After a long hiatus, `rack-contrib` is back under active maintenance!

# Reporting bugs

If you think you've found a problem in a `rack-contrib` middleware, please

accept our apologies. Nothing's perfect, although with your help, we can

get closer.

When reporting a bug, please provide:

\* The version of the `rack-contrib` gem (or git hash of the repo) that you

are running;

\* A tiny `config.ru` which demonstrates how you're using the middleware;

\* An example request which triggers the bug;

\* A description of what you're seeing happen (so that we can tell that we're

seeing the same problem when we reproduce it); and

\* A description of what you expect to see happen.

Note that, in general, the core maintainers of `rack-contrib` are caretakers

of the codebase, not the fixers-of-bugs. If you wish to see a bug fixed,

you will have a far better time if you submit a pull request (see below)

rather than reporting a bug.

# Submitting patches

New functionality and bug fixes are always welcome. To maintain the quality

of the codebase, however, there are a number of things that all patches must

have before they can be landed:

\* Test cases. A bugfix must have a test case which fails prior to the

bugfix being applied, and which passes afterwards. Feature additions must

have test cases which exercise all features and edge cases.

\* Documentation. Most bugfixes won't require documentation changes

(although some will), but all feature enhancements and new middleware will

\*definitely\* need to have documentation written. Many existing

middlewares aren't well documented, we know that, but we're trying to

make sure things don't get any \*worse\* as new things get added.

\* Adhere to existing coding conventions. The existing code isn't in a great

place, but if you diverge from how things are done at the moment the patch

won't get accepted as-is.

\* Support Ruby 2.2 and higher. We maintain the same Ruby version

compatibility as Rack itself. We use [Travis CI test

runs](https://travis-ci.org/rack/rack-contrib) to validate this.

\* Require no external dependencies. Some existing middleware depends on

additional gems in order to function; we feel that this is an

anti-pattern, and so no patches will be accepted which add additional

external gems.

We will not outright reject patches which do not meet these standards,

however \*someone\* will have to do the work to bring the patch up to scratch

before it can be landed.

# Release frequency

\* Bugfix releases (incrementing `Z` in version `X.Y.Z`), which do not change

documented behaviour in any way, may be released as soon as the bugfix

is landed.

\* Minor releases (incrementing `Y` in version `X.Y.Z`), which change

documented behaviour in ways which are entirely backwards compatible,

should be released each month, in the first few days of the month

(assuming there are any features outstanding).

\* Major releases (incrementing `X` in version `X.Y.Z`), which make changes

to documented behaviour in ways which mean that existing users of the gem

may have to change something about the way they use the gem, should be

released no less than six months apart, and ideally far less often than

that.