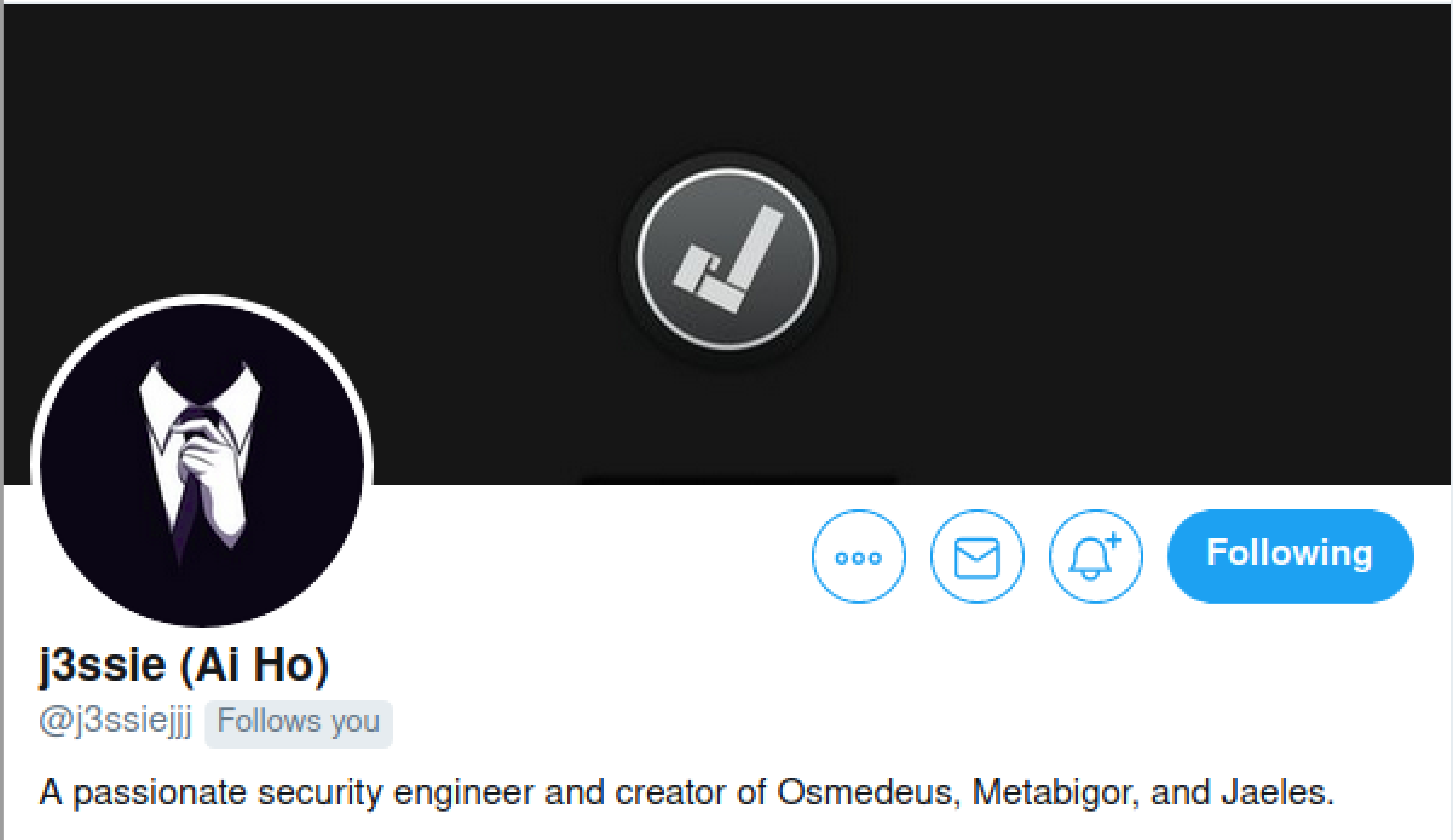
OWASP Tbilisi Chapter 3rd Meeting

Jaeles - The Swiss Army knife for automated Web Application Testing

@GochaOqradze

jaeles ავტორის შესახებ



j3ssie (Ai Ho)

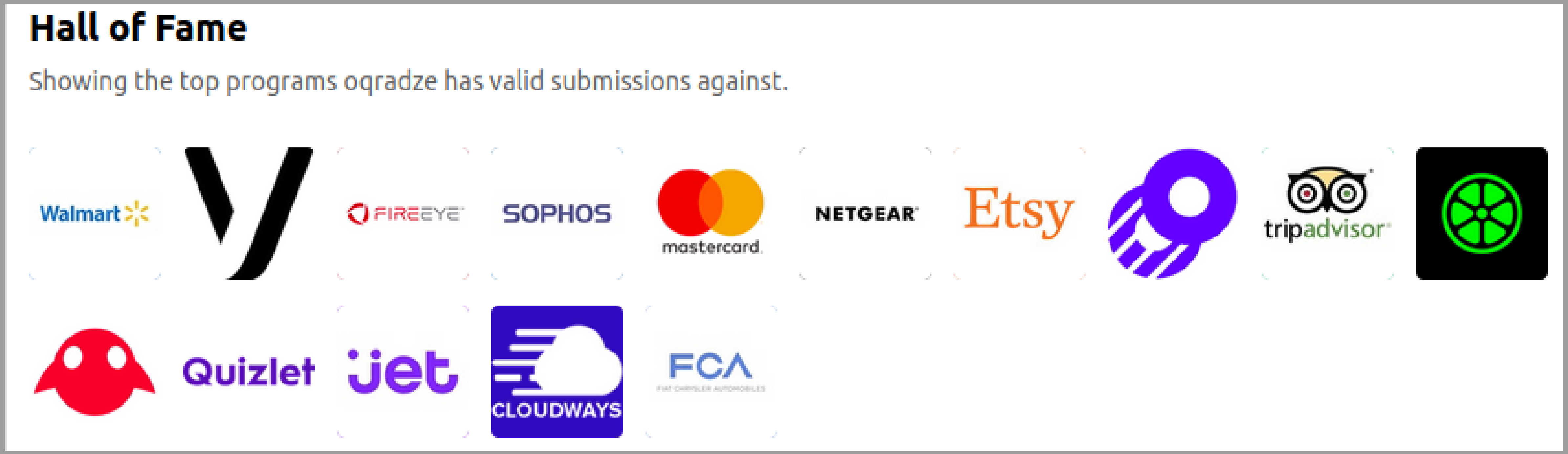
@j3ssiejjj Follows you

A passionate security engineer and creator of Osmedeus, Metabigor, and Jaeles.

https://github.com/jaeles-project

whoami:

- @GochaOqradze
- დამწყები Bug Bounty hunter
- Dji დრონების პილოტი
- https://bugcrowd.com/oqradze



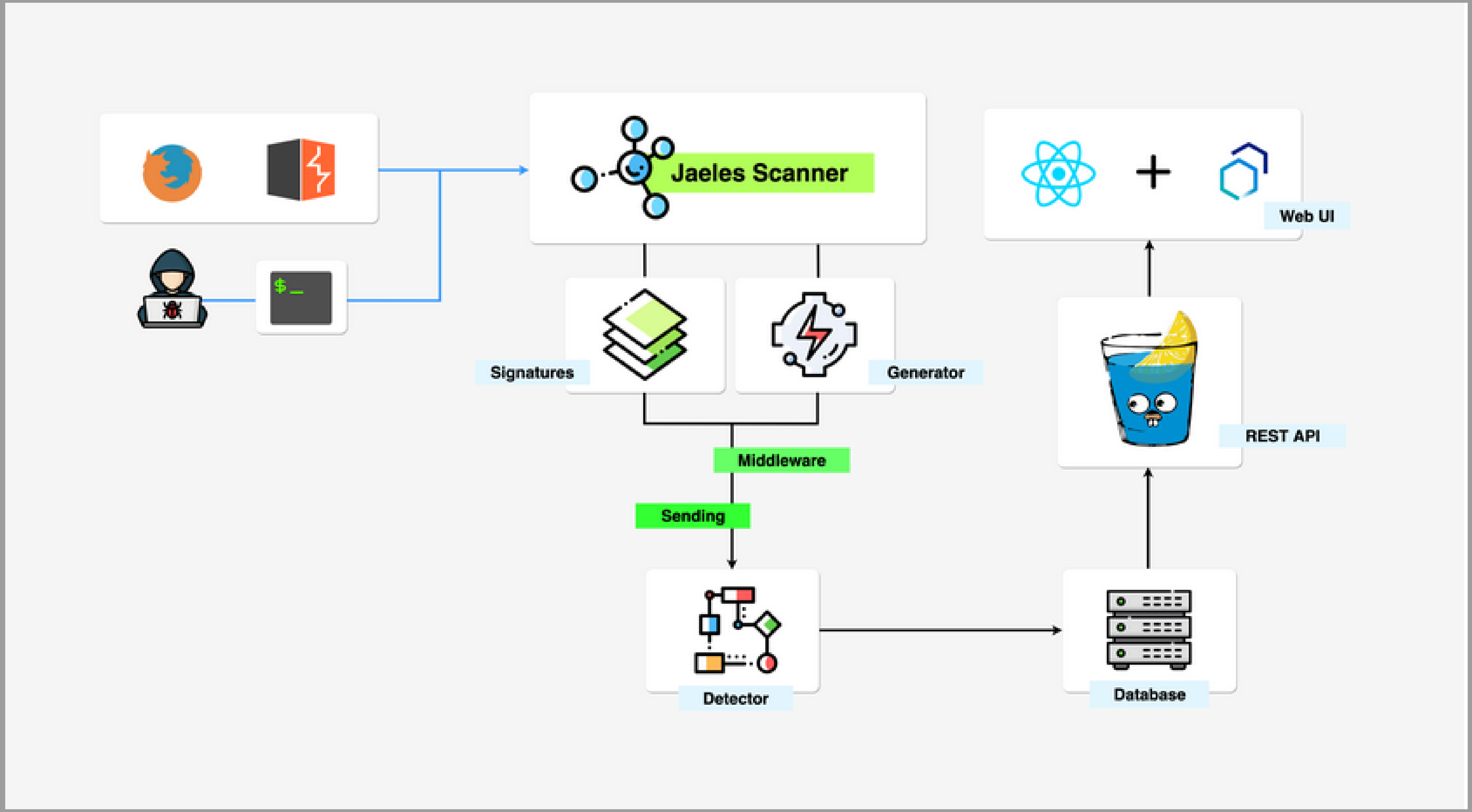
Hall of Fame

Showing the top programs oqradze has valid submissions against.

https://github.com/ghsec/webHunt https://github.com/ghsec/ghsec-jaeles-signatures

Outline

- რატომ?
- არქიტექტურა
- პრაქტიკა



https://github.com/jaeles-project/jaeles

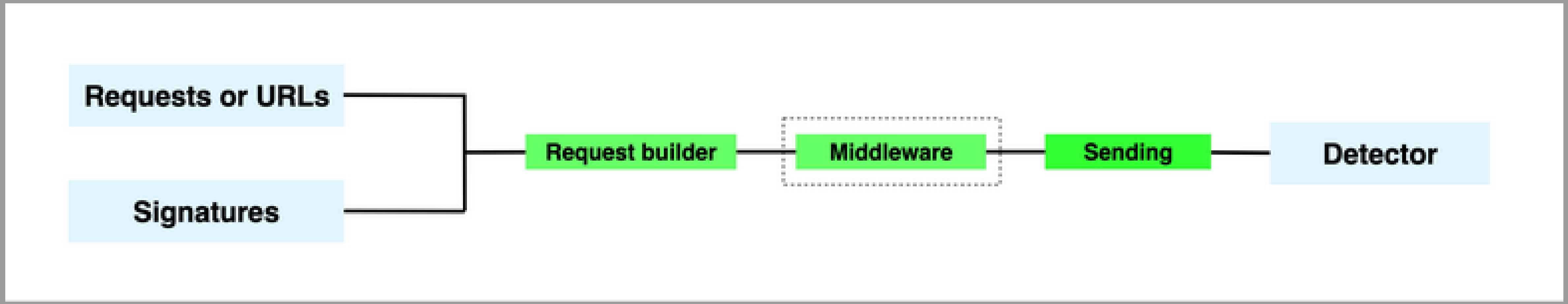
რატომ jaeles?

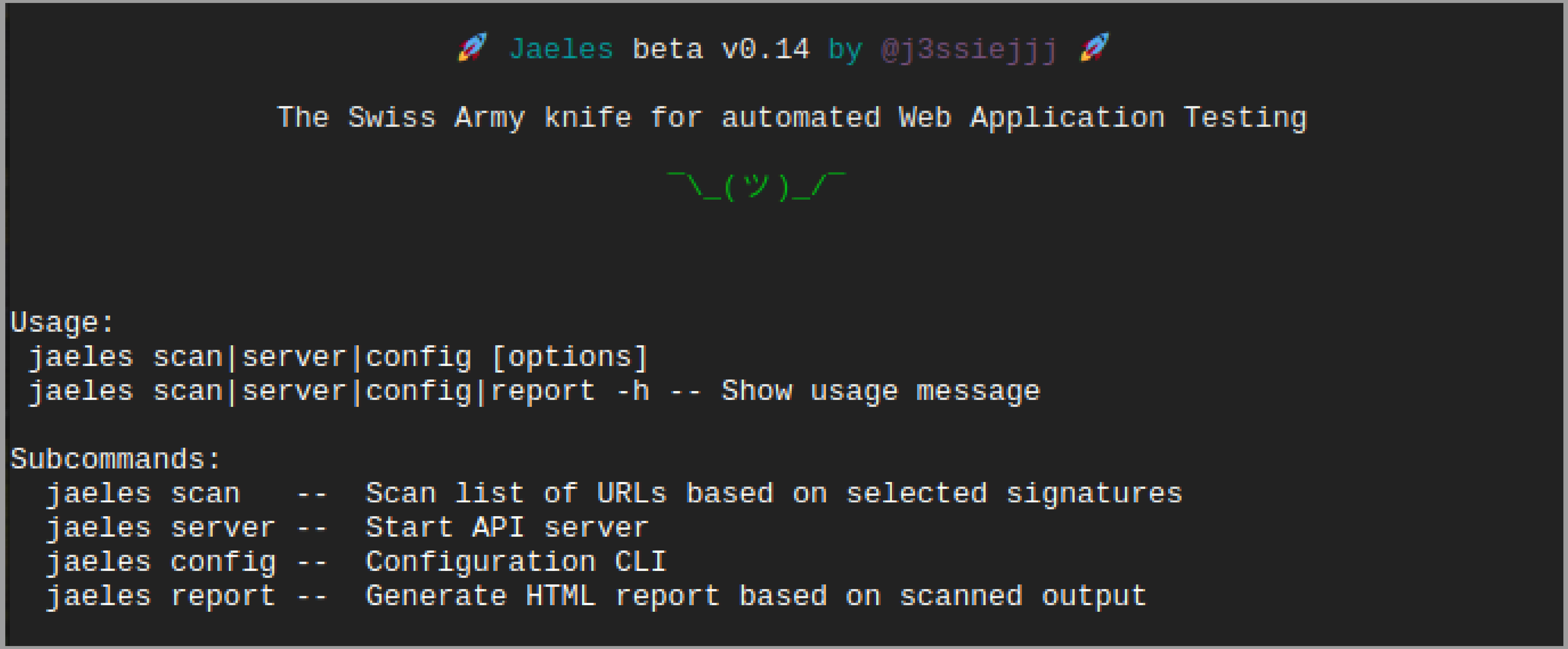
- შექმა სკანერი რომელიც გაძლევს მეტ შესაძლებლობებს.
- შეამოწმო ერთი ან რამოდენიმე {{.host}} -ი სხვადა სხვა სისუსტეებზე
- შესაძლებელია მარტივად გაავრცელო შექმნილი სიგნატურები
- შექმნა სკანერი რომელსაც სრულიად გააკონტროლო
- მოახდინო სხვა ხელსაწყოებთან ინტეგრაცია მარტივად

შედეგი დამოკიდებულია "{{.skill}}"-ებზე

- შეამოწმო CVE-ზე
- Fuzzing-ი
- Directory bruteForce / Content Discovery
- ტექნოლოგიების ანაბეჭდები "FingerPrint"
- Probing HTTP
- მონიტორინგი
- მეტი სხვა

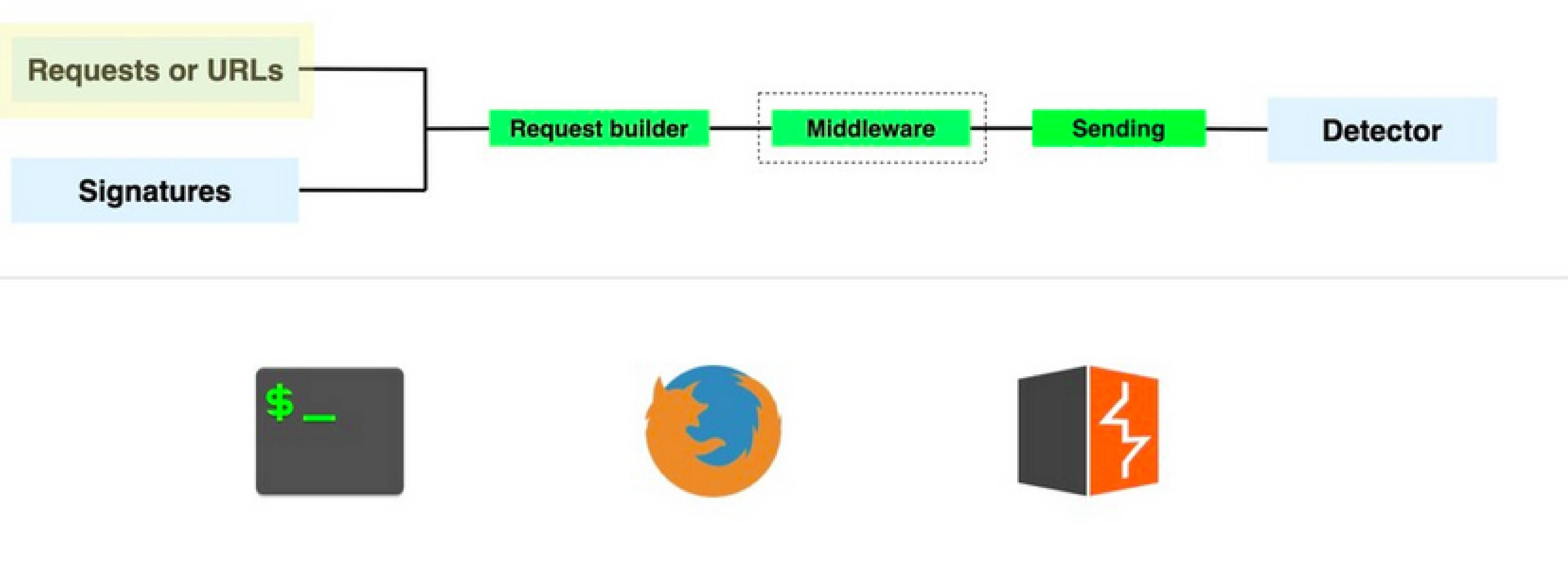
არქიტექტურა





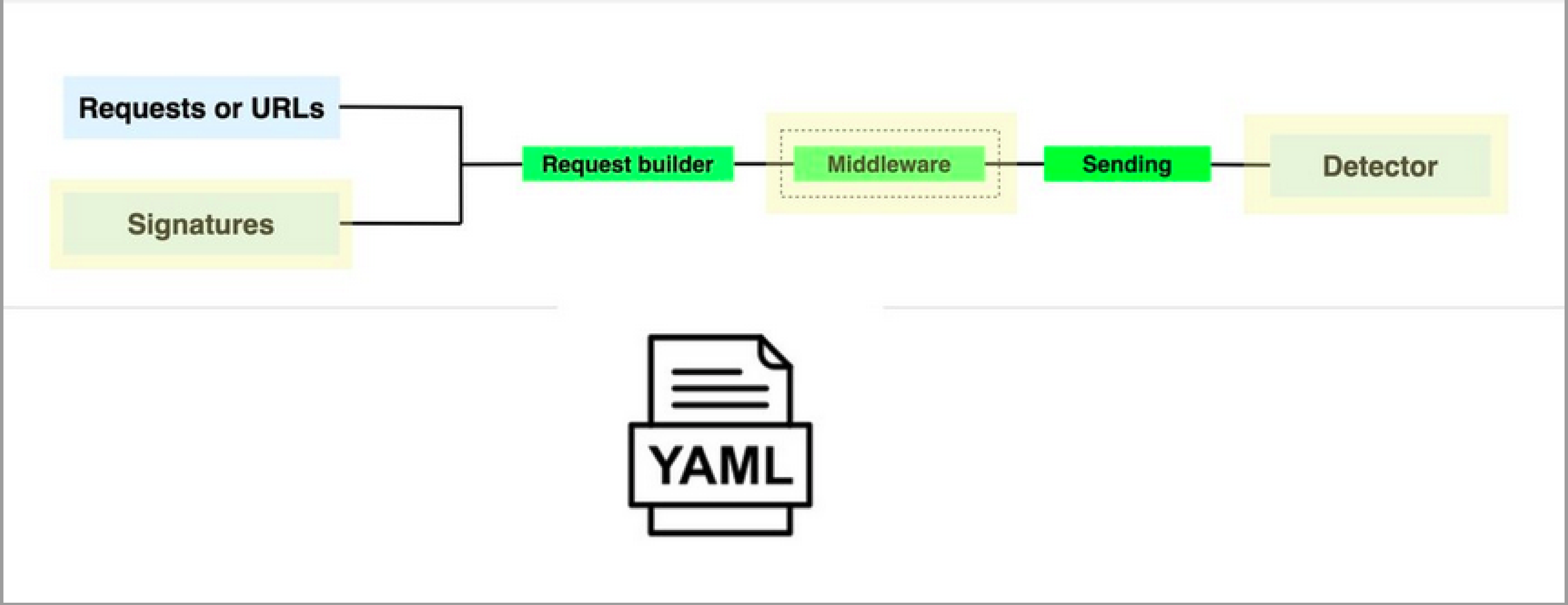
```
                    🚀 Jaeles beta v0.14 by @j3ssiejjj 🚀

            The Swiss Army knife for automated Web Application Testing

                                ¯\_(ツ)_/¯


Usage:
 jaeles scan|server|config [options]
 jaeles scan|server|config|report -h -- Show usage message

Subcommands:
  jaeles scan   --  Scan list of URLs based on selected signatures
  jaeles server --  Start API server
  jaeles config --  Configuration CLI
  jaeles report --  Generate HTML report based on scanned output
```

არქიტექტურა



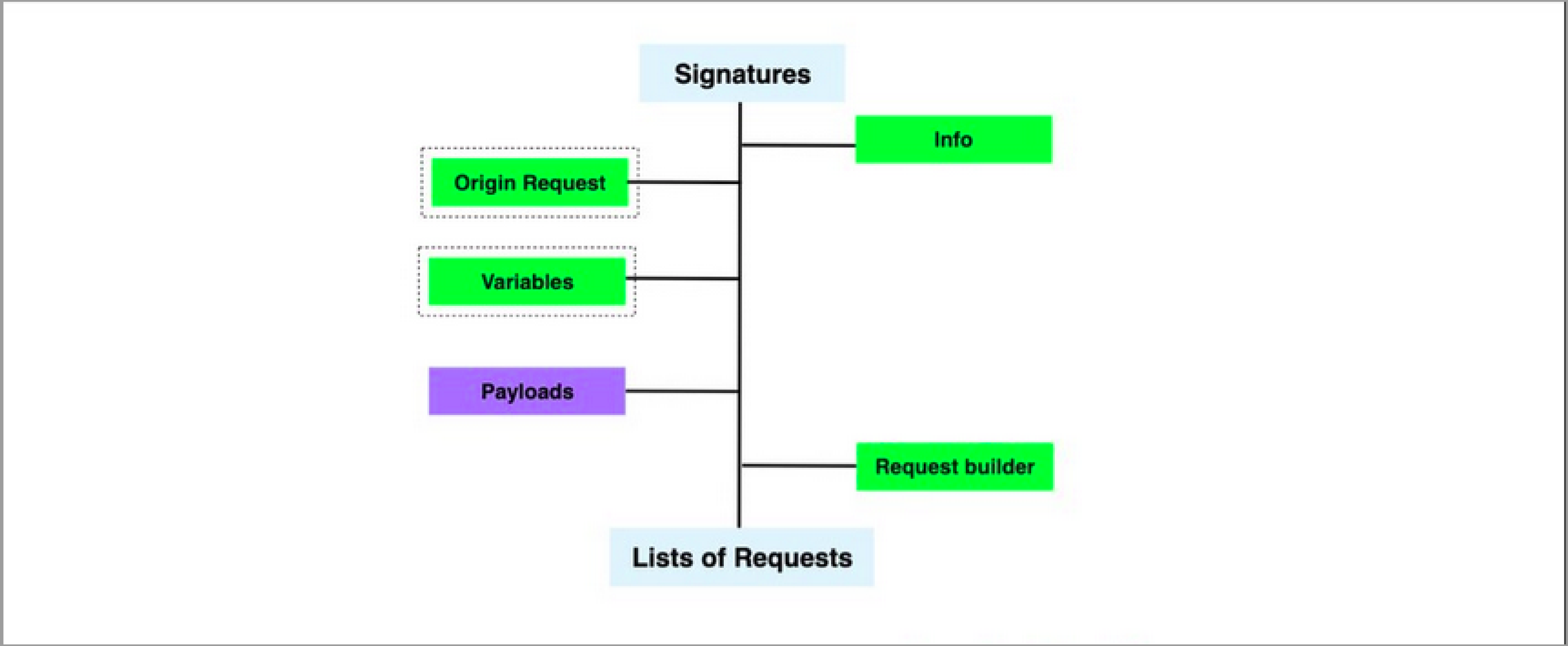
request-ის ან URL -ის დამუშავება სხვა და სხვა გზით

არქიტექტურა



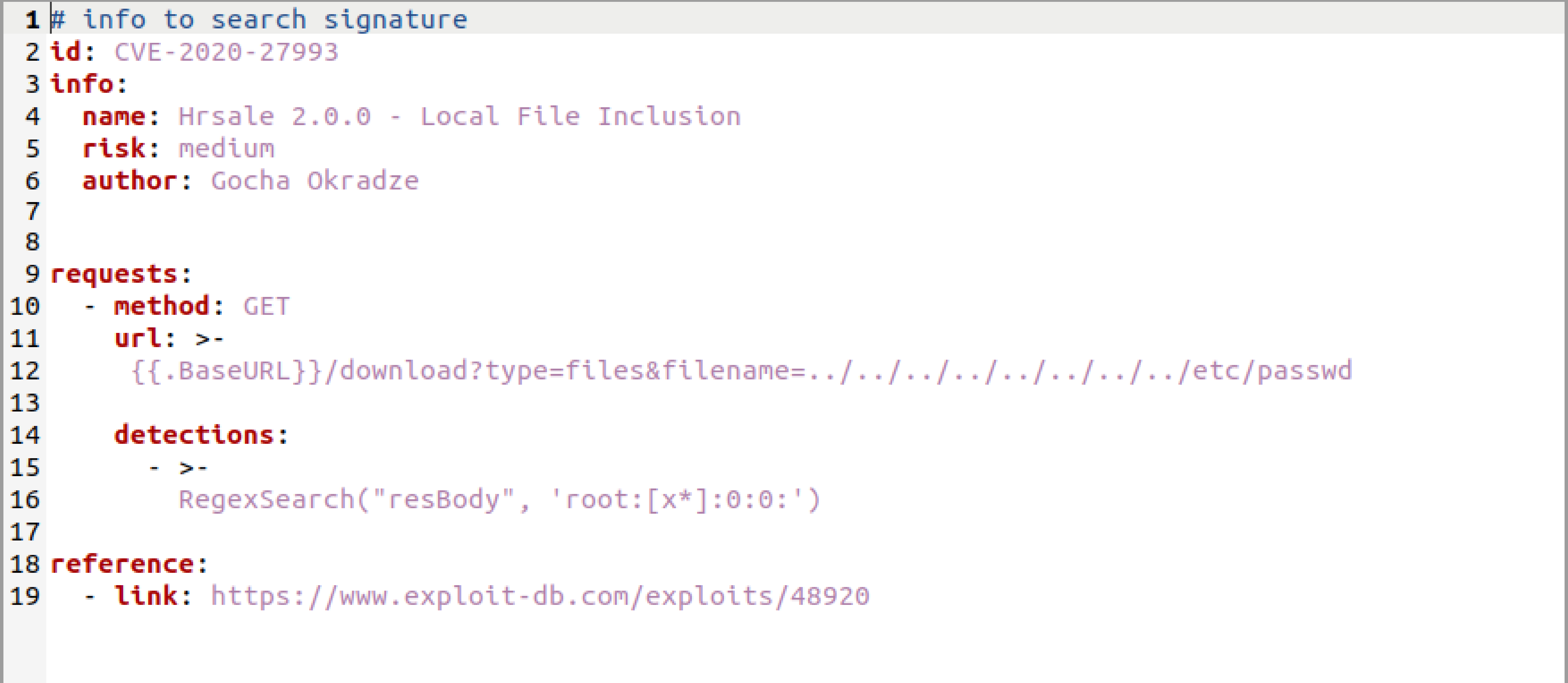
სიგნატურები იწერება YAML ფორმატში

სიგნატურა



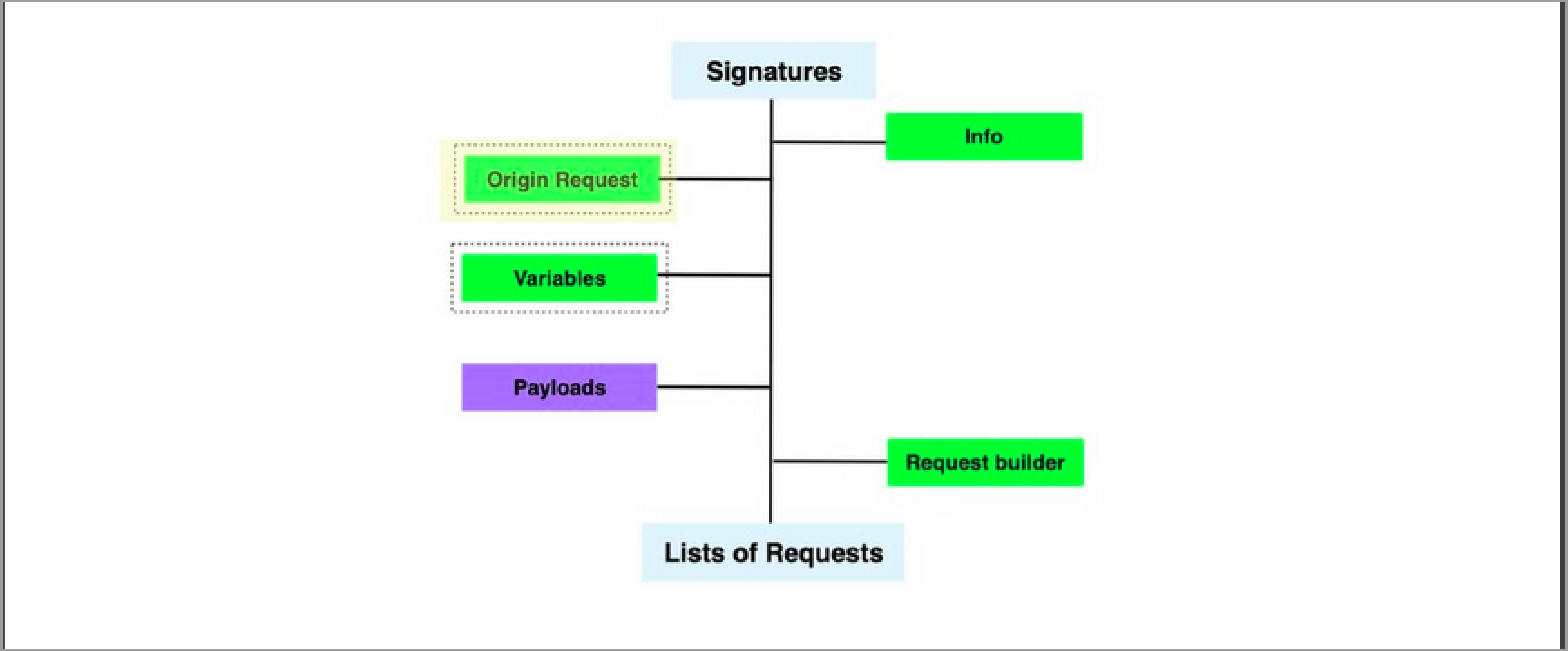
3 ტიპის სიგნატურა: single, list, fuzz

სიგნატურა {{.info}}

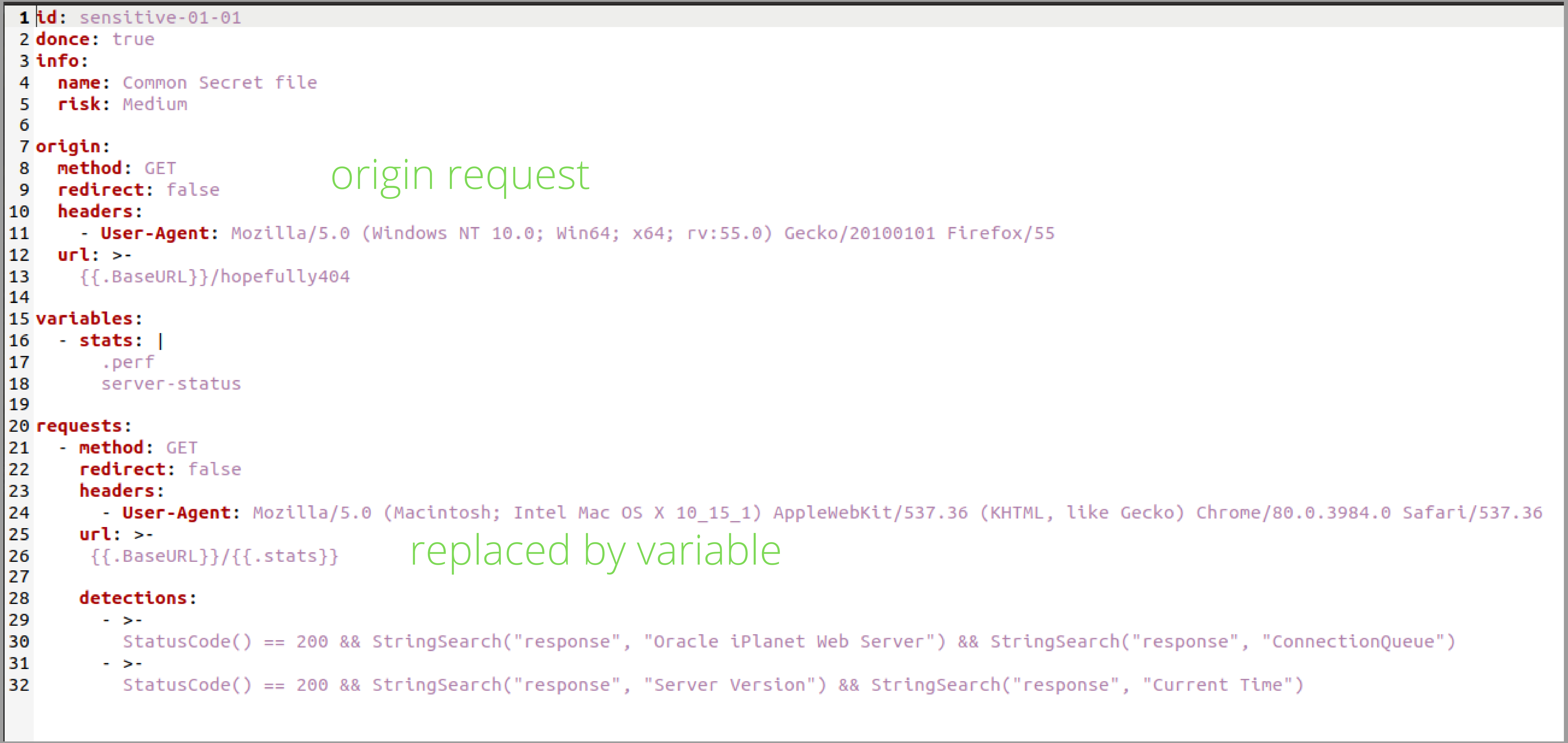


```
# info to search signature
id: CVE-2020-27993
info:
  name: Hrsale 2.0.0 - Local File Inclusion
  risk: medium
  author: Gocha Okradze


requests:
  - method: GET
    url: >-
     {{.BaseURL}}/download?type=files&filename=../../../../../../../../etc/passwd

    detections:
      - >-
        RegexSearch("resBody", 'root:[x*]:0:0:')

reference:
  - link: https://www.exploit-db.com/exploits/48920
```

სიგნატურა: original request



ორიგინალი request-ის შედარება detections ნაწილში

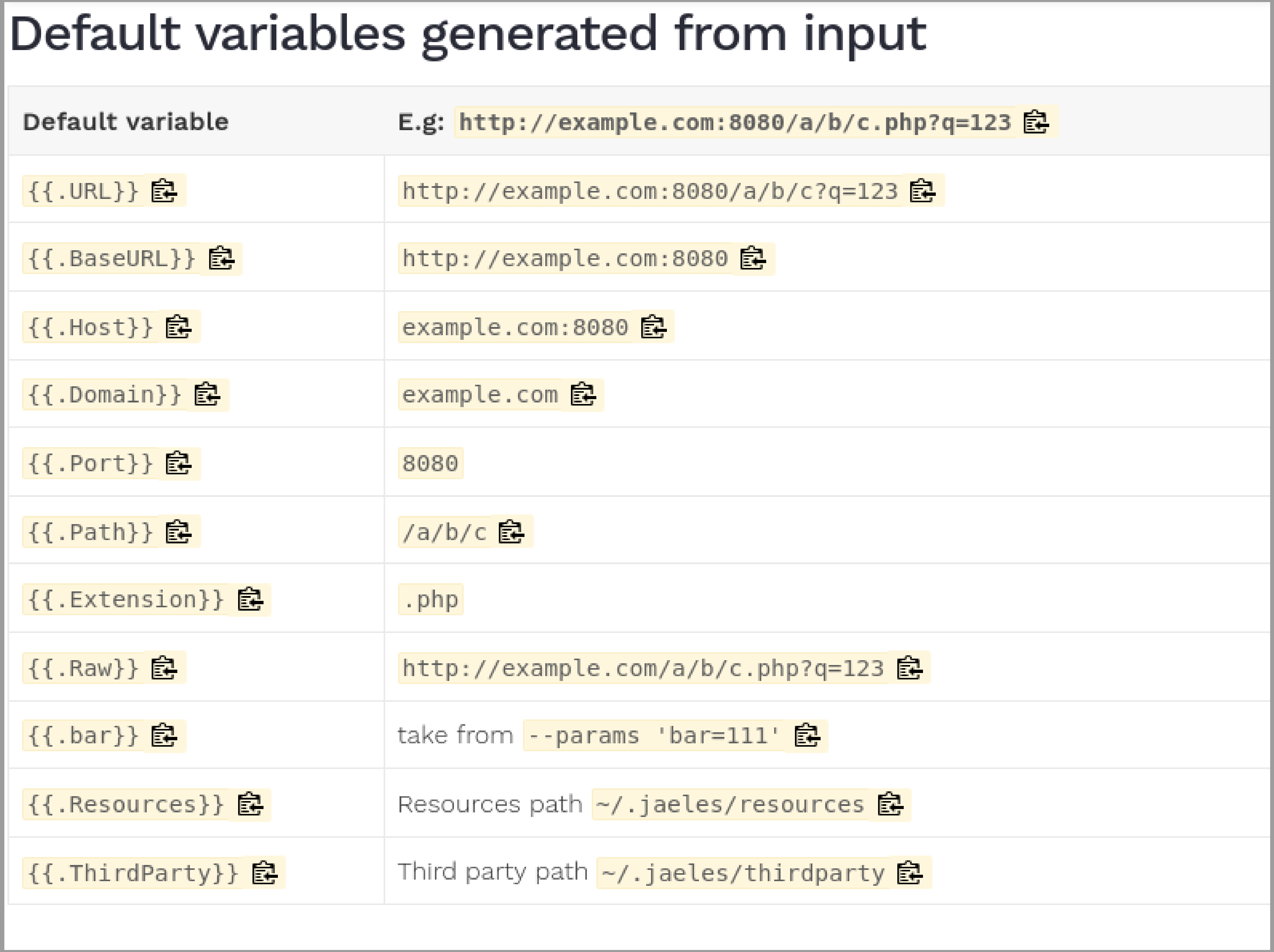


```yaml
id: sensitive-01-01
donce: true
info:
  name: Common Secret file
  risk: Medium

origin:
  method: GET
  redirect: false
  headers:
    - User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:55.0) Gecko/20100101 Firefox/55
  url: >-
    {{.BaseURL}}/hopefully404

variables:
  - stats: |
      .perf
      server-status

requests:
  - method: GET
    redirect: false
    headers:
      - User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/80.0.3984.0 Safari/537.36
    url: >-
     {{.BaseURL}}/{{.stats}}

    detections:
      - >-
        StatusCode() == 200 && StringSearch("response", "Oracle iPlanet Web Server") && StringSearch("response", "ConnectionQueue")
      - >-
        StatusCode() == 200 && StringSearch("response", "Server Version") && StringSearch("response", "Current Time")
```

origin request

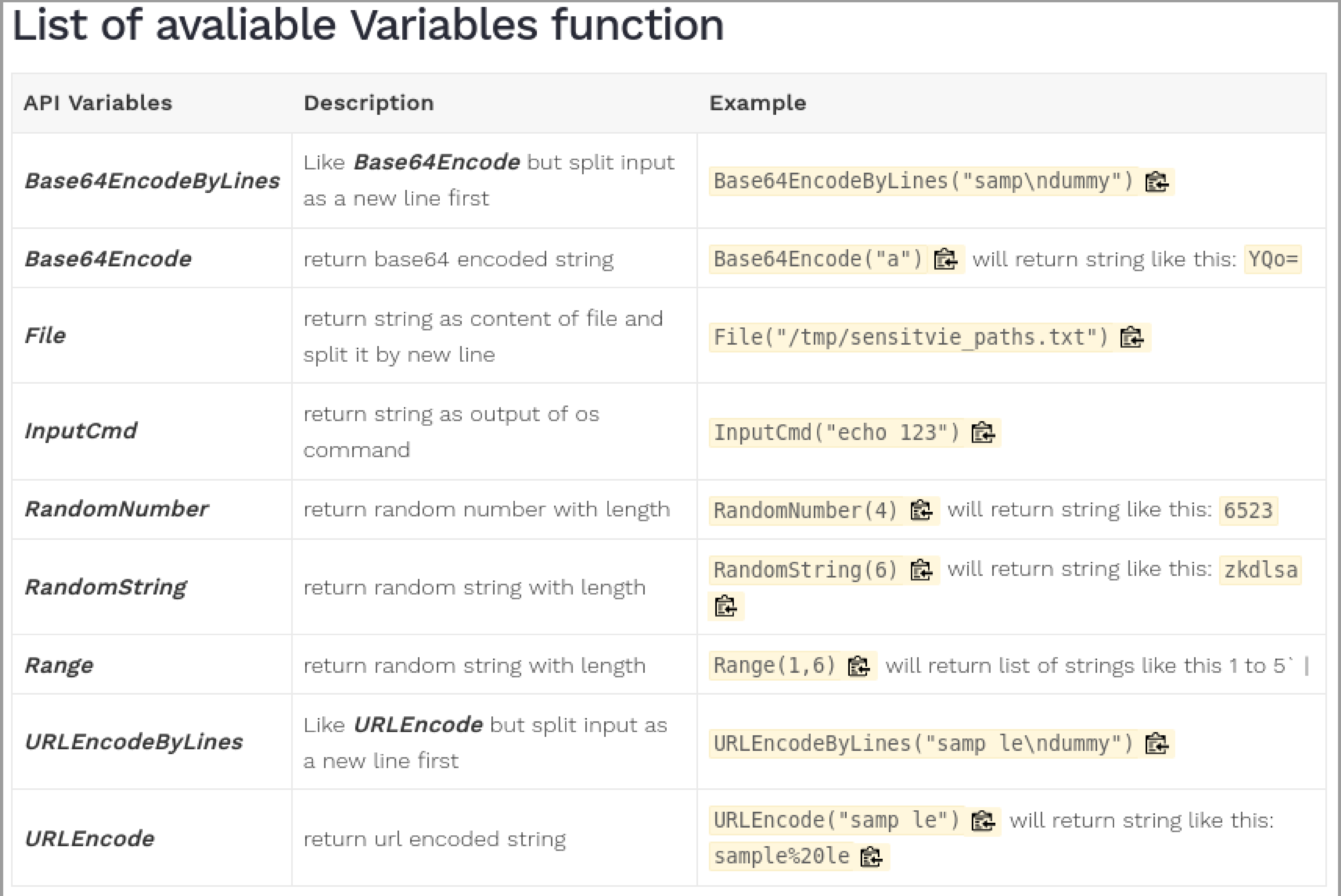
replaced by variable

სიგნატურა: variables და payloads



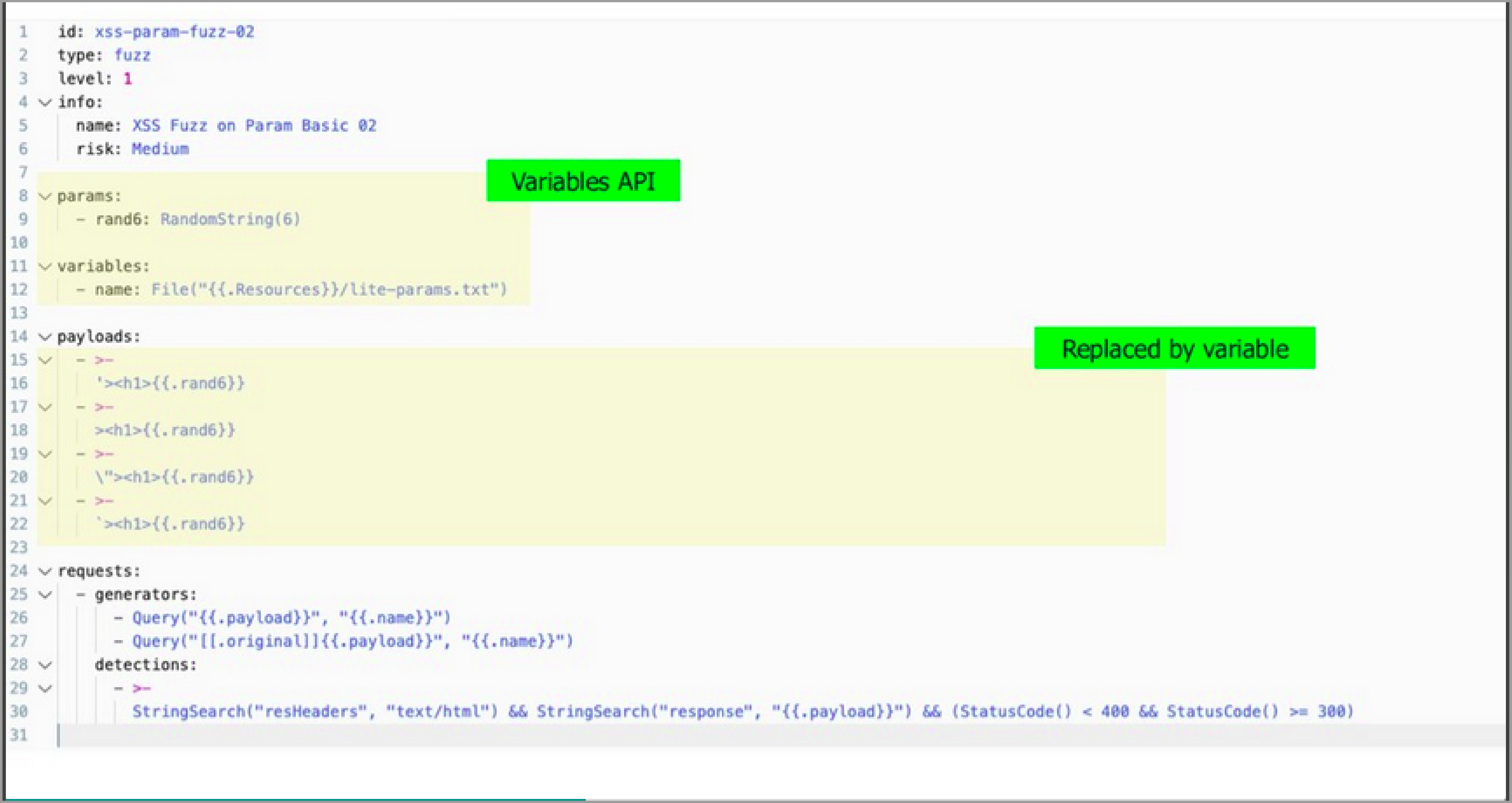
Default variables generated from input

Default variable	E.g: `http://example.com:8080/a/b/c.php?q=123`
`{{.URL}}`	`http://example.com:8080/a/b/c?q=123`
`{{.BaseURL}}`	`http://example.com:8080`
`{{.Host}}`	`example.com:8080`
`{{.Domain}}`	`example.com`
`{{.Port}}`	`8080`
`{{.Path}}`	`/a/b/c`
`{{.Extension}}`	`.php`
`{{.Raw}}`	`http://example.com/a/b/c.php?q=123`
`{{.bar}}`	take from `--params 'bar=111'`
`{{.Resources}}`	Resources path `~/.jaeles/resources`
`{{.ThirdParty}}`	Third party path `~/.jaeles/thirdparty`



List of avaliable Variables function

API Variables	Description	Example
Base64EncodeByLines	Like ***Base64Encode*** but split input as a new line first	`Base64EncodeByLines("samp\ndummy")`
Base64Encode	return base64 encoded string	`Base64Encode("a")` will return string like this: `YQo=`
File	return string as content of file and split it by new line	`File("/tmp/sensitvie_paths.txt")`
InputCmd	return string as output of os command	`InputCmd("echo 123")`
RandomNumber	return random number with length	`RandomNumber(4)` will return string like this: `6523`
RandomString	return random string with length	`RandomString(6)` will return string like this: `zkdlsa`
Range	return random string with length	`Range(1,6)` will return list of strings like this 1 to 5` |
URLEncodeByLines	Like ***URLEncode*** but split input as a new line first	`URLEncodeByLines("samp le\ndummy")`
URLEncode	return url encoded string	`URLEncode("samp le")` will return string like this: `sample%20le`

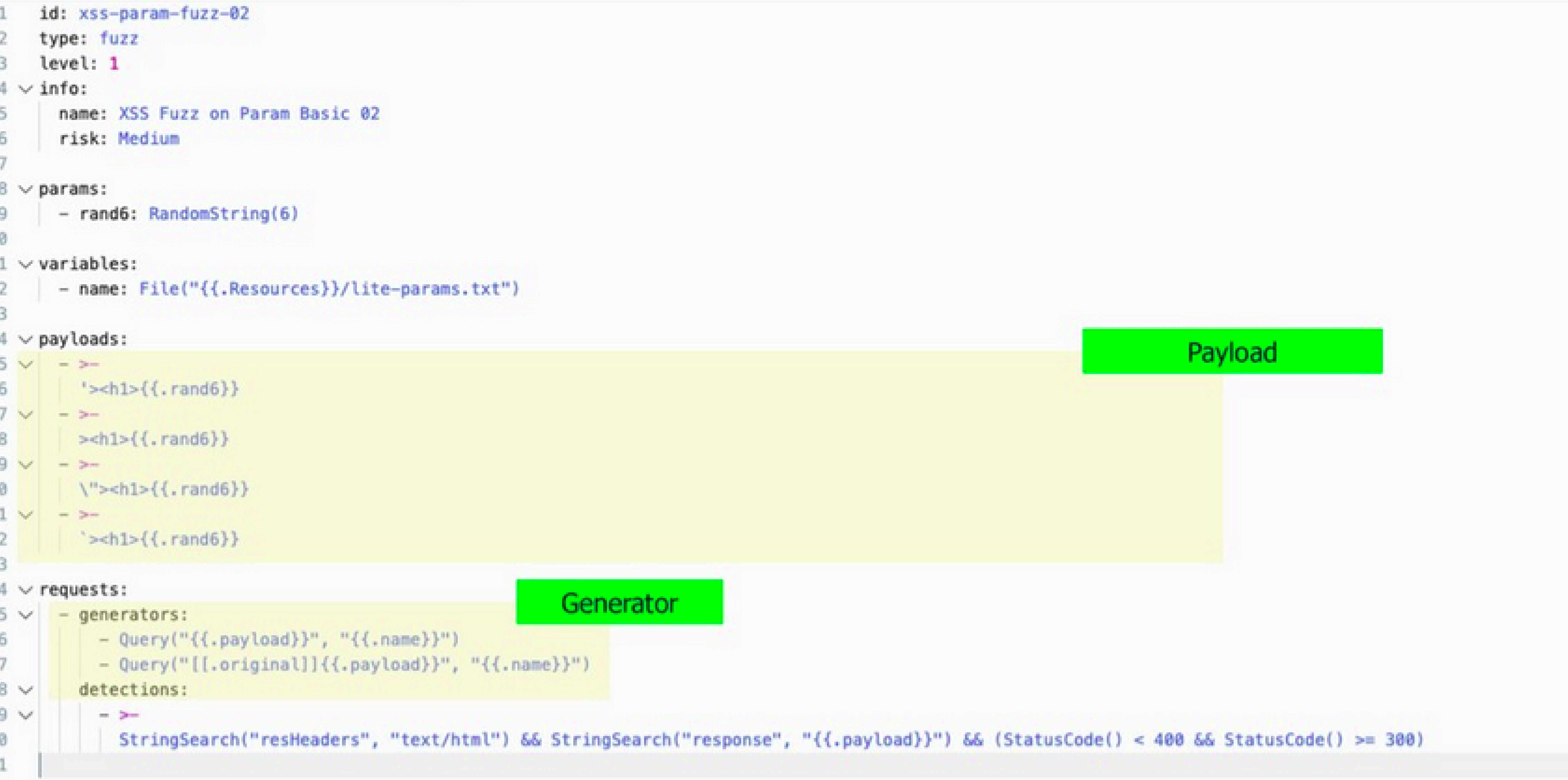


```yaml
id: xss-param-fuzz-02
type: fuzz
level: 1
info:
  name: XSS Fuzz on Param Basic 02
  risk: Medium

params:
  - rand6: RandomString(6)

variables:
  - name: File("{{.Resources}}/lite-params.txt")

payloads:
  - >-
    '><h1>{{.rand6}}
  - >-
    ><h1>{{.rand6}}
  - >-
    \"><h1>{{.rand6}}
  - >-
    `><h1>{{.rand6}}

requests:
  - generators:
      - Query("{{.payload}}", "{{.name}}")
      - Query("[[.original]]{{.payload}}", "{{.name}}")
    detections:
      - >-
        StringSearch("resHeaders", "text/html") && StringSearch("response", "{{.payload}}") && (StatusCode() < 400 && StatusCode() >= 300)
```

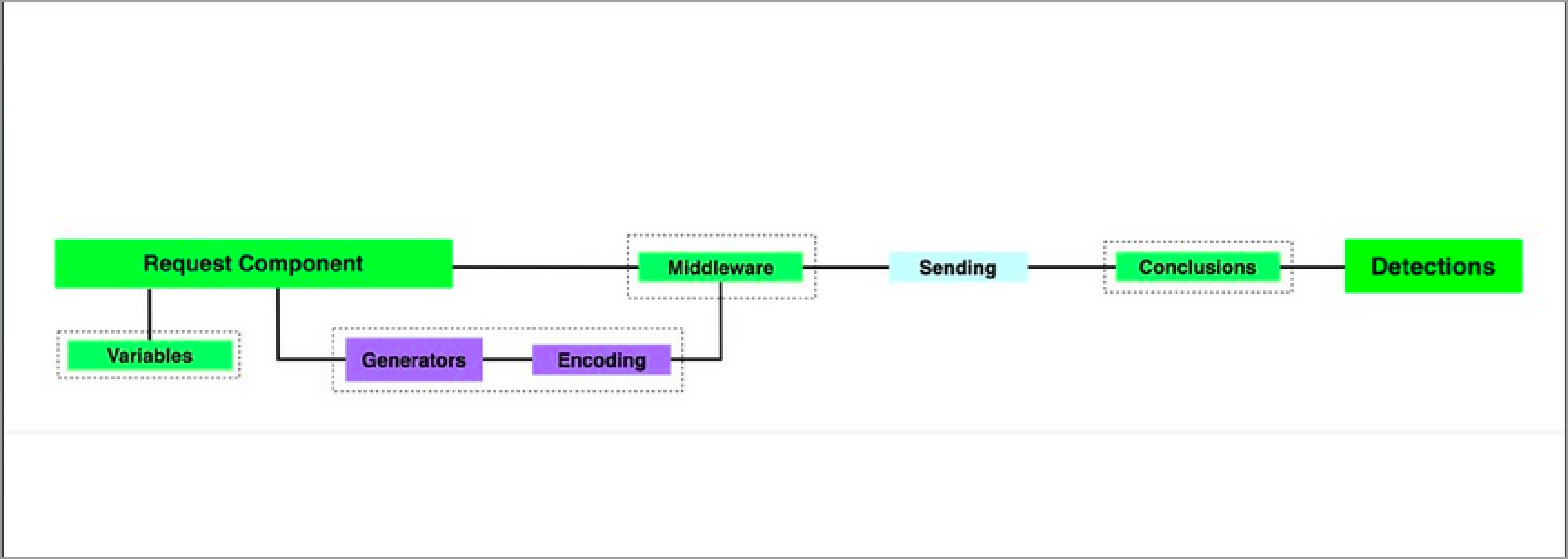
Variables API

Replaced by variable

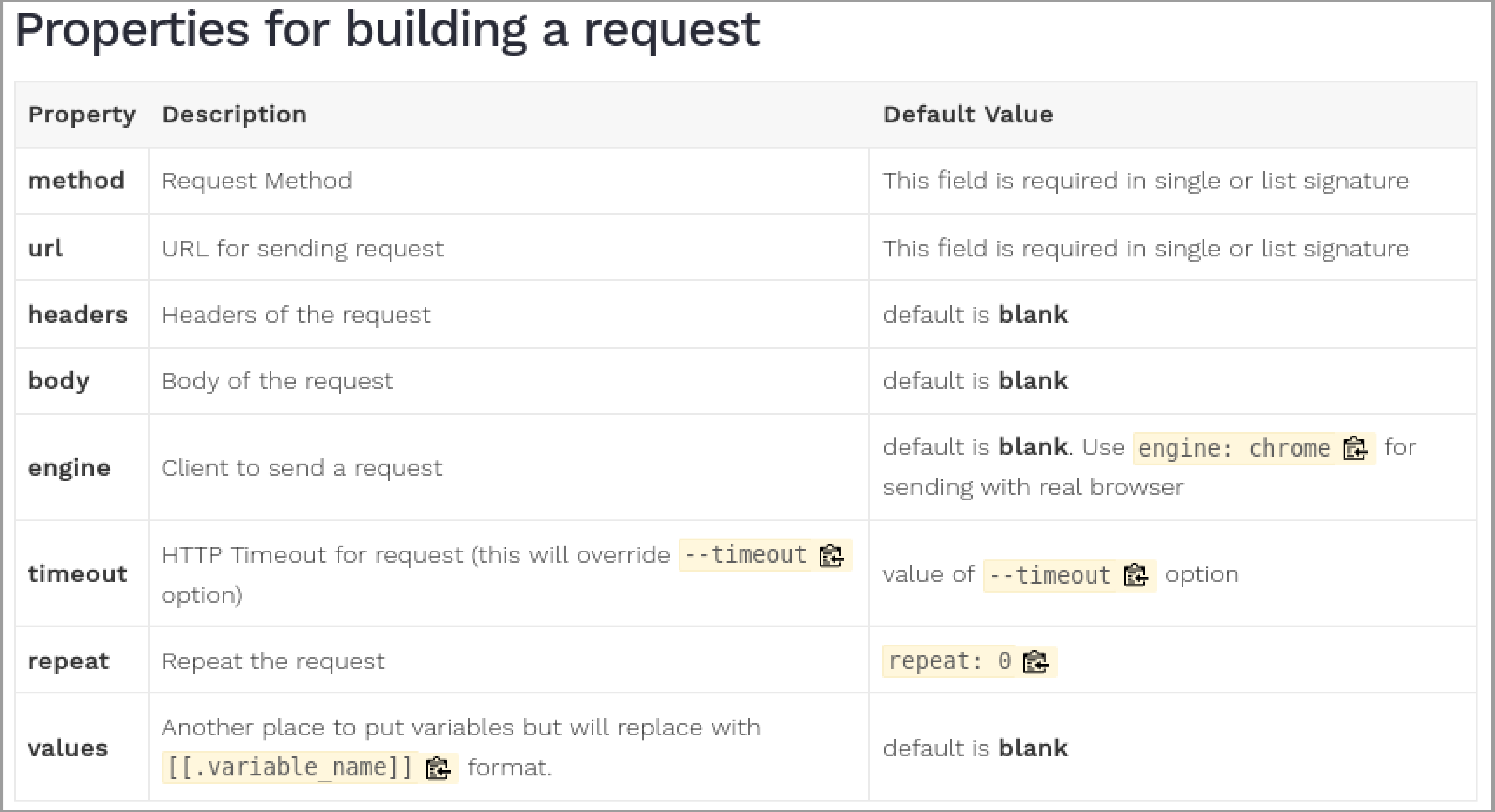
payload list-ის მიბმის შესაძლებლოა variable api-ის დახმარებით



```yaml
id: xss-param-fuzz-02
type: fuzz
level: 1
info:
  name: XSS Fuzz on Param Basic 02
  risk: Medium

params:
  - rand6: RandomString(6)

variables:
  - name: File("{{.Resources}}/lite-params.txt")

payloads:
  - >-
    '><h1>{{.rand6}}
  - >-
    ><h1>{{.rand6}}
  - >-
    \"><h1>{{.rand6}}
  - >-
    `><h1>{{.rand6}}

requests:
  - generators:
      - Query("{{.payload}}", "{{.name}}")
      - Query("[[.original]]{{.payload}}", "{{.name}}")
    detections:
      - >-
        StringSearch("resHeaders", "text/html") && StringSearch("response", "{{.payload}}") && (StatusCode() < 400 && StatusCode() >= 300)
```



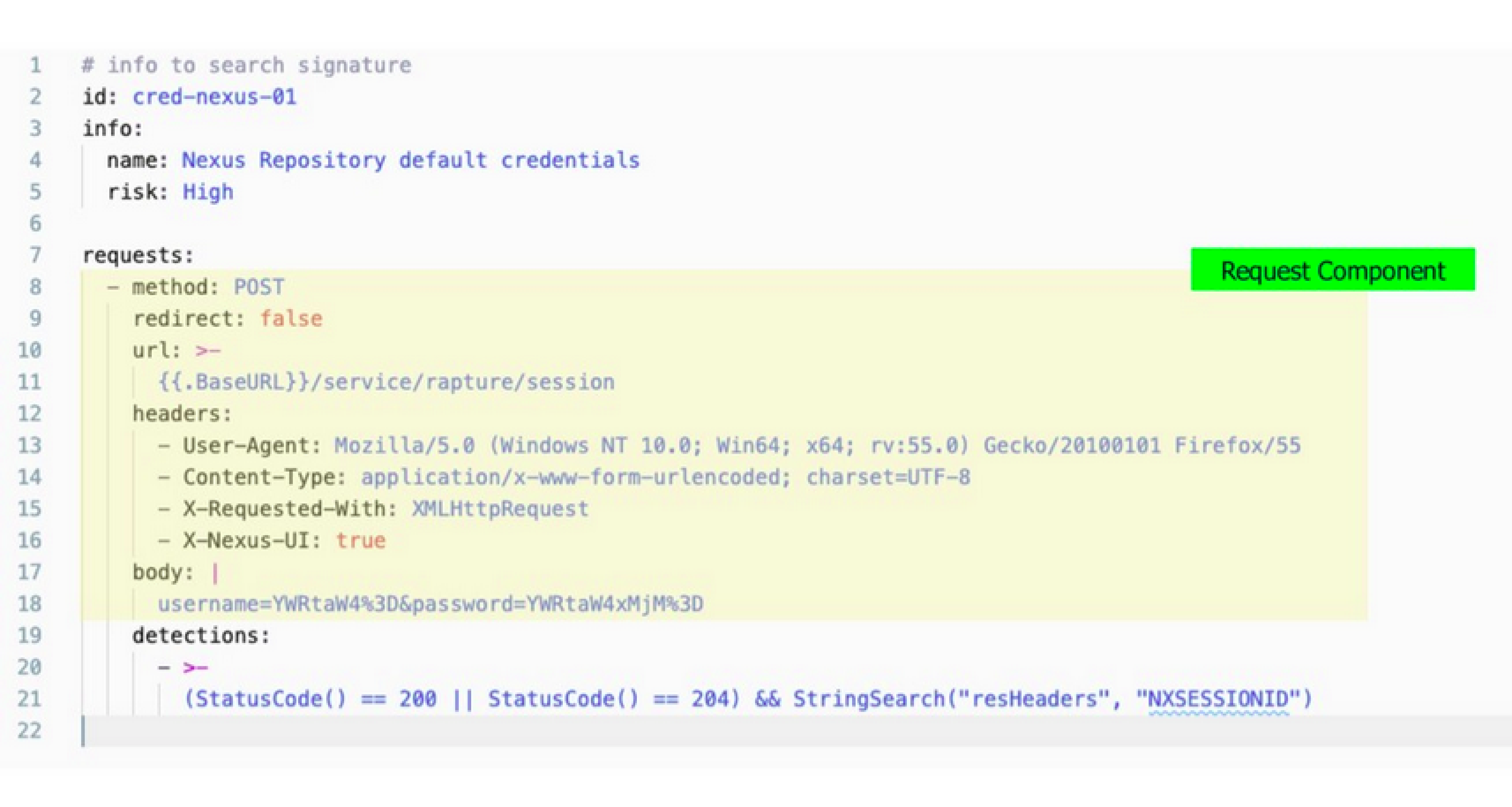
მომხმარებლის მიერ მოთხოვნილი request building , გაგზავნა და შემდგომში სისუტის აღმოჩენა



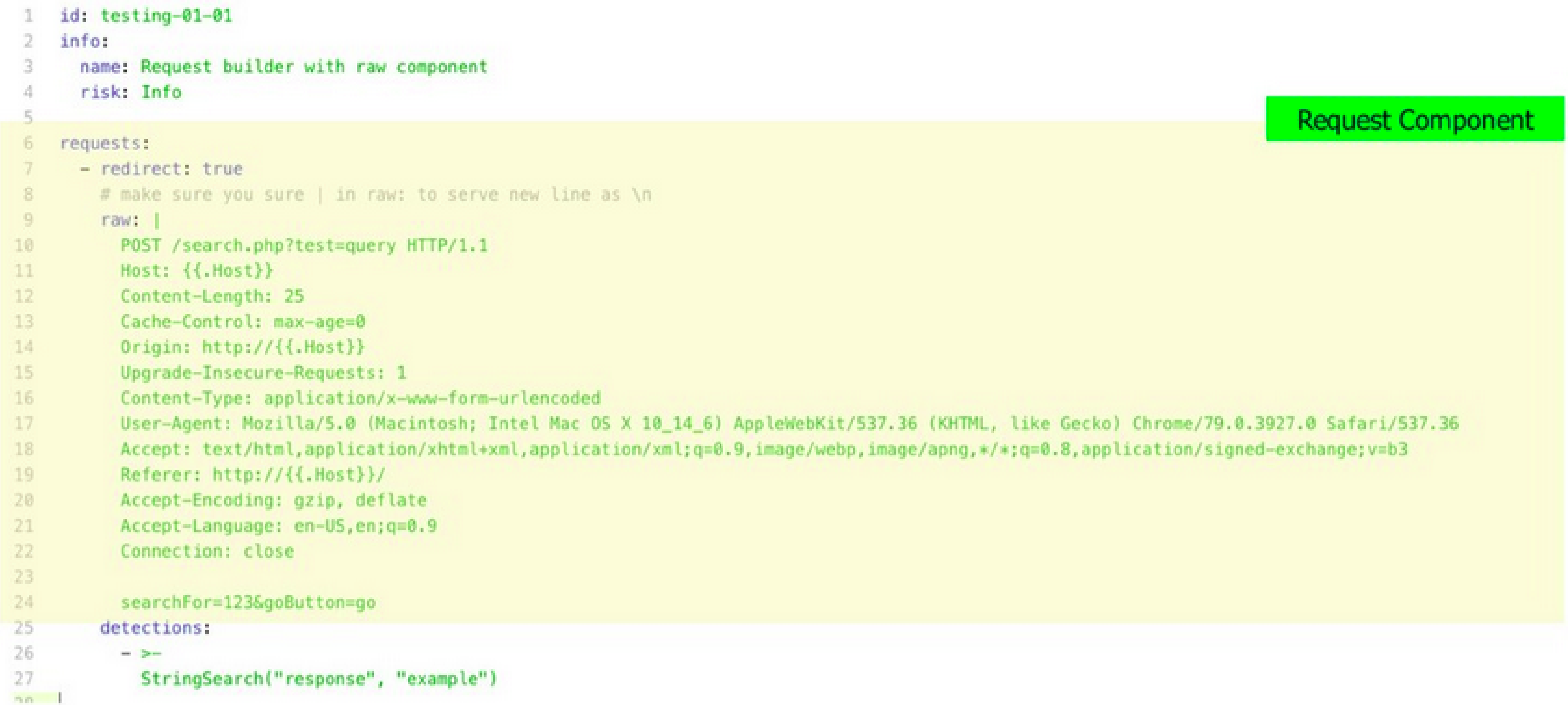
Properties for building a request

Property	Description	Default Value
method	Request Method	This field is required in single or list signature
url	URL for sending request	This field is required in single or list signature
headers	Headers of the request	default is **blank**
body	Body of the request	default is **blank**
engine	Client to send a request	default is **blank**. Use `engine: chrome` for sending with real browser
timeout	HTTP Timeout for request (this will override `--timeout` option)	value of `--timeout` option
repeat	Repeat the request	`repeat: 0`
values	Another place to put variables but will replace with `[[.variable_name]]` format.	default is **blank**

request **დეტალური ინფო**

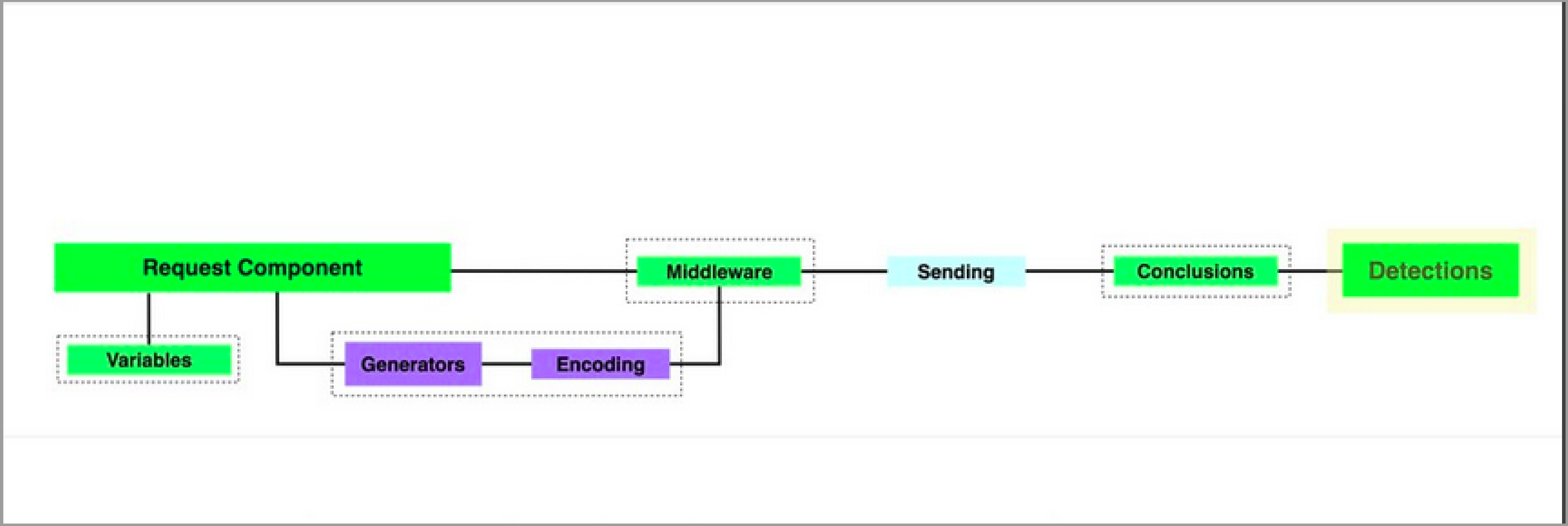


```yaml
# info to search signature
id: cred-nexus-01
info:
  name: Nexus Repository default credentials
  risk: High

requests:
  - method: POST
    redirect: false
    url: >-
      {{.BaseURL}}/service/rapture/session
    headers:
      - User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:55.0) Gecko/20100101 Firefox/55
      - Content-Type: application/x-www-form-urlencoded; charset=UTF-8
      - X-Requested-With: XMLHttpRequest
      - X-Nexus-UI: true
    body: |
      username=YWRtaW4%3D&password=YWRtaW4xMjM%3D
    detections:
      - >-
        (StatusCode() == 200 || StatusCode() == 204) && StringSearch("resHeaders", "NXSESSIONID")

```

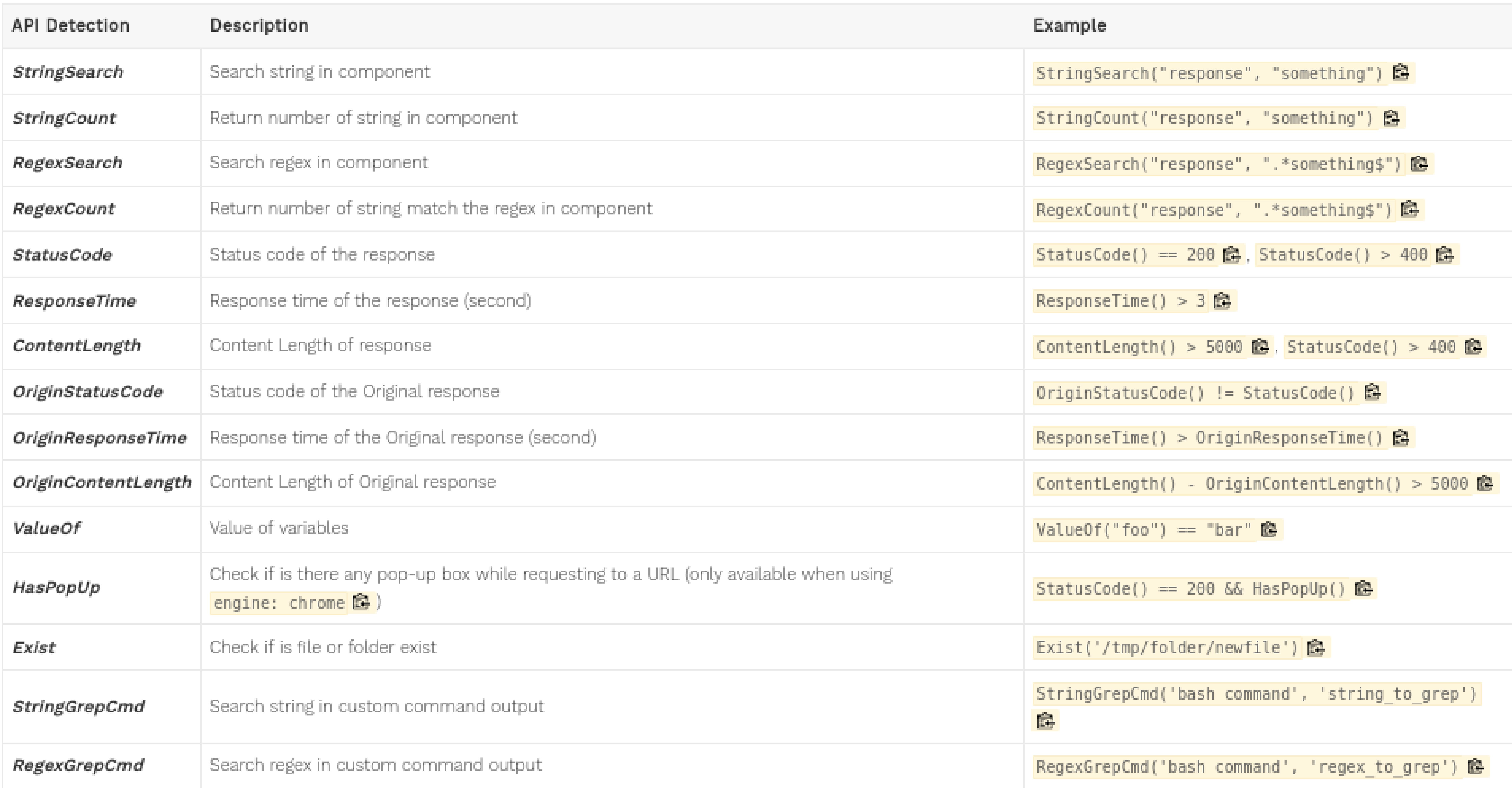


Request Component

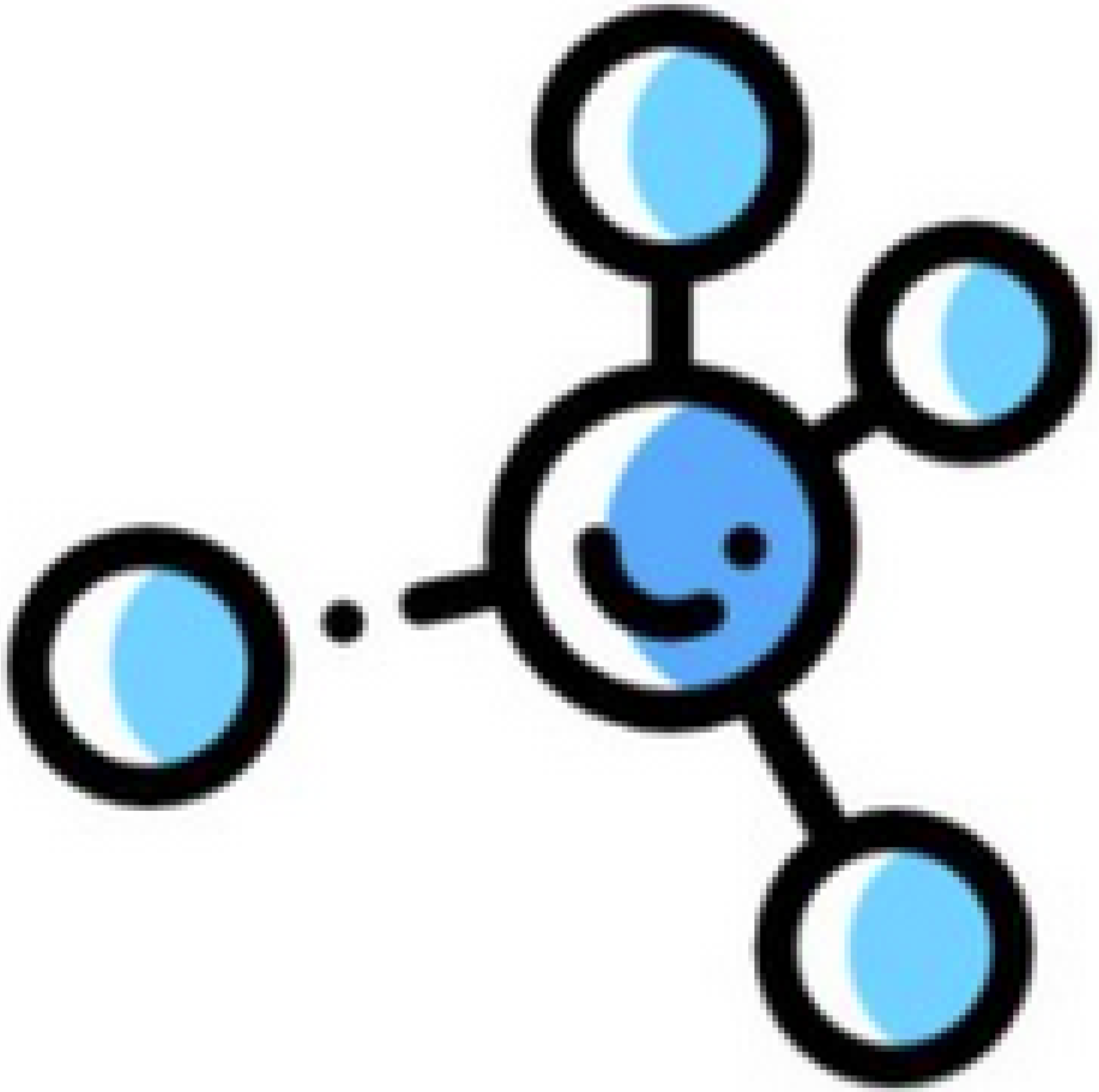
```
id: testing-01-01
info:
  name: Request builder with raw component
  risk: Info

requests:
  - redirect: true
    # make sure you sure | in raw: to serve new line as \n
    raw: |
      POST /search.php?test=query HTTP/1.1
      Host: {{.Host}}
      Content-Length: 25
      Cache-Control: max-age=0
      Origin: http://{{.Host}}
      Upgrade-Insecure-Requests: 1
      Content-Type: application/x-www-form-urlencoded
      User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.3927.0 Safari/537.36
      Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3
      Referer: http://{{.Host}}/
      Accept-Encoding: gzip, deflate
      Accept-Language: en-US,en;q=0.9
      Connection: close

      searchFor=123&goButton=go
    detections:
      - >-
        StringSearch("response", "example")
```



სისუსტეების აღმოსაჩენად ლოგიკის გამოყენება



API Detection	Description	Example
StringSearch	Search string in component	`StringSearch("response", "something")`
StringCount	Return number of string in component	`StringCount("response", "something")`
RegexSearch	Search regex in component	`RegexSearch("response", ".*something$")`
RegexCount	Return number of string match the regex in component	`RegexCount("response", ".*something$")`
StatusCode	Status code of the response	`StatusCode() == 200`, `StatusCode() > 400`
ResponseTime	Response time of the response (second)	`ResponseTime() > 3`
ContentLength	Content Length of response	`ContentLength() > 5000`, `StatusCode() > 400`
OriginStatusCode	Status code of the Original response	`OriginStatusCode() != StatusCode()`
OriginResponseTime	Response time of the Original response (second)	`ResponseTime() > OriginResponseTime()`
OriginContentLength	Content Length of Original response	`ContentLength() - OriginContentLength() > 5000`
ValueOf	Value of variables	`ValueOf("foo") == "bar"`
HasPopUp	Check if is there any pop-up box while requesting to a URL (only available when using `engine: chrome`)	`StatusCode() == 200 && HasPopUp()`
Exist	Check if is file or folder exist	`Exist('/tmp/folder/newfile')`
StringGrepCmd	Search string in custom command output	`StringGrepCmd('bash command', 'string_to_grep')`
RegexGrepCmd	Search regex in custom command output	`RegexGrepCmd('bash command', 'regex_to_grep')`



Demo