

Mirroring a site running AmuseWiki

Contents

- Download the whole site (the easy way) 3
 - Windows 3
- Private sites 3
- Advanced 3
 - Filtering 3
 - Building ZIM file 4
 - Be nice with the servers 4

Starting with AmuseWiki version 2.031, released on October 14, 2017, each AmuseWiki site provides a `/mirror/` path offering a static version of the site, suitable for mirroring, backup and batch download. See e.g. <https://amusewiki.org/mirror>

Starting with AmuseWiki version 2.2, released on March 20, 2018, the list of files to download is provided on two URLs: `/mirror.txt` (basic version) and `/mirror.ts.txt` (advanced)

E.g. <https://amusewiki.org/mirror.txt> and <https://amusewiki.org/mirror.ts.txt>

Download the whole site (the easy way)

If you have a GNU/Linux box, `wget` is already installed and mirroring is as easy as running this command (using <https://amusewiki.org> as example):

```
wget -q -O - https://amusewiki.org/mirror.txt | wget -x -N -q -i -
```

Explanation:

The first `wget` call will download the list of file and pipe it (`-O -`) to the second call which is going to download the piped list (`-i -`), create the needed directories (`-x`) and check the timestamps (`-N`), so it will not download again the files if not modified. All this is happening quietly (`-q`).

Windows

If you don't have `wget` installed or you can't pipe commands, the procedure is a bit different.

First you need to install `wget`. See <https://www.gnu.org/software/wget/>, <https://www.gnu.org/software/wget/faq.html#download> and <https://eternallybored.org/misc/wget/>

Please keep in mind that this is a command line utility, so you are going to need the Windows command prompt.

Go to the directory where you want to create the mirror. Download <https://amusewiki.org/mirror.txt> and fetch that list:

```
wget https://amusewiki.org/mirror.txt
wget -x -N -i mirror.txt
```

And that's it.

Private sites

Private sites are not exposing `/mirror/` for obvious reasons. However, they can be mirrored with `wget` providing the credentials to the HTTP authentication.

```
wget -q -O - --user=user --password=password \
https://private.amusewiki.org/mirror.txt | \
wget --user=user --password=password -x -N -q -i -
```

Advanced

Filtering

Creative people can also additionally filter the file list to exclude formats they don't want or get only a specific format, editing (locally or on the fly) the file list passed to `wget`.

Example: download all the EPUB files and put them in the current directory (no directory tree):

```
wget -q -O - https://amusewiki.org/mirror.txt | grep '\.epub$' | wget -N -i -
```

Building ZIM file

Mirror can be converted to ZIM file format for offline reading.
Download all files, excluding bare HTML format:

```
wget -q -O - https://amusewiki.org/mirror.txt | \  
  grep -v '\.bare.html$' | \  
  wget -x -N -q -i -
```

Compile ZIM file using zimwriterfs:

```
zimwriterfs -w index.html \  
  -f site_files/favicon.ico \  
  -l EN \  
  -t Amusewiki \  
  -d "Amusewiki" \  
  -c "Amusewiki" \  
  -p "Amusewiki" \  
  amusewiki.org/mirror/ amuse.zim
```

Be nice with the servers

The above described techniques are good for a one-time job, they don't create much traffic if there are no changes, but they still hammer the sites with a lot of requests.

For this purpose, another file list is provided at `/mirror.ts.txt`, which include the timestamp of the files (without the full URL). The format is: one filename, hash symbol, timestamp. One file per line. E.g.:

```
titles.html#1525363603  
topics.html#1525363603  
authors.html#1525363603
```

This can be easily parsed and a client can check the local timestamp before doing the request.

See <https://github.com/melmothx/amusewiki/blob/master/script/mirror-site.pl> for a simple (and usable) implementation.

AmuseWiki

Mirroring a site running AmuseWiki

amusewiki.org