file3
[root@server1 /test]# ls
```

```
file.tar.bz2 file1 file2 file3
```

```
[root@server1 /test]# file file.tar.bz2
```

```
file.tar.bz2: bzip2 compressed data, block size = 900k
```

```
[root@server1 /test]# tar tvjf file.tar.bz2
-> 내용 확인
```

```
[root@server1 /test]# rm -f file1 file2 file3
[root@server1 /test]# ls
```

```
file.tar.bz2
```

```
[root@server1 /test]# tar xvjf file.tar.bz2
[root@server1 /test]# ls
```

```
file.tar.bz2 file1 file2 file3
```

[참고] jar CMD(Java Archive)

(선수작업) JDK/SDK 설치

```
# which jar
```

```
# yum list 'java-*
```

```
# yum -y install java-1.8.0-openjdk java-1.8.0-openjdk-devel
```

■ 파일 이름: ABCD.jar

[jar 명령어 형식] JDK/SDK 설치시 존재

(생성) # jar cvf file.jar file1 file2 file3

(확인) # jar tvf file.jar

(해제) # jar xvf file.jar

(간단한 실습)

```
# cd /test
```

```
# jar cvf file.jar file?
```

```
# jar tvf file.jar
```

```
# rm -f file?
```

```
# jar xvf file.jar
```

```
# find / -name '*.jar' -type f
```


압축 + 아카이빙(zip/unzip명령어)

• zip/unzip 명령어

```
# zip file.zip file1 file2 file3
# unzip -l file.zip
# unzip file.zip
```

<http://cafe.daum.net/bcsolaris>

4 zip/unzip CMD

NAME

zip - package and compress (archive) files

DESCRIPTION

zip is a compression and file packaging utility for Unix, VMS, MSDOS, OS/2, Windows 9x/NT/XP, Minix, Atari, Macintosh, Amiga, and Acorn RISC OS. It is analogous to a combination of the Unix commands tar(1) and compress(1) and is compatible with PKZIP (Phil Katz's ZIP for MSDOS systems).

A companion program (unzip(1L)) unpacks zip archives. The zip and unzip(1L) programs can work with archives produced by PKZIP (supporting most PKZIP features up to PKZIP version 4.6), and PKZIP and PKUNZIP can work with archives produced by zip (with some exceptions, notably streamed archives, but recent changes in the zip file standard may facilitate better compatibility). zip version 3.0 is compatible with PKZIP 2.04 and also supports the Zip64 extensions of PKZIP 4.5 which allow archives as well as files to exceed the previous 2 GB limit (4 GB in some cases). zip also now supports bzip2 compression if the bzip2 library is included when zip is compiled. Note that PKUNZIP 1.10 cannot extract files produced by PKZIP 2.04 or zip 3.0. You must use PKUNZIP 2.04g or unzip 5.0p1 (or later versions) to extract them.

OPTIONS

-l

--to-crlf

Translate the Unix end-of-line character LF into the MSDOS convention CR LF. This option should not be used on binary files. This option can be used on Unix if the zip file is intended for PKUNZIP under MSDOS. If the input files already contain CR LF, this option adds an extra CR. This is to ensure that unzip -a on Unix will get back an exact copy of the original file, to undo the effect of zip -l. See -ll for how binary files are handled.

zip 명령어는 여러 운영체제와 호환이 가능하다. 예를 들어 호환이 가능하다는 것은 Linux에서 zip 명령어를 이용하여 압축한 파일을 Windows에서 압축을 해제할 수 있고, windows에서 zip 방식으로 압축한 파일을 Linux에서 unzip 명령어를 사용하여 풀 수 있다는 것이다. zip 명령어를 사용하여 압축을 할 때는 특별한 옵션없이 압축할 파일 이름과 압축될 파일들의 이름들을 나열하면 된다. 압축된 내용을 확인하기 위해서는 unzip 명령어의 -l 옵션을 사용하여 압축된 파일들의 목록을 확인할 수 있다. 압축을 해제할 때는 unzip 명령어를 사용하여 압축된 파일의 이름을 지정하면 압축이 해제된다.

[zip 명령어 형식](압축) # **zip [-r] file.zip file1 file2 file3**(확인) # **unzip -l file.zip**(해제) # **unzip file.zip**

[참고] 윈도우 압축 툴과 호환이 가능

* (예) 알~~집~~, 빵~~집~~, 윈~~집~~(Winzip),**[실습] zip/unzip 명령어 실습**

■ 실습 시스템

- server1

■ 실습 시나리오

- zip/unzip 명령어 기본 사용법 실습
- zip 명령어를 통한 디렉토리 압축 + 아카이빙

[EX1] zip/unzip 명령어 기본 사용법 실습**[root@server1 /test]# cd /test****[root@server1 /test]# ls**

file1 file2 file3

[root@server1 /test]# zip file.zip file1 file2 file3

```

adding: file1 (deflated 62%)
adding: file2 (deflated 73%)
adding: file3 (deflated 73%)

```

[root@server1 /test]# ls -l

```

-rw-r--r-- 1 root root 195K Jan 15 10:30 file.zip
-rw-r--r-- 1 root root 2.0K Jan 15 10:30 file1
-rw-r--r-- 1 root root 354K Jan 14 12:38 file2
-rw-r--r-- 1 root root 354K Jan 14 12:38 file3

```

[root@server1 /test]# unzip -l file.zip

| Archive: | file.zip |
|----------|----------------------|
| Length | Date Time Name |
| 2027 | 01-15-14 10:30 file1 |
| 362031 | 01-14-14 12:38 file2 |
| 362031 | 01-14-14 12:38 file3 |
| 726089 | 3 files |

[root@server1 /test]# rm -rf file?**[root@server1 /test]# unzip file.zip**

```

Archive: file.zip
  inflating: file1
  inflating: file2
  inflating: file3

```

[root@server1 /test]# ls

-> 정보 확인

[EX2] zip 명령어를 통한 디렉토리 압축 + 아카이빙

```
[root@server1 /test]# cd /test
[root@server1 /test]# cp -L -r /usr/share/doc /test
[root@server1 /test]# du -sh doc
```

```
86M doc
```

```
[root@server1 /test]# zip -r all.zip doc
```

```
adding: sysconfig/ (stored 0%)
adding: sysconfig/rhn/ (stored 0%)
adding: sysconfig/rhn/sources (deflated 21%)
adding: sysconfig/ip6tables-config (deflated 60%)
adding: sysconfig/xinetd (deflated 35%)
adding: sysconfig/system-config-securityle
..... (중략) .....
```

```
[root@server1 /test]# unzip -l all.zip
```

```
..... (중략) .....
 391  01-15-14 10:34  sysconfig/wpa_supplicant
   20  01-15-14 10:34  sysconfig/hidd
 610  01-15-14 10:34  sysconfig/syslog
 304  01-15-14 10:34  sysconfig/system-config-users
   83  01-15-14 10:34  sysconfig/network
1929  01-15-14 10:34  sysconfig/named
-----
246363                118 files
```

```
[root@server1 /test]# rm -rf doc
[root@server1 /test]# unzip all.zip
```

-> 출력 내용 생략

```
[root@server1 /test]# ls
```

-> 정보 확인

(실무 예) 인터넷상에 받은 파일 압축 해제 방법

| 압축 파일 형식 | 압축 툴 | 압축 해제 방법 | |
|----------------------------------|-------------------|---------------------------------|--------------------------|
| file. gz | ---- gzip ----> | # gunzip file.gz | (# gzip -d file.gz) |
| file. bz2 | ---- bzip2 ----> | # bunzip2 file.bz2 | (# bzip2 -d file.bz2) |
| file. xz | ---- xz -----> | # unxz file.xz | (# xz -d file.xz) |
| file. tar.gz file.tgz | ---- tar/gzip -> | # tar x vzf file.tar.gz | (# tar xvf file.tar.gz) |
| file. tar.bz2 file.tbz | ---- tar/bzip2-> | # tar x vjf file.tar.bz2 | (# tar xvf file.tar.bz2) |
| file. tar.xz file.txz | ---- tar/xz ----> | # tar x vJf file.tar.xz | (# tar xvf file.tar.xz) |
| file. zip | ---- zip -----> | # unzip file.zip | |
| file. jar | ---- jar -----> | # jar xvf file.jar | |

[추가적인 실습] 아파치 사이트에서 파일을 다운로드 받고 압축을 해제한다.
아파치 웹서버 소스 코드 다운로드

<https://httpd.apache.org/download.cgi>
* 다운로드 파일: **httpd-2.4.62.tar.bz2**
* 다운로드 디렉토리: /test
* 압축 해제

For details, see the [Official Announcement](#) and the [CHANGES_2.4](#) and [CHANGES_2.4.62](#) lists.

- Source: [httpd-2.4.62.tar.bz2](#) [PGP] [SHA256] [SHA512]
- Source: [httpd-2.4.62.tar.gz](#) [PGP] [SHA256] [SHA512]

[추가적인 실습] 아파치 사이트에서 파일을 다운로드 받고 압축을 해제한다.
톰캣 소스 코드 다운로드

<https://tomcat.apache.org/download-90.cgi>
* 다운로드 파일: **apache-tomcat-9.0.97.zip**
* 다운로드 디렉토리: /test
* 압축 해제

Binary Distributions

- Core:
 - [zip \(pgp, sha512\)](#)
 - [tar.gz \(pgp, sha512\)](#)
 - [32-bit Windows.zip \(pgp, sha512\)](#)
 - [64-bit Windows.zip \(pgp, sha512\)](#)
 - [32-bit/64-bit Windows Service Installer \(pgp, sha512\)](#)

■ 압축 파일의 종류

<https://docs.fileformat.com/ko/>

<https://docs.fileformat.com/ko/executable/msi/>

■ 압축 파일의 종류

https://www.egovframe.go.kr/wiki/doku.php?id=egovframework:rte2:fdl:압축파일의_종류:압축파일의_종류

| 압축 파일 | 설 명 |
|-----------|--|
| .alz | 이스트소프트에서 개발한 압축 형식입니다.. 분할 압축을 할 경우 확장자는 (ALZip 으로 생성) alZ, a00, a01...형식으로 생성 |
| .ace | ACE, WinAce에서 이용하는 압축 형식입니다. 분할 압축을 할 경우 확장자는 ace, c00, c01, ... 형식으로 생성됨 |
| .arc | DOS용 프로그램 pkarc.com, ppxarc.com 에서 사용되는 압축형식 |
| .arj | DOS용 프로그램 arj.exe, 윈도우용 프로그램 WinArj에서 이용하는 압축형식. 분할 압축을 할 경우 확장자는 arj, a01, a02,... 형식으로 생기게 됨. |
| .b64 | 인터넷에서 문서를 주고 받을때 사용하는 형식으로 BASE64MIME 형식으로 인코딩된 파일임. |
| .bh | BinHex Format. E-Mail 로 이진(binary)형태의 파일을 보내기 위해 사용하는 형식임. |
| .bhz | 인터넷에서 문서를 주고 받을때 사용하는 형식으로 BASE64MIME 형식으로 인코딩된 파일임. |
| .bin | Macintosh용이며, MacBinary Format.Aladdin StuffIt Expander 에서 지원함. |
| .bz2 | UNIX용의 bzip2 에서 사용하는 압축 형식 입니다. 파일 하나만 압축 할 수 있으므로 주로 .tar와 함께 사용되며 이 경우 .tar.bz2 의 확장자를 갖음. (bzip2 로 생성) |
| .cab | Microsoft Cabinet 파일. 마이크로소프트에서 사용하는 압축 형식임. |
| .ear | 내부적으로 zip 압축 알고리즘을 사용하는 파일 형식임. |
| .enc | E-Mail 로 이진(binary)형태의 파일을 보내기 위해 사용하는 형식임. |
| .gz | UNIX용의 gzip에 에서 사용하는 압축 형식.파일 하나만 압축 할 수 있으므로 주로 .tar와 함께 사용되며 이 경우 .tar.gz 의 확장자를 갖고 줄여서 .tgz확장자를 사용하기도 함. |
| .ha | PPMC를 개선한 압축 파일 형식 |
| .hqx | 맥에서 제작된 파일을 인터넷에서 문서를 주고 받을때 사용하는 형식임. (BinHex로 생성) |
| .ice | DOS용 프로그램 ice.exe 에서 사용하는 압축 형식임. 실제 파일 내용은 lha와 동일함. |
| .img | Disk image를 저장해둔 파일로, Falk Huth에 의해 만들어진 img.exe라는 프로그램을 이용하여 파일들을 추출해 낼 수 있음. |
| .jar | 자바의 jar.exe 에서 사용하는 압축 형식. 내부적으로 zip 압축 알고리즘을 사용함. |
| .lha, lzh | DOS용 프로그램 lha.exe, lharc.exe에서 사용하던 압축 형식. Lempel-Ziv 알고리즘을 사용함 |
| .mim | 인터넷에서 문서를 주고 받을때 사용하는 형식. |
| .pak | DOS용 프로그램 pak.exe에서 사용하던 압축 형식. |
| .rar | DOS용 프로그램 rar.exe와 윈도우용 프로그램 winrar.exe에서 사용하는 압축 형식. 분할 압축을 할 경우 확장자는 rar, r00, r01,... 형식으로 생기게 됨. |
| .sit | Macintosh에서 이용되는 압축 Format.WinArj, Aladdin StuffIt Expander , WinPack등의 윈도우용 압축 프로그램에서 이를 지원 함. |
| .tar | UNIX 명령 tar를 이용해 생성되는 파일 형식으로 실제로는 압축은 되지 않고 여러 파일을 하나로 묶어주기만 함. 보통 tar로 묶은 후 gz으로 압축하며 이 경우 .tar.gz 의 확장자를 갖고 줄여서 .tgz확장자를 사용하기도 함. |
| .tgz | UNIX 에서 tar로 묶은 파일을 gzip으로 압축한 파일 형식. |
| .uue | UU Encoded Format. E-Mail 로 이진(binary)형태의 파일을 보내기 위해 사용하는 형식. |
| .war | 내부적으로 zip압축 알고리즘을 사용하는 파일 형식. |
| .xxe | XXEncode Format. E-Mail 로 이진(binary)형태의 파일을 보내기 위해 사용하는 형식. |
| .z | UNIX용의 compress, uncompress에서 사용하는 파일 형식. |
| .zip | 도스용 프로그램 pkzip.exe, pkunzip.exe에서 사용하는 파일 형식. |
| .zoo | DOS용 프로그램 zoo.exe 에서 사용되는 파일 형식. |
| .001 | rzjoin 으로 분할된 파일 형식. 압축이 아닌 단순히 001, 002..... 순서로만 분할된 파일. |