AutomationSTAR 2024

Fuzz Testing Web APIs: Overview of Existing Tools

Andrea Arcuri

Kristiania Norway

Web Services

- Providing APIs (Application Programming Interfaces) over network, remote servers
- Communications over UDP/TCP, with protocols like HTTP
- Different types of data transfer formats
  - JSON, XML, HTML, plain text, etc.
- Permanent storage:
  - eg, SQL/NoSQL databases
- **REST API**s most common type of web services
  - others are *SOAP, GraphQL and gRPC*

Resource Summary
- Files
- About
- Changes
- Children
- Parents
- Permissions
- Revisions
- Apps
- Comments
- Replies
- Properties
- Channels
- Realtime
- Teamdrives
- Standard Features

API Reference

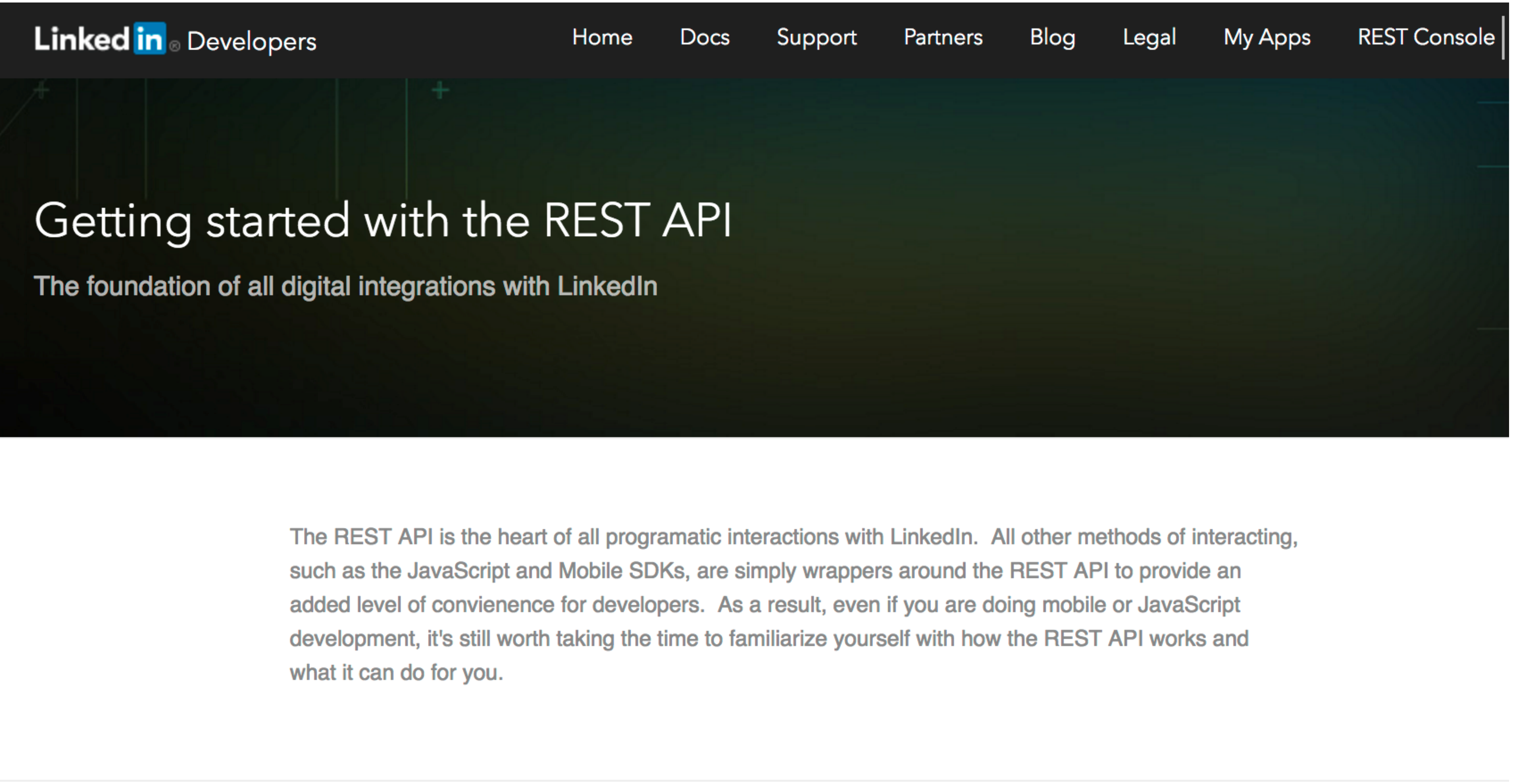
This API reference is organized by resource type. Each resource type has one or more data representations and one or more methods.

Resource types

Files

For Files Resource details, see the resource representation page.

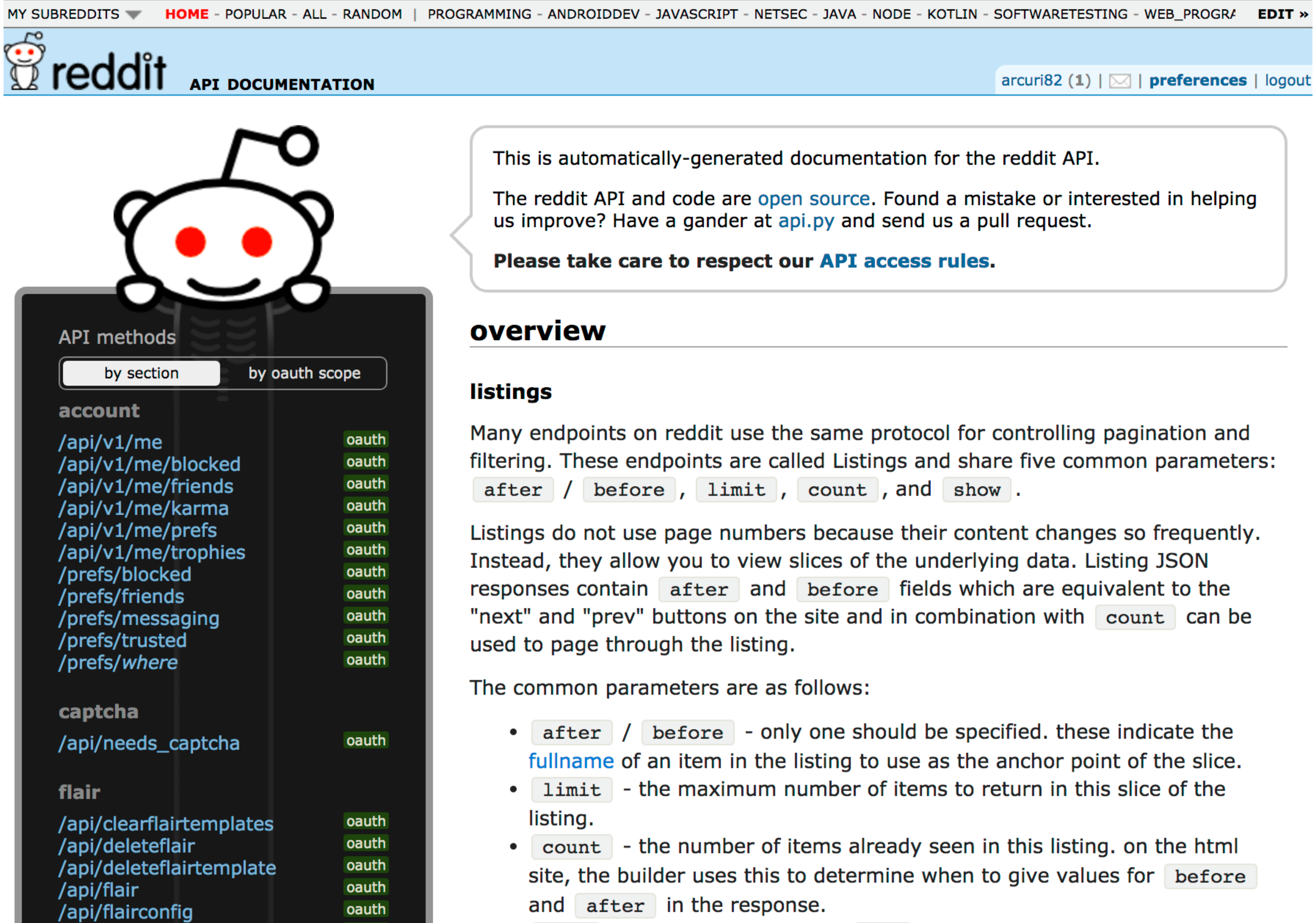
Method	HTTP request	Description
URIs relative to https://www.googleapis.com/drive/v2, unless otherwise noted		
get	`GET /files/`*`fileId`*	Gets a file's metadata by ID.
insert	`POST https://www.googleapis.com/upload/drive/v2/files` and `POST /files`	Insert a new file.
patch	`PATCH /files/`*`fileId`*	Updates file metadata. This method supports patch semantics.
update	`PUT https://www.googleapis.com/upload/drive/v2/files/`*`fileId`* and `PUT /files/`*`fileId`*	Updates file metadata and/or content.
copy	`POST /files/`*`fileId`*`/copy`	Creates a copy of the specified file.



Getting started with the REST API

The foundation of all digital integrations with LinkedIn

The REST API is the heart of all programatic interactions with LinkedIn. All other methods of interacting, such as the JavaScript and Mobile SDKs, are simply wrappers around the REST API to provide an added level of convienence for developers. As a result, even if you are doing mobile or JavaScript development, it's still worth taking the time to familiarize yourself with how the REST API works and what it can do for you.


MY SUBREDDITS
HOME - POPULAR - ALL - RANDOM | PROGRAMMING - ANDROIDDEV - JAVASCRIPT - NETSEC - JAVA - NODE - KOTLIN - SOFTWARETESTING - WEB_PROGRA
EDIT »
reddit
API DOCUMENTATION
arcuri82 (1) | | preferences | logout

This is automatically-generated documentation for the reddit API.

The reddit API and code are open source. Found a mistake or interested in helping us improve? Have a gander at api.py and send us a pull request.

Please take care to respect our API access rules.

API methods

by section | by oauth scope

account	
/api/v1/me	oauth
/api/v1/me/blocked	oauth
/api/v1/me/friends	oauth
/api/v1/me/karma	oauth
/api/v1/me/prefs	oauth
/api/v1/me/trophies	oauth
/prefs/blocked	oauth
/prefs/friends	oauth
/prefs/messaging	oauth
/prefs/trusted	oauth
/prefs/*where*	oauth
captcha	
/api/needs_captcha	oauth
flair	
/api/clearflairtemplates	oauth
/api/deleteflair	oauth
/api/deleteflairtemplate	oauth
/api/flair	oauth
/api/flairconfig	oauth

overview

listings

Many endpoints on reddit use the same protocol for controlling pagination and filtering. These endpoints are called Listings and share five common parameters: `after` / `before`, `limit`, `count`, and `show`.

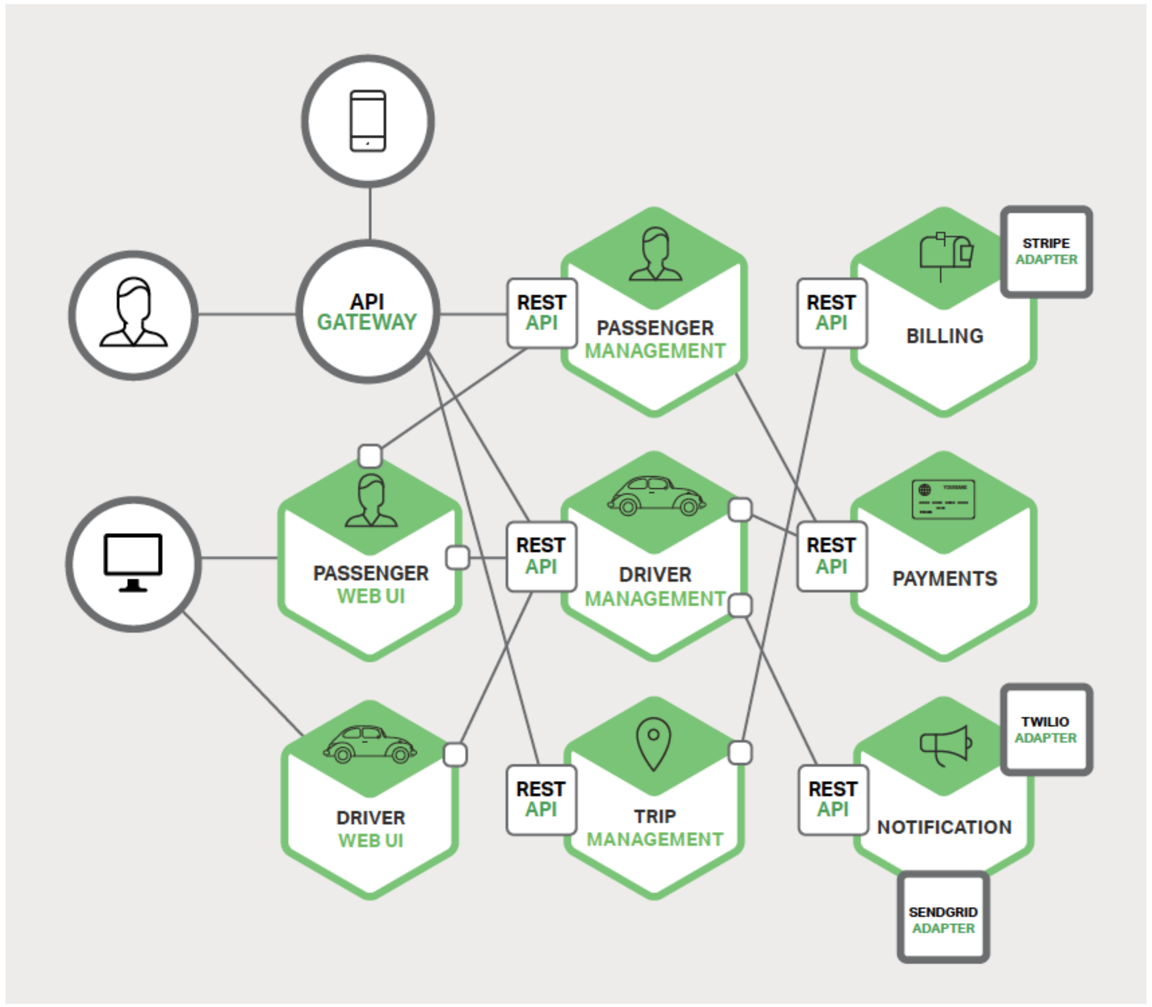
Listings do not use page numbers because their content changes so frequently. Instead, they allow you to view slices of the underlying data. Listing JSON responses contain `after` and `before` fields which are equivalent to the "next" and "prev" buttons on the site and in combination with `count` can be used to page through the listing.

The common parameters are as follows:

- `after` / `before` - only one should be specified. these indicate the fullname of an item in the listing to use as the anchor point of the slice.
- `limit` - the maximum number of items to return in this slice of the listing.
- `count` - the number of items already seen in this listing. on the html site, the builder uses this to determine when to give values for `before` and `after` in the response.

REST in Microservices

- Common trend in enterprises
- Split application in many small web services, often REST
- Easier to scale and maintain



REST Testing Challenges

- How to choose **query** and **path** parameters?
- How to prepare **body payloads** (e.g. JSON)?
- How to choose data to insert into **SQL** databases?
- Goals:
  - **Finding faults** (eg crashes)
  - **Maximize code coverage** (eg, regression tests)
- Writing high coverage tests *by hand* for every single endpoint is time consuming

What about Automated Test Generation for RESTful APIs?

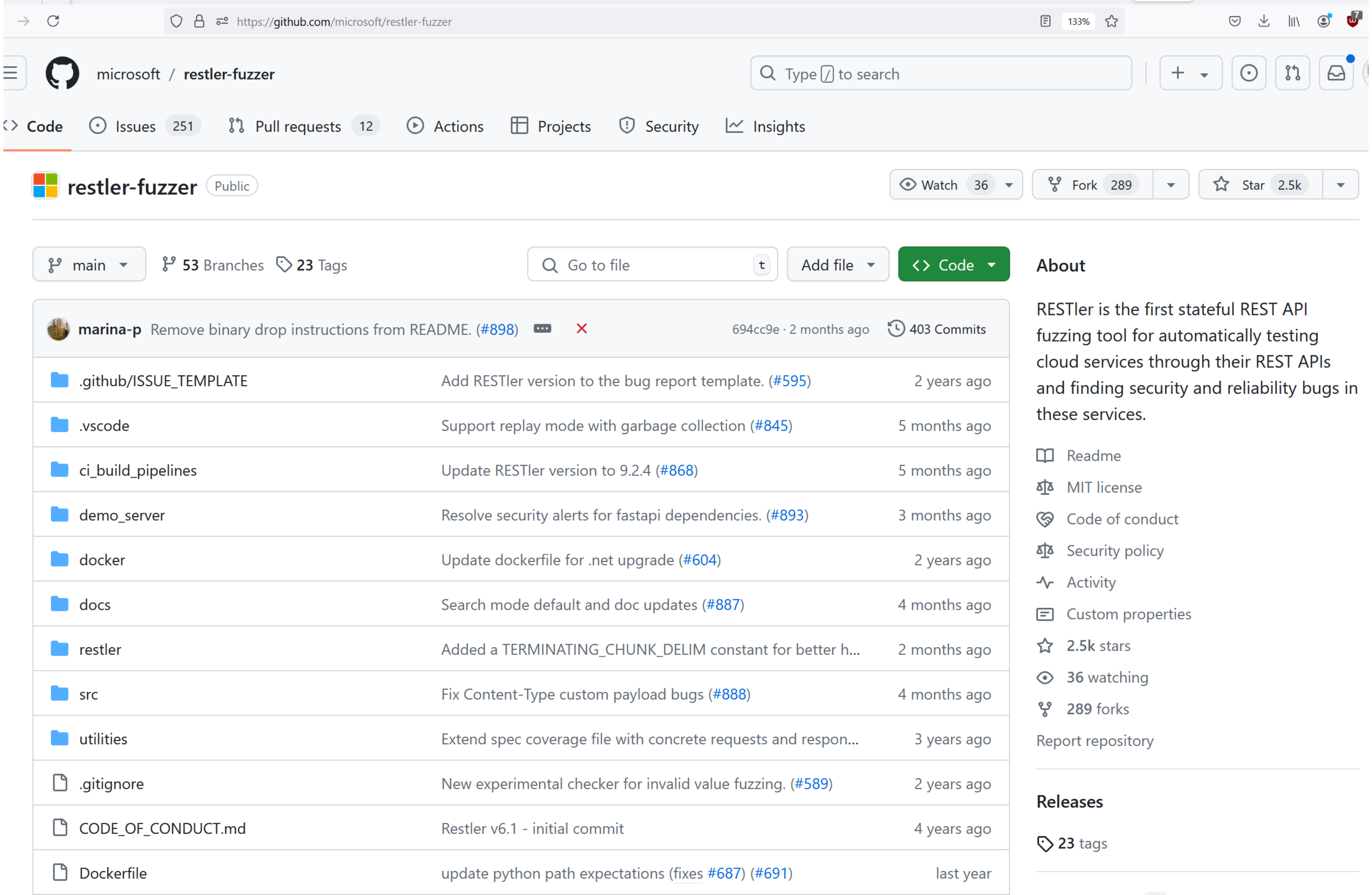
- Automatically write all the test cases
- Not just execution, but choice of all the inputs
- Hard, complex problem

2 Uses of Generated Tests

- If automated oracles: **automatically detect faults**
  - e.g., HTTP response giving 500
- No oracles / faults: **regressing testing**
  - Tests can be added to Git, to capture current behavior of system
  - If in future introduce new bug that breaks functionality, regression tests will start to fail

Fuzzers

- Tools that automatically generate test inputs
- Different strategies: from **random** inputs to advanced **AI** techniques
- Used in many different domains
  - eg, parser libraries and unit testing
- REST fuzzing is a more recent development
  - eg, Restler, Schemathesis, RESTest, Fuzz-Lightyear and EvoMaster


https://github.com/microsoft/restler-fuzzer

microsoft / restler-fuzzer

restler-fuzzer Public

marina-p Remove binary drop instructions from README. (#898) 694cc9e · 2 months ago 403 Commits

.github/ISSUE_TEMPLATE	Add RESTler version to the bug report template. (#595)	2 years ago
.vscode	Support replay mode with garbage collection (#845)	5 months ago
ci_build_pipelines	Update RESTler version to 9.2.4 (#868)	5 months ago
demo_server	Resolve security alerts for fastapi dependencies. (#893)	3 months ago
docker	Update dockerfile for .net upgrade (#604)	2 years ago
docs	Search mode default and doc updates (#887)	4 months ago
restler	Added a TERMINATING_CHUNK_DELIM constant for better h...	2 months ago
src	Fix Content-Type custom payload bugs (#888)	4 months ago
utilities	Extend spec coverage file with concrete requests and respon...	3 years ago
.gitignore	New experimental checker for invalid value fuzzing. (#589)	2 years ago
CODE_OF_CONDUCT.md	Restler v6.1 - initial commit	4 years ago
Dockerfile	update python path expectations (fixes #687) (#691)	last year

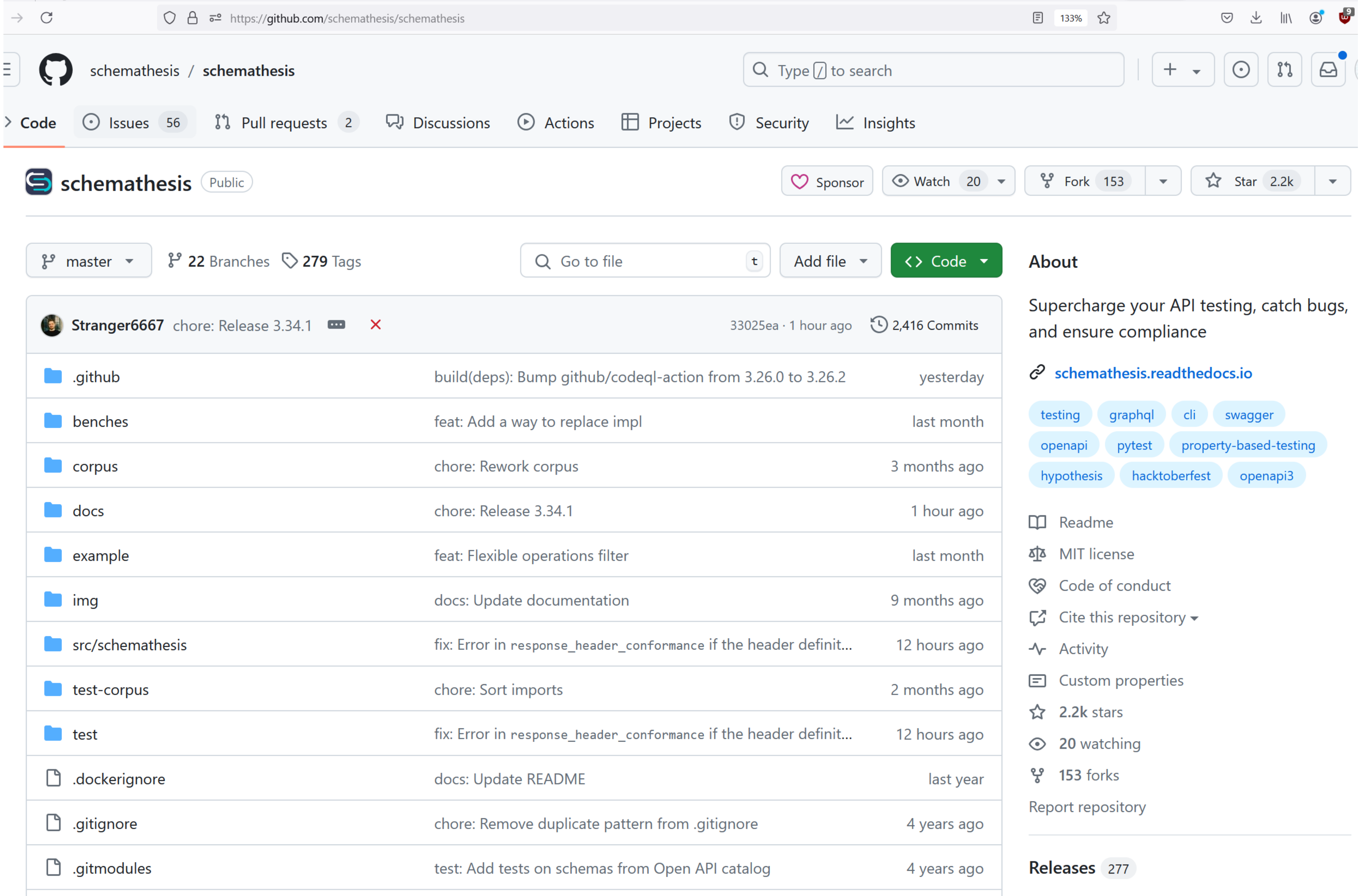
About

RESTler is the first stateful REST API fuzzing tool for automatically testing cloud services through their REST APIs and finding security and reliability bugs in these services.

Readme
MIT license
Code of conduct
Security policy
Activity
Custom properties
2.5k stars
36 watching
289 forks
Report repository

Releases

23 tags


https://github.com/schemathesis/schemathesis

schemathesis / schemathesis

Type / to search

Code · Issues 56 · Pull requests 2 · Discussions · Actions · Projects · Security · Insights

schemathesis Public

Sponsor · Watch 20 · Fork 153 · Star 2.2k

master · 22 Branches · 279 Tags · Go to file · Add file · Code

Stranger6667 chore: Release 3.34.1 · 33025ea · 1 hour ago · 2,416 Commits

.github	build(deps): Bump github/codeql-action from 3.26.0 to 3.26.2	yesterday
benches	feat: Add a way to replace impl	last month
corpus	chore: Rework corpus	3 months ago
docs	chore: Release 3.34.1	1 hour ago
example	feat: Flexible operations filter	last month
img	docs: Update documentation	9 months ago
src/schemathesis	fix: Error in `response_header_conformance` if the header definit...	12 hours ago
test-corpus	chore: Sort imports	2 months ago
test	fix: Error in `response_header_conformance` if the header definit...	12 hours ago
.dockerignore	docs: Update README	last year
.gitignore	chore: Remove duplicate pattern from .gitignore	4 years ago
.gitmodules	test: Add tests on schemas from Open API catalog	4 years ago

About

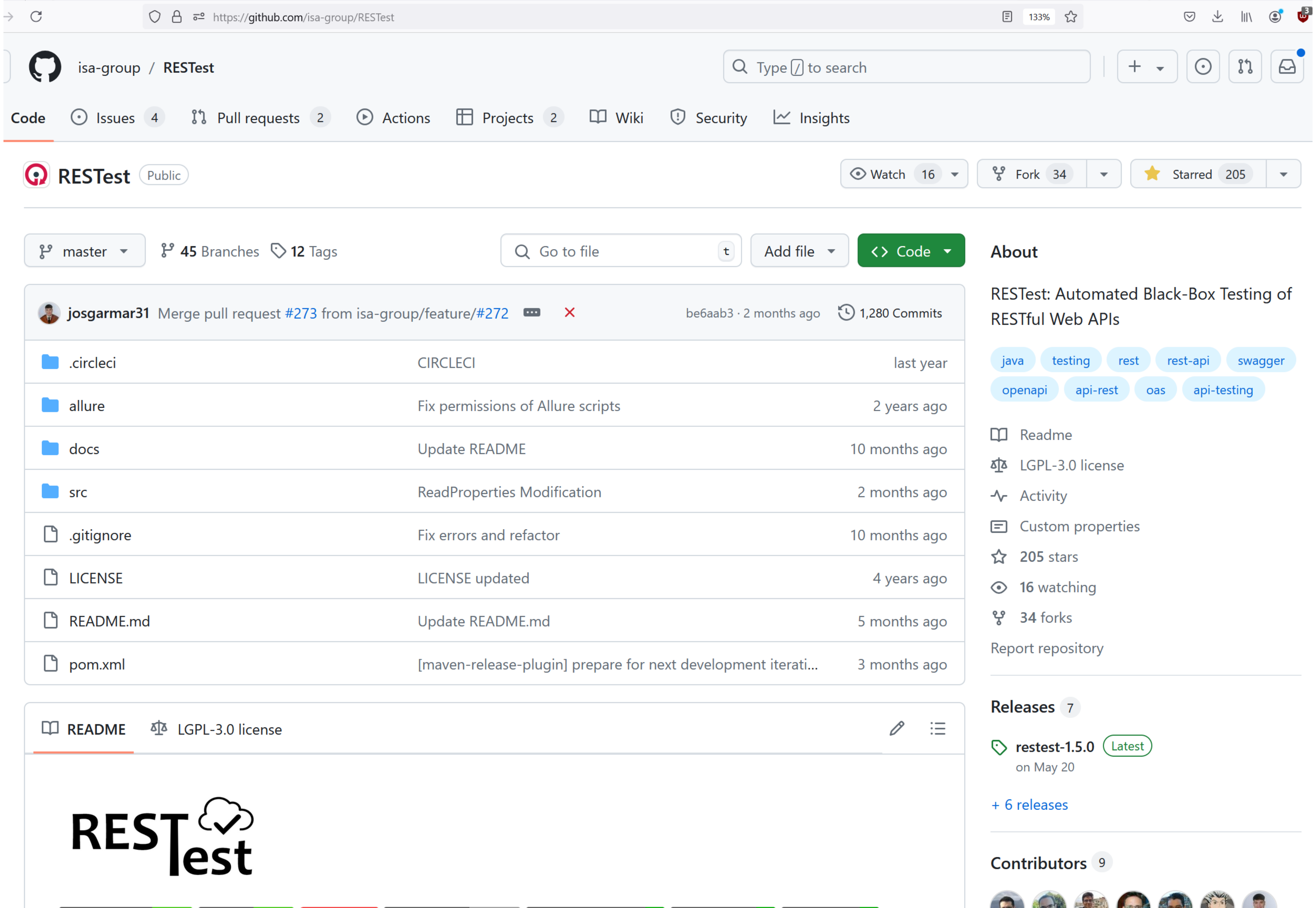
Supercharge your API testing, catch bugs, and ensure compliance

schemathesis.readthedocs.io

testing · graphql · cli · swagger · openapi · pytest · property-based-testing · hypothesis · hacktoberfest · openapi3

Readme
MIT license
Code of conduct
Cite this repository
Activity
Custom properties
2.2k stars
20 watching
153 forks
Report repository

Releases 277



https://github.com/isa-group/RESTest

isa-group / RESTest

Code · Issues 4 · Pull requests 2 · Actions · Projects 2 · Wiki · Security · Insights

RESTest Public

Watch 16 · Fork 34 · Starred 205

master · 45 Branches · 12 Tags · Go to file · Add file · Code

josgarmar31 Merge pull request #273 from isa-group/feature/#272 · be6aab3 · 2 months ago · 1,280 Commits

.circleci	CIRCLECI	last year
allure	Fix permissions of Allure scripts	2 years ago
docs	Update README	10 months ago
src	ReadProperties Modification	2 months ago
.gitignore	Fix errors and refactor	10 months ago
LICENSE	LICENSE updated	4 years ago
README.md	Update README.md	5 months ago
pom.xml	[maven-release-plugin] prepare for next development iterati...	3 months ago

README · LGPL-3.0 license

RESTest

About

RESTest: Automated Black-Box Testing of RESTful Web APIs

java · testing · rest · rest-api · swagger · openapi · api-rest · oas · api-testing

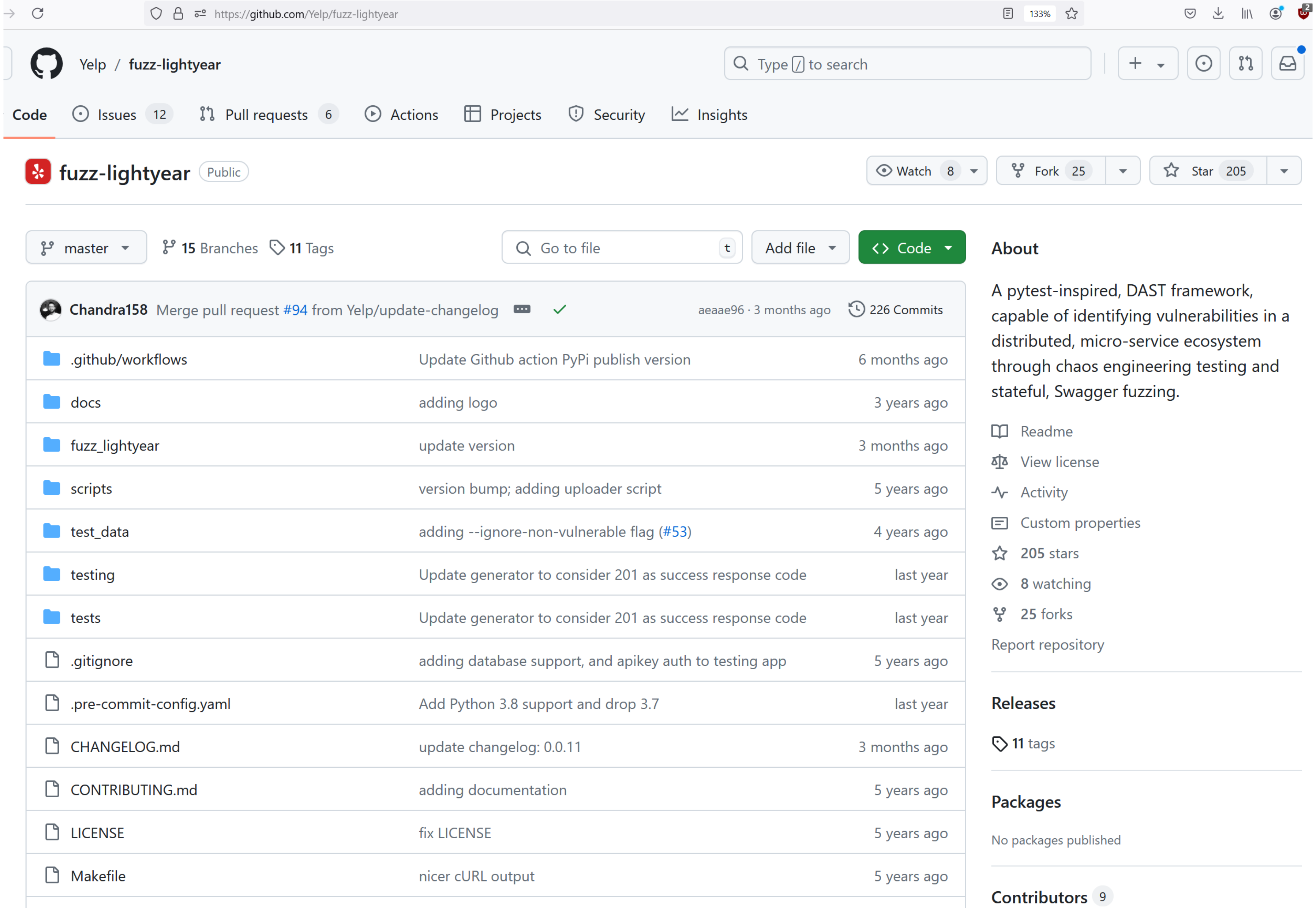
Readme
LGPL-3.0 license
Activity
Custom properties
205 stars
16 watching
34 forks
Report repository

Releases 7

restest-1.5.0 Latest
on May 20

+ 6 releases

Contributors 9



Chandra158 Merge pull request #94 from Yelp/update-changelog ✓ aeaae96 · 3 months ago 226 Commits

.github/workflows	Update Github action PyPi publish version	6 months ago
docs	adding logo	3 years ago
fuzz_lightyear	update version	3 months ago
scripts	version bump; adding uploader script	5 years ago
test_data	adding --ignore-non-vulnerable flag (#53)	4 years ago
testing	Update generator to consider 201 as success response code	last year
tests	Update generator to consider 201 as success response code	last year
.gitignore	adding database support, and apikey auth to testing app	5 years ago
.pre-commit-config.yaml	Add Python 3.8 support and drop 3.7	last year
CHANGELOG.md	update changelog: 0.0.11	3 months ago
CONTRIBUTING.md	adding documentation	5 years ago
LICENSE	fix LICENSE	5 years ago
Makefile	nicer cURL output	5 years ago

About

A pytest-inspired, DAST framework, capable of identifying vulnerabilities in a distributed, micro-service ecosystem through chaos engineering testing and stateful, Swagger fuzzing.

Readme
View license
Activity
Custom properties
205 stars
8 watching
25 forks
Report repository

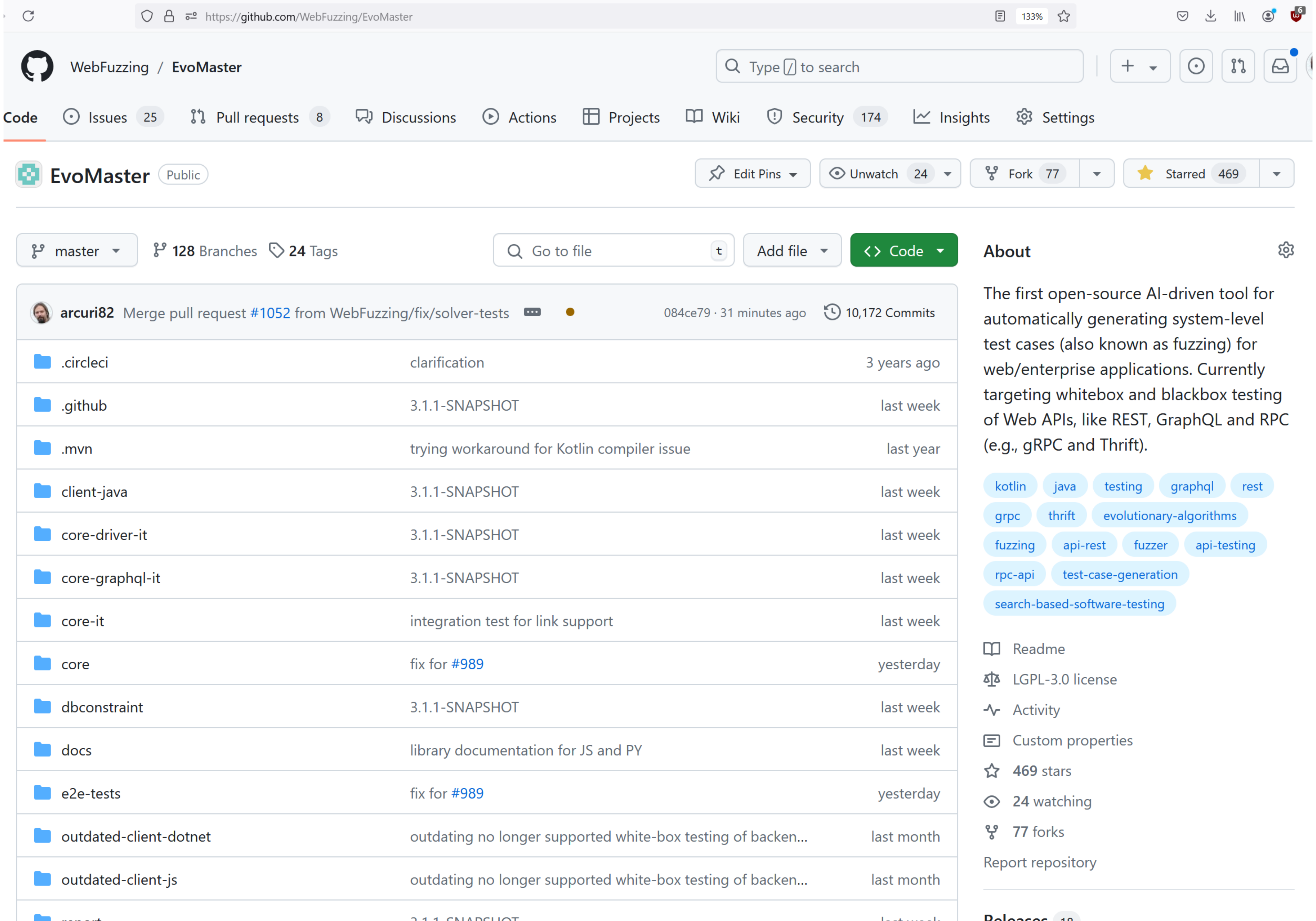
Releases

11 tags

Packages

No packages published

Contributors 9



https://github.com/WebFuzzing/EvoMaster

WebFuzzing / EvoMaster

Code · Issues 25 · Pull requests 8 · Discussions · Actions · Projects · Wiki · Security 174 · Insights · Settings

EvoMaster Public

Edit Pins · Unwatch 24 · Fork 77 · Starred 469

master · 128 Branches · 24 Tags · Go to file · Add file · Code

arcuri82 Merge pull request #1052 from WebFuzzing/fix/solver-tests · 084ce79 · 31 minutes ago · 10,172 Commits

.circleci	clarification	3 years ago
.github	3.1.1-SNAPSHOT	last week
.mvn	trying workaround for Kotlin compiler issue	last year
client-java	3.1.1-SNAPSHOT	last week
core-driver-it	3.1.1-SNAPSHOT	last week
core-graphql-it	3.1.1-SNAPSHOT	last week
core-it	integration test for link support	last week
core	fix for #989	yesterday
dbconstraint	3.1.1-SNAPSHOT	last week
docs	library documentation for JS and PY	last week
e2e-tests	fix for #989	yesterday
outdated-client-dotnet	outdating no longer supported white-box testing of backen...	last month
outdated-client-js	outdating no longer supported white-box testing of backen...	last month

About

The first open-source AI-driven tool for automatically generating system-level test cases (also known as fuzzing) for web/enterprise applications. Currently targeting whitebox and blackbox testing of Web APIs, like REST, GraphQL and RPC (e.g., gRPC and Thrift).

kotlin · java · testing · graphql · rest · grpc · thrift · evolutionary-algorithms · fuzzing · api-rest · fuzzer · api-testing · rpc-api · test-case-generation · search-based-software-testing

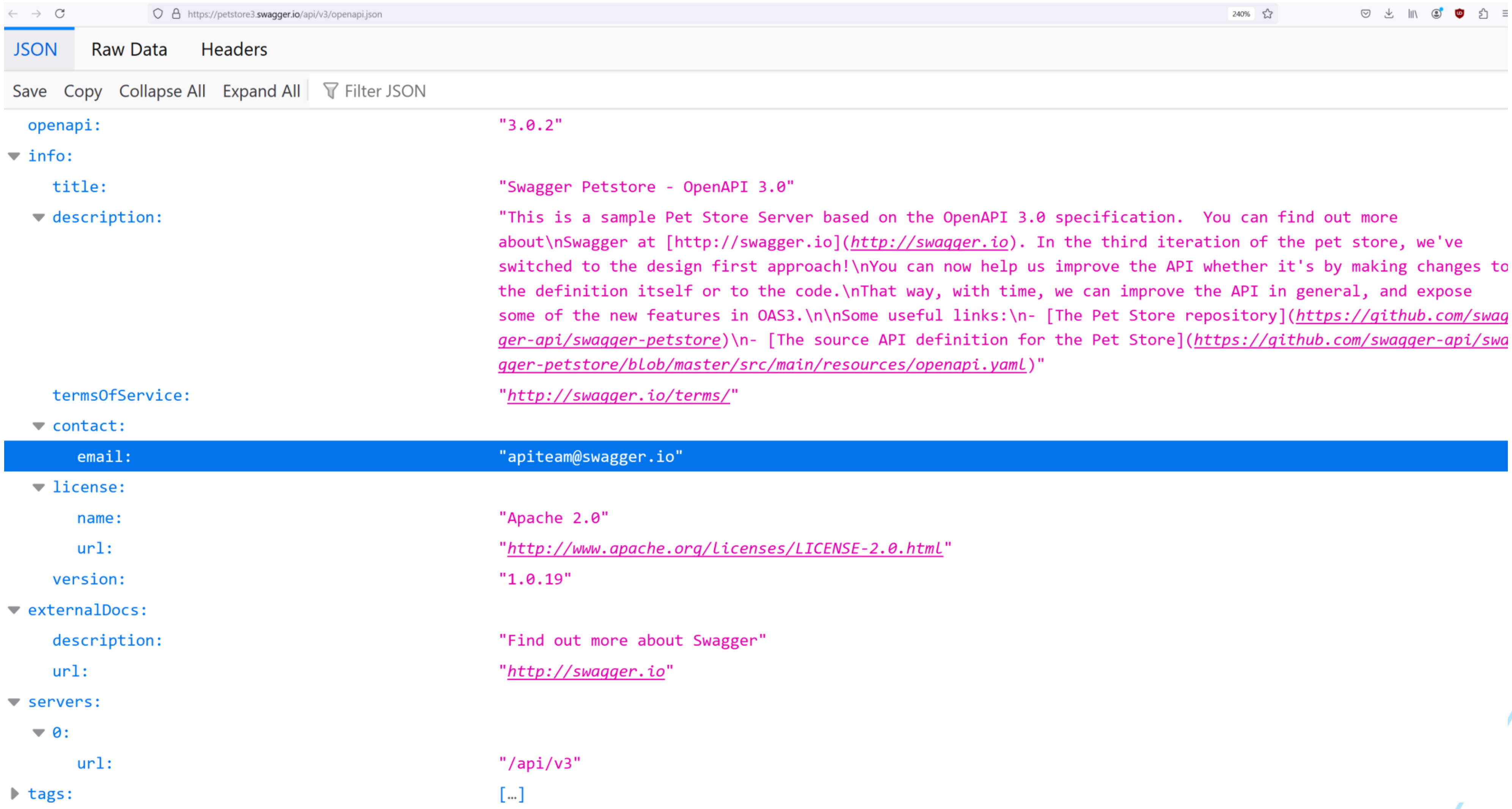
Readme
LGPL-3.0 license
Activity
Custom properties
469 stars
24 watching
77 forks
Report repository

Input: OpenAPI/Swagger Schema

- Need to know what endpoints are available, and their parameters
- Schema defining the APIs
- OpenAPI is the most popular one
- Defined as JSON file, or YAML

Example: PetStore

Online schema at https://petstore3.swagger.io/api/v3/openapi.json


https://petstore3.swagger.io/api/v3/openapi.json

JSON Raw Data Headers

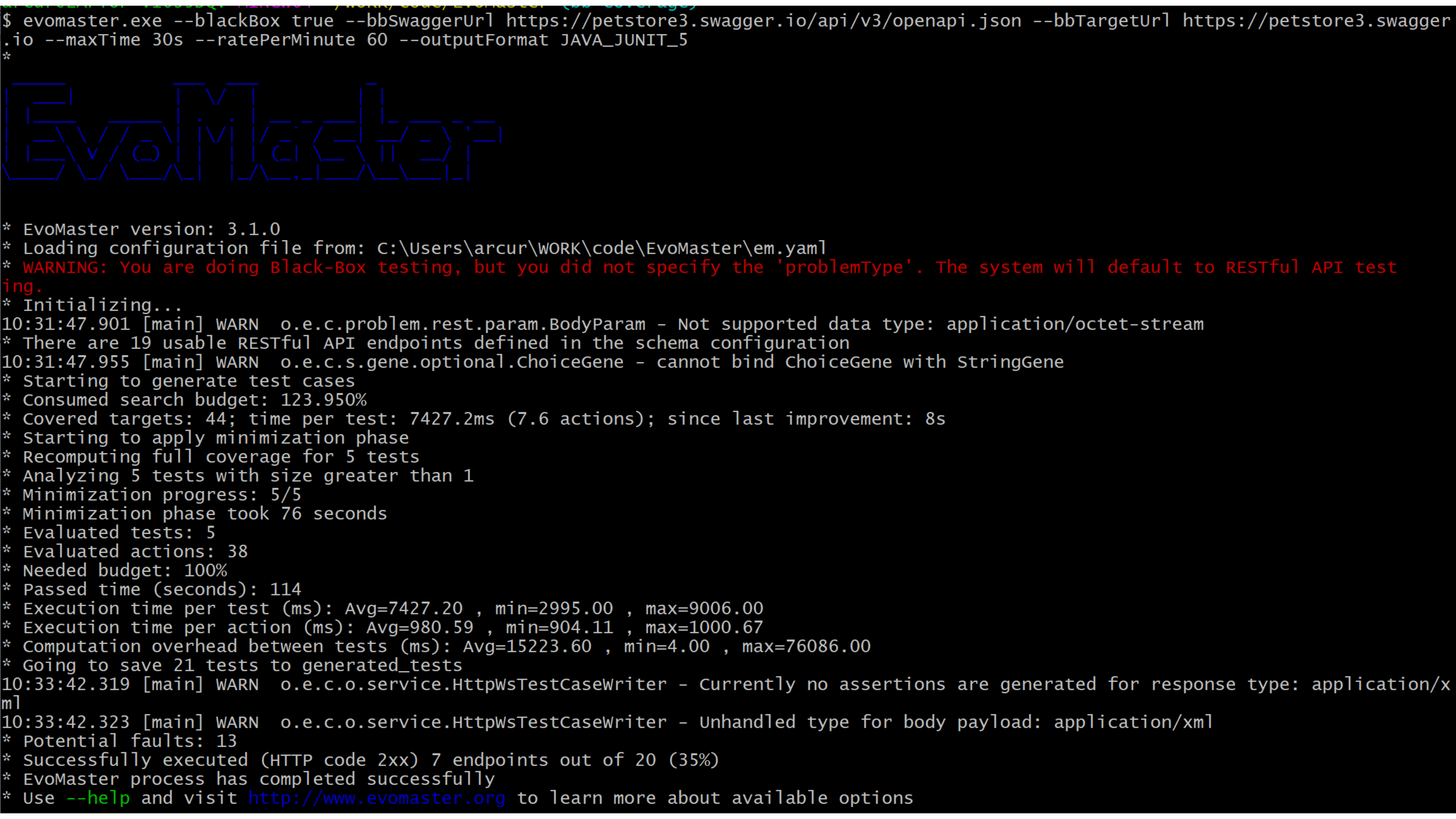
Save Copy Collapse All Expand All Filter JSON

```
openapi:                      "3.0.2"
▼ info:
    title:                    "Swagger Petstore - OpenAPI 3.0"
  ▼ description:              "This is a sample Pet Store Server based on the OpenAPI 3.0 specification.  You can find out more
                              about\nSwagger at [http://swagger.io](http://swagger.io). In the third iteration of the pet store, we've
                              switched to the design first approach!\nYou can now help us improve the API whether it's by making changes to
                              the definition itself or to the code.\nThat way, with time, we can improve the API in general, and expose
                              some of the new features in OAS3.\n\nSome useful links:\n- [The Pet Store repository](https://github.com/swag
                              ger-api/swagger-petstore)\n- [The source API definition for the Pet Store](https://github.com/swagger-api/swa
                              gger-petstore/blob/master/src/main/resources/openapi.yaml)"
    termsOfService:           "http://swagger.io/terms/"
  ▼ contact:
      email:                  "apiteam@swagger.io"
  ▼ license:
      name:                   "Apache 2.0"
      url:                    "http://www.apache.org/licenses/LICENSE-2.0.html"
    version:                  "1.0.19"
▼ externalDocs:
    description:              "Find out more about Swagger"
    url:                      "http://swagger.io"
▼ servers:
  ▼ 0:
      url:                    "/api/v3"
▶ tags:                       […]
```

#AutomationSTAR

What Can Expect?

- All these tools will analyze the schema
- Send requests with many different strategies
  - there is lot of research in academia on this
- Check if any error in the API can be identified
- Output executable test cases
  - in different formats, eg Java and Python



```
$ evomaster.exe --blackBox true --bbSwaggerUrl https://petstore3.swagger.io/api/v3/openapi.json --bbTargetUrl https://petstore3.swagger
.io --maxTime 30s --ratePerMinute 60 --outputFormat JAVA_JUNIT_5
*
 _____            ___  ___          _
|  ___|           |  \/  |         | |
| |____   _____   | .  . | __ _ ___| |_ ___ _ __
|  __\ \ / / _ \  | |\/| |/ _` / __| __/ _ \ '__|
| |___\ V / (_) | | |  | | (_| \__ \ ||  __/ |
\____/ \_/ \___/  \_|  |_/\__,_|___/\__\___|_|


* EvoMaster version: 3.1.0
* Loading configuration file from: C:\Users\arcur\WORK\code\EvoMaster\em.yaml
* WARNING: You are doing Black-Box testing, but you did not specify the 'problemType'. The system will default to RESTful API test
ing.
* Initializing...
10:31:47.901 [main] WARN  o.e.c.problem.rest.param.BodyParam - Not supported data type: application/octet-stream
* There are 19 usable RESTful API endpoints defined in the schema configuration
10:31:47.955 [main] WARN  o.e.c.s.gene.optional.ChoiceGene - cannot bind ChoiceGene with StringGene
* Starting to generate test cases
* Consumed search budget: 123.950%
* Covered targets: 44; time per test: 7427.2ms (7.6 actions); since last improvement: 8s
* Starting to apply minimization phase
* Recomputing full coverage for 5 tests
* Analyzing 5 tests with size greater than 1
* Minimization progress: 5/5
* Minimization phase took 76 seconds
* Evaluated tests: 5
* Evaluated actions: 38
* Needed budget: 100%
* Passed time (seconds): 114
* Execution time per test (ms): Avg=7427.20 , min=2995.00 , max=9006.00
* Execution time per action (ms): Avg=980.59 , min=904.11 , max=1000.67
* Computation overhead between tests (ms): Avg=15223.60 , min=4.00 , max=76086.00
* Going to save 21 tests to generated_tests
10:33:42.319 [main] WARN  o.e.c.o.service.HttpWsTestCaseWriter - Currently no assertions are generated for response type: application/x
ml
10:33:42.323 [main] WARN  o.e.c.o.service.HttpWsTestCaseWriter - Unhandled type for body payload: application/xml
* Potential faults: 13
* Successfully executed (HTTP code 2xx) 7 endpoints out of 20 (35%)
* EvoMaster process has completed successfully
* Use --help and visit http://www.evomaster.org to learn more about available options
```

Success Calls: Random but Valid Data

```
@Test @Timeout(60)
public void test_1() throws Exception {

    given().accept("application/xml")
            .contentType("application/json")
            .body(" { " +
                " \"id\": 940, " +
                " \"name\": \"doggie\", " +
                " \"photoUrls\": [ " +
                " \"yHQXry\", " +
                " \"AZOgWb5y\", " +
                " \"GROBCmON\" " +
                " ], " +
                " \"tags\": [ " +
                " {}, " +
                " { " +
                " \"name\": \"nosupgc\" " +
                " } " +
                " ], " +
                " \"status\": \"pending\" " +
                " } ")
            .post(baseUrlOfSut + "/api/v3/pet")
            .then()
            .statusCode(200)
            .assertThat()
            .contentType("application/xml");
}
```

#AutomationSTAR

Crashing with 500

```java
@Test @Timeout(60)
public void test_4_with500() throws Exception {
   ExpectationHandler expectationHandler = expectationHandler();

   ValidatableResponse res_0 = given().accept("application/xml")
         .get(baseUrlOfSut + "/api/v3/user/8WlY1")
         .then()
         .statusCode(500)
         .assertThat()
         .contentType("application/xml");

   expectationHandler.expect(ems)
      .that(sco, Arrays.asList(200, 400, 404).contains(res_0.extract().statusCode()));
}
```

Invalid response (eg status code not declared in schema)

```
@Test @Timeout(60)
public void test_8() throws Exception {
    ExpectationHandler expectationHandler = expectationHandler();

    ValidatableResponse res_0 = given().accept("application/json")
            .contentType("application/json")
            .body(" null ")
            .post(baseUrlOfSut + "/api/v3/store/order")
            .then()
            .statusCode(400)
            .assertThat()
            .contentType("application/json")
            .body(containsString("No Order provided. Try again?"));

    expectationHandler.expect(ems)
        .that(sco, Arrays.asList(200, 405).contains(res_0.extract().statusCode()));
}
```

Experience With EvoMaster

- Author’s of EvoMaster
- Academic tool, started in 2016
  - Around 3M Euro in funding from ERC and NFR
- Applied on many open-source APIs
  - found thousands of bugs
- Only tool supporting *white-box* testing
  - but only for JVM
- Academic collaborations with industry

EvoMaster at Meituan

- Fortune500 Chinese enterprise
- EvoMaster used daily on hundreds of microservices, for millions of lines of code
- White-box testing Thrift RPC APIs

美团是一家科技零售公司

美团以“零售 + 科技”的战略践行“帮大家吃得更好，生活更好”的公司使命。

自2010年3月成立以来，美团持续推动服务零售和商品零售在需求侧和供给侧的数字化升级，和广大合作伙伴一起努力为消费者提供品质服务。2018年9月20日，美团在港交所挂牌上市。美团始终以客户为中心，不断加大在新技术上的研发投入。美团会和大家一起努力，更好承担社会责任，更多创造社会价值。

EvoMaster at Volkswagen

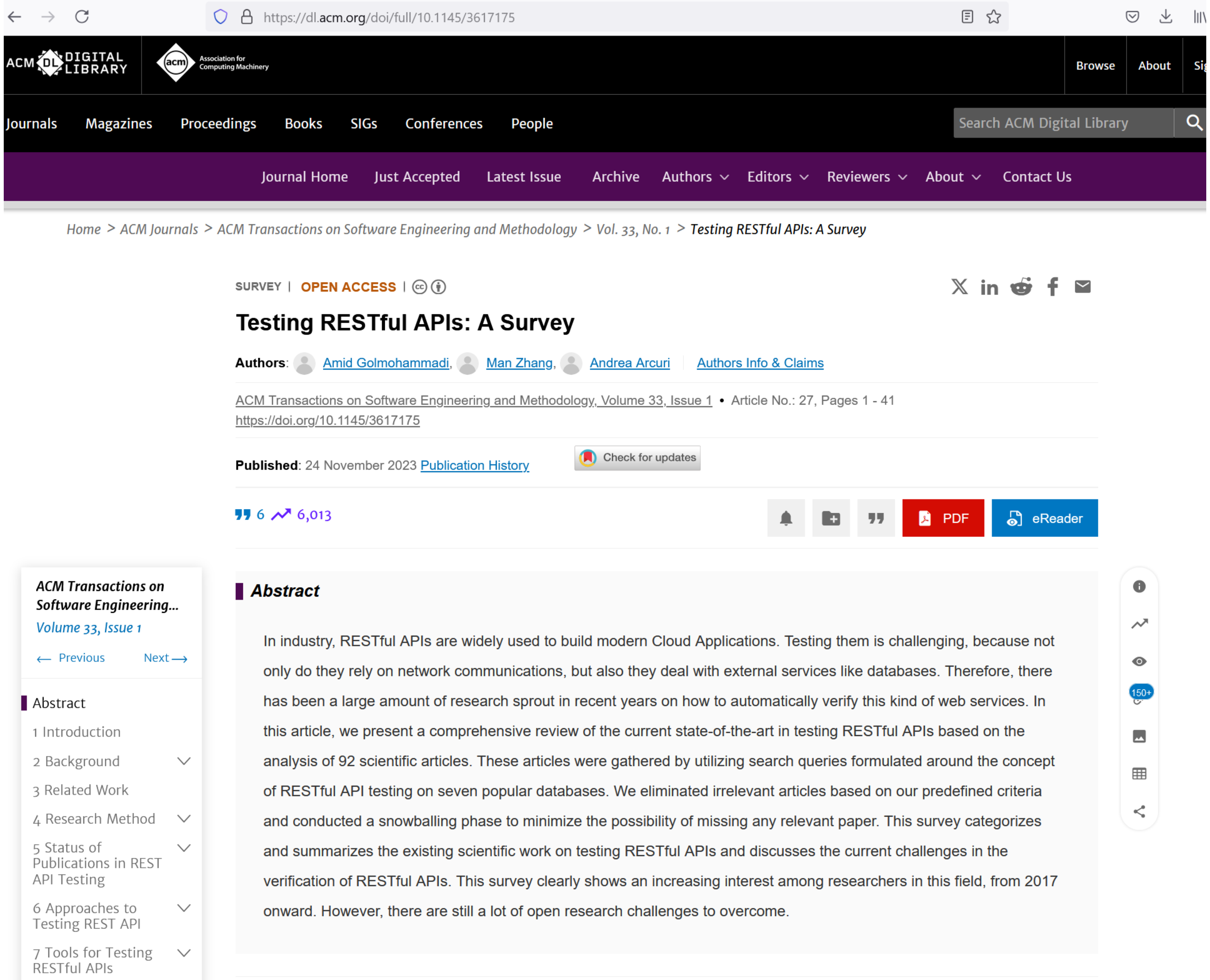
- Fortune500 German automobile manufacturer
- Recent collaboration
- Black-box fuzzing REST APIs
- Still in evaluation phase
  - some needed features are still under development

https://www.volkswagen.no/no.html
Meny
Velg forhandler
Les mer
Bygg din ID.7 GTX stasjonsvogn

ID.7 GTX stasjonsvogn med fire-hjulstrekk: Fra kr 579 400

Challenges

- Lot of research in academia for better test generation strategies
- Cover larger parts of API code
- Find more faults (and fault types)
    - not all faults have same severity
- Test readability
    - testers still need to look at generated tests



Home > ACM Journals > ACM Transactions on Software Engineering and Methodology > Vol. 33, No. 1 > Testing RESTful APIs: A Survey

SURVEY | OPEN ACCESS |

Testing RESTful APIs: A Survey

Authors: Amid Golmohammadi, Man Zhang, Andrea Arcuri | Authors Info & Claims

ACM Transactions on Software Engineering and Methodology, Volume 33, Issue 1 • Article No.: 27, Pages 1 - 41
https://doi.org/10.1145/3617175

Published: 24 November 2023 Publication History

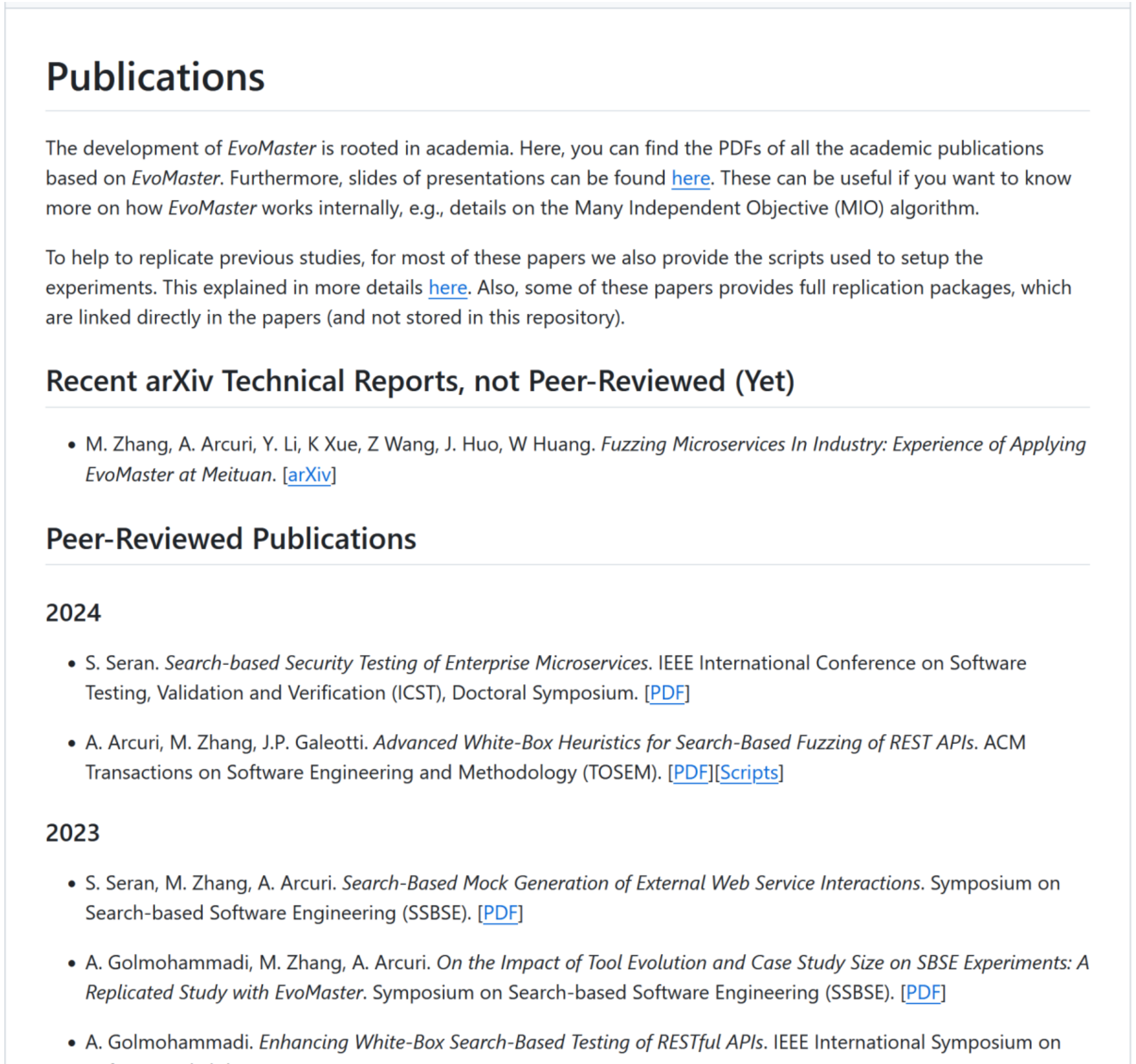
ACM Transactions on Software Engineering...
Volume 33, Issue 1
← Previous Next →

Abstract
1 Introduction
2 Background
3 Related Work
4 Research Method
5 Status of Publications in REST API Testing
6 Approaches to Testing REST API
7 Tools for Testing RESTful APIs

Abstract

In industry, RESTful APIs are widely used to build modern Cloud Applications. Testing them is challenging, because not only do they rely on network communications, but also they deal with external services like databases. Therefore, there has been a large amount of research sprout in recent years on how to automatically verify this kind of web services. In this article, we present a comprehensive review of the current state-of-the-art in testing RESTful APIs based on the analysis of 92 scientific articles. These articles were gathered by utilizing search queries formulated around the concept of RESTful API testing on seven popular databases. We eliminated irrelevant articles based on our predefined criteria and conducted a snowballing phase to minimize the possibility of missing any relevant paper. This survey categorizes and summarizes the existing scientific work on testing RESTful APIs and discusses the current challenges in the verification of RESTful APIs. This survey clearly shows an increasing interest among researchers in this field, from 2017 onward. However, there are still a lot of open research challenges to overcome.

https://github.com/WebFuzzing/EvoMaster/blob/master/docs/publications.md



Publications

The development of *EvoMaster* is rooted in academia. Here, you can find the PDFs of all the academic publications based on *EvoMaster*. Furthermore, slides of presentations can be found here. These can be useful if you want to know more on how *EvoMaster* works internally, e.g., details on the Many Independent Objective (MIO) algorithm.

To help to replicate previous studies, for most of these papers we also provide the scripts used to setup the experiments. This explained in more details here. Also, some of these papers provides full replication packages, which are linked directly in the papers (and not stored in this repository).

Recent arXiv Technical Reports, not Peer-Reviewed (Yet)

- M. Zhang, A. Arcuri, Y. Li, K Xue, Z Wang, J. Huo, W Huang. *Fuzzing Microservices In Industry: Experience of Applying EvoMaster at Meituan*. [arXiv]

Peer-Reviewed Publications

2024

- S. Seran. *Search-based Security Testing of Enterprise Microservices*. IEEE International Conference on Software Testing, Validation and Verification (ICST), Doctoral Symposium. [PDF]
- A. Arcuri, M. Zhang, J.P. Galeotti. *Advanced White-Box Heuristics for Search-Based Fuzzing of REST APIs*. ACM Transactions on Software Engineering and Methodology (TOSEM). [PDF][Scripts]

2023

- S. Seran, M. Zhang, A. Arcuri. *Search-Based Mock Generation of External Web Service Interactions*. Symposium on Search-based Software Engineering (SSBSE). [PDF]
- A. Golmohammadi, M. Zhang, A. Arcuri. *On the Impact of Tool Evolution and Case Study Size on SBSE Experiments: A Replicated Study with EvoMaster*. Symposium on Search-based Software Engineering (SSBSE). [PDF]
- A. Golmohammadi. *Enhancing White-Box Search-Based Testing of RESTful APIs*. IEEE International Symposium on

If you want to go into the low-level details of how these techniques work

#AutomationSTAR

Conclusion

- Many success stories about fuzzing
- REST fuzzing (and partially GraphQL and RPC) is getting momentum
- *Several open-source tools are available, to try out, today!*
  - we are biased about EvoMaster, but Schemathesis and Restler are good alternatives

Questions?

Email andrea.arcuri@kristiani.no

Social www.linkedin.com/in/arcuri82/