

wget2compile Vagrant virtual machine documentation

Release 0.1 - 2022-07-11T18:50:13+02:00

DirCom (MOY 1666, CNRS)

Jul 11, 2022

1	Project's overview	3
2	<i>wget2</i>: new 2.x branch of <i>wget</i> development	5
3	<i>wget2</i> compilation for <i>Linux x86_64</i>	7
4	Vagrant virtual <i>Linux x86_64</i> machine for <i>wget2</i> compilation	9
4.1	Virtual machine starting and provisionning	9
4.2	Result of compilation	9
4.3	Partial provisionning	9
5	<i>wget</i> and <i>wget2</i> command-line options	11
5.1	Already used options in v1	11
5.2	Useful yet unused v1 options	11
5.3	Differences between v1 and v2	12
6	Other sections	13
6.1	Annexes	13
6.2	ToDo-list	19
6.3	Changelog	20

See also

README in project's root!

README.md file standing at project's root directory may contain additional useful pieces of information, but its content is not included hereafter.

Present doc. version

0.1+ (0.1 - 2022-07-11T18:50:13+02:00, *next*)

Authors *DirCom* (MOY 1666, [CNRS](#))

Project's overview

At [DirCom](#), we have numerous old websites to archive and/or turn into only static files (greatly simplifying their hosting stack and reducing their inherent security wholes). We started using *wget* for such statication.

wget2 is the new fresh branch of development of *wget*, and current project's goal is to test if its new features could help us better than 1.x old branch.

Usually, such foundational tool is provided by your *Linux* distribution, but *Debian*-based ones have not yet integrated this new branch. It turns out that, at least at the time of writing of this documentation, we must compile project's sources ourselves to get a fully usable binary on these platforms.

Hence usage of a virtual machine for hosting all needed material for this compilation, and automate its steps in some reproducible way. Using such automatically provisioned virtual machine also allows us to not pollute our work environment, and easy switch between *Linux*-based and *Windows-with-WSL*-based workstations.

wget2: new 2.x branch of wget development

GNU wget project page

<https://www.gnu.org/software/wget/>

wget2 is the new fresh branch of development of *wget*, whose development is done on *GitLab.com*:
<https://gitlab.com/gnuwget/wget2>

First non-beta release was published as *v2.0.0* on September 2021: <https://gitlab.com/gnuwget/wget2/-/tags>

New features are listed in both project's wiki: <https://gitlab.com/gnuwget/wget2/-/wikis/home>
and usual *NEWS* file: <https://gitlab.com/gnuwget/wget2/-/blob/master/NEWS>

See also

Chapter 5 for listing of respective CLI options of *wget* and *wget2*.

wget2 compilation for *Linux x86_64*

Note *Vagrant* virtual machine *provisioning* already contains *wget2* compilation as its final stage. See [Chapter 4](#).

wget2 compilation could be achieved through following steps:

1. install required compilation tools, such as *cc*, *ld*, *GNU make*, etc.;

See also

Vagrant provisioning scripts.

For details on packages, libraries and so on needed to provide required environment for *wget2* compilation on *Debian* stable, see *Vagrant* virtual machine provisioning scripts, in project's directory `vagrant/bash/`.

2. clone *wget2* git repository:

```
# Clone repo. in eg. 'wget2-git-repo' dir.
$ git clone https://gitlab.com/gnuwget/wget2.git wget2-git-repo
# Put git working directory to latest tag
$ cd wget2-git-repo
$ git checkout "v2.0.0"
```

3. bootstrap its development environment, through *wget2* sources bootstrap script:

```
$ cd wget2-git-repo
$ ./bootstrap
```

4. classic `./configure && make && make install` for C project:

```
$ cd wget2-git-repo
# Set static libraries linking and install path, eg. in $HOME/wget2 dir.
$ LDFLAGS="-static" ./configure --prefix=$HOME/wget2
$ make
$ make install
# Result of compilation and installation now stands in $HOME/wget2
$ ls $HOME/wget2
bin include lib share
$ ls $HOME/wget2/bin
wget2 wget2_noinstall
```

Vagrant virtual *Linux x86_64* machine for *wget2* compilation

Present project aims to provide a virtual machine, based on *Debian x86_64*, for compiling *wget2* in some single fat binary with all its libraries statically linked into. This virtual machine is managed by *Vagrant*, through its *Vagrantfile*, and provisionned with several *Bash* scripts standing in `vagrant/bash/` directory. These provisioning scripts roughly follow [Chapter 3](#).

4.1 Virtual machine starting and provisioning

If you just start using present project and have not yet started *Vagrant* virtual machine, the only command you need to run in project's root directory is:

```
# Start virtual machine and provision it, including wget2 compilation steps
$ vagrant up
```

Else, if you already started and/or provision the virtual machine, you always could re-run its full provisioning (which is idempotent) through one of following commands:

- if virtual machine is currently stopped:

```
# Restart virtual machine and re-provision it, including wget2 compilation
steps
$ vagrant up --provision
```

- else if virtual machine is currently running:

```
# Re-provision virtual machine, including wget2 compilation steps
$ vagrant provision
```

4.2 Result of compilation

Result of *wget2* compilation is shared by guest to host in project's `shared/` subdirectory.

4.3 Partial provisioning

Project's *Vagrantfile* and *Bash* scripts behavior allow applying only some given specific provisioning step, thanks `--provision-with=<step>` CLI option (for notably `up` and `provision` subcommands). These partial provisioning steps are:

Num	Step	Description
1	<code>init</code>	Distrib. packages upgrade, facilities and Python installations

2	libdeps	wget2 development librairies installation
3	wget	wget2 repo. clone, checkout, env. bootstrap, compilation and install

For example, to re-run last step (wget) on some already running virtual machine:

```
$ vagrant provision --provision-with=wget
```

wget and *wget2* command-line options

See also

- [wget \(v1\) options](#)
- [wget2 \(v2\) options](#)

5.1 Already used options in v1

- `--adjust-extension`
- `--convert-links`
- `--domains=${DOMAINS}`
- `--level=${DEPTH}`
- `--load-cookies=${COOKIES_FILEPATH}`
- `--no-host-directories`
- `--no-parent`
- `--no-remove-listing`
- `--page-requisites`
- `--random-wait`
- `--recursive`
- `--save-cookies=${COOKIES_FILEPATH}`
- `--timestamping`
- `--user-agent=Mozilla`
- `--wait=${WAIT}`

5.2 Useful yet unused v1 options

- `--base=${BASE_URL}`
- `--cut-dirs=${N}`

5.3 Differences between v1 and v2

Option	v1	v2
--bind-interface	NA	X
--cache	NA	X
--check-certificate	NA	X
--check-hostname	NA	X
--chunk-size	NA	X
--ciphers	X	NA
--clobber	NA	X
--cookies	NA	X
--cookies-suffix	NA	X
--cut-file-get-vars	NA	X
--cut-url-get-vars	NA	X
--default-http-port	NA	X
--default-https-port	NA	X
--directories	NA	X
--dns-cache	NA	X
--dns-cache-preload	NA	X
--download-attr	NA	X

Todo

Investigate following v2 options:

- --download-attr
-

Other sections

6.1 Annexes

6.1.1 *wget* and *wget2* command-line options

Following command-line options listings could be roughly obtained by running this shell command:

```
# For wget2, replace with:
#     WGET=./shared/wget2/bin/wget2
# Prefix its call to set LANG env. to C, defaulting help message in English.
LANG=C WGET=wget $WGET --help \
|grep -P -e
"^[[:blank:]]+-((([:alpha:]]),|[:blank:]]+)?)(-+[:alpha:]]{2,}[[:blank:]]+)?.*"
\
|uniq \
|sort
```

wget (v1) options

-4,	--inet4-only	connect only to IPv4 addresses
-6,	--inet6-only	connect only to IPv6 addresses
-A,	--accept=LIST	comma-separated list of accepted extensions
-a,	--append-output=FILE	append messages to FILE
	--accept-regex=REGEX	regex matching accepted URLs
	--ask-password	prompt for passwords
	--auth-no-challenge	send Basic HTTP authentication information
	--backups=N	before writing file X, rotate up to N backup
files		
-b,	--background	go to background after startup
-B,	--base=URL	resolves HTML input-file links (-i -F)
	--bind-address=ADDRESS	bind to ADDRESS (hostname or IP) on local host
	--body-data=STRING	send STRING as data. --method MUST be set
	--body-file=FILE	send contents of FILE. --method MUST be set
	--ca-certificate=FILE	file with the bundle of CAs
	--ca-directory=DIR	directory where hash list of CAs is stored
-c,	--continue	resume getting a partially-downloaded file
	--certificate=FILE	client certificate file
	--certificate-type=TYPE	client certificate type, PEM or DER
	--ciphers=STR	Set the priority string (GnuTLS) or cipher list
	string (OpenSSL) directly.	
	--compression=TYPE	choose compression, one of auto, gzip and none.
	(default: none)	
	--config=FILE	specify config file to use
	--connect-timeout=SECS	set the connect timeout to SECS
	--content-disposition	honor the Content-Disposition header when

```

--content-on-error      output the received content on server errors
--convert-file-only    convert the file part of the URLs only (usually
known as the basename)
--crl-file=FILE        file with bundle of CRLs
--cut-dirs=NUMBER      ignore NUMBER remote directory components
-d, --debug            print lots of debugging information
-D, --domains=LIST     comma-separated list of accepted domains
--default-page=NAME    change the default page name (normally
--delete-after         delete files locally after downloading them
--dns-timeout=SECS    set the DNS lookup timeout to SECS
-E, --adjust-extension save HTML/CSS documents with proper extensions
-e, --execute=COMMAND  execute a `.wgetrc'-style command
--exclude-domains=LIST comma-separated list of rejected domains
-F, --force-html       treat input file as HTML
--follow-ftp           follow FTP links from HTML documents
--follow-tags=LIST     comma-separated list of followed HTML tags
--ftp-password=PASS    set ftp password to PASS
--ftps-clear-data-connection
will be in plaintext  cipher the control channel only; all the data
--ftps-fallback-to-ftp
the target server     fall back to FTP if FTPS is not supported in
--ftps-implicit        use implicit FTPS (default port is 990)
--ftps-resume-ssl      resume the SSL/TLS session started in
the control connection when
--ftp-user=USER        set ftp user to USER
--header=STRING        insert STRING among the headers
-h, --help            print this help
-H, --span-hosts      go to foreign hosts when recursive
--hsts-file            path of HSTS database (will override default)
--http-password=PASS   set http password to PASS
--https-only           only follow secure HTTPS links
--http-user=USER       set http user to USER
--ignore-case          ignore case when matching files/directories
--ignore-length        ignore 'Content-Length' header field
--ignore-tags=LIST     comma-separated list of ignored HTML tags
-I, --include-directories=LIST
list of allowed directories
-i, --input-file=FILE  download URLs found in local or external FILE
-K, --backup-converted
before converting file X, back up as X.orig
-k, --convert-links    make links in downloaded HTML or CSS point to
--keep-session-cookies
load and save session (non-permanent) cookies
--limit-rate=RATE     limit download rate to RATE
-l, --level=NUMBER     maximum recursion depth (inf or 0 for infinite)
--load-cookies=FILE   load cookies from FILE before session
--local-encoding=ENC  use ENC as the local encoding for IRIs
-L, --relative         follow relative links only
--max-redirect         maximum redirections allowed per page
--method=HTTPMethod   use method "HTTPMethod" in the request
-m, --mirror           shortcut for -N -r -l inf --no-remove-listing
-nc, --no-clobber      skip downloads that would download to
-nd, --no-directories  don't create directories
-nH, --no-host-directories
don't create host directories
--no-cache             disallow server-cached data
--no-check-certificate
don't validate the server's certificate
--no-config            do not read any config file
--no-cookies           don't use cookies
--no-dns-cache         disable caching DNS lookups
--no-glob              turn off FTP file name globbing
--no-hsts              disable HSTS
--no-http-keep-alive  disable HTTP keep-alive (persistent connections)
--no-if-modified-since
don't use conditional if-modified-since get
--no-iri              turn off IRI support
--no-netrc             don't try to obtain credentials from .netrc
--no-passive-ftp      disable the "passive" transfer mode
--no-proxy             explicitly turn off proxy
    
```

```

--no-remove-listing      don't remove '.listing' files
--no-use-server-timestamps  don't set the local file's timestamp by
--no-warc-compression    do not compress WARC files with GZIP
--no-warc-digests       do not calculate SHA1 digests
--no-warc-keep-log       do not store the log file in a WARC record
-np, --no-parent         don't ascend to the parent directory
-N, --timestamping       don't re-retrieve files unless newer than
-nv, --no-verbose        turn off verbosity, without being quiet
-O, --output-document=FILE  write documents to FILE
-o, --output-file=FILE    log messages to FILE
--password=PASS          set both ftp and http password to PASS
-P, --directory-prefix=PREFIX  save files to PREFIX/..
--pinnedpubkey=FILE/HASHES  Public key (PEM/DER) file, or any number
--post-data=STRING       use the POST method; send STRING as the data
--post-file=FILE         use the POST method; send contents of FILE
-p, --page-requisites    get all images, etc. needed to display HTML page
--prefer-family=FAMILY   connect first to addresses of specified family,
--preserve-permissions   preserve remote file permissions
--private-key=FILE       private key file
--private-key-type=TYPE  private key type, PEM or DER
--progress=TYPE          select progress gauge type
--protocol-directories   use protocol name in directories
--proxy-password=PASS    set PASS as proxy password
--proxy-user=USER        set USER as proxy username
-q, --quiet              quiet (no output)
-Q, --quota=NUMBER       set retrieval quota to NUMBER
--random-wait            wait from 0.5*WAIT...1.5*WAIT secs between
retrievals
--read-timeout=SECS      set the read timeout to SECS
--referer=URL            include 'Referer: URL' header in HTTP request
--regex-type=TYPE        regex type (posix|pcre)
--rejected-log=FILE      log reasons for URL rejection to FILE
--reject-regex=REGEX     regex matching rejected URLs
--remote-encoding=ENC    use ENC as the default remote encoding
--report-speed=TYPE      output bandwidth as TYPE. TYPE can be bits
--restrict-file-names=OS restrict chars in file names to ones OS allows
--retr-symlinks          when recursing, get linked-to files (not dir)
--retry-connrefused      retry even if connection is refused
--retry-on-http-error=ERRORS  comma-separated list of HTTP errors to retry
-r, --recursive          specify recursive download
-R, --reject=LIST         comma-separated list of rejected extensions
--save-cookies=FILE      save cookies to FILE after session
--save-headers           save the HTTP headers to file
--secure-protocol=PR     choose secure protocol, one of auto, SSLv2,
--show-progress          display the progress bar in any verbosity mode
--spider                 don't download anything
-S, --server-response    print server response
--start-pos=OFFSET       start downloading from zero-based position OFFSET
--strict-comments        turn on strict (SGML) handling of HTML comments
--trust-server-names     use the name specified by the redirection
-T, --timeout=SECONDS    set all timeout values to SECONDS
-t, --tries=NUMBER       set number of retries to NUMBER (0 unlimits)
--unlink                 remove file before clobber
--use-askpass=COMMAND    specify credential handler for requesting
--user=USER              set both ftp and http user to USER
-U, --user-agent=AGENT   identify as AGENT instead of Wget/VERSION
-v, --verbose            be verbose (this is the default)
-V, --version            display the version of Wget and exit
--waitretry=SECONDS      wait 1..SECONDS between retries of a retrieval
--warc-cdx               write CDX index files
--warc-dedup=FILENAME    do not store records listed in this CDX file
--warc-file=FILENAME     save request/response data to a .warc.gz file
--warc-header=STRING     insert STRING into the warcinfo record
--warc-max-size=NUMBER   set maximum size of WARC files to NUMBER

```

```

--warc-tempdir=DIRECTORY location for temporary files created by the
-w, --wait=SECONDS      wait SECONDS between retrievals
--xattr                 turn on storage of metadata in extended file
attributes
-X, --exclude-directories=LIST list of excluded directories
-x, --force-directories force creation of directories
    
```

wget2 (v2) options

```

-4 --inet4-only          Use IPv4 connections only. (default: off)
-6 --inet6-only          Use IPv6 connections only. (default: off)
-A --accept              Comma-separated list of file name suffixes or
-a --append-output       File where messages are appended to, '-' for STDOUT
--accept-regex           Regex matching accepted URLs.
--ask-password           Print prompt for password
--auth-no-challenge      send Basic HTTP Authentication before challenge
--backups                Make backups instead of overwriting/increasing
-b --background          Go to background immediately after startup. If no
-B --base                Base for relative URLs read from input-file
--bind-address           Bind to sockets to local address.
--bind-interface         Bind sockets to the input Network Interface.
--body-data              Data to be sent in a request.
--body-file              File with data to be sent in a request.
--ca-certificate          File with bundle of PEM CA certificates.
--cache                  Enabled using of server cache. (default: on)
--ca-directory           Directory with PEM CA certificates.
-c --continue            Continue download for given files. (default: off)
--certificate            File with client certificate.
--certificate-type        Certificate type: PEM or DER (known as ASN1).
--check-certificate       Check the server's certificate. (default: on)
--check-hostname        Check the server's certificate's hostname.
--chunk-size             Download large files in multithreaded chunks.
--clobber                Enable file clobbering. (default: on)
--compression            Customize Accept-Encoding with
--config                 Path to initialization file (default:
~/config/wget/wget2rc)
--connect-timeout        Connect timeout in seconds.
--content-disposition    Save response body even on error status.
--content-on-error       Convert only filename part of embedded URLs.
--convert-file-only      Enable use of cookies. (default: on)
--cookies                Load public suffixes from file.
--cookie-suffixes        File with PEM CRL certificates.
--crl-file               Skip creating given number of directory
--cut-dirs                Cut HTTP GET vars from file names. (default: off)
--cut-file-get-vars      Cut HTTP GET vars from URLs. (default: off)
--cut-url-get-vars
-d --debug                Print debugging messages.(default: off)
-D --domains             Comma-separated list of domains to follow.
--default-http-port      Set default port for HTTP. (default: 80)
--default-https-port     Set default port for HTTPS. (default: 443)
--default-page           Default file name. (default: index.html)
--delete-after           Don't save downloaded files. (default: off)
--directories             Create hierarchy of directories when retrieving
--dns-cache              Caching of domain name lookups. (default: on)
--dns-cache-preload      File to be used to preload the DNS cache.
--dns-timeout            DNS lookup timeout in seconds.
--download-attr          Recognize HTML5 download attributes.
-E --adjust-extension    Append extension to saved file (.html or .css).
-e --execute             Wget compatibility option, not needed for Wget
--egd-file               File to be used as socket for random data from
--exclude-domains        Comma-separated list of domains NOT to follow.
-F --force-html          Treat input file as HTML. (default: off)
--filter-mime-type       Specify a list of mime types to be saved or ignored
    
```

```

--filter-urls      Apply the accept and reject filters on the URL
--follow-tags      Scan additional tag/attributes for URLs,
--force-atom       Treat input file as Atom Feed. (default: off) (NEW!)
--force-css        Treat input file as CSS. (default: off) (NEW!)
--force-metalink   Treat input file as Metalink. (default: off) (NEW!)
--force-progress   Force progress bar.
--force-rss        Treat input file as RSS Feed. (default: off) (NEW!)
--force-sitemap    Treat input file as Sitemap. (default: off) (NEW!)
--fsync-policy     Use fsync() to wait for data being written to
--gnupg-homedir    Specify a directory to use as the GnuPG home
--header           Insert input string as a HTTP header in
-h --help          Print this help.
--host-directories Create host directories when retrieving
--hpkp-file        Set file for storing HPKP data
--hpkp             Use HTTP Public Key Pinning (HPKP). (default: on)
-H --span-hosts    Span hosts that were not given on the
--hsts-file        Set file for HSTS caching. (default: ~/.wget-hsts)
--hsts-preload-file Set name for the HSTS Preload file (DAFSA format).
--hsts-preload     Use HTTP Strict Transport Security (HSTS).
--hsts             Use HTTP Strict Transport Security (HSTS).
--html-extension   Obsoleted by --adjust-extension
--http2-only       Only use HTTP/2 protocol, error if server doesn't offer
it. (default: off)
--http2-request-window
--http2            Use HTTP/2 protocol if possible. (default: on)
--http-keep-alive  Keep connection open for further requests.
--http-password    Password for HTTP Authentication.
--http-proxy-password
--http-proxy       Set HTTP proxy/proxies, overriding environment
--http-proxy-user  Username for HTTP Proxy Authentication.
--https-enforce    Use secure HTTPS instead of HTTP. Legal types are
--https-only       Do not follow non-secure URLs. (default: off).
--https-proxy      Set HTTPS proxy/proxies, overriding environment
--http-user        Username for HTTP Authentication.
--hyperlink        Enable terminal hyperlink support
--if-modified-since Do not send If-Modified-Since header in -N mode.
--ignore-case      Ignore case when matching files. (default: off)
--ignore-length    Ignore content-length header field
--ignore-tags      Ignore tag/attributes for URL scanning,
-I --include-directories
-i --input-file     File where URLs are read from, - for STDIN.
--input-encoding   Character encoding of the file contents read with
--input-file. (default: local encoding)
--iri              Wget dummy option, you can't switch off
-K --backup-converted When converting, keep the original file with
-k --convert-links  Convert embedded URLs to local URLs.
--keep-extension   If file exists: Use pattern 'basename_N.ext'
--keep-session-cookies
--limit-rate       Limit rate of download per second, 0 = no limit.
(default: 0)
--list-plugins     Lists all the plugins in the plugin search paths.
-l --level          Maximum recursion depth. (default: 5)
--load-cookies     Load cookies from file.
--local-db         Read or load databases
--local-encoding   Character encoding of environment and filenames.
--local-plugin     Loads a plugin with a given path.
--max-redirect     Max. number of redirections to follow.
--max-threads      Max. concurrent download threads.
--metalink         Follow a metalink file instead of storing it
--method           HTTP method to use for request.
-m --mirror         Turn on mirroring options -r -N -l inf
--netrc-file       Set file for login/password to use instead of
--netrc            Load credentials from ~/.netrc if not given.
-N --timestamping  Just retrieve younger files than the local ones.

```

```

--ocsp-date          Check if OCSP response date is too old.
--ocsp-file          Set file for OCSP chaching.
--ocsp-nonce         Allow nonce checking when verifying OCSP
--ocsp-server        Set OCSP server address.
--ocsp-stapling      Use OCSP stapling to verify the server's
--ocsp               Use OCSP server access to verify server's
-O --output-document File where downloaded content is written to,
-o --output-file     File where messages are printed to,
--parent             Ascend above parent directory. (default: on)
--password           Password for Authentication.
-P --directory-prefix Set directory prefix.
--plugin-dirs        Specify alternative directories to look
--plugin-help        Print help message for all loaded plugins
--plugin             Load a plugin with a given name.
--plugin-opt         Forward an option to a loaded plugin.
--post-data          Data to be sent in a POST request.
--post-file          File with data to be sent in a POST request.
-p --page-requisites Download all necessary files to display a
--prefer-family      Prefer IPv4 or IPv6. (default: none)
--private-key        File with private key.
--private-key-type   Type of the private key (PEM or DER).
--progress           Type of progress bar (bar, none).
--protocol-directories
--proxy              Enable support for *_proxy environment variables.
-q --quiet           Print no messages except debugging messages.
-Q --quota           Download quota, 0 = no quota. (default: 0)
--random-file        File to be used as source of random data.
--random-wait        Wait 0.5 up to 1.5*  
--wait> seconds between
--read-timeout       Read and write timeout in seconds.
--referer            Include Referer: url in HTTP request.
--regex-type         Regular expression type. Possible types are
--reject-regex       Regex matching rejected URLs.
--remote-encoding    Character encoding of remote files
--report-speed        Output bandwidth as TYPE. TYPE can be bytes
--restrict-file-names
--retry-connrefused Consider "connection refused" a transient error.
--retry-on-http-error
--robots             Respect robots.txt standard for recursive
-r --recursive       Recursive download. (default: off)
-R --reject          Comma-separated list of file name suffixes or
--save-content-on    Specify a list of response codes that requires it's
--save-cookies       Save cookies to file.
--save-headers       Save the response headers in front of the response
--secure-protocol    Set protocol to be used (auto, SSLv3, TLSv1, TLSv1_1,
TLSv1_2, TLS1_3, PFS).
--signature-extensions
--spider             Enable web spider mode. (default: off)
-S --server-response Print the server response headers. (default: off)
--start-pos          Start downloading at zero-based position, 0 = option
disabled. (default: 0)
--stats-dns=[FORMAT:]FILE
--stats-dns          Print DNS stats. (default: off)
--stats-ocsp=[FORMAT:]FILE
--stats-ocsp         Print OCSP stats. (default: off)
--stats-server=[FORMAT:]FILE
--stats-server       Print server stats. (default: off)
--stats-site=[FORMAT:]FILE
--stats-site         Print site stats. (default: off)
--stats-tls=[FORMAT:]FILE
--stats-tls          Print TLS stats. (default: off)
--strict-comments    A dummy option. Parsing always works non-strict.
-s --verify-sig      Download .sig file and verify. (default: off)
--tcp-fastopen       Enable TCP Fast Open (TFO). (default: on)
--tls-false-start    Enable TLS False Start (needs GnuTLS 3.5+).
    
```

--tls-resume	Enable TLS Session Resumption. (default: off)
--tls-session-file	Set file for TLS Session caching.
--trust-server-names	On redirection use the server's filename.
-T --timeout	General network timeout in seconds.
-t --tries	Number of tries for each download. (default 20)
--unlink	Remove files before clobbering. (default: off)
--use-askpass	Prompt for a user and password using
--user	Username for Authentication.
--use-server-timestamps	
-U --user-agent	HTTP User Agent string.
--verify-save-failed	Save target files even when their signatures fail
-v --verbose	Print more messages. (default: on)
-V --version	Display the version of Wget and exit.
--waitretry	Wait up to number of seconds after error
-w --wait	Wait number of seconds between downloads
--xattr	Save extended file attributes. (default: off)
-X --exclude-directories	
-x --force-directories	Create hierarchy of directories when not

6.2 ToDo-list

- document (for usage) new wget2 CLI options, eg. `--keep-extension`
See:
 - project's wiki on GitLab: <https://gitlab.com/gnuwget/wget2/-/wikis/home>
 - announce on project's mailing-list: <https://lists.gnu.org/archive/html/info-gnu/2021-09/msg00011.html>
 - changelog: <https://gitlab.com/gnuwget/wget2/-/blob/42204ee9c906e5afe39a2b5581af0b4e06fb1460/NEWS>
 - discussion on HackerNews about 1.99.2: <https://news.ycombinator.com/item?id=20895226>
 - Phoronix: https://www.phoronix.com/scan.php?page=news_item&px=GNU-Wget2-2.0-Released
- enable other interesting compilation options, currently unavailable (see output of `./shared/wget2/bin/wget2 --version`):
`-iconv -ntlm -opie -hsts`

6.2.1 Automatic recall of *todo* items included in documentation

Todo

Investigate following v2 options:

- `--download-attr`

(The *original entry* is located in `/home/work/boulot/DIRCOM/PROJETS/wget2/compil/work/doc/src/options.rst`, line 66.)

6.3 Changelog

6.3.1 *next* (in development)

- Add following compression algorithms support:
 - *LZMA* (+lзма feature)
 - *lzip* (+lzip feature)
 - *bzip2* (+bzip2 feature)
- Compilation environment:
 - add new *python* executable in `$PATH`, as symbolic link to *python3*, to better allow some *wget2* libraries to be compiled and installed
 - install *libiconv* in `/usr/local/` instead of its own dedicated directory
 - set more compilation options in Bash constants available into `vagrant/bash/parts/wget2/const.sh`
 - clone *wget2* git repository in `/opt` instead of `/var/local`, like *libiconv*
- Documentation:
 - upgrade *Sphinx* to latest 5.0.* version and *rinohype* to 0.5.4
 - include CHANGELOG in rendered documents
 - include current datetime in release version (matching seconds precision)
 - move PDF document from project's `doc/rinoh/` directory to `doc/pdf/`

6.3.2 0.1 (2022-06-27)

- Initial release, compiling *wget2* at v2.0.1 for *Linux x86_64* with following features (see `wget2 --version` output):

```
+digest +https +ssl/gnutls +ipv6 +iri +large-file +nls -ntlm -opie +psl -hsts  
+iconv +idn2 +zlib -lzma +brotldec +zstd -bzip2 -lzip +http2 +gpgme
```

- genindex
- search

