

Open-Source Report for TCP Connections

[Flask]

General Information & Licensing

Code Repository	https://github.com/pallets/flask.git
License Type	BSD
License Description	<ul style="list-style-type: none">• Framework is available to use for free with or without modifications as long as none of the restrictions below are violated
License Restrictions	<ul style="list-style-type: none">• 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 python-flask nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

<https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1a/src/werkzeug/serving.py#L107>

WSGIRequestHandle:

<https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1a/src/werkzeug/serving.py#L148>

- write() declaration:
<https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1a/src/werkzeug/serving.py#L248>
- execute() declaration:
<https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1a/src/werkzeug/serving.py#L319>
- Call to DechunkedInput:
<https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1a/src/werkzeug/serving.py#L214>
- Call to write()

Stack trace from app start to page load on client side:

__call__, app.py:2548

<https://github.com/pallets/flask/blob/066a35dd322f689ec07d7c0e82b19eacadac3c6b/src/flask/app.py#L2546>

The previously created wsgi server is retrieving the application we have created and wrapping it for its application.

execute, serving.py:322

<https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1a/src/werkzeug/serving.py#L319>

This function is checking to see if the headers have been created yet or not. If they have been created it will send the response. If they have not yet been created it will create the headers accordingly and then proceed to send the proper response.

run_wsgi, serving.py:335

<https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1a/src/werkzeug/serving.py#L237>

Check if the headers that it has been sent are correct if they write a 100 level response and continue. If they are not correct, set the status and header values to None.

handle_one_request, server.py:415

<https://github.com/python/cpython/blob/91a8e002c21a5388c5152c5a4871b9a2d85f0fc1/Lib/http/server.py#L391>

This method is used to read and respond to a single request on the socket. This will parse the request and retrieve useful information like content length, body, etc.

handle, server.py:427

<https://github.com/python/cpython/blob/91a8e002c21a5388c5152c5a4871b9a2d85f0fc1/Lib/http/server.py#L428>

Purpose of this method is to handle multiple requests at once. It achieves this by calling the handle_one_request method multiple times.

handle, serving.py:363

<https://github.com/pallets/werkzeug/blob/3115aa6a6276939f5fd6efa46282e0256ff21f1a/src/werkzeug/serving.py#L358>

Checks to see if there is a dropped connection. In the case of a dropped connection this method will ignore the dropped connection.

__init__, socketserver.py:747

<https://github.com/python/cpython/blob/91a8e002c21a5388c5152c5a4871b9a2d85f0fc1/Lib/socketserver.py#L748>

Sets up BaseRequestHandler class variables, and attempts to call socketserver's handle method or finish the server's duties.

finish_request, socketserver.py:360

<https://github.com/python/cpython/blob/91a8e002c21a5388c5152c5a4871b9a2d85f0fc1/Lib/socketserver.py#L358>

Calls self.RequestHandlerClass to take care of finishing up post processing of the request.

process_request_thread, socketserver.py:683

<https://github.com/python/cpython/blob/91a8e002c21a5388c5152c5a4871b9a2d85f0fc1/Lib/socketserver.py#L696>

Attempts to finish a given request or accept and handle any raised errors.

run, threading.py:910

<https://github.com/enthought/Python-2.7.3/blob/69fe0ffd2d85b4002cacae1f28ef2eb0f25e16ae/Lib/threading.py#L501>

Invokes a callable object to represent the thread's activity, attempts to send to target with given keys/args

_bootstrap_inner, threading.py:973

<https://github.com/enthought/Python-2.7.3/blob/69fe0ffd2d85b4002cacae1f28ef2eb0f25e16ae/Lib/threading.py#L533>

Sets up threading class variables and attempts to start thread for communicating with the socket.

_bootstrap, threading.py:930

<https://github.com/enthought/Python-2.7.3/blob/69fe0ffd2d85b4002cacae1f28ef2eb0f25e16ae/Lib/threading.py#L510>

Wrapper method for bootstrap that prevents unseen exceptions caused by threads waking up after their app's context has been removed.

