Chaos Engineering Report

14 July 2022

Contents

Summary

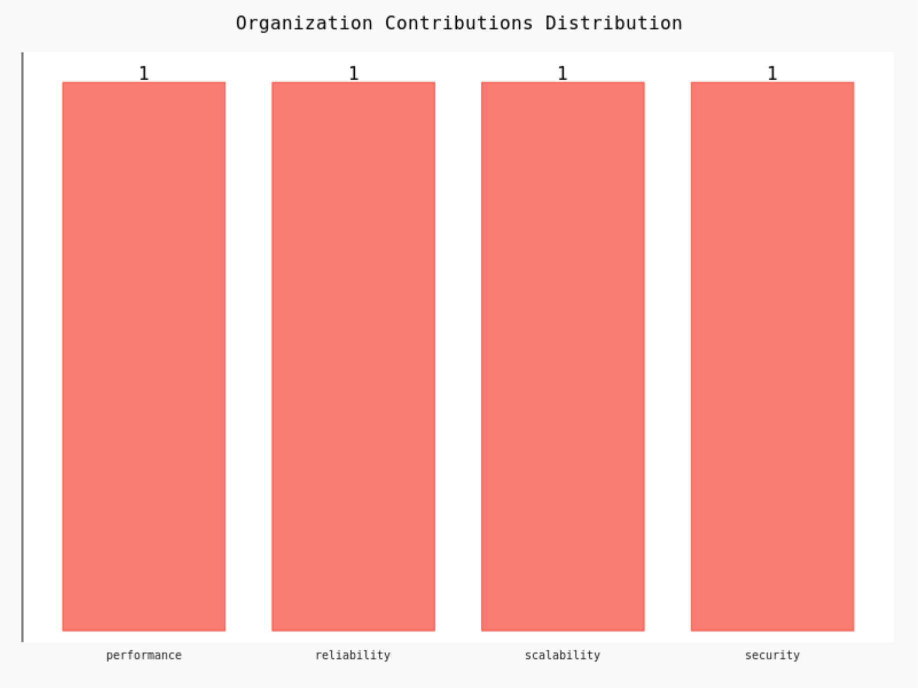
This report aggregates 1 experiments spanning over the following subjects:

flask, *python*

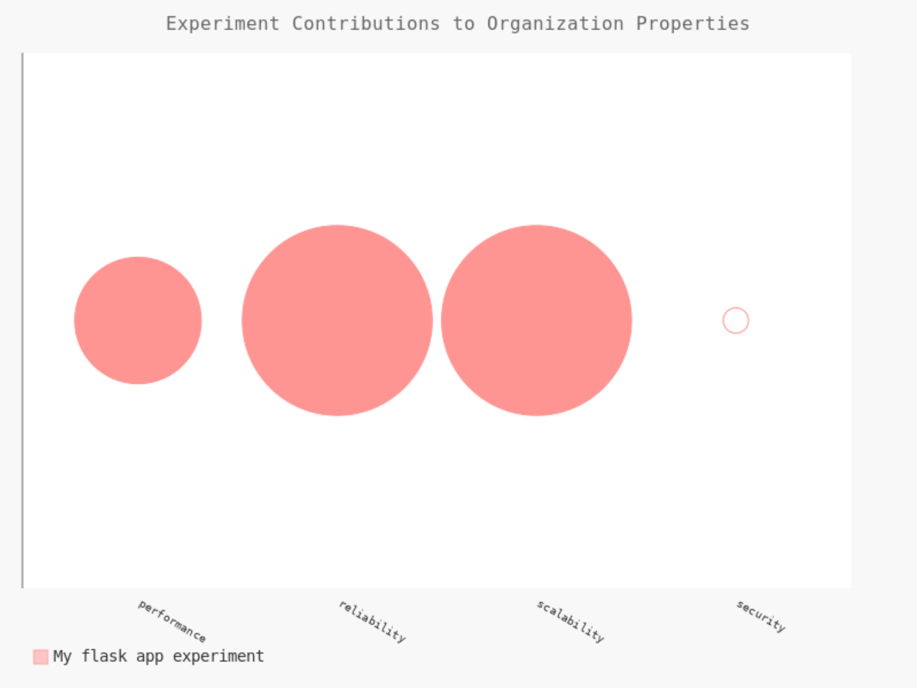
Contributions

Contributions surface the properties this collection of experiments target in order to improve the confidence and trust in the system.

The aggregated 1 experiments cover 4 distinct contributions:



Experiment Contributions Breakdown The breakdown of impacts each experiment contributes to for these system properties[1]:

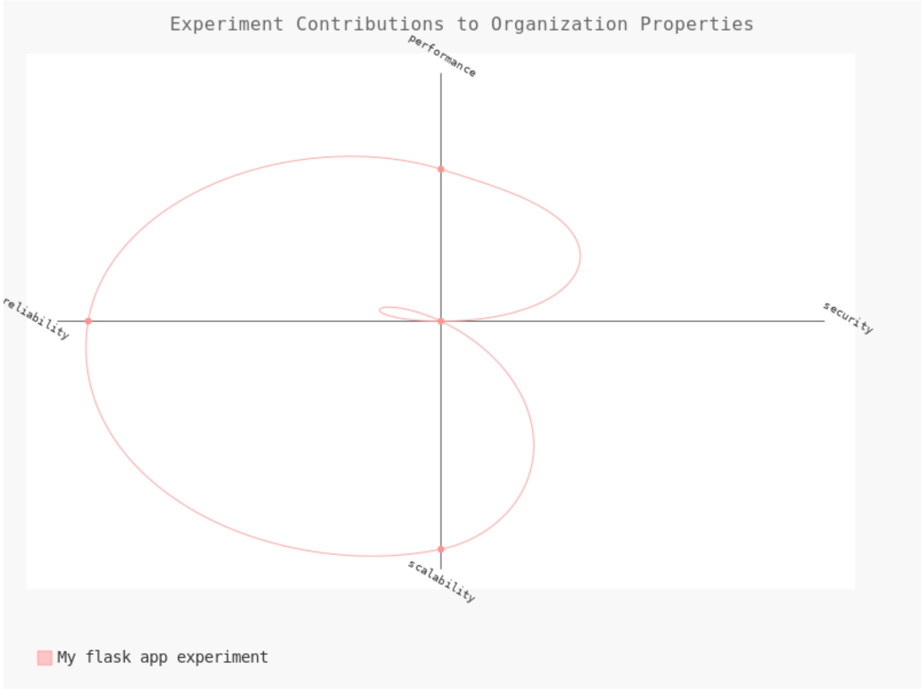


Sparse lines indicate a given experiment focused on a specific subset of contributions.

Sparse columns indicate that a given contribution was seldom impacted by this collection of experiments.

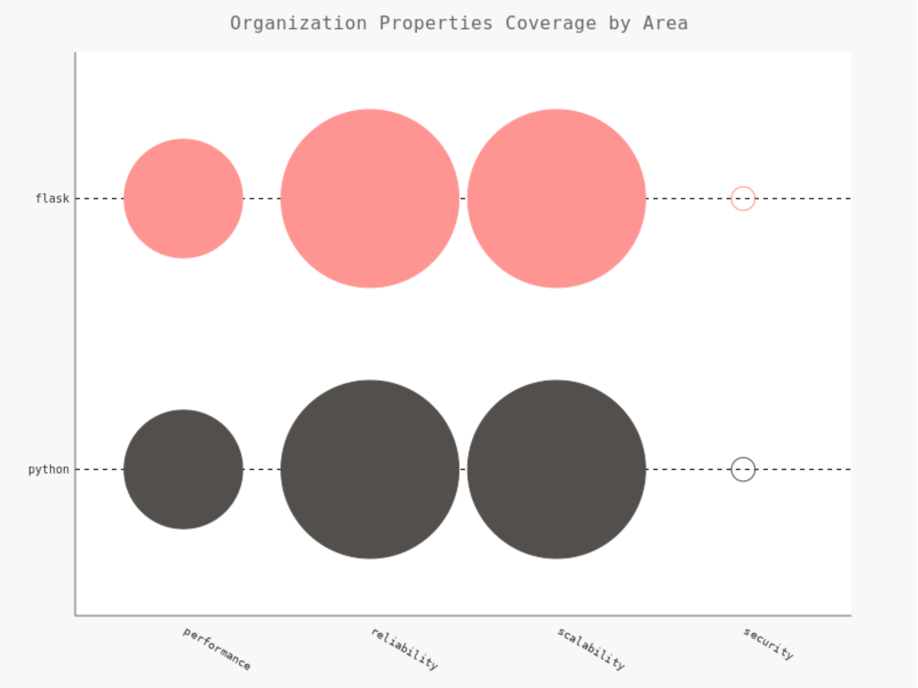
[1] Empty dots indicate an experiment is explicitely not addressing a given contribution while missing dots indicate no data for a given contribution.

Another view of the same data:



Areas Contributions Breakdown

The distribution of areas impacted by these contributions[2]:



Sparse lines indicate a given tag contributes only to a specific subset of system property.

Sparse columns indicate that a given contribution impacts moderately areas covered by the experiments.

[2] Empty dots indicate a tag is explicitely not addressing a given contribution while missing dots indicate no data for a given contribution.

Experiment

My flask app experiment

sample demo to determine resiliency in existing flask web app

Summary

Status	completed
Tagged	python, flask
Executed From	ip-172-31-40-90
Platform	Linux-5.13.0-1029-aws-x86_64-with-glibc2.29
Started	Thu, 14 Jul 2022 10:09:22 GMT
Completed	Thu, 14 Jul 2022 10:09:22 GMT
Duration	0 seconds

Definition

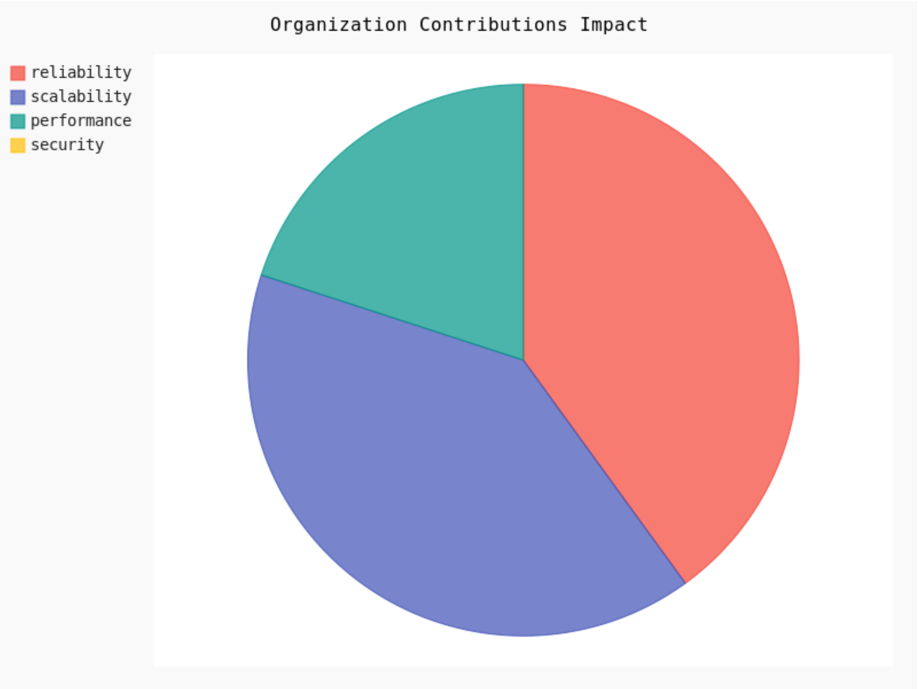
The experiment was made of 0 actions, to vary conditions in your system, and 2 probes, to collect objective data from your system during the experiment.

Organization Contributions

An estimate of relative system properties this experiment contributes to:

Contribution	Impact
reliability	high
scalability	high
performance	medium
security	none

The *"none"* value for an impact indicates that this experiment explicitely does not address that specific property.



Steady State Hypothesis

The steady state hypothesis this experiment tried was “**Website is OK**”.

Before Run

The steady state was verified

Probe	Tolerance	Verified
website must be up and running	200	True

After Run

The steady state was verified

Probe	Tolerance	Verified
website must be up and running	200	True

Method

The experiment method defines the sequence of activities that help gathering evidence towards, or against, the hypothesis.

The following activities were conducted as part of the experimental's method:

Type	Name
probe	website must return 500
probe	website must return 200 with json response

Result

The experiment was conducted on Thu, 14 Jul 2022 10:09:22 GMT and lasted roughly 0 seconds.

Probe - website must return 500

Status	succeeded
Background	False
Started	Thu, 14 Jul 2022 10:09:22 GMT
Ended	Thu, 14 Jul 2022 10:09:22 GMT
Duration	0 seconds

The probe provider that was executed:

Type	http
URL	http://localhost:5002/return-failure
Method	GET
Timeout	[3, 5]
Arguments	N/A

Probe - website must return 200 with json response

Status	succeeded
Background	False
Started	Thu, 14 Jul 2022 10:09:22 GMT
Ended	Thu, 14 Jul 2022 10:09:22 GMT
Duration	0 seconds

The probe provider that was executed:

Type	http
URL	http://localhost:5002/json-example
Method	GET

Timeout	[3, 5]
Arguments	N/A

Appendix

Probe - website must return 500

The *probe* returned the following result:

```
{'body': '<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"\n'
         '  "http://www.w3.org/TR/html4/loose.dtd">\n'
         '<html>\n'
         '  <head>\n'
         '    <title>json.decoder.JSONDecodeError: Expecting property name '
         'enclosed in double quotes: line 1 column 2 (char 1) // Werkzeug '
         'Debugger</title>\n'
         '    <link rel="stylesheet" '
         'href="?__debugger__=yes&amp;cmd=resource&amp;f=style.css"\n'
         '        type="text/css">\n'
         '    <!-- We need to make sure this has a favicon so that the '
         'debugger does\n'
         '         not accidentally trigger a request to /favicon.ico which '
         'might\n'
         "         change the application's state. -->\n"
         '    <link rel="shortcut icon"\n'
         '        '
         'href="?__debugger__=yes&amp;cmd=resource&amp;f=console.png">\n'
         '    <script '
         'src="?__debugger__=yes&amp;cmd=resource&amp;f=debugger.js"></script>\n'
         '    <script type="text/javascript">\n'
         '      var TRACEBACK = 140352018412544,\n'
         '          CONSOLE_MODE = false,\n'
         '          EVALEX = true,\n'
         '          EVALEX_TRUSTED = false,\n'
         '          SECRET = "u9Ix64QzWbIx9m60Fd4G";\n'
         '    </script>\n'
         '  </head>\n'
         '  <body style="background-color: #fff">\n'
         '    <div class="debugger">\n'
         '<h1>json.decoder.JSONDecodeError</h1>\n'
         '<div class="detail">\n'
         '  <p class="errormsg">json.decoder.JSONDecodeError: Expecting '
         'property name enclosed in double quotes: line 1 column 2 (char '
         '1)</p>\n'
         '</div>\n'
         '<h2 class="traceback">Traceback <em>(most recent call '
         'last)</em></h2>\n'
         '<div class="traceback">\n'
         '  <h3></h3>\n'
         '  <ul><li><div class="frame" id="frame-140352018413888">\n'
         '  <h4>File <cite '
         'class="filename">"/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py"</
         '      line <em class="line">2091</em>,\n'
         '      in <code class="function">__call__</code></h4>\n'
         '  <div class="source "><pre class="line before"><span class="ws">    '
         '</span>def __call__(self, environ: dict, start_response: t.Callable) '
         '-&gt; t.Any:</pre>\n'
         '<pre class="line before"><span class="ws">        '
         '</span>&quot;&quot;&quot;The WSGI server calls the Flask application '
         'object as the</pre>\n'
         '<pre class="line before"><span class="ws">        </span>WSGI '
         'application. This calls :meth:`wsgi_app`, which can be</pre>\n'
         '<pre class="line before"><span class="ws">        </span>wrapped to '
         'apply middleware.</pre>\n'
         '<pre class="line before"><span class="ws">        '
         '</span>&quot;&quot;&quot;</pre>\n'
         '<pre class="line current"><span class="ws">        </span>return '
         'self.wsgi_app(environ, start_response)</pre></div>\n'
         '</div>\n'
         '\n'
         '<li><div class="frame" id="frame-140352018413936">\n'
         '  <h4>File <cite '
         'class="filename">"/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py"</
         '      line <em class="line">2076</em>,\n'
         '      in <code class="function">wsgi_app</code></h4>\n'
         '  <div class="source "><pre class="line before"><span '
         'class="ws">            </span>try:</pre>\n'
         '<pre class="line before"><span class="ws">                '
         '</span>ctx.push()</pre>\n'
         '<pre class="line before"><span class="ws">                '
         '</span>response = self.full_dispatch_request()</pre>\n'
         '<pre class="line before"><span class="ws">            </span>except '
         'Exception as e:</pre>\n'
         '<pre class="line before"><span class="ws">                '
         '</span>error = e</pre>\n'
         '<pre class="line current"><span class="ws">                '
         '</span>response = self.handle_exception(e)</pre>\n'
         '<pre class="line after"><span class="ws">            </span>except:  '
         '# noqa: B001</pre>\n'
         '<pre class="line after"><span class="ws">                '
         '</span>error = sys.exc_info()[1]</pre>\n'
         '<pre class="line after"><span class="ws">                '
```

```
'</span>raise</pre>\n'
'<pre class="line after"><span class="ws">            </span>return '
'response(environ, start_response)</pre>\n'
'<pre class="line after"><span class="ws">        '
'</span>finally:</pre></div>\n'
'</div>\n'
'\n'
'<li><div class="frame" id="frame-140352018414080">\n'
'  <h4>File <cite '
'class="filename">"/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py"</
'      line <em class="line">2073</em>,\n'
'      in <code class="function">wsgi_app</code></h4>\n'
'  <div class="source "><pre class="line before"><span '
'class="ws">        </span>ctx = self.request_context(environ)</pre>\n'
'<pre class="line before"><span class="ws">        </span>error: '
't.Optional[BaseException] = None</pre>\n'
'<pre class="line before"><span class="ws">        </span>try:</pre>\n'
'<pre class="line before"><span class="ws">            '
'</span>try:</pre>\n'
'<pre class="line before"><span class="ws">                '
'</span>ctx.push()</pre>\n'
'<pre class="line current"><span class="ws">                '
'</span>response = self.full_dispatch_request()</pre>\n'
'<pre class="line after"><span class="ws">            </span>except '
'Exception as e:</pre>\n'
'<pre class="line after"><span class="ws">                '
'</span>error = e</pre>\n'
'<pre class="line after"><span class="ws">                '
'</span>response = self.handle_exception(e)</pre>\n'
'<pre class="line after"><span class="ws">            </span>except:  '
'# noqa: B001</pre>\n'
'<pre class="line after"><span class="ws">                '
'</span>error = sys.exc_info()[1]</pre></div>\n'
'</div>\n'
'\n'
'<li><div class="frame" id="frame-140352018414272">\n'
'  <h4>File <cite '
'class="filename">"/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py"</
'      line <em class="line">1519</em>,\n'
'      in <code class="function">full_dispatch_request</code></h4>\n'
'  <div class="source "><pre class="line before"><span '
'class="ws">            </span>rv = self.preprocess_request()</pre>\n'
'<pre class="line before"><span class="ws">            </span>if rv '
'is None:</pre>\n'
'<pre class="line before"><span class="ws">                </span>rv '
'= self.dispatch_request()</pre>\n'
```

```
'<pre class="line before"><span class="ws">        </span>except '
'Exception as e:</pre>\n'
'<pre class="line before"><span class="ws">            </span>rv = '
'self.handle_user_exception(e)</pre>\n'
'<pre class="line current"><span class="ws">        </span>return '
'self.finalize_request(rv)</pre>\n'
'<pre class="line after"><span class="ws"></span> </pre>\n'
'<pre class="line after"><span class="ws">    </span>def '
'finalize_request(</pre>\n'
'<pre class="line after"><span class="ws">        </span>self,</pre>\n'
'<pre class="line after"><span class="ws">        </span>rv: '
't.Union[ResponseReturnValue, HTTPException],</pre>\n'
'<pre class="line after"><span class="ws">        '
'</span>from_error_handler: bool = False,</pre></div>\n'
'</div>\n'
'\n'
'<li><div class="frame" id="frame-140352018414320">\n'
'  <h4>File <cite '
'class="filename">"/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py"</
'      line <em class="line">1540</em>,\n'
'      in <code class="function">finalize_request</code></h4>\n'
'  <div class="source "><pre class="line before"><span '
'class="ws"></span> </pre>\n'
'<pre class="line before"><span class="ws">        '
'</span>:internal:</pre>\n'
'<pre class="line before"><span class="ws">        '
'</span>&quot;&quot;&quot;</pre>\n'
'<pre class="line before"><span class="ws">        </span>response = '
'self.make_response(rv)</pre>\n'
'<pre class="line before"><span class="ws">        </span>try:</pre>\n'
'<pre class="line current"><span class="ws">            '
'</span>response = self.process_response(response)</pre>\n'
'<pre class="line after"><span class="ws">            '
'</span>request_finished.send(self, response=response)</pre>\n'
'<pre class="line after"><span class="ws">        </span>except '
'Exception:</pre>\n'
'<pre class="line after"><span class="ws">            </span>if not '
'from_error_handler:</pre>\n'
'<pre class="line after"><span class="ws">                '
'</span>raise</pre>\n'
'<pre class="line after"><span class="ws">            '
'</span>self.logger.exception(</pre></div>\n'
'</div>\n'
'\n'
'<li><div class="frame" id="frame-140352018414368">\n'
'  <h4>File <cite '
```

```
'class="filename">"/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py"</
'      line <em class="line">1885</em>,\n'
'      in <code class="function">process_response</code></h4>\n'
'  <div class="source "><pre class="line before"><span '
'class="ws">            </span>response = '
'self.ensure_sync(func)(response)</pre>\n'
'<pre class="line before"><span class="ws"></span> </pre>\n'
'<pre class="line before"><span class="ws">        </span>for name in '
'chain(request.blueprints, (None,)):</pre>\n'
'<pre class="line before"><span class="ws">            </span>if name '
'in self.after_request_funcs:</pre>\n'
'<pre class="line before"><span class="ws">                </span>for '
'func in reversed(self.after_request_funcs[name]):</pre>\n'
'<pre class="line current"><span class="ws">                    '
'</span>response = self.ensure_sync(func)(response)</pre>\n'
'<pre class="line after"><span class="ws"></span> </pre>\n'
'<pre class="line after"><span class="ws">        </span>if not '
'self.session_interface.is_null_session(ctx.session):</pre>\n'
'<pre class="line after"><span class="ws">            '
'</span>self.session_interface.save_session(self, ctx.session, '
'response)</pre>\n'
'<pre class="line after"><span class="ws"></span> </pre>\n'
'<pre class="line after"><span class="ws">        </span>return '
'response</pre></div>\n'
'</div>\n'
'\n'
'<li><div class="frame" id="frame-140352018414416">\n'
'  <h4>File <cite '
'class="filename">"/home/ubuntu/chaostk/lib/python3.8/site-packages/pdchaos/middlew
'      line <em class="line">14</em>,\n'
'      in <code class="function">_after_request</code></h4>\n'
'  <div class="source "><pre class="line before"><span '
'class="ws"></span> </pre>\n'
'<pre class="line before"><span class="ws"></span>def '
'_after_request(response):</pre>\n'
'<pre class="line before"><span class="ws">    '
'</span>&quot;&quot;&quot;Runs after each request.</pre>\n'
'<pre class="line before"><span class="ws">    </span>See: '
'https://flask.palletsprojects.com/en/1.1.x/api/#flask.Flask.after_request</pre>\n'
'<pre class="line before"><span class="ws">    '
'</span>&quot;&quot;&quot;</pre>\n'
'<pre class="line current"><span class="ws">    </span>attack = '
'json.loads(request.headers.get(HEADER_ATTACK)) if (HEADER_ATTACK in '
'request.headers) else None</pre>\n'
'<pre class="line after"><span class="ws">    </span>attack_ctx = '
'{ATTACK_KEY_ROUTE: request.path}</pre>\n'
```

```
'<pre class="line after"><span class="ws"></span> </pre>\n'
'<pre class="line after"><span class="ws">    '
'</span>chaos.attack(attack, attack_ctx)</pre>\n'
'<pre class="line after"><span class="ws">    </span>return '
'response</pre>\n'
'<pre class="line after"><span class="ws"></span> </pre></div>\n'
'</div>\n'
'\n'
'<li><div class="frame" id="frame-140352018414128">\n'
'  <h4>File <cite '
'class="filename">"/usr/lib/python3.8/json/__init__.py"</cite>,\n'
'      line <em class="line">357</em>,\n'
'      in <code class="function">loads</code></h4>\n'
'  <div class="source "><pre class="line before"><span '
'class="ws">        </span>del kw[&#x27;encoding&#x27;]</pre>\n'
'<pre class="line before"><span class="ws"></span> </pre>\n'
'<pre class="line before"><span class="ws">    </span>if (cls is None '
'and object_hook is None and</pre>\n'
'<pre class="line before"><span class="ws">            '
'</span>parse_int is None and parse_float is None and</pre>\n'
'<pre class="line before"><span class="ws">            '
'</span>parse_constant is None and object_pairs_hook is None and not '
'kw):</pre>\n'
'<pre class="line current"><span class="ws">        </span>return '
'_default_decoder.decode(s)</pre>\n'
'<pre class="line after"><span class="ws">    </span>if cls is '
'None:</pre>\n'
'<pre class="line after"><span class="ws">        </span>cls = '
'JSONDecoder</pre>\n'
'<pre class="line after"><span class="ws">    </span>if object_hook '
'is not None:</pre>\n'
'<pre class="line after"><span class="ws">        '
'</span>kw[&#x27;object_hook&#x27;] = object_hook</pre>\n'
'<pre class="line after"><span class="ws">    </span>if '
'object_pairs_hook is not None:</pre></div>\n'
'</div>\n'
'\n'
'<li><div class="frame" id="frame-140352018413984">\n'
'  <h4>File <cite '
'class="filename">"/usr/lib/python3.8/json/decoder.py"</cite>,\n'
'      line <em class="line">337</em>,\n'
'      in <code class="function">decode</code></h4>\n'
'  <div class="source "><pre class="line before"><span class="ws">    '
'</span>def decode(self, s, _w=WHITESPACE.match):</pre>\n'
'<pre class="line before"><span class="ws">        '
'</span>&quot;&quot;&quot;Return the Python representation of ``s`` '
```

```
'(a ``str`` instance</pre>\n'
'<pre class="line before"><span class="ws">        </span>containing '
'a JSON document).</pre>\n'
'<pre class="line before"><span class="ws"></span> </pre>\n'
'<pre class="line before"><span class="ws">        '
'</span>&quot;&quot;&quot;</pre>\n'
'<pre class="line current"><span class="ws">        </span>obj, end = '
'self.raw_decode(s, idx=_w(s, 0).end())</pre>\n'
'<pre class="line after"><span class="ws">        </span>end = _w(s, '
'end).end()</pre>\n'
'<pre class="line after"><span class="ws">        </span>if end != '
'len(s):</pre>\n'
'<pre class="line after"><span class="ws">            </span>raise '
'JSONDecodeError(&quot;Extra data&quot;, s, end)</pre>\n'
'<pre class="line after"><span class="ws">        </span>return '
'obj</pre>\n'
'<pre class="line after"><span class="ws"></span> </pre></div>\n'
'</div>\n'
'\n'
'<li><div class="frame" id="frame-140352018414512">\n'
'  <h4>File <cite '
'class="filename">"/usr/lib/python3.8/json/decoder.py"</cite>,\n'
'      line <em class="line">353</em>,\n'
'      in <code class="function">raw_decode</code></h4>\n'
'  <div class="source "><pre class="line before"><span '
'class="ws">        </span>This can be used to decode a JSON document '
'from a string that may</pre>\n'
'<pre class="line before"><span class="ws">        </span>have '
'extraneous data at the end.</pre>\n'
'<pre class="line before"><span class="ws"></span> </pre>\n'
'<pre class="line before"><span class="ws">        '
'</span>&quot;&quot;&quot;</pre>\n'
'<pre class="line before"><span class="ws">        </span>try:</pre>\n'
'<pre class="line current"><span class="ws">            </span>obj, '
'end = self.scan_once(s, idx)</pre>\n'
'<pre class="line after"><span class="ws">        </span>except '
'StopIteration as err:</pre>\n'
'<pre class="line after"><span class="ws">            </span>raise '
'JSONDecodeError(&quot;Expecting value&quot;, s, err.value) from '
'None</pre>\n'
'<pre class="line after"><span class="ws">        </span>return obj, '
'end</pre></div>\n'
'</div>\n'
'</ul>\n'
'  <blockquote>json.decoder.JSONDecodeError: Expecting property name '
'enclosed in double quotes: line 1 column 2 (char 1)</blockquote>\n'
```

```
'</div>\n'
'\n'
'<div class="plain">\n'
'    <p>\n'
'      This is the Copy/Paste friendly version of the traceback.\n'
'    </p>\n'
'    <textarea cols="50" rows="10" name="code" readonly>Traceback '
'(most recent call last):\n'
'  File '
'&quot;/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py&quot;, '
'line 2091, in __call__\n'
'    return self.wsgi_app(environ, start_response)\n'
'  File '
'&quot;/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py&quot;, '
'line 2076, in wsgi_app\n'
'    response = self.handle_exception(e)\n'
'  File '
'&quot;/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py&quot;, '
'line 2073, in wsgi_app\n'
'    response = self.full_dispatch_request()\n'
'  File '
'&quot;/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py&quot;, '
'line 1519, in full_dispatch_request\n'
'    return self.finalize_request(rv)\n'
'  File '
'&quot;/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py&quot;, '
'line 1540, in finalize_request\n'
'    response = self.process_response(response)\n'
'  File '
'&quot;/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py&quot;, '
'line 1885, in process_response\n'
'    response = self.ensure_sync(func)(response)\n'
'  File '
'&quot;/home/ubuntu/chaostk/lib/python3.8/site-packages/pdchaos/middleware/contrib/
'line 14, in _after_request\n'
'    attack = json.loads(request.headers.get(HEADER_ATTACK)) if '
'(HEADER_ATTACK in request.headers) else None\n'
'  File &quot;/usr/lib/python3.8/json/__init__.py&quot;, line 357, in '
'loads\n'
'    return _default_decoder.decode(s)\n'
'  File &quot;/usr/lib/python3.8/json/decoder.py&quot;, line 337, in '
'decode\n'
'    obj, end = self.raw_decode(s, idx=_w(s, 0).end())\n'
'  File &quot;/usr/lib/python3.8/json/decoder.py&quot;, line 353, in '
'raw_decode\n'
'    obj, end = self.scan_once(s, idx)\n'
```

```
'json.decoder.JSONDecodeError: Expecting property name enclosed in '
'double quotes: line 1 column 2 (char 1)</textarea>\n'
'</div>\n'
'<div class="explanation">\n'
'  The debugger caught an exception in your WSGI application.  You '
'can now\n'
'  look at the traceback which led to the error.  <span '
'class="nojavascript">\n'
'  If you enable JavaScript you can also use additional features such '
'as code\n'
'  execution (if the evalex feature is enabled), automatic pasting of '
'the\n'
'  exceptions and much more.</span>\n'
'</div>\n'
'      <div class="footer">\n'
'        Brought to you by <strong class="arthur">DON\'T '
'PANIC</strong>, your\n'
'        friendly Werkzeug powered traceback interpreter.\n'
'      </div>\n'
'    </div>\n'
'\n'
'    <div class="pin-prompt">\n'
'      <div class="inner">\n'
'        <h3>Console Locked</h3>\n'
'        <p>\n'
'          The console is locked and needs to be unlocked by entering '
'the PIN.\n'
'          You can find the PIN printed out on the standard output of '
'your\n'
'          shell that runs the server.\n'
'        <form>\n'
'          <p>PIN:\n'
'            <input type=text name=pin size=14>\n'
'            <input type=submit name=btn value="Confirm Pin">\n'
'        </form>\n'
'      </div>\n'
'    </div>\n'
'  </body>\n'
'</html>\n'
'\n'
'<!--\n'
'\n'
'Traceback (most recent call last):\n'
'  File '
'"/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py", '
'line 2091, in __call__\n'
```

```
     '    return self.wsgi_app(environ, start_response)\n'
     '  File '
     '"/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py", '
     'line 2076, in wsgi_app\n'
     '    response = self.handle_exception(e)\n'
     '  File '
     '"/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py", '
     'line 2073, in wsgi_app\n'
     '    response = self.full_dispatch_request()\n'
     '  File '
     '"/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py", '
     'line 1519, in full_dispatch_request\n'
     '    return self.finalize_request(rv)\n'
     '  File '
     '"/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py", '
     'line 1540, in finalize_request\n'
     '    response = self.process_response(response)\n'
     '  File '
     '"/home/ubuntu/chaostk/lib/python3.8/site-packages/flask/app.py", '
     'line 1885, in process_response\n'
     '    response = self.ensure_sync(func)(response)\n'
     '  File '
     '"/home/ubuntu/chaostk/lib/python3.8/site-packages/pdchaos/middleware/contrib/flask
     'line 14, in _after_request\n'
     '    attack = json.loads(request.headers.get(HEADER_ATTACK)) if '
     '(HEADER_ATTACK in request.headers) else None\n'
     '  File "/usr/lib/python3.8/json/__init__.py", line 357, in loads\n'
     '    return _default_decoder.decode(s)\n'
     '  File "/usr/lib/python3.8/json/decoder.py", line 337, in decode\n'
     '    obj, end = self.raw_decode(s, idx=_w(s, 0).end())\n'
     '  File "/usr/lib/python3.8/json/decoder.py", line 353, in '
     'raw_decode\n'
     '    obj, end = self.scan_once(s, idx)\n'
     'json.decoder.JSONDecodeError: Expecting property name enclosed in '
     'double quotes: line 1 column 2 (char 1)\n'
     '\n'
     '-->\n',
 'headers': {'Connection': 'close',
       'Content-Type': 'text/html; charset=utf-8',
       'Date': 'Thu, 14 Jul 2022 10:09:22 GMT',
       'Server': 'Werkzeug/2.0.2 Python/3.8.10',
       'X-XSS-Protection': '0'},
 'status': 500}
```

Probe - website must return 200 with json response

The *probe* returned the following result:

```
{'body': {'name': 'Karan'},
 'headers': {'Content-Length': '22',
             'Content-Type': 'application/json',
             'Date': 'Thu, 14 Jul 2022 10:09:22 GMT',
             'Server': 'Werkzeug/2.0.2 Python/3.8.10'},
 'status': 200}
```