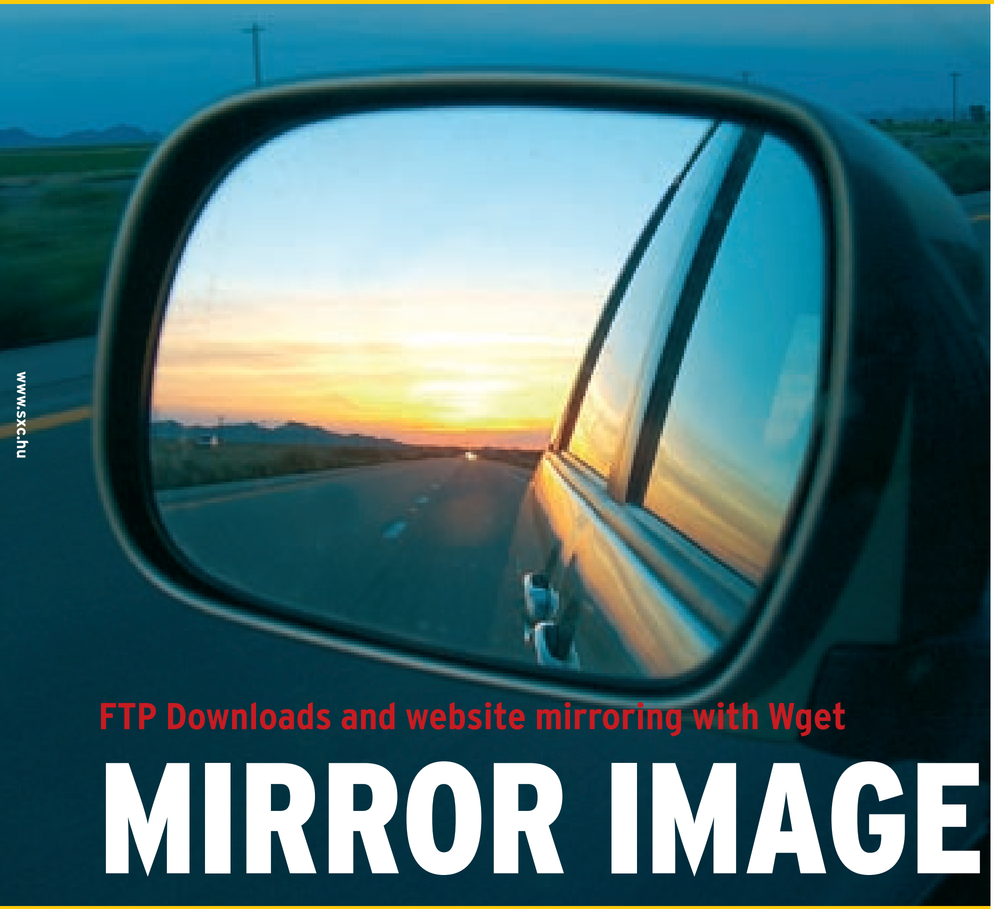


FTP Downloads and website mirroring with Wget

MIRROR IMAGE

Wget downloads files and even whole websites from the command line. BY HEIKE JURZIK

Any number of GUI-based download managers allow users to download files and whole websites. At the command line, you'll need a tool like Wget. Wget handles downloads quickly and without a lot of pointing and clicking. Wget "speaks" HTTP, HTTPS, and FTP; it can continue interrupted transfers, and it even has an update function that only updates files that have changed.

All-Rounder

The generic syntax for Wget is as follows:

```
wget URL
```

Wget gives you command line output to let you know what it is doing (Figure 1): in our example, the tool is establishing a connection to a web server (standard port 80) and downloading the *index.html* file to a local directory, ignoring embedded images and not following links. If you do not want to view the fairly verbose output at the console, you might like to specify the *-q* (for quiet) option. As this tells Wget to suppress the output of error messages and basic information, however, you might prefer a compromise, which you can achieve by entering *wget -nv*. This option tells the program to write less output to your console but still provide some information.

To tell Wget to follow local links on the server and mirror the data recursively, just add the *-r* parameter. It makes sense to specify the recursion depth while doing so. You need to go down one level to get both *index.html* and all embedded links (such as images or other HTML pages):

```
wget -r -l 1 ↩
www.linux-magazine.com
```

If you set the level depth to *-l 2*, Wget will dig deeper by one level. In other words, if *index.html* contains a link to *images.html*, the download manager will now follow the links on this page.

A folder on the local hard disk is created for each URL, but you can change this behavior by adding another option. You can specify *-nH* ("no host") to store all the results in the current directory.

Wget can modify the links in individual HTML files. For example, if you set the *-k* flag, Wget will handle references to images, stylesheets, HTML pages from the same server, and so on. Wget references links to files that it has already downloaded by means of a **relative path**, whereas files that have not been stored on the local disk will keep their full URLs.

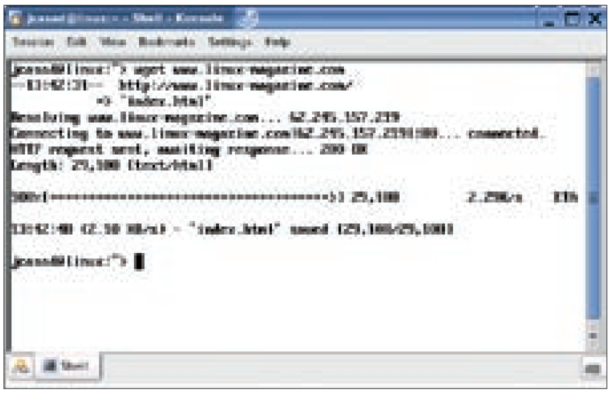
Don't Panic!

Even if a large download is interrupted, there is no need to panic, and no need to start over from scratch. Wget gives you the *-c* (for continue) option, which picks up from where the previous download left off. It does not matter whether you made the original download attempt using Wget or a graphical download manager – the tool compares the fragments with the original and just carries on from there. Wget gives you a lot of information while doing so; it might report:

```
The file is already fully ↩
retrieved; nothing to do.
```

In cases where you repeatedly download the same data, it makes sense to specify the *-N* option, which tells Wget to compare the size and date of each file with the local copy:

```
$ wget -N ↩
ftp://ftp.debian.de/debian-cd/↩
3.1_r0a/i386/iso-cd/debian-↩
31r0a-i386-binary-1.iso
...
The sizes do not match ↩
(local 7935840) - retrieving.
```



```
[illegible]@linux:~> wget www.linux-magazine.com
--13:42:31--  http://www.linux-magazine.com/
           => `index.html'
Resolving www.linux-magazine.com... 62.245.157.219
Connecting to www.linux-magazine.com|62.245.157.219|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 29,100 [text/html]

100%[=====================================>] 29,100        2.29K/s    ETA [illegible]

13:42:40 (2.30 KB/s) - `index.html' saved [29,100/29,100]

[illegible]@linux:~> 
```

Figure 1: The simplest form of the Wget command ignores embedded images and does not follow links.

If nothing has changed, the download manager will say something like:

```
Server file not newer than ↵
local file "index.html" ↵
-- not retrieving.
```

But don't worry if you forget the option: Wget does not normally overwrite local files, but creates a backup with a serial number (*index.html.1*, *index.html.2* etc.) instead.

Specifying File Types

If you only want to download files of a specific type, and you try to pass an asterisk (*) to Wget as a **wildcard**, the tool will simply display the following error message:

```
wget ↵
www.linux-magazine.com/*.jpg
...
HTTP request sent, ↵
awaiting response... ↵
404 Not Found
14:24:09 ERROR 404: Not Found.
```

Instead, you need to state the file type, or a list of file types, with the *-A* option, for example:

```
wget -r -l 1 ↵
-A jpg,png,gif ...
```

If you watch the output closely, you will notice that Wget first downloads the *index.html* but then removes the file again to leave only the images.

The whole thing also works in the opposite direction, allowing users to ignore specific file types using the *-R* parameter. Again, the parameter expects a list of file types that you do not want to transfer to your local disk. The following command

```
wget -R avi,mpg,wmv ...
```

keeps the disk memory hogs at bay.

Frugal

You have several options for restricting Wget. For example, if you have a slow Internet connection, and prefer not to use up all of your bandwidth for the download, you can restrict the bandwidth by specifying the *--limit-rate=* option. You additionally need to specify the volume in Kbytes per second, as in:

```
wget--limit-rate↵
=20k ...
```

The parameter also understands values in Mbytes; for 10 MBps you would type:

```
wget--limit-rate↵
=10m ...
```

If you prefer to restrict the total download volume, specify the *-Q* parameter instead. Again, you need to state the volume of data; the option understands values in bytes, Kbytes or Mbytes. For example, the following command:

```
wget -Q40m
```

restricts the download volume to 40 Mbytes.

Fully Automatic

If you have difficulty remembering the parameters for Wget, or if you think doing so is a waste of your time, you can use a hidden configuration file in your home directory to define your own preferences. To create a configuration file, copy the global */etc/wgetrc* template to your home directory

```
cp /etc/wgetrc ~/.wgetrc
```

and then launch an editor to modify the file. The file has entries for all the command line options, although they are disabled by default. To set the *-N* parameter as a default, simply remove the pound sign (#) at the start of the following line

```
#timestamping = off
```

and replace *off* with *on*. Figure 2 shows a sample configuration file for Wget.

Secure Download

Wget can also mirror data from servers where you need to authenticate by providing a username and password. To use this option, pass your credentials to the program when you launch it:

```
wget --http-user=username ↵
--http-passwd=password ...
```

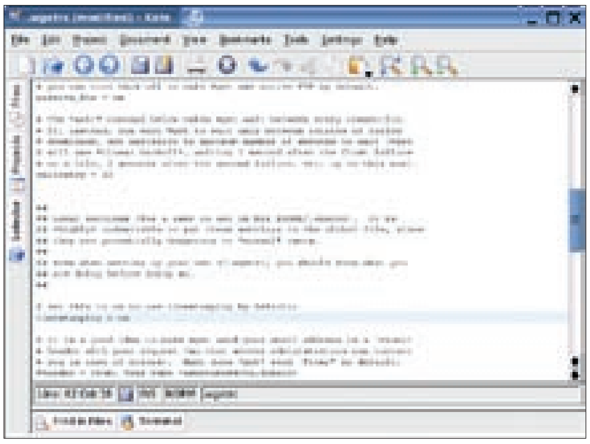


Figure 2: This could be your very own "~/.wgetrc"; comments are indicated with pound signs.

If you do this, you should beware of other inquisitive users: if another user runs the *ps* command to display the active processes on the machine, the Wget command will be listed along with the clear text username and password. As a workaround, you could use a *~/.wgetrc* configuration file in your home directory. Enter the following, for example:

```
http_user = username
http_passwd = password
```

and keep the file private by entering

```
chmod 600 ~/.wgetrc
```

Perfect Combination

Wget does not expect user input but instead just gets on with the job in the background. This is a big advantage if you need to run Wget on a remote machine via an SSH session. To do this, first establish a SSH session, and then launch the Screen program by entering

```
screen
```

at the prompt. After typing the Wget command and options to start downloading, you can press [Ctrl-A], [D] to quit Screen. You can then log off, as any processes you have launched will continue to run. The next time you log on to the remote machine, simply type

```
screen -r
```

to reestablish the Screen session. You can now check whether Wget performed as expected and relaunch the download if necessary. ■