
sprockets.mixins.sentry

Release 2.0.1.dev3+ga15299b

Dec 05, 2018

Contents

1	Installation	3
2	Documentation	5
3	Requirements	7
4	Example	9
5	License	11
5.1	API	11
5.2	Release History	13
	Python Module Index	15

A RequestHandler mixin for sending exceptions to Sentry

CHAPTER 1

Installation

`sprockets.mixins.sentry` is available on the [Python Package Index](#) and can be installed via `pip` or `easy_install`:

```
pip install sprockets.mixins.sentry
```


CHAPTER 2

Documentation

<https://sprocketsmixinsentry.readthedocs.org>

CHAPTER 3

Requirements

- raven
- tornado

CHAPTER 4

Example

This examples demonstrates how to use `sprockets.mixins.sentry`.

```
from sprockets.mixins import sentry
from tornado import web

class RequestHandler(sentry.SentryMixin, web.RequestHandler):
    """Requires a ``SENTRY_DSN`` environment variable is set with the
    DSN value provided by sentry.

    The Mixin should catch unhandled exceptions and report them to Sentry.

    """
    def get(self, *args, **kwargs):
        raise ValueError("This should send an error to sentry")
```


Copyright (c) 2015-2018 AWeber Communications All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of Sprockets nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

5.1 API

`mixins.sentry`

A RequestHandler mixin for sending exceptions to Sentry

```
class sprockets.mixins.sentry.SanitizeEmailsProcessor (client)
```

```
    Remove all email addresses from the payload sent to sentry.
```

class sprockets.mixins.sentry.**SentryMixin** (*args, **kwargs)

Report unexpected exceptions to Sentry.

Mix this in over a `tornado.web.RequestHandler` to report unhandled exceptions to Sentry so that you can figure out what went wrong. In order to use this mix-in, all that you have to do is define the `SENTRY_DSN` environment variable that contains your projects Sentry DSN. Whenever a request comes it and the environment variable is set, this mix-in will create a new `raven.base.Client` instance and make it available via the `sentry_client` property.

If an exception is caught by `_handle_request_exception()`, then it will be reported to Sentry in all it's glory.

sentry_client

The `raven.base.Client` instance or `None` if sentry reporting is disabled. You can modify attributes of the client as required for your application – for example, you can add new modules by adding to `self.sentry_client.include_paths`.

sentry_extra

A `dict` of extra information to pass to sentry when an exception is reported.

sentry_tags

A `dict` of tag and value pairs to associated with any reported exceptions.

sprockets.mixins.sentry.**get_client** (*application*)

Retrieve the sentry client for *application*.

Parameters **application** (`tornado.web.Application`) – application to retrieve the sentry client for.

Returns a `raven.base.Client` instance or `None`

Return type `raven.base.Client`

sprockets.mixins.sentry.**install** (*application*, **kwargs)

Call this to install a sentry client into a Tornado application.

Parameters

- **application** (`tornado.web.Application`) – the application to install the client into.
- **kwargs** – keyword parameters to pass to the `raven.base.Client` initializer.

Returns `True` if the client was installed by this call and `False` otherwise.

This function should be called to initialize the Sentry client for your application. It will be called automatically with the default parameters by `SentryMixin` if you do not call it during the creation of your application. You should install the client explicitly so that you can set at least the following properties:

- **include_paths** list of python modules to include in tracebacks. This function ensures that `raven`, `sprockets`, `sys`, and `tornado` are included but you probably want to include additional packages.
- **release** the version of the application that is running

See the [raven documentation](#) for additional information.

5.1.1 Examples

The following application will report errors to sentry if you export the `SENTRY_DSN` environment variable and make a request to `http://localhost:8000/whatever` provided that *whatever* is not an integer.


```

import sys
import signal

from tornado import ioloop, web

from sprockets.mixins import sentry

class Handler(sentry.SentryMixin, web.RequestHandler):

    def initialize(self, **kwargs):
        tags = kwargs.pop('tags', dict())
        super().initialize(**kwargs)
        self.sentry_tags.update(tags)

    def get(self, status_code):
        self.set_status(int(status_code))

def make_application(app_tags):
    application = web.Application([(r'/(\\S+)', Handler)])
    sentry.install(application, include_paths=[__name__], tags=app_tags)
    return application

def stop(signo, frame):
    iol = ioloop.IOLoop.current()
    iol.add_callback_from_signal(iol.stop)

if __name__ == '__main__':
    app_tags = {}
    for arg in sys.argv[1:]:
        name, _, value = arg.partition('=')
        app_tags[name] = value

    signal.signal(signal.SIGINT, stop)
    app = make_application(app_tags)
    app.listen(8000)
    ioloop.IOLoop.current().start()

```

5.2 Release History

- 2.0.0 (8-Dec-2018)
 - Add support for Tornado 5
 - Drop Tornado<5 support
 - Drop Python<3.5 support
- 1.2.0
 - Extend raven pin so that we can use python 3.7
 - Advertise python 3.7 support
 - Drop python 3.4 from support matrix

- Remove unused import of `urllib.parse`
- 1.1.2
 - Add email sanitization processor
- 1.1.1
 - Fix password scrubbing in URLs.
 - Remove support for python 2.6, 3.2, and 3.3
- 1.1.0
 - Move raven client initialization into `sprockets.mixins.sentry.install`
 - Add support for setting raven.Client options when calling `install` on the application.
 - The sentry “environment” is set to the `$ENVIRONMENT` environment variable if it is set.
- 1.0.0
 - Work around [getsentry/raven-python#735](#)
- 0.4.0 (16-Dec-2015)
 - Ignore `web.Finish` exceptions
- 0.3.0 (13-Jul-2015)
 - Add `sprockets.mixins.sentry.SentryMixin.sentry_extra`
 - Add `sprockets.mixins.sentry.SentryMixin.sentry_tags`
 - Improve module reporting in Sentry messages
 - Improved documentation
- 0.2.0 (22-Jun-2015)
 - Stop reporting `tornado.web.HTTPError`
- 0.1.0 (13-May-2015)
 - Initial public release

S

`sprockets.mixins.sentry`, [11](#)

E

environment variable
 [SENTRY_DSN](#), 12

G

[get_client\(\)](#) (in module *sprockets.mixins.sentry*), 12

I

[install\(\)](#) (in module *sprockets.mixins.sentry*), 12

S

[SanitizeEmailsProcessor](#) (class in *sprockets.mixins.sentry*), 11

[sentry_client](#) (*sprockets.mixins.sentry.SentryMixin* attribute), 12

[SENTRY_DSN](#), 12

[sentry_extra](#) (*sprockets.mixins.sentry.SentryMixin* attribute), 12

[sentry_tags](#) (*sprockets.mixins.sentry.SentryMixin* attribute), 12

[SentryMixin](#) (class in *sprockets.mixins.sentry*), 11

[sprockets.mixins.sentry](#) (module), 11