# 7-Zip – File Archiver

## Introduction

An open-source application and one of the best file archivers in Windows, Linux and MacOS.

The program supports **7z**, **XZ**, **BZIP2**, **GZIP**, **TAR**, **ZIP**, WIM, ARJ, CAB, CHM, CPIO, CramFS, DEB, DMG, FAT, HFS, **ISO**, LZH, LZMA, MBR, MSI, NSIS, NTFS, RAR, RPM, SquashFS, UDF, VHD, WIM, XAR, Z, etc.

Home page: <https://www.7-zip.org/>

Manuals:

* 7-zip.chm (in 7-Zip folder)

## Released Versions

Along with the very famous fully-functional UI application, 7-Zip has some variants for command line uses.

|  |  |  |
| --- | --- | --- |
| **Version** | **Description** | **Note** |
| 7z.exe | Fully-functional UI application | Main features:   * High compression ratio in the new 7z format * Supported formats:   + Packing / unpacking: 7z, XZ, BZIP2, GZIP, TAR, ZIP and WIM.   + Unpacking only: APFS, AR, ARJ, Base64, CAB, CHM, CPIO, CramFS, DMG, EXT, FAT, GPT, HFS, IHEX, ISO, LZH, LZMA, MBR, MSI, NSIS, NTFS, QCOW2, RAR, RPM, SquashFS, UDF, UEFI, VDI, VHD, VHDX, VMDK, XAR and Z * Fast compression and decompression * Self-extracting capability for 7z format * Strong AES-256 encryption in 7z and ZIP formats * Integration with Windows Shell * Powerful File Manager * Powerful command line version * Localizations for 90 languages |
| 7za.exe | Standalone CLI version of 7-Zip | Main features:   * High compression ratio in 7z format * Supported formats:   + Packing / unpacking: 7z, XZ, ZIP, GZIP, BZIP2 and TAR   + Unpacking only: Z, lzma, CAB * Highest compression ratio for ZIP and GZIP formats. * Fast compression and decompression * Strong AES-256 encryption in 7z and ZIP formats. |
| 7zr.exe | Reduced version of 7za.exe | * Support only 7z format. |

### CLI

7-Zip (a) 23.01 (x86) : Copyright (c) 1999-2023 Igor Pavlov : 2023-06-20

Usage: **7za <command> [<switches>...] <archive\_name> [<file\_names>...] [@listfile]**

<Commands>

**a** : Add files to archive

**b** : Benchmark

**d** : Delete files from archive

**e** : Extract files from archive (without using directory names)

**h** : Calculate hash values for files

**i** : Show information about supported formats

**l** : List contents of archive

**rn** : Rename files in archive

**t** : Test integrity of archive

**u** : Update files to archive

**x** : eXtract files with full paths

<Switches>

**--** : Stop switches and @listfile parsing

**-ai[r[-|0]]{@listfile|!wildcard}** : Include archives

**-ax[r[-|0]]{@listfile|!wildcard}** : eXclude archives

**-ao{a|s|t|u}** : set Overwrite mode

**-an** : disable archive\_name field

**-bb[0-3]** : set output log level

**-bd** : disable progress indicator

**-bs{o|e|p}{0|1|2}** : set output stream for output/error/progress line

**-bt** : show execution time statistics

**-i[r[-|0]]{@listfile|!wildcard}** : Include filenames

**-m{Parameters}** : set compression Method

**-mmt[N]** : set number of CPU threads

**-mx[N]** : set compression level: -mx1 (fastest) ... -mx9 (ultra)

**-o{Directory}** : set Output directory

**-p{Password}** : set Password

**-r[-|0]** : Recurse subdirectories for name search

**-sa{a|e|s}** : set Archive name mode

**-scc{UTF-8|WIN|DOS}** : set charset for console input/output

**-scs{UTF-8|UTF-16LE|UTF-16BE|WIN|DOS|{id}}** : set charset for list files

**-scrc[CRC32|CRC64|SHA1|SHA256|\*]** : set hash function for x, e, h commands

**-sdel** : delete files after compression

**-seml[.]** : send archive by email

**-sfx[{name}]** : Create SFX archive

**-si[{name}]** : read data from stdin

**-slp** : set Large Pages mode

**-slt** : show technical information for l (List) command

**-snh** : store hard links as links

**-snl** : store symbolic links as links

**-sni** : store NT security information

**-sns[-]** : store NTFS alternate streams

**-so** : write data to stdout

**-spd** : disable wildcard matching for file names

**-spe** : eliminate duplication of root folder for extract command

**-spf[2]** : use fully qualified file paths

**-ssc[-]** : set sensitive case mode

**-sse** : stop archive creating, if it can't open some input file

**-ssp** : do not change Last Access Time of source files while archiving

**-ssw** : compress shared files

**-stl** : set archive timestamp from the most recently modified file

**-stm{HexMask}** : set CPU thread affinity mask (hexadecimal number)

**-stx{Type}** : exclude archive type

**-t{Type}** : Set type of archive

**-u[-][p#][q#][r#][x#][y#][z#][!newArchiveName]** : Update options

**-v{Size}[b|k|m|g]** : Create volumes

**-w[{path}]** : assign Work directory. Empty path means a temporary directory

**-x[r[-|0]]{@listfile|!wildcard}** : eXclude filenames

**-y** : assume Yes on all queries

## Speed Improvement

### Multithread

7-Zip supports multithread compression for some formats.

But only a few come with multithreaded decompression support, including **BZIP2**, **LZMA2**, ZIP (?), 7z (?), XR (?).

Refs:

* <https://sourceforge.net/p/sevenzip/discussion/45797/thread/136f029b/>
* <https://copyprogramming.com/howto/fastest-way-to-extract-zip-archive>
* <https://copyprogramming.com/howto/how-to-make-7-zip-faster>
* <https://peazip.github.io/peazip-compression-benchmark.html>

### Alternatives

* UberZip: <https://matthicks.com/2008/01/14/multithreaded-unzip/>
* QuickUnzip: <https://github.com/lukehutch/quickunzip>

## 7-Zip vs Python Lib

### Single Thread

Below is a simple Python code to compare decompression speed between 7-Zip and Python lib in single thread.

Surprisingly, Python lib is faster. Although this test is too small to prove anything, it somehow shows that Python lib is not slow at all.

import subprocess

import time

import zipfile

import tarfile

def extract\_zip(filepath: str, destdir: str):

    with zipfile.ZipFile(filepath, 'r') as file:

        file.extractall(destdir)

def extract\_targz(filepath: str, destdir: str):

    with tarfile.open(filepath) as file:

        file.extractall(destdir)

path\_to\_7z = '7za.exe'

def extract\_7z(filepath: str, destdir: str):

    # -bb3: Set output log level to max

    # -bso2: Set log level for output stream to max

    # -mmt4: Set number of cores to 4

    cmd = f'{path\_to\_7z} x -bb3 -bso2 "{filepath}" -o"{destdir}"'

    subprocess.run(cmd, shell=True, check=True)

def calc\_exec\_time(start, end, unit='ms'):

    exec\_time = 0

    match unit:

        case 's':

            exec\_time = end - start

        case 'ms':

            exec\_time = (end - start) \* 1000

    return round(exec\_time, 3)

filepath = 'testfile\\testfile1.zip'

destdir = 'testfile'

start = time.time()

extract\_zip(filepath, destdir)

end = time.time()

print(f'Extract ZIP: {calc\_exec\_time(start, end)}')       # 22176.752, 23166.24

time.sleep(10)

start = time.time()

extract\_7z(filepath, destdir)

end = time.time()

print(f'Extract 7Z: {calc\_exec\_time(start, end)}')        # 25969.803, 26970.559, 27776.067

### Multi Thread

Python – Multithread unzip program: <https://superfastpython.com/multithreaded-unzip-files/>

# Aria2 – Download Accelerator

## Introduction

Most users know about downloading files from CLI using wget or curl. However, aria2 has a few advantages over them all:

* Improve download speed by downloading from more than one source in the single session.
* Allow to pause and resume downloads.
* Support various protocols, including HTTP/HTTPS, FTP, SFTP, BitTorrent and Metalink.

Note that aria2 is originally for Linux, but it can be installed and run in Windows easily.

## CLI

Usage: aria2c [OPTIONS] [URI | MAGNET | TORRENT\_FILE | METALINK\_FILE]...

Options:

 -v, --version                Print the version number and exit.

                              Tags: #basic

 -h, --help[=TAG|KEYWORD]     Print usage and exit.

                              The help messages are classified with tags.

                              A tag starts with "#". For example, "--help=#http" gets the usage for the options tagged with "#http".

If non-tag word is given, print the usage for the options whose name includes that word.

                              Possible Values: #basic, #advanced, #http, #https, #ftp, #metalink, #bittorrent, #cookie, #hook,

#file, #rpc, #checksum, #experimental, #deprecated, #help, #all

                              Default: #basic

                              Tags: #basic, #help

 -l, --log=LOG                The file name of the log file. If '-' is specified, log is written to stdout.

                              Possible Values: /path/to/file, -

                              Tags: #basic

 -d, --dir=DIR                The directory to store the downloaded file.

                              Possible Values: /path/to/directory

                              Default: E:\V2X\vcm\_dev\_tool\package

                              Tags: #basic, #file

 -o, --out=FILE               The file name of the downloaded file. It is always relative to the directory given in -d

                              option. When the -Z option is used, this option will be ignored.

                              Possible Values: /path/to/file

                              Tags: #basic, #http, #ftp, #file

 -s, --split=N                Download a file using N connections.

                              If more than N URIs are given, first N URIs are used and remaining URLs are used for backup.

                              If less than N URIs are given, those URLs are used more than once so that N connections total

are made simultaneously. The number of connections to the same host is restricted by the

                              --max-connection-per-server option. See also the

                              --min-split-size option.

                              Possible Values: 1-\*

                              Default: 5

                              Tags: #basic, #http, #ftp

 --file-allocation=METHOD     Specify file allocation method.

                              'none' doesn't pre-allocate file space. 'prealloc' pre-allocates file space before download begins.

                              This may take some time depending on the size of the file.

                              If you are using newer file systems such as ext4 (with extents support), btrfs, xfs or NTFS

                              (MinGW build only), 'falloc' is your best choice. It allocates large(few GiB) files almost instantly.

                              Don't use 'falloc' with legacy file systems such as ext3 and FAT32 because it

                              takes almost the same time as 'prealloc' and it blocks aria2 entirely until allocation finishes.

                              'falloc' may not be available if your system doesn't have posix\_fallocate() function.

                              'trunc' uses ftruncate() system call or platform-specific counterpart to truncate a file

                              to a specified length.

                              Possible Values: none, prealloc, trunc, falloc

                              Default: prealloc

                              Tags: #basic, #file

 -V, --check-integrity[=true|false] Check file integrity by validating piece hashes or a hash of entire file. This option has effect

only in BitTorrent, Metalink downloads with checksums or HTTP(S)/FTP downloads with --checksum option.

                              If piece hashes are provided, this option can detect damaged portions of a file and re-download them.

                              If a hash of entire file is provided, hash check is only done when file has been already download.

                              This is determined by file length. If hash check fails, file is re-downloaded from scratch.

                              If both piece hashes and a hash of entire file are provided, only piece hashes are used.

                              Possible Values: true, false

                              Default: false

                              Tags: #basic, #metalink, #bittorrent, #file, #checksum

 -c, --continue[=true|false]  Continue downloading a partially downloaded file.

                              Use this option to resume a download started by a web browser or another program

                              which downloads files sequentially from the beginning. Currently this option is only

                              applicable to http(s)/ftp downloads.

                              Possible Values: true, false

                              Default: false

                              Tags: #basic, #http, #ftp

 -i, --input-file=FILE        Downloads URIs found in FILE. You can specify multiple URIs for a single entity: separate

                              URIs on a single line using the TAB character.

                              Reads input from stdin when '-' is specified. Additionally, options can be specified after each

                              line of URI. This optional line must start with one or more white spaces and have one option per

                              single line. See INPUT FILE section of man page for details. See also --deferred-input option.

                              Possible Values: /path/to/file, -

                              Tags: #basic

 -j, --max-concurrent-downloads=N       Set maximum number of parallel downloads for every static (HTTP/FTP) URL, torrent and metalink.

                                        See also --split and --optimize-concurrent-downloads options.

                                        Possible Values: 1-\*

                                        Default: 5

                                        Tags: #basic

 -Z, --force-sequential[=true|false]    Fetch URIs in the command-line sequentially and download each URI in a separate session, like

                                        the usual command-line download utilities.

                                        Possible Values: true, false

                                        Default: false

                                        Tags: #basic

 -x, --max-connection-per-server=NUM    The maximum number of connections to one server for each download.

                                        Possible Values: 1-16

                                        Default: 1

                                        Tags: #basic, #http, #ftp

 -k, --min-split-size=SIZE    aria2 does not split less than 2\*SIZE byte range.

                              For example, let's consider downloading 20MiB file. If SIZE is 10M, aria2 can split file into 2

                              range [0-10MiB) and [10MiB-20MiB) and download it using 2 sources(if --split >= 2, of course).

                              If SIZE is 15M, since 2\*15M > 20MiB, aria2 does not split file and download it using 1 source.

                              You can append K or M(1K = 1024, 1M = 1024K).

                              Possible Values: 1048576-1073741824

                              Default: 20M

                              Tags: #basic, #http, #ftp

 --ftp-user=USER              Set FTP user. This affects all URLs.

                              Tags: #basic, #ftp

 --ftp-passwd=PASSWD          Set FTP password. This affects all URLs.

                              Tags: #basic, #ftp

 --http-user=USER             Set HTTP user. This affects all URLs.

                              Tags: #basic, #http

 --http-passwd=PASSWD         Set HTTP password. This affects all URLs.

                              Tags: #basic, #http

 --load-cookies=FILE          Load Cookies from FILE using the Firefox3 format and Mozilla/Firefox(1.x/2.x)/Netscape format.

                              Possible Values: /path/to/file

                              Tags: #basic, #http, #cookie

 -S, --show-files[=true|false]   Print file listing of .torrent, .meta4 and .metalink file and exit. More detailed

                                information will be listed in case of torrent file.

                                Possible Values: true, false

                                Default: false

                                Tags: #basic, #metalink, #bittorrent

 --max-overall-upload-limit=SPEED   Set max overall upload speed in bytes/sec. 0 means unrestricted.

                                    You can append K or M(1K = 1024, 1M = 1024K).

                                    To limit the upload speed per torrent, use --max-upload-limit option.

                                    Possible Values: 0-\*

                                    Default: 0

                                    Tags: #basic, #bittorrent

 -u, --max-upload-limit=SPEED       Set max upload speed per each torrent in bytes/sec. 0 means unrestricted.

                                    You can append K or M(1K = 1024, 1M = 1024K).

                                    To limit the overall upload speed, use --max-overall-upload-limit option.

                                    Possible Values: 0-\*

                                    Default: 0

                                    Tags: #basic, #bittorrent

 -T, --torrent-file=TORRENT\_FILE    The path to the .torrent file.

                                    Possible Values: /path/to/file

                                    Tags: #basic, #bittorrent

 --listen-port=PORT...              Set TCP port number for BitTorrent downloads.

                                    Multiple ports can be specified by using ',', for example: "6881,6885". You can also use '-'

                                    to specify a range: "6881-6999". ',' and '-' can be used together.

                                    Possible Values: 1024-65535

                                    Default: 6881-6999

                                    Tags: #basic, #bittorrent

 --enable-dht[=true|false]    Enable IPv4 DHT functionality. It also enables UDP tracker support. If a private flag is set

                              in a torrent, aria2 doesn't use DHT for that download even if ``true`` is given.

                              Possible Values: true, false

                              Default: true

                              Tags: #basic, #bittorrent

 --dht-listen-port=PORT...    Set UDP listening port used by DHT(IPv4, IPv6) and UDP tracker. Multiple ports can be specified

                              by using ',', for example: "6881,6885". You can also use '-' to specify a range: "6881-6999".

                              ',' and '-' can be used together.

                              Possible Values: 1024-65535

                              Default: 6881-6999

                              Tags: #basic, #bittorrent

 --enable-dht6[=true|false]   Enable IPv6 DHT functionality.

                              Use --dht-listen-port option to specify port number to listen on. See also --dht-listen-addr6 option.

                              Possible Values: true, false

                              Default: false

                              Tags: #basic, #bittorrent

 --dht-listen-addr6=ADDR      Specify address to bind socket for IPv6 DHT. It should be a global unicast IPv6 address of the host.

                              Tags: #basic, #bittorrent

 -M, --metalink-file=METALINK\_FILE  The file path to the .meta4 and .metalink file. Reads input from stdin when '-' is specified.

                                    Possible Values: /path/to/file, -

                                    Tags: #basic, #metalink

URI, MAGNET, TORRENT\_FILE, METALINK\_FILE:

 You can specify multiple HTTP(S)/FTP URIs. Unless you specify -Z option, all URIs must point to the same file or downloading will fail.

 You can also specify arbitrary number of BitTorrent Magnet URIs, torrent/metalink files stored in a local drive.

Please note that they are always treated as a separate download.

 You can specify both torrent file with -T option and URIs. By doing this, download a file from both torrent swarm and HTTP/FTP server

at the same time, while the data from HTTP/FTP are uploaded to the torrent swarm.

For single file torrents, URI can be a complete URI pointing to the resource or if URI ends with '/', 'name' in torrent file is added.

For multi-file torrents, 'name' and 'path' in torrent are added to form a URI for each file.

 Make sure that URI is quoted with single(') or double(") quotation if it contains "&" or any characters that have special meaning in shell.

About the number of connections

 Since 1.10.0 release, aria2 uses 1 connection per host by default and has 20MiB segment size restriction.

So whatever value you specify using -s option, it uses 1 connection per host. To make it behave like 1.9.x, use

 --max-connection-per-server=4 --min-split-size=1M.

Refer to man page for more information.

Refs:

* <https://linuxconfig.org/aria2-all-in-one-command-line-download-tool>
* <https://calomel.org/aria2.html>
* <https://superuser.com/a/1149786>

### Tips

* **Maximum speed**

# download using 16 simultaneous connections

$ aria2c -s16 -x16 http://abc.exe

Note: Some servers suddenly choke the transfer speed drastically, especially if several connections are made. Given the diversity of the server domain, it is sometime faster and sometimes slower. aria2 just checks socket is readable, and if it is, aria2 reads it. So, it's **absolutely normal if you observe fewer simultaneous connections than expected**.

## Installation

### Linux

On Ubuntu, Debian, and Linux Mint:

$ sudo apt install aria2

On CentOS, Fedora, AlmaLinux, and Red Hat:

$ sudo dnf install aria2

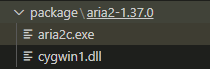
On Arch Linux and Manjaro:

$ sudo pacman -S aria2

### Windows

Guide: <https://www.tutorialexample.com/install-aria2-on-win10-to-download-files-a-beginner-guide/>

My experience: This is enough to run aria2 in Windows



## UI Client

Aria2 Web UI: <https://i12bretro.wordpress.com/category/aria2-web-ui/>