

NAME

netstiff – powerful and easy tool to check for Web and FTP updates

SYNOPSIS

netstiff [*options*] [*command*]

DESCRIPTION

Netstiff (formerly known as webdiff) is a powerful and easy-to-use tool which checks for Web page and/or FTP site updates.

For the Web, updates are recognized using several test criteria (`diff`, `html`, `size`, `date`, `md5sum`, `regexp`). The FTP update checker is only able to `diff` on directory listings and files and to compare `size` and `date` of files. Please see the configurator for more information on these methods.

Without a given command, netstiff will only check for updates of the specified URIs and only print those that changed, one per line, so that the output of netstiff can easily be used by further scripts (e.g. `wget` or `mail`). If no configuration file exists, the configurator is launched instead.

Netstiff exits after all configured Web pages are checked. Occuring warnings and errors leave a message in the log file (`~/.netstiff/lastlog`) and on `stderr`. Use it with `cron` if you want to check for updates regularly.

COMMANDS

You can only pass one command to netstiff. It has to be the last argument in the argument list.

Commands may be shortened down to one character (e.g. ‘`c`’ instead of ‘`configure`’). Leading dashes are ignored.

If you start netstiff without command, ‘`get`’ will be used.

configure

Use this command if you want to start the configurator, the interactive configuration tool of netstiff. Of course, you may also edit the configuration file in `~/.netstiff/config` by hand. Using the configurator is recommended if you are a new netstiff user, because it explains the possible test methods, validates your regexps, etc. Nevertheless, the configuration file format is very easy.

The configurator will not initialize the netstiff cache for added URIs, meaning: it will not download anything. To do so, you have to rerun netstiff. This is a feature.

If the config file does not exist, the configuration tool is started automatically.

diff Use this command if you want to see the differences between the last two versions of the Web pages that are configured with test method `diff` or `html`. See *diff(1)*.

get Use this command if you want netstiff to fetch the data from the specified URIs and show you only those that changed, one per line. The behaviour is already described in **DESCRIPTION**.

help Use this command to get usage information about netstiff. To be honest, this manual page in conjunction with the configurator is a better documentation.

reset Use this command after you noticed all differences with the **diff** command (see above), so that **diff** will not show you the same changes again and again.

version

This command will display version number and copyright.

OPTIONS

You may pass the following options.

--no-stderr, -S

Use this option to suppress warning and error messages on `stderr`. Thus the messages can only be seen in the log file.

--workdir DIR, -W DIR

Use this option if you want to specify another working directory. The working directory is the directory where netstiff reads the configuration file, stores the downloaded data and writes it logs. See **BUGS**.

RESTRICTIONS

There is no special case to handle status codes other than 200. In practice, netstiff will neither follow redirections nor will it notice any 4xx or 5xx error code. The resulting error pages are treated like usual Web pages. No logged message. Please check on your own.

USAGE EXAMPLE

You want to add a new URI netstiff should check for updates.

```
netstiff conf
```

The configurator is not described here. I know some weaknesses in usability, but you can get along with it. When you are seeing your shell prompt again, you know that netstiff should retrieve an initial version of the Web page you specified.

```
netstiff
```

After some weeks in the sun you want to see if something has changed. So you let netstiff check for updates.

```
netstiff
```

It is printing an URI! Let's see the changes!

```
netstiff diff
```

Oh, it is so much, that it does not fit on a screen!

```
netstiff d | pager
```

Now you are satisfied because you read all the changes. So you finally do

```
netstiff reset
```

and netstiff forgets about the changes.

RETURN VALUE

The number of errors are returned. So exit code 0 is success.

BUGS

The regular expression stuff is using the *eval* function of Ruby. This means that you are able to do non-regex-related stuff using special strings as 'regular expressions'. This is a big security issue when using netstiff as a backend for e.g. Web applications. So do NOT do it and NEVER start netstiff on foreign, unchecked configurations (-W can be dangerous).

Feel free to send feedback, bug reports, etc.

AUTHOR AND COPYRIGHT

© 2004, 2007 Stephan Beyer <s-beyer@gmx.net>, GNU GPL