Project Part 2

Part 1: Black Box Test Cases

Section 1: Client Side Bypassing

1. Upon addition of new patient, patient can be assigned any gender
   Unique ID: 5.1-1
   ASVS:
   5.1.4 Verify that structured data is strongly typed and validated against a defined schema including allowed characters, length and pattern
   CWE: CWE-20
   URL: http://localhost/interface/new/new_comprehensive_save.php
   Input Field: Sex
   Initial User Input: Male
   Malicious Input: Unassigned

   Steps:
   - Login to OpenEMR using valid credentials
   - Navigate to Patients tab and then New Patient
   - Create a valid patient using by entering the mandatory inputs in the valid format, a valid sex in this case
   - Submit the form
   - Navigate to OWASP ZAP and look for the post request containing the details of the form. It can be found at POST request for the given URL.
   - Right click on the POST request and create a breakpoint
   - Navigate to Patients tab and create a new patient again
   - This time on submitting valid inputs, OWASP ZAP will intercept and request.
   - Navigate to breakpoint and modify the value of ‘form_sex’ to any custom value, ‘Unassigned’ in this case
   - Forward the request to the server

   Expected Inputs:
   - 500 Internal Server Error
   - Server Side sanitization of input

2. A new patient can be created with a blank first name and last name fields
   Unique ID: 5.1-2

ASVS:
5.2.7 Verify that the application sanitizes, disables, or sandboxes user-supplied Scalable Vector Graphics (SVG) scriptable content, especially as they relate to XSS resulting from inline scripts, and foreignObject.
CWE: CWE-159
URL: http://localhost/interface/new/new_comprehensive_save.php
Input Field: First Name, Last Name
Initial User Input: 'John', 'Doe'
Malicious Input: '', ''

Steps:

- Login to OpenEMR using valid credentials
- Navigate to Patients tab and then New Patient
- Create a valid patient using by entering the mandatory inputs in the valid format, a first and last name in this case
- Submit the form
- Navigate to OWASP ZAP and look for the post request containing the details of the form. It can be found at POST request for the given URL.
- Right click on the POST request and create a breakpoint
- Navigate to Patients tab and create a new patient again
- This time on submitting valid inputs, OWASP ZAP will intercept and request.
- Navigate to breakpoint and modify the value of 'form_fname' and 'form_lname' to empty fields
- Forward the request to the server

Expected Inputs:

- 500 Internal Server Error
- Server Side sanitization of input

3. Date of Birth of the patient is accepting alphanumeric values
Unique ID: 5.1-3
ASVS:
5.1.3 Verify that all input (HTML form fields, REST requests, URL parameters, HTTP headers, cookies, batch files, RSS feeds, etc) is validated using positive validation
CWE: CWE-20: Improper Input Validation
URL: http://localhost/interface/new/new_comprehensive_save.php
Input Field: Date of Birth
Initial User Input: Current/Past Date
Malicious Input: Any date in future

Steps:

- Login to OpenEMR using valid credentials

- Navigate to Patients tab and then New Patient
- Create a valid patient using by entering the mandatory inputs in the valid format, a valid date of birth in this case
- Submit the form
- Navigate to OWASP ZAP and look for the post request containing the details of the form. It can be found at POST request for the given URL.
- Right click on the POST request and create a breakpoint
- Navigate to Patients tab and create a new patient again
- This time on submitting valid inputs, OWASP ZAP will intercept and request.
- Navigate to breakpoint and modify the value of 'form_DOB' to any alphanumeric value of date
- Forward the request to the server

Expected Inputs:

- 500 Internal Server Error
- Server Side sanitization of input.

4. Message length not validated in the POST request
   Unique ID: 5.1-6
   ASVS:
   5.2.2 Verify that unstructured data is sanitized to enforce safety measures such as allowed characters and length.
   CWE: CWE-138
   URL: http://localhost/interface/main/messages/messages.php
   Input Field: Check or Reference Number
   Initial User Input: Valid reference number
   Malicious Input: Any alphanumeric input of unspecified length

   Steps:
   - Login to OpenEMR using valid credentials
   - Navigate to patients tab
   - Search for a patient using the details provided and select the patient
   - Go to Visits->Create new visit and create a valid visit for the patient (if none is created)
   - Go to Fees -> Payments
   - Enter the payment details with valid inputs
   - Submit the form.
   - Navigate to OWASP ZAP and look for the post request containing the details of the form. It can be found at POST request for the given URL.
   - Right click on the POST request and create a breakpoint
   - Navigate to Patients tab and create a new patient visit again
   - This time on submitting valid inputs, OWASP ZAP will intercept and request.

- Navigate to breakpoint and modify the value of ‘form_source’ to any alphanumeric value of any length.
- Forward the request to the server

Expected Inputs:

- 500 Internal Server Error
- Error message specifying maximum input length

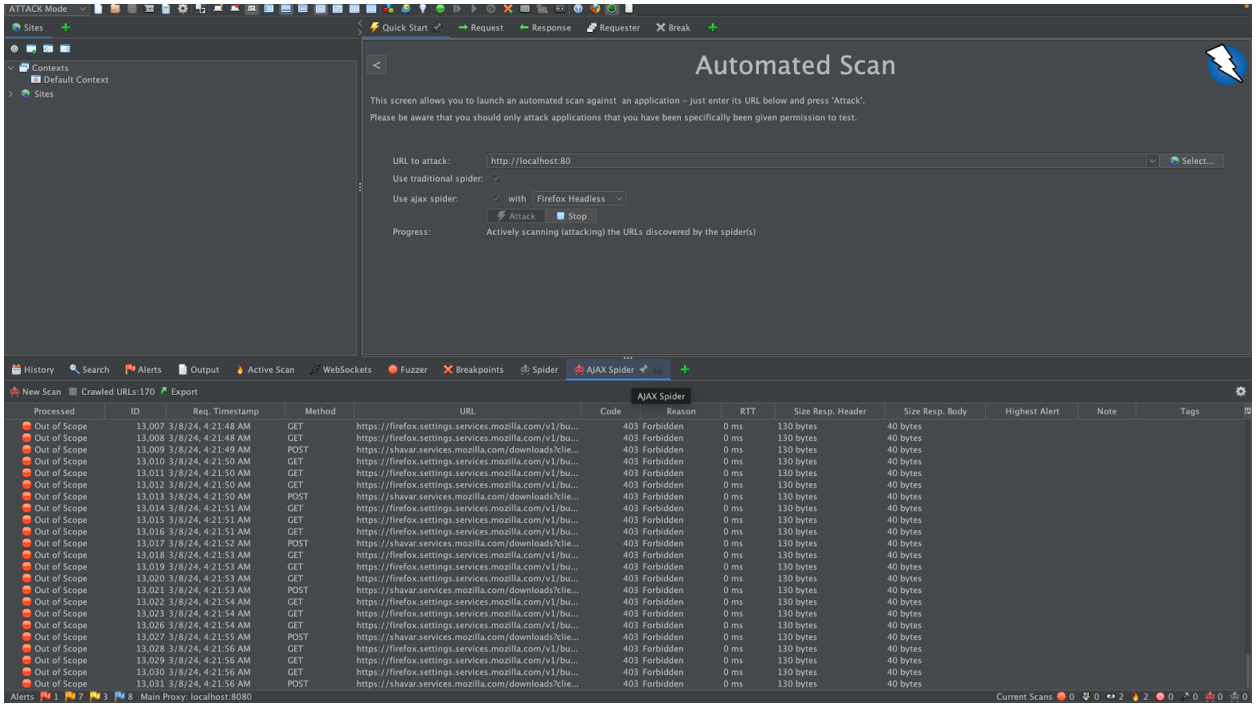
5. SQL Injection in login form
   Unique ID: 5.1-6
   ASVS:
   5.3.4 Verify that data selection or database queries (e.g. SQL, HQL, ORM, NoSQL) use parameterized queries, ORMs, entity frameworks, or are otherwise protected from database injection attacks
   CWE: CWE-89
   URL: http://localhost/interface/main/main_screen.php?auth=login&site=default
   Input Field: admin
   Initial User Input: valid admin input
   Malicious Input: SQL Injection statement

   Steps:
   - Login to OpenEMR using valid credentials
   - Navigate to patients tab
   - Search for a patient using the details provided and select the patient
   - Go to Fees -> Payments
   - Enter the payment details with valid inputs
   - Submit the form.
   - Navigate to OWASP ZAP and look for the post request containing the details of the form. It can be found at POST request for the given URL.
   - Right click on the POST request and create a breakpoint
   - Navigate to Patients tab and create a new patient visit again
   - This time on submitting valid inputs, OWASP ZAP will intercept and request.
   - Navigate to breakpoint and modify the value of ‘form_source’ to any alphanumeric value of any length.
   - Forward the request to the server

   Expected Inputs:
   - 500 Internal Server Error
   - Error message specifying maximum input length

Section 2: Automated Fuzzing

Total ZAP Automated Scan Time: 1 hour 30 mins
Total Time for finding True Positive: 5 hours
Total Time for planning the black box test cases: 2 hours
Time taken per true positive: 1 hour 20 minutes
The actual screenshots of results as well as fuzzing as included in the second document



2. Vulnerable dependencies

Github Checker total vulnerable dependencies: 25

Dependency name	CVE	Nature	Safer version
mongoose	CVE-2023-3696	Transitive	v5.13.20
xmldom	CVE-2022-39353	Transitive	N/A
	CVE-2021-32796		
angular	CVE-2024-21490	Direct	N/A
	CVE-2023-26116		
	CVE-2023-26117		
	CVE-2023-26118		
	CVE-2022-25844		
	CVE-2022-25869		
jszip	CVE-2022-48285	Transitive	v3.8.0
	CVE-2021-23413		v3.7.0
ckeditor4	CVE-2023-4771	Direct	v4.24.0-LTS
	CVE-2024-24816		
	CVE-2024-24815		
qs	CVE-2022-24999	Transitive	v6.5.3
xml2js	CVE-2023-0842	Transitive	v0.5.0
tar	CVE-2021-37713	Transitive	v4.4.18
	CVE-2021-32804		v3.2.2
postcss	CVE-2023-44270	Transitive	v8.4.31
got	CVE-2022-33987	Transitive	v11.8.5

Snyk total vulnerable dependencies: 6,486

Dependency name	CVE	Nature	Safer Version
xmldom	CVE-2022-39353	Transitive	N/A
	CVE-2022-37616		
	CVE-2021-32796		
mongoose	CVE-2023-3696	Transitive	v5.13.20
qs	CVE-2022-24999	Transitive	v6.2.4
xml2js	CVE-2023-0842	Transitive	v0.5.0
inflight	N/A	Transitive	N/A
jszip	CVE-2021-23413	Transitive	v3.7.0
	CVE-2022-48285		v3.8.0
ckeditor4	CVE-2024-24816	Direct	v4.24.0
	CVE-2024-24815		v4.24.0
	CVE-2023-4771		N/A
angular	CVE-2024-21490	Direct	N/A
	CVE-2022-25844		
	CVE-2023-26116		
	CVE-2023-26117		
	CVE-2023-26118		
	CVE-2022-25869		
i18next	N/A	Transitive	v19.8.5 v19.8.3 v19.5.5
dompdf	CVE-2024-25117	Direct	v2.0.4

- **Explain why you think the results differ among the two tools and write a comparison report. What do you think are the strengths and weaknesses of each tool from both technical and usability standpoints.)**

GitHub Dependabot offers a seamless integration with GitHub's version control system, providing a user-friendly interface for managing dependencies within GitHub repositories. This tight coupling streamlines the workflow for developers using GitHub's platform. However, Dependabot's major limitation is its exclusivity to GitHub repositories, which makes it non compatible to alternative version control systems such as BitBucket, GitLab.

In contrast, Synk distinguishes itself by supporting multiple version control systems, including BitBucket and GitLab. Synk's strength lies in its extensive vulnerability database, enabling a more comprehensive scan that may uncover potential security vulnerabilities that GitHub Dependabot might overlook. Synk also provides detailed dependency paths, which is particularly useful for comprehending and mitigating vulnerabilities, especially in the case of transitive dependencies.

While Synk's is versatile, its broad integration options and extensive feature set may introduce some complexity and a learning curve for new users. The choice between GitHub Dependabot and Synk ultimately hinges upon the development team's specific requirements, including the version control system in use, the desired level of security coverage, and the team's familiarity with each tool's user interface.

The results differ due to the specific focus and integrations of each tool, the size and scope of their vulnerability databases, and the extent to which they address both direct and transitive dependencies. Github checker tool could have improved their design to provide a dependency path to better understand and mitigate the vulnerabilities. Moreover, there are some vulnerabilities that were found by Synk but not github checker, we believe that is because of the vulnerability database.

- **Compare the output of these two tools with that of the Bomber tool you ran during a classroom exercise.**

The comparison between GitHub Checks, Snyk, and the Bomber tool reveals distinct approaches to vulnerability management and reporting within the software development lifecycle (SDLC). GitHub Checks and Snyk provide integrated solutions that enable seamless identification and remediation of security vulnerabilities within the developers' workflow. Both tools leverage industry-standard vulnerability data, including Common Vulnerabilities and Exposures (CVE) identifiers, vulnerability titles, descriptions, Common Weakness Enumeration (CWE) classifications, and severity ratings, facilitating consistent vulnerability assessment across platforms.

The Bomber tool's output is detailed and structured, often in JSON format, providing a comprehensive breakdown of each vulnerability, which is ideal for automated processing and in-depth analysis. This contrasts with GitHub Checks and Snyk, which present more streamlined and user-friendly information, focusing on summarizing key details and remediation advice directly within their interfaces for quick decision-making by developers. While the Bomber tool caters to technical users requiring detailed data for extensive analysis or integration into complex workflows, GitHub Checks and Snyk are designed to provide accessible, actionable insights, optimizing ease of use and immediacy for developers in their daily work.

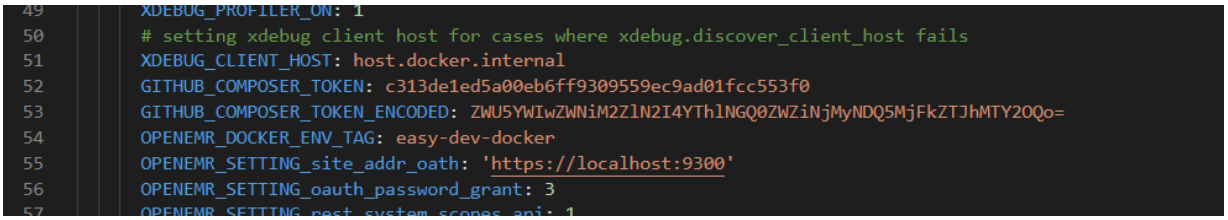
3. Secret Detection

Tool 1: Gitleaks

1) **Secret type:** API Key
   Exposed Secret:

ZWU5YWIwZWNiM2ZlN2I4YThlNGQ0ZWZiNjMyNDQ5MjFkZTJhMTY2OQo=

Screenshot of code:



```
49        XDEBUG_PROFILER_ON: 1
50        # setting xdebug client host for cases where xdebug.discover_client_host fails
51        XDEBUG_CLIENT_HOST: host.docker.internal
52        GITHUB_COMPOSER_TOKEN: c313de1ed5a00eb6ff9309559ec9ad01fcc553f0
53        GITHUB_COMPOSER_TOKEN_ENCODED: ZWU5YWIwZWNiM2ZlN2I4YThlNGQ0ZWZiNjMyNDQ5MjFkZTJhMTY2OQo=
54        OPENEMR_DOCKER_ENV_TAG: easy-dev-docker
55        OPENEMR_SETTING_site_addr_oath: 'https://localhost:9300'
56        OPENEMR_SETTING_oauth_password_grant: 3
57        OPENEMR_SETTING_rest_system_scopes_api: 1
```

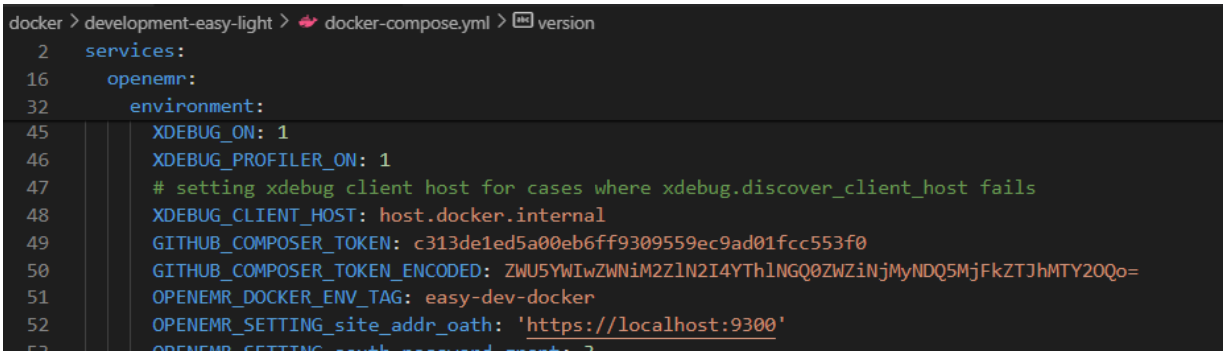
Tool generated output:


```
{
    "Description": "Detected a Generic API Key, potentially exposing access to various services and sensitive operations.",
    "StartLine": 52,
    "EndLine": 52,
    "StartColumn": 23,
    "EndColumn": 0,
    "Match": "TOKEN_ENCODED: ZWU5YWIwZWNiM2ZlN2I4YThlNGQ0ZWZiNjMyNDQ5MjFkZTJhMTY2OQo=",
    "Secret": "ZWU5YWIwZWNiM2ZlN2I4YThlNGQ0ZWZiNjMyNDQ5MjFkZTJhMTY2OQo=",
    "File": "docker/development-easy-arm32/docker-compose.yml",
    "SymlinkFile": "",
    "Commit": "fe61175c8f0cc33552004ddc122b0722043f0d29",
    "Entropy": 4.4834447,
    "Author": "Brady Miller",
    "Email": "brady.g.miller@gmail.com",
    "Date": "2021-03-20T23:16:32Z",
    "Message": "docker dev fix (#4294)",
    "Tags": [],
    "RuleID": "generic-api-key",
    "Fingerprint": "fe61175c8f0cc33552004ddc122b0722043f0d29:docker/development-easy-arm32/docker-compose.yml:generic-api-key:52"
},
```

2) **Secret type: API key**
Exposed Secret:

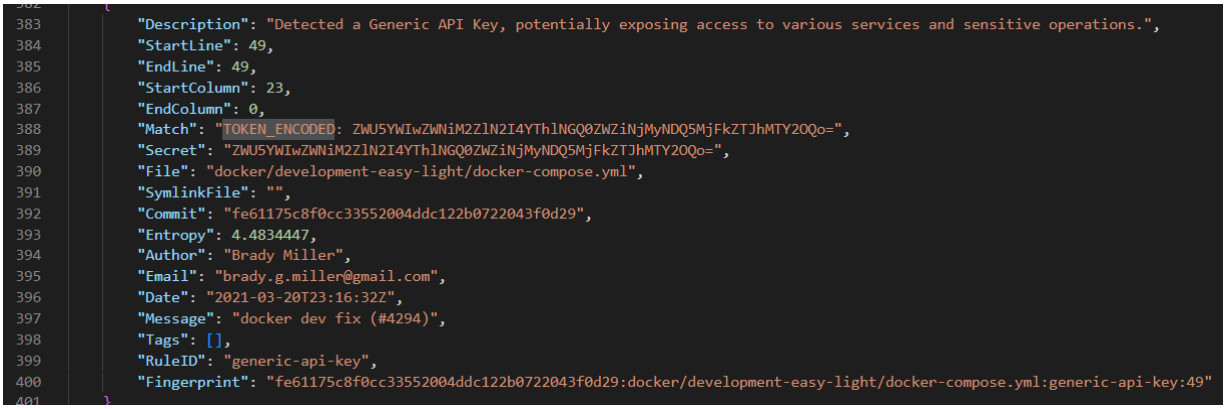
ZWU5YWIwZWNiM2ZlN2I4YThlNGQ0ZWZiNjMyNDQ5MjFkZTJhMTY2OQo=

Screenshot of code:



```
docker > development-easy-light > docker-compose.yml > version
  2   services:
 16     openemr:
 32       environment:
 45         XDEBUG_ON: 1
 46         XDEBUG_PROFILER_ON: 1
 47         # setting xdebug client host for cases where xdebug.discover_client_host fails
 48         XDEBUG_CLIENT_HOST: host.docker.internal
 49         GITHUB_COMPOSER_TOKEN: c313de1ed5a00eb6ff9309559ec9ad01fcc553f0
 50         GITHUB_COMPOSER_TOKEN_ENCODED: ZWU5YWIwZWNiM2ZlN2I4YThlNGQ0ZWZiNjMyNDQ5MjFkZTJhMTY2OQo=
 51         OPENEMR_DOCKER_ENV_TAG: easy-dev-docker
 52         OPENEMR_SETTING_site_addr_oath: 'https://localhost:9300'
```

Tool generated output:



```
383       "Description": "Detected a Generic API Key, potentially exposing access to various services and sensitive operations.",
384       "StartLine": 49,
385       "EndLine": 49,
386       "StartColumn": 23,
387       "EndColumn": 0,
388       "Match": "TOKEN_ENCODED: ZWU5YWIwZWNiM2ZlN2I4YThlNGQ0ZWZiNjMyNDQ5MjFkZTJhMTY2OQo=",
389       "Secret": "ZWU5YWIwZWNiM2ZlN2I4YThlNGQ0ZWZiNjMyNDQ5MjFkZTJhMTY2OQo=",
390       "File": "docker/development-easy-light/docker-compose.yml",
391       "SymlinkFile": "",
392       "Commit": "fe61175c8f0cc33552004ddc122b0722043f0d29",
393       "Entropy": 4.4834447,
394       "Author": "Brady Miller",
395       "Email": "brady.g.miller@gmail.com",
396       "Date": "2021-03-20T23:16:32Z",
397       "Message": "docker dev fix (#4294)",
398       "Tags": [],
399       "RuleID": "generic-api-key",
400       "Fingerprint": "fe61175c8f0cc33552004ddc122b0722043f0d29:docker/development-easy-light/docker-compose.yml:generic-api-key:49"
401   }
```

3) **Secret type:** API key

Exposed Secret:

ZWU5YWIwZWNiM2ZlN2I4YThlNGQ0ZWZiNjMyNDQ5MjFkZTJhMTY2OQo=

Screenshot of code:

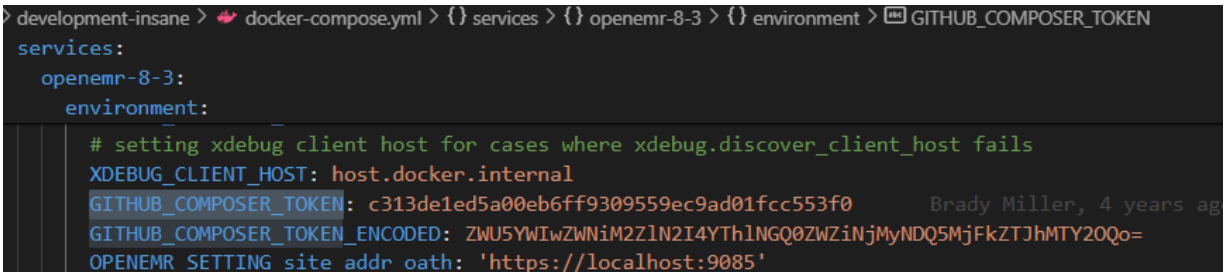
```
docker > development-easy > docker-compose.yml > version
  2  services:
 16    openemr:
 33      environment:
 47        XDEBUG_PROFILER_ON: 1
 48        # setting xdebug client host for cases where xdebug.discover_client_host fails
 49        XDEBUG_CLIENT_HOST: host.docker.internal
 50        GITHUB_COMPOSER_TOKEN: c313de1ed5a00eb6ff9309559ec9ad01fcc553f0
 51        GITHUB_COMPOSER_TOKEN_ENCODED: ZWU5YWIwZWNiM2ZlN2I4YThlNGQ0ZWZiNjMyNDQ5MjFkZTJhMTY2OQo=
 52        OPENEMR_DOCKER_ENV_TAG: easy-dev-docker
 53        OPENEMR_SETTING_site_addr_oath: 'https://localhost:9300'
 54        OPENEMR_SETTING_oauth_password_grant: 3
```

Tool generated output:


```
422  {
423      "Description": "Detected a Generic API Key, potentially exposing access to various services and sensitive operations.",
424      "StartLine": 50,
425      "EndLine": 50,
426      "StartColumn": 23,
427      "EndColumn": 0,
428      "Match": "TOKEN_ENCODED: ZWU5YWIwZWNiM2ZlN2I4YThlNGQ0ZWZiNjMyNDQ5MjFkZTJhMTY2OQo=",
429      "Secret": "ZWU5YWIwZWNiM2ZlN2I4YThlNGQ0ZWZiNjMyNDQ5MjFkZTJhMTY2OQo=",
430      "File": "docker/development-easy/docker-compose.yml",
431      "SymlinkFile": "",
432      "Commit": "fe61175c8f0cc33552004ddc122b0722043f0d29",
433      "Entropy": 4.4834447,
434      "Author": "Brady Miller",
435      "Email": "brady.g.miller@gmail.com",
436      "Date": "2021-03-20T23:16:32Z",
437      "Message": "docker dev fix (#4294)",
438      "Tags": [],
439      "RuleID": "generic-api-key",
440      "Fingerprint": "fe61175c8f0cc33552004ddc122b0722043f0d29:docker/development-easy/docker-compose.yml:generic-api-key:50"
441  },
442  {
```

4) **Secret type:** API key
Exposed Secret:
ZWU5YWIwZWNiM2ZlN2I4YThlNGQ0ZWZiNjMyNDQ5MjFkZTJhMTY2OQo=

Screenshot of code:



Tool generated output:


```
{
    "Description": "Detected a Generic API Key, potentially exposing access to various services and sensitive operations.",
    "StartLine": 169,
    "EndLine": 169,
    "StartColumn": 23,
    "EndColumn": 0,
    "Match": "TOKEN_ENCODED: ZWU5YWIwZWNiM2ZlN2I4YThlNGQ0ZWZiNjMyNDQ5MjFkZTJhMTY2OQo=",
    "Secret": "ZWU5YWIwZWNiM2ZlN2I4YThlNGQ0ZWZiNjMyNDQ5MjFkZTJhMTY2OQo=",
    "File": "docker/development-insane/docker-compose.yml",
    "SymlinkFile": "",
    "Commit": "fe61175c8f0cc33552004ddc122b0722043f0d29",
    "Entropy": 4.4834447,
    "Author": "Brady Miller",
    "Email": "brady.g.miller@gmail.com",
    "Date": "2021-03-20T23:16:32Z",
    "Message": "docker dev fix (#4294)",
    "Tags": [],
    "RuleID": "generic-api-key",
    "Fingerprint": "fe61175c8f0cc33552004ddc122b0722043f0d29:docker/development-insane/docker-compose.yml:generic-api-key:169"
}
```

5) **Secret type:** Authentication Token

Exposed Secret:

QkJENjI5MzUtRjc2Ri00RUM4LTg4MzQtQkRBQTc1REFEOEFCOjlBMzVEMzEzLUU3QkMtNDFDOS04OTMzLTNBM0Q3Mzk1M0Y3Mw==

Screenshot of code:

```
class Easipro
{
    public function __construct()
    {
        // direct to easipro dev server
        $this->server = "https://www.assessmentcenter.net/ac_api/2014-01/";

        // credential token for easipro dev server
        $this->devAuth = "QkJENjI5MzUtRjc2Ri00RUM4LTg4MzQtQkRBQTc1REFEOEFCOjlBMzVEMzEzLUU3QkMtNDFDOS04OTMzLTNBM0Q3Mzk1M0Y3Mw==";
    }

    // Collect list of forms (returns json)
    public function listForms()
    {
        $response = oeHttp::usingHeaders(['Authorization' => 'Basic ' . $this->devAuth])->get($this->server . 'Forms/.json');
        $data = $response->body();
```

Tool generated output:


```
{
    "Description": "Detected a Generic API Key, potentially exposing access to various services and s
    "StartLine": 27,
    "EndLine": 27,
    "StartColumn": 20,
    "EndColumn": 128,
    "Match": "Auth = \"QkJENjI5MzUtRjc2Ri00RUM4LTg4MzQtQkRBQTc1REFEOEFCOjlBMzVEMzEzLUU3QkMtNDFDOS04OT
    "Secret": "QkJENjI5MzUtRjc2Ri00RUM4LTg4MzQtQkRBQTc1REFEOEFCOjlBMzVEMzEzLUU3QkMtNDFDOS04OTMzLTNBM0
    "File": "src/Easipro/Easipro.php",
    "SymlinkFile": "",
    "Commit": "9c5de9ce761c44061927f22a75f641749841db1c",
    "Entropy": 4.7440248,
    "Author": "StrongTSQ",
    "Email": "StrongTSQ@gmail.com",
    "Date": "2020-02-04T19:41:09Z",
    "Message": "easipro feature, 1st commit\n\nSee following PR for details:\nhttps://github.com/open
    "Tags": [],
    "RuleID": "generic-api-key",
    "Fingerprint": "9c5de9ce761c44061927f22a75f641749841db1c:src/Easipro/Easipro.php:generic-api-key:
},
```

6) **Secret type:**Token
Exposed Secret: SHBhCq4BLOpxJww3DG6g7dtKcuhG62zcK

Screenshot of code:

Tool generated output:


```
{
    "Description": "Detected a Generic API Key, potentially exposing access to various services and sensitive operations.",
    "StartLine": 1,
    "EndLine": 1,
    "StartColumn": 6,
    "EndColumn": 0,
    "Match": "token: SHBhCq4BLOpxJww3DG6g7dtKcuhG62zcK",
    "Secret": "SHBhCq4BLOpxJww3DG6g7dtKcuhG62zcK",
    "File": "public/assets/i18next-9-0-1/.coveralls.yml",
    "SymlinkFile": "",
    "Commit": "215c131cfcf749a5456ecaf4c493cedb647272c3",
    "Entropy": 4.6201515,
    "Author": "Brady Miller",
    "Email": "brady.g.miller@gmail.com",
    "Date": "2017-11-13T17:21:47Z",
    "Message": "added dwv package and dependencies (#1216)",
    "Tags": [],
    "RuleID": "generic-api-key",
    "Fingerprint": "215c131cfcf749a5456ecaf4c493cedb647272c3:public/assets/i18next-9-0-1/.coveralls.yml:generic-api-key:1"
},
```

7) **Secret type:** Private key

Exposed Secret:

-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEArmjyxQOgKQ17D4IEG0XvDP3Q58oq+6bqRkDCgFlrj6GkSbbt
9NKRwgA+ewjKO5uef8jJuFSWGsOTCE9E5kIZnjzdRKCXDackDWo52DfbxgDmVRp3
g95BjEUzoVk717TZ6nxZxLNqQHmckmPFBZI3lWX3nFYflaSgosL4ODbuCr1o/i7R
2ofMkz86Q23kAmXCRfiugmObhXrNZbNjOrGUdiAx9DN5N0dX+ywIJ6UbWnHuqbXE
+Vbhy6hEfur3AzWDQOdPy/2IezTba2KLAuKJbteGnt0wJ/ar311fqTdJ8r1HJ8Qq
6MNjuMBC+k4ku7F2HY60axG9RP7jfUqbL5N/bwIDAQABAoIBAAWlrEaoyWTK5PMp
mIxxhT38cekvaJjhXLCNpD6Xpp3t8+3XGHlW4BVSALPnIvUvIsCsdaucQYf4UShx
uJ9kAIRZICj6sillAgvVJq5TkLAmk06qHriMV7789qEAzKk9V9j50cZGsOgJiAJN
bLGzfE1AEfZj63ohZHogqfzX+nkqfy62tuQkrw5mHCHAxuHzIXKe5D/3ugOJAt7/
K7rpcsbFjj0N43INm5ffn4jJXX7Ko/y8asBSFEmmWiKp39wlcd6A/us18eYjFCIP
6GowncGruFxQSQoZK+603nWqw1CdrZlQIaCdD1AtDkP8YZUzJWDQjdr2o35Ef/Ib
UGnCxmECgYEA1LC8QYO9n4Qjkovyvf+t8tlsBm1e5DD0wsDAQRFxiORLeOJsETIG
BYrehIYsTDjSGtl/SqHhnprmMo/mczfjvVbcxNAq518b1KQiw61uiXCLj1lAlcc+
WKyPVCgjIaJQowjYRo7guy6XGWWprZK7RaU+Sqz3JYt2vQ3KIPyISSkCgYEA0ey0
uQMdwY7YwYzn48/OUnhU0MFL5YEJfbwblMKQMDfK8yUQ796F44L53F9+Uxf05+o4
89vSUugOGYg6+znpDhBM9w2SzAxL54PtM+YZCw7EXJswWHYrK+JcgHYS1SWZAtMx
BsO5pjgUmaapGhNmUM51uxzeN5XSrjQ8X4dZXtcCgYEAxRswq8pqLe1ohgQaJohP
bVKfMOmk2fagreexg9uliwZ/Bv1WxyzDtdk3VDt/y/eQhv9vXvvcun60vMDtAXQX
Iz3fhKmiSR+0DAE8knkkikx5mZROLUAjgEwS3BwCSsZA0oZ84A7DJ9UKhKfZcpkb
4nQ74TqgI/Q2Swf/fUhrCNkCgYEAn9PRkN8vg5Pnsg0+7HLmCmI1aS8SQu1otYNH
mI49y4hfC5pX99j7JUouRDusaUVfIEdzDy95RcgKHCRdi7L+nseIYbzC3HosPFt0
eMQiowmeNUeumlt16RaSCPhcxE243q6+sADK0yP1gzqhmwnUJ4fZmRmUksuOOTXq
FW1ufMcCgYBTqhDgLjj2kM5ZCL6TGZ63zQ0IxBgYGwUPsa3JwecDkQOPfTc37zte
cZ9K9WuMJccMfblgJl7k2YL9Jp91KeHRvuraI2kWwII4VynuVAnJVKxxNEsnZn27
YWDo52SqT/WajC99HKjnw3aD2PCZ5tma/VQejrZv8XVZMbPlkUipCg==
-----END RSA PRIVATE KEY-----

Screenshot of code:

contrib > util > docker > ldap-ssl-certs-keys > easy > ca-key.pem

Brady Miller, 4 years ago | 1 author (Brady Miller)

```
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEArmjyxQOgKQ17D4IEG0XvDP3Q58oq+6bqRkDCgFlrj6GkSbbt
9NKRwgA+ewjKO5uef8jJuFSWGsOTCE9E5kIZnjzdRKCXDackDWo52DfbxgDmVRp3
g95BjEUzoVk717TZ6nxZxLNqQHmckmPFBZI3lWX3nFYfIaSgosL4ODbuCr1o/i7R
2ofMkz86Q23kAmXCRfiugmObhXrNZbNjOrGUdiAx9DN5N0dX+ywIJ6UbWnHuqbXE
+Vbhy6hEfur3AzWDQOdPy/2IezTba2KLAuKJbteGnt0wJ/ar311fqTdJ8r1HJ8Qq
6MNjuMBC+k4ku7F2HY60axG9RP7jfUqbL5N/bwIDAQABAoIBAAWlrEaoyWTK5PMp
mIxxhT38cekvaJjhXLCNpD6Xpp3t8+3XGHlW4BVSALPnIvUvIsCsdaucQYf4UShx
uJ9kAIRZICj6sillAgvVJq5TkLAmk06qHriMV7789qEAzKk9V9j50cZGs0gJiAJN
bLGzfE1AEfZj63ohZHogqfzX+nkqfy62tuQkrw5mHCHAxuHzIXKe5D/3ugOJAt7/
K7rpcsbFjj0N43INm5ffn4jJXX7Ko/y8asBSFEmmWiKp39wlcd6A/us18eYjFCIP
6GowncGruFxQSQoZK+603nWqw1CdrZlQIaCdD1AtDkP8YZUzJWDQjdr2o35Ef/Ib
UGnCxmECgYEA1LC8QYO9n4Qjkovyvf+t8tlsBm1e5DD0wsDAQRFxiORLeOJsETIG
BYrehIYsTDjSGtl/SqHhnprmMo/mczfjvVbcxNAq518b1KQiw61uiXCLj1lAlcc+
WKyPVCgjIaJQowjYRo7guy6XGWWprZK7RaU+Sqz3JYt2vQ3KIPyISSkCgYEA0ey0
uQMdwY7YwYzn48/OUnhU0MFL5YEJfbwblMKQMDfK8yUQ796F44L53F9+Uxf05+o4
89vSUugOGYg6+znpDhBM9w2SzAxL54PtM+YZCw7EXJswWHYrK+JcgHYS1SWZAtMx
BsO5pjgUmaapGhNmUM51uxzeN5XSrjQ8X4dZXtcCgYEAxRswq8pqLe1ohgQaJohP
bVKfMOmk2fagreexg9uliwZ/Bv1WxyzDtdk3VDt/y/eQhv9vXvvcun60vMDtAXQX
Iz3fhKmiSR+0DAE8knkkikx5mZROLUAjgEwS3BwCSsZA0oZ84A7DJ9UKhKfZcpkb
4nQ74TqgI/Q2Swf/fUhrCNkCgYEAn9PRkN8vg5Pnsg0+7HLmCmI1aS8SQu1otYNH
mI49y4hfC5pX99j7JUouRDusaUVfIEdzDy95RcgKHCRdi7L+nseIYbzC3HosPFt0
eMQiowmeNUeumlt16RaSCPhcxE243q6+sADK0yP1gzqhmwnUJ4fZmRmUksuOOTXq
FW1ufMcCgYBTqhDgLjj2kM5ZCL6TGZ63zQ0IxBgYGwUPsa3JwecDkQOPfTc37zte
cZ9K9WuMJccMfblgJl7k2YL9Jp91KeHRvuraI2kWwII4VynuVAnJVKxxNEsnZn27
YWDo52SqT/WajC99HKjnw3aD2PCZ5tma/VQejrZv8XVZMbPlkUipCg==
-----END RSA PRIVATE KEY-----
```

Tool generated output:


```
{
    "Description": "Identified a Private Key, which may compromise cryptographic security and sensitive data encryption.",
    "StartLine": 1,
    "EndLine": 27,
    "StartColumn": 1,
    "EndColumn": 29,
    "Match": "-----BEGIN RSA PRIVATE KEY-----\nMIIEpAIBAAKCAQEArmjyxQOgKQ17D4IEG0XvDP3Q58oq+6bqRkDCgFlrj6GkSbbt\n9NKRwgA+ewjKO5uef8jJuFSWGsOTCE9E5kIZnjz
    "Secret": "-----BEGIN RSA PRIVATE KEY-----\nMIIEpAIBAAKCAQEArmjyxQOgKQ17D4IEG0XvDP3Q58oq+6bqRkDCgFlrj6GkSbbt\n9NKRwgA+ewjKO5uef8jJuFSWGsOTCE9E5kIZnj
    "File": "contrib/util/docker/ldap-ssl-certs-keys/easy/ca-key.pem",
    "SymlinkFile": "",
    "Commit": "62c506acb492c087671a965d304c856afb6c8f59",
    "Entropy": 6.0210257,
    "Author": "Brady Miller",
    "Email": "brady.g.miller@gmail.com",
    "Date": "2020-08-19T19:32:39Z",
    "Message": "Ldap tls support (#3870)\n\n* ldap tls support\n\n* updates",
    "Tags": [],
    "RuleID": "private-key",
    "Fingerprint": "62c506acb492c087671a965d304c856afb6c8f59:contrib/util/docker/ldap-ssl-certs-keys/easy/ca-key.pem:private-key:1"
},
```

8) **Secret type:** Private Key

Exposed Secret:


```
-----BEGIN RSA PRIVATE KEY-----
MIIEpQIBAAKCAQEAxitr10bBdY7zHhrXE8u9tFXJk71wSIl63tBo0insHMonDYhc
XXZte2DPRnTB/k+j6AkjktMxtTUbF6tkbhZIKcfWSBv7onfqSLsc9BkILlfDLDgy
6NEMNmKCQwqfgvo5V8FiDPIBeVINevf3Nzk/JLzPewUN1negHkG1GUTXt835JDSY
HILLBk4HLrCVR9lnuy9Lm4IDQ4EJm9KN1+04ofzBQ2W3p25DpAzqaL3QTVHfK/rQ
tvPQR09dslKeYfB7/rpzE1cPf8AcsW2AsBWDZROFEdtJ7eWga9LX4s/r28f5MdDC
L6lWwcGblcOl6LSn4DZJ6IEVQMVxvBIfQAkanwIDAQABAoIBAQC3umA0aiRI8uzS
d336m4XZYUQPlq44gkYJuF3GpBmFo6LLeCZyfK8KmXnyz6NgPcQwB/EUWnwWjhL9
QeIN65iiK+WuW0tUmSr8nYY9hcvA85s/Lyd8FPzmmzQ8Jeg3zfhGNUPLzr+xFHLt
FGL+rzOswuT2IHxw393yz2aw89OwU0LzegmIKm4tNPIBDVwYqrbldUaf5wUubO1J
Xk1ytWulJcx5+QlHHkWKebjjoW5xdH+iKF5Ppd9dX8Ct+6SzzxLR2HVnMPN1UT34
wwUI5rjaq02I0osI1WFL9qgHBG5ZAUn85sMMyK/sXAbQVtzSPlYwifMvn9TOWCmU
dPNfZy/BAoGBAPPzY0TtzBVBHcQRW1TCsVBw4NGKrae5RNO6ifxlh5wF1Y5KxKCq
5zXpP9HUKiOhHwdlXgDa8H9a/Kwm+L4gUZ7Z0z9O9EEds08M2nIUnh8wKrQGdEQV
ETflk+Xf0WcZ64Lncks9QiRt2IktrdSdtilS1id1AhssmhXqoh4oqoO/AoGBAM/1
KI0mxIlSfcSQ6KWiIKgOWxUbktYaG6swDTALnrZP/0SIbLJ96drz8sLATw67WGys
O8eBKHoLD+9IKmLEgnoHPacEa9ykxrPOrWHV84OhqjPzFmpcQ20kAZekvgrAwkOl
jYFrLj5PXECetPaSNsBzrsNUtKW6rSKGJisLjaEhAoGBAKn6DFEa3/hvRq426cxF
XAcQn5Vha15SWQKuFzcmBzdPWV2r+xV3/2IxA6bQjgwFV/IvR/CQnMIgQ64v+WGA
Y16rp+WUvdwhNa7+5bZXZ1WT2i1V0e0eQMER1dt1+BI+nTq4yYpjRJLdduMeUOC0
gxiH/2++t5j3muVqd0tMSoNBAoGAOKczOvD6zeZZOPm5R5ZZiy8OBTwTzpwiOYrH
rPUdcdFtW4BHDIFc+jdHRWMaporHViO9zWYUXtswpiFk2q9qVFAwrZV4xQ1mzIlb
lrGwu4WJVsm8q5EjFVKf1G4fD9LfmYo6eK14VaVNHpWd+yZuIBPj4nbrF9M5Y5Rk
RYnKNcECgYEAnAoATpLcHa5OH5bDtO8PxftsoUHr+W6pSeYPeZLlRWsFNZnI6cnA
qyJh6YNrLDJaBFcFq9HmYgVVaYa15P5vB7KpweepgJJZ7Vdws+78ZA7fmyR+cKDj
PG++Yr9Lj8sJILQzuqcTA/XvX0cJb6f4oK/YbmXwEa8DFxnG8R6TcN8=
-----END RSA PRIVATE KEY-----
```

Screenshot of code:

contrib > util > docker > ldap-ssl-certs-keys > easy > client-key.pem

Brady Miller, 4 years ago | 1 author (Brady Miller)

```
1  -----BEGIN RSA PRIVATE KEY-----
2  MIIEpQIBAAKCAQEAxitr10bBdY7zHhrXE8u9tFXJk71wSIl63tBo0insHMonDYhc
3  XXZte2DPRnTB/k+j6AkjktMxtTUbF6tkbhZIKcfWSBv7onfqSLsc9BkILlfDLDgy
4  6NEMNmKCQwqfgvo5V8FiDPIBeVINevf3Nzk/JLzPewUN1negHkG1GUTXt835JDSY
5  HILLBk4HLrCVR9lnuy9Lm4lDQ4EJm9KN1+04ofzBQ2W3p25DpAzqaL3QTVHfK/rQ
6  tvPQR09dslKeYfB7/rpzE1cPf8AcsW2AsBWDZROFEdtJ7eWga9LX4s/r28f5MdDC
7  L6lWwcGblcOl6LSn4DZJ6IEVQMVxvBIfQAkanwIDAQABAoIBAQC3umA0aiRI8uzS
8  d336m4XZYUQPlq44gkYJuF3GpBmFo6LLeCZyfK8KmXnyz6NgPcQwB/EUWnwWjhL9
9  QelN65iiK+WuW0tUmSr8nYY9hcvA85s/Lyd8FPzmmzQ8Jeg3zfhGNUPLzr+xFHLt
10 FGL+rzOswuT2IHxw393yz2aw89OwU0LzegmIKm4tNPIBDVwYqrbldUaf5wUubO1J
11 Xk1ytWulJcx5+QlHHkWKebjjoW5xdH+iKF5Ppd9dX8Ct+6SzzxLR2HVnMPN1UT34
12 wwUI5rjaq02I0osl1WFL9qgHBG5ZAUn85sMMyK/sXAbQVtzSPlYwifMvn9TOWCmU
13 dPNfZy/BAoGBAPPzY0TtzBVBHcQRW1TCsVBw4NGKrae5RNO6ifxlh5wF1Y5KxKCq
14 5zXpP9HUKiOhHwdlXgDa8H9a/Kwm+L4gUZ7Z0z9O9EEds08M2nIUnh8wKrQGdEQV
15 ETflk+Xf0WcZ64Lncks9QiRt2IktrdSdtilS1id1AhssmhXqoh4oqoO/AoGBAM/1
16 KI0mxIlSfcSQ6KWiIKgOWxUbktYaG6swDTALnrZP/0SIbLJ96drz8sLATw67WGys
17 O8eBKHoLD+9IKmLEgnoHPacEa9ykxrPOrWHV84OhqjPzFmpcQ20kAZekvgrAwkOl
18 jYFrLj5PXECetPaSNsBzrsNUtKW6rSKGJisLjaEhAoGBAKn6DFEa3/hvRq426cxF
19 XAcQn5Vha15SWQKuFzcmBzdPWV2r+xV3/2IxA6bQjgwFV/lvR/CQnMIgQ64v+WGA
20 Y16rp+WUvdwhNa7+5bZXZ1WT2i1V0e0eQMER1dt1+BI+nTq4yYpjRJLdduMeUOC0
21 gxiH/2++t5j3muVqd0tMSoNBAoGAOKczOvD6zeZZOPm5R5ZZiy8OBTwTzpwiOYrH
22 rPUdcdFtW4BHDIFc+jdHRWMaporHViO9zWYUXtswpiFk2q9qVFAwrZV4xQ1mzIlb
23 lrGwu4WJVsm8q5EjFVKf1G4fD9LfmYo6eK14VaVNHpWd+yZuIBPj4nbrF9M5Y5Rk
24 RYnKNcECgYEAnAoATpLcHa5OH5bDtO8PxftsoUHr+W6pSeYPeZllRWsFNZnI6cnA
25 qyJh6YNrLDJaBFcFq9HmYgVVaYa15P5vB7KpweepgJJZ7Vdws+78ZA7fmyR+cKDj
26 PG++Yr9Lj8sJILQzuqcTA/XvX0cJb6f4oK/YbmXwEa8DFxnG8R6TcN8=
27 -----END RSA PRIVATE KEY-----
28
```

Tool generated output:


```
{
  "Description": "Identified a Private Key, which may compromise cryptographic security and sensitive data encryption.",
  "StartLine": 1,
  "EndLine": 27,
  "StartColumn": 1,
  "EndColumn": 29,
  "Match": "-----BEGIN RSA PRIVATE KEY-----\nMIIEpQIBAAKCAQEAxitr10bBdY7zHhrXE8u9tFXJk71wSIl63tBo0insHMonDYhc\nXXZte2DPRnTB/k+j6AkjktMxtTUbF
  "Secret": "-----BEGIN RSA PRIVATE KEY-----\nMIIEpQIBAAKCAQEAxitr10bBdY7zHhrXE8u9tFXJk71wSIl63tBo0insHMonDYhc\nXXZte2DPRnTB/k+j6AkjktMxtTUb
  "File": "contrib/util/docker/ldap-ssl-certs-keys/easy/client-key.pem",
  "SymlinkFile": "",
  "Commit": "62c506acb492c087671a965d304c856afb6c8f59",
  "Entropy": 6.01207,
  "Author": "Brady Miller",
  "Email": "brady.g.miller@gmail.com",
  "Date": "2020-08-19T19:32:39Z",
  "Message": "Ldap tls support (#3870)\n\n* ldap tls support\n\n* updates",
  "Tags": [],
  "RuleID": "private-key",
  "Fingerprint": "62c506acb492c087671a965d304c856afb6c8f59:contrib/util/docker/ldap-ssl-certs-keys/easy/client-key.pem:private-key:1"
}
```

9) **Secret type:** Private Key

Exposed Secret:

```
-----BEGIN RSA PRIVATE KEY-----
MIIEpQIBAAKCAQEA1O3a0T78PopUxuBRsARC7gnONEwgtkoDyu0vt5j2p1ydhMAP
VNsZ5sSwaECgiZ8uzjDgMiQFCUj+lxTn9W9nMQPc7g/o25lrGtQQqQTkGcvJXnac
y1ToSe5QYrDRya8WMRDzMPFVNL9esuwoGgbzZWixfq2coJx9u25V0uqPwvvPS0Xh
DOOo1t/Urc/Gt9TAXaDHDHovSAtWUv7c3POmGQfO1Ogydwa047yYMu1U0CH3Xrtq
fPv19D6BVs4U4XHsbyrCP/9s+P6uBKeNNRrcBRIhKvcFVLGGTZ5l1sI1O8T4fYJx
yYXZczJSWMBK2l6Dw708iAFEfUUILfdWeXVuEQIDAQABAoIBAQDHS93crkRw0Q8q
mjk7M7HllT3xCzQvCrXkEP7xrHVpDrHM1+tBtFXY/k5TNfJM/bZUfcDKjZY9K5pH
lOTLf7spUIFQPYnpSP7LvKPfX1mUn35aMVDurAlDcZDB3thtGrj9sey96ZSj617a
v1mFiu30A/rmI9+PIQo6Dbq80JlAmLSRjpq0LGYz/hezXK4iuvkE1nKD7rLPtkHB
llhFbQ5d0yhvIMw7YA+tiTeAAF+x+UPsp5fMZ9cfBU31jPYkc2YorgRKLwp3qcby
yPvrD0fzln9xvojjRc6GyKA2qPlSiLjYLIdpcDV/+RXFPZzfgfzBs0ensfdwZJyf
uTjmqfHVAoGBAPdsZ5u7/VG3RXbgpfrlhGPvir4YRHD8kkFyQ4Cv9t9vEIV6ciln
9j4u9eurk5pimG71a4gt5Ju00FAf5RoWWQeNzrdWm66ZRjIS8YluTwP/1W4eSP51
ReDWw34AZZJkmK3KiNRB2Ul0PIPpfXW0af1GIaMhVrFBoTtKxPGgPxmHAoGBANxP
Wlpa4dCWn2nPf6/JgwO5spgXYeP6lHC7DdfzfT4CykzroJo71MAms6W1XvrIx777
pDnzPqayUFPZnlCWlMl4GglRDRWqcbah2nqZ5ekpP1dOCyUkpVPfPDMsmZwCv0kQ
RyvUioqPuW5WNOOKsn89FTXabe5XPP9DWjjgy8GnAoGBAIxjpD5HYqs1NGMJgORl
ysLxrmXCW0NI7KvZ69dfANMyr8ss+Q3pgV0zRDKevjfBbgC3nHY+pBJ9fB2am2I7
//Q7UGnS6K8fJSdQfGCgiJEaXr4GzSAk5qt28KHUE84G6mLiItVdL69wFs46dAeP
gPxHCNsJFamqNZPxDuKTLHqNAoGAaTVX/6hBCHy1ujEGWCtSbrgCitE3xUiIFOGv
Nb6ow7FPbLZlpuIFspGl3wyjQ/4YLaM8hRQdijvDUpetG+0dNq0tOd/D8OHJlh7r
F/tgqR0yi33QordCYLBNrl6RKMjDEqA8bLj+RDdopUFz1moI3eyMsYATWWu9zGUc
s+beRycCgYEAs0L8+OuoUsp7z9wB9vfTMZopEt7DJ4eG9owoo0xOL/RxWNLMvKrB
+fajHq/nY5ByLojTuQCm6RpykEkJjB+buS6e11omJHEJw9KynYDgOcL+B35QyMzE
oSGrVrKxjm0UKsJ6UWF86DIo8vkr/IUK/6C8KkIiPa0MMSUHiudLz0M=
-----END RSA PRIVATE KEY-----
```

Screenshot of code:

contrib > util > docker > ldap-ssl-certs-keys > easy > server-key.pem

Brady Miller, 4 years ago | 1 author (Brady Miller)

```
-----BEGIN RSA PRIVATE KEY-----
MIIEpQIBAAKCAQEA1O3a0T78PopUxuBRsARC7gnONEwgtkoDyu0vt5j2p1ydhMAP
VNsZ5sSwaECgiZ8uzjDgMiQFCUj+lxTn9W9nMQPc7g/o25lrGtQQqQTkGcvJXnac
y1ToSe5QYrDRya8WMRDzMPFVNL9esuwoGgbzZWixfq2coJx9u25V0uqPwvvPS0Xh
DOOo1t/Urc/Gt9TAXaDHDHovSAtWUv7c3POmGQfO1Ogydwa047yYMu1U0CH3Xrtq
fPv19D6BVs4U4XHsbyrCP/9s+P6uBKeNNRrcBRIhKvcFVLGGTZ5l1sI1O8T4fYJx
yYXZczJSWMBK2l6Dw708iAFEfUUILfdWeXVuEQIDAQABAoIBAQDHS93crkRw0Q8q
mjk7M7HllT3xCzQvCrXkEP7xrHVpDrHM1+tBtFXY/k5TNfJM/bZUfcDKjZY9K5pH
lOTLf7spUIFQPYnpSP7LvKPfX1mUn35aMVDurAlDcZDB3thtGrj9sey96ZSj617a
v1mFiu30A/rmI9+PIQo6Dbq80JlAmLSRjpq0LGYz/hezXK4iuvkE1nKD7rLPtkHB
llhFbQ5d0yhvIMw7YA+tiTeAAF+x+UPsp5fMZ9cfBU31jPYkc2YorgRKLwp3qcby
yPvrD0fzln9xvojjRc6GyKA2qPlSiLjYLIdpcDV/+RXFPZzfgfzBs0ensfdwZJyf
uTjmqfHVAoGBAPdsZ5u7/VG3RXbgpfrlhGPvir4YRHD8kkFyQ4Cv9t9vEIV6ciln
9j4u9eurk5pimG71a4gt5Ju00FAf5RoWWQeNzrdWm66ZRjIS8YluTwP/1W4eSP51
ReDWw34AZZJkmK3KiNRB2Ul0PIPpfXW0af1GIaMhVrFBoTtKxPGgPxmHAoGBANxP
Wlpa4dCWn2nPf6/JgwO5spgXYeP6lHC7DdfzfT4CykzroJo71MAms6W1XvrIx777
pDnzPqayUFPZnlCWlMl4GglRDRWqcbah2nqZ5ekpP1dOCyUkpVPfPDMsmZwCv0kQ
RyvUioqPuW5WNOOKsn89FTXabe5XPP9DWjjgy8GnAoGBAIxjpD5HYqs1NGMJgORl
ysLxrmXCW0NI7KvZ69dfANMyr8ss+Q3pgV0zRDKevjfBbgC3nHY+pBJ9fB2am2I7
//Q7UGnS6K8fJSdQfGCgiJEaXr4GzSAk5qt28KHUE84G6mLiItVdL69wFs46dAeP
gPxHCNsJFamqNZPxDuKTLHqNAoGAaTVX/6hBCHy1ujEGWCtSbrgCitE3xUiIFOGv
Nb6ow7FPbLZlpuIFspGI3wyjQ/4YLaM8hRQdijvDUpetG+0dNq0tOd/D8OHJlh7r
F/tgqR0yi33QordCYLBNrl6RKMjDEqA8bLj+RDdopUFz1moI3eyMsYATWWu9zGUc
s+beRycCgYEAs0L8+OuoUsp7z9wB9vfTMZopEt7DJ4eG9owoo0xOL/RxWNLMvKrB
+fajHq/nY5ByLojTuQCm6RpykEkJjB+buS6e11omJHEJw9KynYDgOcL+B35QyMzE
oSGrVrKxjm0UKsJ6UWF86DIo8vkr/IUK/6C8KkIiPa0MMSUHiudLz0M=
-----END RSA PRIVATE KEY-----
```

Tool generated output:


```
{
    "Description": "Identified a Private Key, which may compromise cryptographic security and sensitive data encryption.",
    "StartLine": 1,
    "EndLine": 27,
    "StartColumn": 1,
    "EndColumn": 29,
    "Match": "-----BEGIN RSA PRIVATE KEY-----\nMIIEpQIBAAKCAQEA1O3a0T78PopUxuBRsARC7gnONEwgtkoDyu0vt5j2p1ydhMAP\nVNsZ5sSwaECgiZ8uzjDgMiQFC
    "Secret": "-----BEGIN RSA PRIVATE KEY-----\nMIIEpQIBAAKCAQEA1O3a0T78PopUxuBRsARC7gnONEwgtkoDyu0vt5j2p1ydhMAP\nVNsZ5sSwaECgiZ8uzjDgMiQF
    "File": "contrib/util/docker/ldap-ssl-certs-keys/easy/server-key.pem",
    "SymlinkFile": "",
    "Commit": "62c506acb492c087671a965d304c856afb6c8f59",
    "Entropy": 6.0220375,
    "Author": "Brady Miller",
    "Email": "brady.g.miller@gmail.com",
    "Date": "2020-08-19T19:32:39Z",
    "Message": "ldap tls support (#3870)\n\n* ldap tls support\n\n* updates",
    "Tags": [],
    "RuleID": "private-key",
    "Fingerprint": "62c506acb492c087671a965d304c856afb6c8f59:contrib/util/docker/ldap-ssl-certs-keys/easy/server-key.pem:private-key:1"
}
```

10) **Secret type:** Private Key

Exposed Secret:

-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEAxedVAO6KskINwKpRXT9wFByBXHYKvgoXKi1tBAdlKsIChy4m
0K3Mq5rAXBy27j2O+mrgRHQHtFflrzrZYINrdf8fZO2xGJgwMjiRlTc5cNR7zFb+
JqjeMDzU/+DjLEmsDclobGSF9XTtmFwodfu8Kg4BUv7AO/EFBNoLev/165Ab7inx
LmSO0yphGxV8oqtElO7820RyZ0HtLHknXLu9iR3VJNXgWey8P336gZ3sytUs6JI4
r2acNpyrWhjiuLYrwSdKPASAEFVRYWOvH0atdjql4ywyig4VbsjcLZKazP/5Wt+H
PKjykEEfbHbzYKA4EU2u3/v+BxqM8ZFMFE5CnwIDAQABAoIBABWH122khh5ipVBi
tztn3W+ysjg293A5dM+35XZcv5rpRJEkakClx/Cbg3YCAbzloZqI6iFIZlYugeSQ
iHnBkn1h5i561ZuRYxzznpMXUP5/f4B5rmjSUmn5FEtKUizNTxSoHPG99f0AXaEH
1bcQTiZdI7O913gt/pfKiRk2o7Ce6gbcN5uSDVeQ57qn3SZesDDxjakxswQMrHPS
gBqF/yTFyzWHDSf/e0M6HPM2ZnzogSngVemtYXG+Mq6h16dzjYQGwwWBeiQL8zcW
BdaNhfkDxs47EiDuqjky5I8lktzFau02zY5CH2iVmx9XnGmfMaTjwyqTJfF6kXi3
IGvUrkECgYEA6ldD10YKPmMLJ7/3K0K0ob1aSnEgbaPe+Hb/7sd2vzZNhBgODPQ0
HnzqKy6pnDtV5TmlbmxBKdL8ycdIw9K/3HMJePppnjfxhPwqV80eYELy2uhpkLS8
gmAL4mfCtSgZk4Ef+6u14iAPT+BEy45PxQfno8ZXOQwFa291LVZ4308CgYEA2DHt
HyhGdkuP6wcdwBipou1ZJTX1VcXpOJ9k1HG4ZyMrqJwaPP237LmNRNQE13Fx/iMO
+pbmt5KsJTAg0vMPViRipRV/CGyHcTt1VygZojQB7c6HvFSuBfUP7I7x7y6xIFEI
ecqNUoyYOKRw7Pcqodp05+bKmCY4ttNN/JkHE7ECgYACj2mCVJWQEOzyRjqhEl1+
1t7BqXK9P+kTtdkd9rfpZuJ56vlcUt6u5pZ4Wk4oEs3hbzvLdQtRNajlW3QED61g
7zDPTY0R2XKeDVM5GliWnKSil59voUnd8saudou+Bu4gmmyLB1k5WzhPmxelUay1
blytgWE6klMM1W7mS+eOKQKBgF2SNkwO9XhLug2CNfknc3xl9/hoAzhNgWSVcGGW
mrZg/YtrjNurAlj/wknw7euFWU4iRGBnIZKAc+tgikRjGNcyxlMnG6U8u95hWqcP
XxDaK5+OjScRC4TejTbYo5PfBKC1EdMMvN/mpBeX7JxTWetVjtW0CSmTGWbfy9gx
L2PxAoGBAMA83lhO3+omsh+MfW5HEC2to6hRB9HTdiFsJWLaQlePYCpalNJeg5WV
dXj/+mxNBO//vZIr+5jGv3S8R0tWq61QLicDId/xB68e5ORuz+GM+rP5CP/An0RD
uDZe2Z9woka08FeDYcfCDLAtS3xVXToycW0wn/gge1AxKeARXXqa
-----END RSA PRIVATE KEY-----

Screenshot of code:

contrib > util > docker > ldap-ssl-certs-keys > insane > ca-key.pem

Brady Miller, 4 years ago | 1 author (Brady Miller)

```
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEAxedVAO6KskINwKpRXT9wFByBXHYKvgoXKi1tBAdlKsIChy4m
0K3Mq5rAXBy27j2O+mrgRHQHtFflrzrZYINrdf8fZO2xGJgwMjiRlTc5cNR7zFb+
JqjeMDzU/+DjLEmsDcIobGSF9XTtmFwodfu8Kg4BUv7AO/EFBNoLev/165Ab7inx
LmSO0yphGxV8oqtElO7820RyZ0HtLHknXLu9iR3VJNXgWey8P336gZ3sytUs6JI4
r2acNpyrWhjiuLYrwSdKPASAEFVRYWOvH0atdjql4ywyig4VbsjcLZKazP/5Wt+H
PKjykEEfbHbzYKA4EU2u3/v+BxqM8ZFMFE5CnwIDAQABAoIBABWH122khh5ipVBi
tztn3W+ysjg293A5dM+35XZcv5rpRJEkakClx/Cbg3YCAbzloZqI6iFIZlYugeSQ
iHnBkn1h5i561ZuRYxzznpMXUP5/f4B5rmjSUmn5FEtKUizNTxSoHPG99f0AXaEH
1bcQTiZdI7O913gt/pfKiRk2o7Ce6gbcN5uSDVeQ57qn3SZesDDxjakxswQMrHPS
gBqF/yTFyzWHD5f/e0M6HPM2ZnzogSngVemtYXG+Mq6h16dzjYQGwwWBeiQL8zcW
BdaNhfkDxs47EiDuqjky5I8lktzFau02zY5CH2iVmx9XnGmfMaTjwyqTJfF6kXi3
IGvUrkECgYEA6ldD10YKPmMLJ7/3K0K0ob1aSnEgbaPe+Hb/7sd2vzZNhBgODPQ0
HnzqKy6pnDtV5TmlbmxBKdL8ycdIw9K/3HMJePppnjfxhPwqV80eYELy2uhpkLS8
gmAL4mfCtSgZk4Ef+6u14iAPT+BEy45PxQfno8ZXOQwFa291LVZ4308CgYEA2DHt
HyhGdkuP6wcdwBipou1ZJTX1VcXpOJ9k1HG4ZyMrqJwaPP237LmNRNQE13Fx/iMO
+pbmt5KsJTAg0vMPViRipRV/CGyHcTt1VygZojQB7c6HvFSuBfUP7I7x7y6xIFEI
ecqNUoyYOKRw7Pcqodp05+bKmCY4ttNN/JkHE7ECgYACj2mCVJWQEOzyRjqhEl1+
1t7BqXK9P+kTtdkd9rfpZuJ56vlcUt6u5pZ4Wk4oEs3hbzvLdQtRNajlW3QED61g
7zDPTY0R2XKeDVM5GliWnKSil59voUnd8saudou+Bu4gmmyLB1k5WzhPmxelUay1
blytgWE6klMM1W7mS+eOKQKBgF2SNkwO9XhLug2CNfknc3xl9/hoAzhNgWSVcGGW
mrZg/YtrjNurAIj/wknw7euFWU4iRGBnIZKAc+tgikRjGNcyxlMnG6U8u95hWqcP
XxDaK5+OjScRC4TejTbYo5PfBKC1EdMMvN/mpBeX7JxTWetVjtW0CSmTGWbfy9gx
L2PxAoGBAMA83IhO3+omsh+MfW5HEC2to6hRB9HTdiFsJWLaQlePYCpalNJeg5WV
dXj/+mxNBO//vZIr+5jGv3S8R0tWq61QLicDId/xB68e5ORuz+GM+rP5CP/An0RD
uDZe2Z9woka08FeDYcfCDLAtS3xVXToycW0wn/gge1AxKeARXXqa
-----END RSA PRIVATE KEY-----
```

Tool generated output:


```
{
    "Description": "Identified a Private Key, which may compromise cryptographic security and sensitive data encryption.",
    "StartLine": 1,
    "EndLine": 27,
    "StartColumn": 1,
    "EndColumn": 29,
    "Match": "-----BEGIN RSA PRIVATE KEY-----\nMIIEowIBAAKCAQEAxedVAO6KskINwKpRXT9wFByBXHYKvgoXKi1tBAdlKsIChy4m\n0K3Mq5rAXBy27j2O+mrgRHQ
    "Secret": "-----BEGIN RSA PRIVATE KEY-----\nMIIEowIBAAKCAQEAxedVAO6KskINwKpRXT9wFByBXHYKvgoXKi1tBAdlKsIChy4m\n0K3Mq5rAXBy27j2O+mrgRH
    "File": "contrib/util/docker/ldap-ssl-certs-keys/insane/ca-key.pem",
    "SymlinkFile": "",
    "Commit": "62c506acb492c087671a965d304c856afb6c8f59",
    "Entropy": 6.027987,
    "Author": "Brady Miller",
    "Email": "brady.g.miller@gmail.com",
    "Date": "2020-08-19T19:32:39Z",
    "Message": "Ldap tls support (#3870)\n\n* ldap tls support\n\n* updates",
    "Tags": [],
    "RuleID": "private-key",
    "Fingerprint": "62c506acb492c087671a965d304c856afb6c8f59:contrib/util/docker/ldap-ssl-certs-keys/insane/ca-key.pem:private-key:1"
},
```

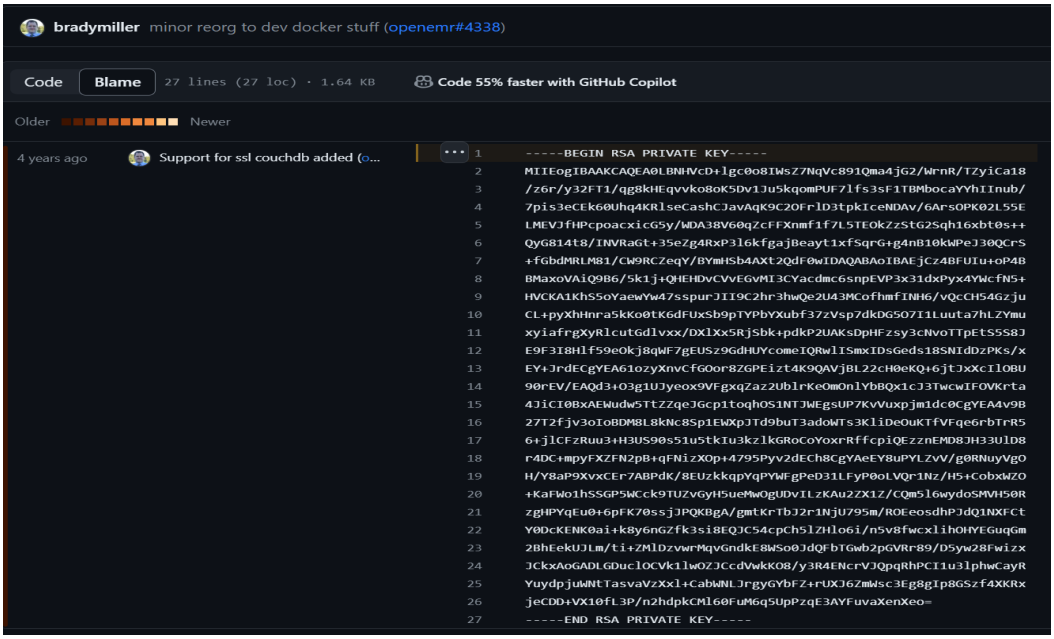
Gitleaks false positive pattern:
Gitleaks often produces false positives primarily in documentation, readme files, and test code. This issue arises because Gitleaks employs a keyword-matching approach within the codebase, identifying terms like "password" or "token." However, the tool tends to interpret any such match as a potential hardcoded password, even when these instances are benign, such as in testing code or documentation. Notably, Gitleaks may flag variables named "password" or “token” as if they were actual hardcoded credentials, leading to a substantial number of inaccuracies. This behavior can be attributed to the tool's keyword-centric analysis, causing a significant volume of false positives that may not impact the production code but can create challenges in accurately identifying genuine security vulnerabilities.

Tool 2: SpectralOps

1) **Secret type:** Private Key
   Exposed Secret:

-----BEGIN RSA PRIVATE KEY-----
MIIEogIBAAKCAQEA0LBNHVcD+lgc0o8IWsZ7NqVc891Qma4jG2/WrnR/TZyiCa18
/z6r/y32FT1/qg8kHEqvvko8oK5Dv1Ju5kqomPUF7lfs3sF1TBMbocaYYhIInub/
7pis3eCEk60Uhq4KRlseCashCJavAqK9C2OFrlD3tpkIceNDAv/6ArsOPK02L55E
LMEVJfHPcpoacxicG5y/WDA38V60qZcFFXnmf1f7L5TEOkZzStG2Sqh16xbt0s++
QyG814t8/INVRaGt+35eZg4RxP3l6kfgajBeayt1xfSqrG+g4nB10kWPeJ30QCrS
+fGbdMRLM81/CW9RCZeqY/BYmHSb4AXt2QdF0wIDAQABAoIBAEjCz4BFUIu+oP4B
BMaxoVAiQ9B6/5k1j+QHEHDvCVvEGvMI3CYacdmc6snpEVP3x31dxPyx4YWcfN5+
HVCKA1KhS5oYaewYw47sspurJII9C2hr3hwQe2U43MCofhmfINH6/vQcCH54Gzju
CL+pyXhHnra5kKo0tK6dFUxSb9pTYPbYXubf37zVsp7dkDG5O7I1Luuta7hLZYmu
xyiafrgXyRlcutGdlvxx/DXlXx5RjSbk+pdkP2UAKsDpHFzsy3cNvoTTpEtS5S8J
E9F3I8Hlf59eOkj8qWF7gEUSz9GdHUYcomeIQRwlISmxIDsGeds18SNIdDzPKs/x
EY+JrdECgYEA61ozyXnvCfGOor8ZGPEizt4K9QAVjBL22cH0eKQ+6jtJxXcIIOBU
90rEV/EAQd3+O3g1UJyeox9VFgxqZaz2UblrKeOmOnlYbBQx1cJ3TwcwIFOVKrta
4JiCI0BxAEWudw5TtZZqeJGcp1toqhOS1NTJWEgsUP7KvVuxpjm1dc0CgYEA4v9B
27T2fjv3oIoBDM8L8kNc8Sp1EWXpJTd9buT3adoWTs3KliDeOuKTfVFqe6rbTrR5
6+jlCFzRuu3+H3US90s51u5tkIu3kzlkGRoCoYoxrRffcpiQEzznEMD8JH33UlD8
r4DC+mpyFXZFN2pB+qFNizXOp+4795Pyv2dECh8CgYAeEY8uPYLZvV/g0RNuyVgO
H/Y8aP9XvxCEr7ABPdK/8EUzkkqpYqPYWFgPeD31LFyP0oLVQr1Nz/H5+CobxWZO
+KaFWo1hSSGP5WCck9TUZvGyH5ueMwOgUDvILzKAu2ZX1Z/CQm5l6wydoSMVH5R
zgHPYqEu0+6pFK70ssjJPQKBgA/gmtKrTbJ2r1NjU795m/ROEeosdhPJdQ1NXFCt
Y0DcKENK0ai+k8y6nGZfk3si8EQJC54cpCh5lZHlo6i/n5v8fwcxlihOHYEGuqGm
2BhEekUJLm/ti+ZMlDzvwrMqvGndkE8WSo0JdQFbTGwb2pGVRr89/D5yw28Fwizx
JCkxAoGADLGDuclOCVk1lwOZJCcdVwkKO8/y3R4ENcrVJQpqRhPCI1u3lphwCayR
YuydpjuWNtTasvaVzXxl+CabWNLJrgyGYbFZ+rUXJ6ZmWsc3Eg8gIp8GSzf4XKRx
jeCDD+VX10fL3P/n2hdpkCMl60FuM6q5UpPzqE3AYFuvaXenXeo=
-----END RSA PRIVATE KEY-----

Screenshot of code:



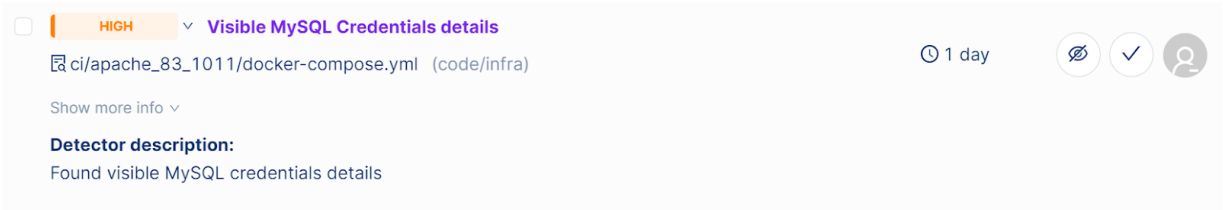
Tool generated output:

2) **Secret type:** SQL Credentials
Exposed Secret: MYSQL_ROOT_PASSWORD: root

Screenshot of code:

```yaml
# docker-compose.yml for travis ci testing
version: '3.1'
services:
  mysql:
    restart: always
    image: mariadb:10.11
    command: ['mysqld','--character-set-server=utf8mb4']
    environment:
      MYSQL_ROOT_PASSWORD: root
  openemr:
    restart: always
    image: openemr/openemr:flex-3.19
    ports:
    - 80:80
    - 443:443
    volumes:
    - ../../:/var/www/localhost/htdocs/openemr
    environment:
      FORCE_NO_BUILD_MODE: "yes"
      EMPTY: "yes"
    depends_on:
    - mysql
```

Tool generated output:

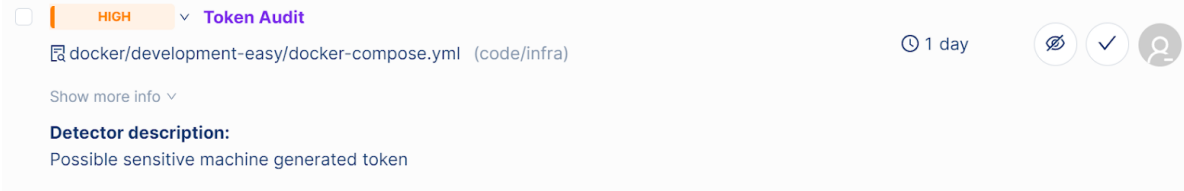


3) **Secret type:** API
Exposed Secret: c313de1ed5a00eb6ff9309559ec9ad01fcc553f0

Screenshot of code:

```
XDEBUG_CLIENT_HOST: host.docker.internal
GITHUB_COMPOSER_TOKEN: c313de1ed5a00eb6ff9309559ec9ad01fcc553f0
GITHUB_COMPOSER_TOKEN_ENCODED: ZWU5YWIwZWNiM2ZlN2I4YThlNGQ0ZWZiNjMyNDQ5MjFkZTJhMTY2OQo=
OPENEMR_DOCKER_ENV_TAG: easy-dev-docker
OPENEMR_SETTING_site_addr_oath: 'https://localhost:9300'
OPENEMR_SETTING_oauth_password_grant: 3
OPENEMR_SETTING_rest_system_scopes_api: 1
OPENEMR_SETTING_rest_api: 1
OPENEMR_SETTING_rest_fhir_api: 1
OPENEMR_SETTING_rest_portal_api: 1
OPENEMR_SETTING_portal_onsite_two_enable: 1
OPENEMR_SETTING_ccda_alt_service_enable: 3
OPENEMR_SETTING_couchdb_host: couchdb
OPENEMR_SETTING_couchdb_port: 6984
OPENEMR_SETTING_couchdb_user: admin
OPENEMR_SETTING_couchdb_pass: password
OPENEMR_SETTING_couchdb_dbase: example
OPENEMR_SETTING_couchdb_ssl_allow_selfsigned: 1
OPENEMR_SETTING_gbl_ldap_host: 'ldap://openldap:389'
OPENEMR_SETTING_gbl_ldap_dn: 'cn={login},dc=example,dc=org'
```

Tool generated output:



4) **Secret type:** Certificate

Exposed Secret:

-----BEGIN CERTIFICATE-----
MIIDKzCCAhMCAQEwDQYJKoZIhvcNAQELBQAwWzELMAkGA1UEBhMCVVMxEzARBgNV
BAgMCldhc2hpbmd0b24xEDAOBgNVBAcMB1NlYXR0bGUxEDAOBgNVBAoMB09wZW5F
TVIxEzARBgNVBAMMCm9wZW5lbXItY2EwHhcNMjAwNzI4MDkxMTExWhcNMzAwNjA2
MDkxMTExWjBcMQswCQYDVQQGEwJVUzETMBEGA1UECAwKV2FzaGluZ3RvbjEQMA4G
A1UEBwwHU2VhdHRsZTEQMA4GA1UECgwHT3BlbkVNUjEUMBIGA1UEAwwLbWFyaWFk
Yi1zc2wwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQC134e9cUELhjVI
olNd8z/A+fY4WnQsrpmC24exK0ZWoliULWpBr7D2ruMHZp/R3oUh04PcH1jfXbl2
V4abwyfEiF2wHGKw+patTyk7xT6n+XETHTQTOwLKMTzOoIIT/+KBu3ABtXKgk4le
/l8+JMgNryXRp1yEBQnBEjdRxu7D/AZtlhWBZimUYLjDjiKGn+QrdslgIiFG0zVF
snnwmnHWRTCtNACUtcIz4t6OvSjHlkNKkSsmKY2/Cr+YDVPvMLQ3HMdncuoDyKy6
g7aN1FWLtcJM8d8sxcOpMe++ac+ShZjQtejBEQDH6gJAN16IYKADrx867IxmduH4
iyPaEJalAgMBAAEwDQYJKoZIhvcNAQELBQADggEBAIqPl/B9J53elamI15TuMQLZ
49kBdqM1LZwiAftsftR+yEFZY0n8p1stR1q5mofaIjD4Ut8cy7/KaDHz2hzsYteK
9t2qltV83RjB+/kColldKAkfqhVuIF370E2rH5D1EL9gT5wn4RFrZ3D8lP/UmulT
Un37xTxbh2J+cw62QgfY2BBPXjCU50HKgHKoFA+ReO30+l04cclvOihN+SEwh0X0
bi2qTcaNDp3AYL3or+qB+yje0tkZ1cogoLqfR7RF1UJxnvqLXpNAZE0YH4iDhxyw
ap2EXdlR2o+nZVwlEQCl3Yqq0cb2nX/B5CSjJgqw3KzP8cXqoqumXt/ydAthXMk=
-----END CERTIFICATE-----

Screenshot of code:

```
1   -----BEGIN CERTIFICATE-----
2   MIIDKzCCAhMCAQEwDQYJKoZIhvcNAQELBQAwWzELMAkGA1UEBhMCVVMxEzARBgNV
3   BAgMCldhc2hpbmd0b24xEDAOBgNVBAcMB1NlYXR0bGUxEDAOBgNVBAoMB09wZW5F
4   TVIxEzARBgNVBAMMCm9wZW5lbXItY2EwHhcNMjAwNzI4MDkxMTExWhcNMzAwNjA2
5   MDkxMTExWjBcMQswCQYDVQQGEwJVUzETMBEGA1UECAwKV2FzaGluZ3RvbjEQMA4G
6   A1UEBwwHU2VhdHRsZTEQMA4GA1UECgwHT3BlbkVNUjEUMBIGA1UEAwwLbWFyaWFk
7   Yi1zc2wwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQC134e9cUELhjVI
8   olNd8z/A+fY4WnQsrpmC24exK0ZWoliULWpBr7D2ruMHZp/R3oUh04PcH1jfXbl2
9   V4abwyfEiF2wHGKw+patTyk7xT6n+XETHTQTOwLKMTzOoIIT/+KBu3ABtXKgk4le
10  /l8+JMgNryXRp1yEBQnBEjdRxu7D/AZtlhWBZimUYLjDjiKGn+QrdslgIiFG0zVF
11  snnwmnHWRTCtNACUtcIz4t6OvSjHlkNKkSsmKY2/Cr+YDVPvMLQ3HMdncuoDyKy6
12  g7aN1FWLtcJM8d8sxcOpMe++ac+ShZjQtejBEQDH6gJAN16IYKADrx867IxmduH4
13  iyPaEJalAgMBAAEwDQYJKoZIhvcNAQELBQADggEBAIqPl/B9J53elamI15TuMQLZ
14  49kBdqM1LZwiAftsftR+yEFZY0n8p1stR1q5mofaIjD4Ut8cy7/KaDHz2hzsYteK
15  9t2qltV83RjB+/kColldKAkfqhVuIF370E2rH5D1EL9gT5wn4RFrZ3D8lP/UmulT
16  Un37xTxbh2J+cw62QgfY2BBPXjCU50HKgHKoFA+ReO30+l04cclvOihN+SEwh0X0
17  bi2qTcaNDp3AYL3or+qB+yje0tkZ1cogoLqfR7RF1UJxnvqLXpNAZE0YH4iDhxyw
18  ap2EXdlR2o+nZVwlEQCl3Yqq0cb2nX/B5CSjJgqw3KzP8cXqoqumXt/ydAthXMk=
19  -----END CERTIFICATE-----
```

Tool generated output:

5) **Secret type:** Private Key
   Exposed Secret:

-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEA0/GRlWU0TyhSTfvb8zKchRdKSBDaH1jqceiPfTLJIwcc9sTv
WxMlmiCYOfxdBHVPC1Z9YB18ujFSR3q7p42mFx4k2kvB69C2SesHISfzSk90CnmB
bcupRJkYlumpy1NI/iOjLkXuttL7K7HNipNjHrv41ffk/kk5uXTFmtrRgldOL1er
K/n+Ma5RNmoxfiL4retmsabWwAAzPcJ/1iOCtY9CXJT0TQ2EfJ0lPWaV9KLYjD45
l3LUBKF1pEWktG93Xcmw9ognxnJGfPVhFIoMFPkQaqCaWXAeHXxtEsDmkt0BzGq+
9Q+wT8aHt5R9ySIsEBp17PyRDVDl7DM67UKLGwIDAQABAoIBAAZBJLix6d9El1ml
yxHcNn4+97Q0uxsdtp1x7XoWW5UrCpHFsgKQCBYKTDO53Mza4WBGRyDk/d9IwVLW
rl79cR9Rhmjiv3BEEn0P8H/r++P/gD8m4sjor7AUjIM3lgkoBglp59Dfw3GlvD/H
2XbzboNjrK4zyJAwfQGaDpLJTlF4UE7TuIs4ODtDChsYxEHWuDXW/ByQePGoAmod
0HkXpTQVW9UzA159lUU77OnTRGF52PG1OnJcXf6LOWnQ4pWZgEl+sbmL3stdxC1r
EK5MDd9f15lE8VzcI5t/ICqFFVjR5GINYzS/6bM9IG9We0whHFi3OifR1x1V2aLs
BKBe2nECgYEA+zhJVU5AWSBZo8r6bzewRW6AtXqFNdx4IIxxpabQfhFFznPi3UGQ
UfiKyR2SdWwYn4s9jhoUSw5OPCtcnO6zZzrMd3LAa+VAMOKmaD6gVW4VezymQ9WJ
60NJOS1MYl1SpjkhEzyImV9A/QUmlwksQuzoedAF2yUnKawySYlIGCcCgYEA1/n2
22723x0TXwopsHUtVTE6QH9Tuo0Z7u/DZ7Lh78NzjpLPIsJLuSiUU2vIt1WuK9Nx
9OBF3hYyZyJNT/szNZjugLJtaemXzbPSzC35C7U4964K1Vmbeyxvm3SDGDWkJIR+
L85aSiVcBXkf/BzUOeevnoXMwjGhzAgwp8Uxue0CgYEAyxrZpx9JxQTzlAoR6QMT
/z50zsRMmJwzftUtOKkiXRon3t8wZoOQY8VWp3zCIS8QxUR0ssOA7qGVPO4txS7z
Wp1eG6Bp05GYES3dLeCcDhWfRnBlV+h3xf4zYAewfsJGhN8hT4UcRyuMxkN4zJl8
8GrLFVjKsz9ZZ0axrV6MEPcCgYB/OcKPdQh8WhZlMAlq5zvzKXKVN+xMwn36YMcN
OTUbFlnmLNlUMnyE8ECf+6hbR1+4dXB1vf+qWg0K8Av3B5UdB5/HtFP7xh0kkb+l
g5dnrDOlw7+h+o06u+xq+PGYu1CcKmhjJP3PG77fgoCy9RCXo8GyckMySiZzmIeA
7Qrj6QKBgQCefIv4PONqU50LMguBskNEU2HJOEn4yVW26ym8KPjwtAtR7c6dA0eH
mhNNY8rlylcZuBYPzejTxxmkZWGULAhEDDeRMnQw680DhcDX2T3Y6cXsf02l7S32
5hKOOY2ZgfKd1wZHJgh49LvPQN4xT6vOnQu8QsIy/ucgA1dyhcLCsg==
-----END RSA PRIVATE KEY-----

Screenshot of code:

```
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEA0/GRlWU0TyhSTfvb8zKchRdKSBDaH1jqceiPfTLJIwcc9sTv
WxMlmiCYOfxdBHVPC1Z9YB18ujFSR3q7p42mFx4k2kvB69C2SesHISfzSk90CnmB
bcupRJkYlumpy1NI/iOjLkXuttL7K7HNipNjHrv41ffk/kk5uXTFmtrRgldOL1er
K/n+Ma5RNmoxfiL4retmsabWwAAzPcJ/1iOCtY9CXJT0TQ2EfJ0lPWaV9KLYjD45
l3LUBKF1pEWktG93Xcmw9ognxnJGfPVhFIoMFPkQaqCaWXAeHXxtEsDmkt0BzGq+
9Q+wT8aHt5R9ySIsEBp17PyRDVDl7DM67UKLGwIDAQABAoIBAAZBJLix6d9EI1ml
yxHcNn4+97Q0uxsdtp1x7XoWW5UrCpHFsgKQCBYKTDO53Mza4WBGRyDk/d9IwVLW
rl79cR9Rhmjiv3BEEn0P8H/r++P/gD8m4sjor7AUjIM3lgkoBglp59Dfw3GlvD/H
2XbzboNjrK4zyJAwfQGaDpLJTlF4UE7TuIs4ODtDChsYxEHWuDXW/ByQePGoAmod
0HkXpTQVW9UzA159lUU77OnTRGF52PG1OnJcXf6LOWnQ4pWZgEl+sbmL3stdxC1r
EK5MDd9f15IE8VzcI5t/ICqFFVjR5GINYzS/6bM9IG9We0whHFi3OifR1x1V2aLs
BKBe2nECgYEA+zhJVU5AWSBZo8r6bzewRW6AtXqFNdx4IIxxpabQfhFFznPi3UGQ
UfiKyR2SdWwYn4s9jhoUSw5OPCtcnO6zZzrMd3LAa+VAMOKmaD6gVW4VezymQ9WJ
60NJOS1MYl1SpjkhEzyImV9A/QUmlwksQuzoedAF2yUnKawySYIlGCcCgYEA1/n2
22723x0TXwopsHUtVTE6QH9Tuo0Z7u/DZ7Lh78NzjpLPIsJLuSiUU2vIt1WuK9Nx
9OBF3hYyZyJNT/szNZjugLJtaemXzbPSzC35C7U4964K1Vmbeyxvm3SDGDWkJIR+
L85aSiVcBXkf/BzUOeevnoXMwjGhzAgwp8Uxue0CgYEAyxrZpx9JxQTzlAoR6QMT
/z50zsRMmJwzftUtOKkiXRon3t8wZoOQY8VWp3zCIS8QxUR0ssOA7qGVPO4txS7z
Wp1eG6Bp05GYES3dLeCcDhWfRnBlV+h3xf4zYAewfsJGhN8hT4UcRyuMxkN4zJl8
8GrLFVjKsz9ZZ0axrV6MEPcCgYB/OcKPdQh8WhZlMAlq5zvzKXKVN+xMwn36YMcN
OTUbFlnmLNlUMnyE8ECf+6hbR1+4dXB1vf+qWg0K8Av3B5UdB5/HtFP7xh0kkb+l
g5dnrDOIw7+h+o06u+xq+PGYu1CcKmhjJP3PG77fgoCy9RCXo8GyckMySiZzmIeA
7Qrj6QKBgQCefIv4PONqU50LMguBskNEU2HJOEn4yVW26ym8KPjwtAtR7c6dA0eH
mhNNY8rlylcZuBYPzejTxxmkZWGULAhEDDeRMnQw680DhcDX2T3Y6cXsf02l7S32
5hKOOY2ZgfKd1wZHJgh49LvPQN4xT6vOnQu8QsIy/ucgA1dyhcLCsg==
-----END RSA PRIVATE KEY-----
```

Tool generated output:

Rest of the leaks were identified as false positives like dummy files, tests file, readme and documentation files etc.

Comparison report reflecting on the similarities and differences between the results of the two tools. Include a discussion of whether the true positive secrets you found are found by both tools.

SpectralOps' secret detection tool identified a total of 176 vulnerabilities within OpenEMR, featuring an intuitive dashboard with severity labels for each vulnerability, enhancing accessibility for users of varying technical levels, from developers to non-technical personnel. The tool also offers comprehensive metric-based reporting for secret detection, with each issue linked for easy review and guidance on mitigation steps. In contrast to Gitleaks, SpectralOps includes an "information" leak label, detected MYSQL credential leaks absent in Gitleaks, and provides convenient links to the GitHub repository files, pinpointing the problematic code lines.

On the other hand, Gitleaks detected 109 secret exposures in OpenEMR, lacking a user interface like SpectralOps but allowing results exportation into a detailed JSON format, which includes information such as the description, file location, and entropy values to gauge severity. Notably, Gitleaks offers an entropy value for each exposure, aiding in severity assessment, but it has generated some false positives, particularly with AWS credentials, which SpectralOps did not encounter.

Comment on how OpenEMR handles secrets. What OpenEmr should do to protect the secrets you found? Do these secrets appear to protect valuable resources/assets? What good design principles does it seem OpenEMR has used to protect other types of secrets (hint: gitignore, config files, contributor documentation, etc)? Refer to the class sides and the resources linked from the course website.

Considering the huge scale of the project, the number of true positives found by the above tools is really small, however, cannot be considered negligible. Although some of the secrets have been removed from the latest version of master branch, they are still found in the previous commit history.

To enhance secret protection, OpenEMR can adopt cloud-native secret stores like AWS Secrets Manager or dedicated vaults, alongside leveraging secret-loading libraries such as .env. Access to configuration files should be restricted, with the removal of sensitive files from public repositories and avoidance of distribution among multiple sources. Implementing strict access controls and utilizing password managers or secure storage alternatives are vital measures, categorizing data and managing access rights based on sensitivity.

OpenEMR employs robust design principles to safeguard various types of secrets. The effective use of .gitignore files prevents accidental commits of sensitive information to version control repositories. Clear contributor documentation, including a code of conduct and guidelines on secure coding practices, ensures contributors handle secrets securely. However, secret management can be improved by use of environment variables files. In the current version of OpenEMR, environment variables are not being used effectively and efficiently. Specifically, all the hardcoded private keys can be added to environment variables and avoided to be uploaded to public repositories.

Actual Results

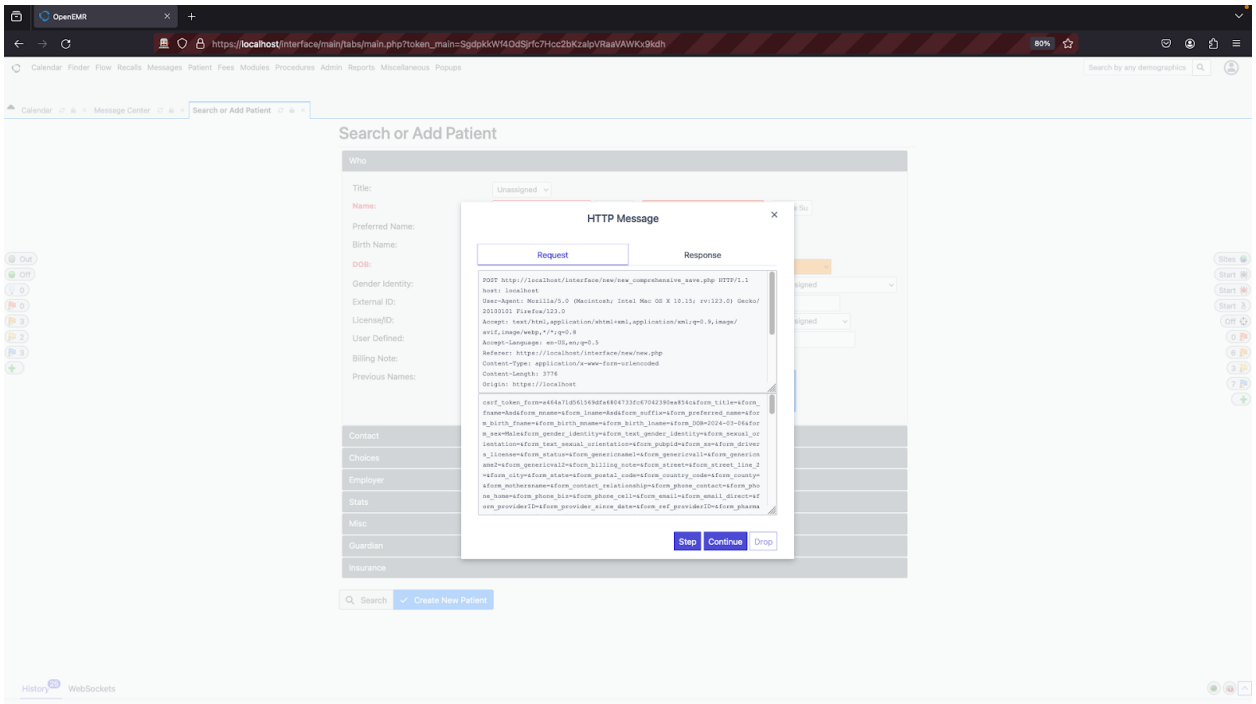
1. **Test Case 1 (True Positive):**
   Ruleset: Injection
   Actual Result: The details were submitted successfully and a patient was created with an unassigned gender
   Mitigation: Firstly, server-side validation should always be enforced, ensuring that all data submitted by the client is thoroughly validated and sanitized on the server before being processed or stored. Additionally, implementing strict access controls and authentication mechanisms can restrict users' abilities to modify sensitive data unless they have appropriate permissions.Furthermore, employing client-side integrity checks, such as checksums or digital signatures, can detect and reject unauthorized modifications made by clients.
2. **Test Case 2 (True Positive)**
   Ruleset: XSS Attack
   Actual Result: The details were submitted successfully and a patient was created with a blank first and last name
   MItigation:
   Implement strong server-side validation to enforce mandatory fields. Require both first and last names with checks that prevent submission if these fields are empty or contain only whitespace. Consider setting minimum character lengths for the names as an added layer of protection.
3. **Test Case 3 (True Positive)**
   Ruleset: Injection
   Actual Result: The details were submitted successfully and a patient was created with a date of birth as 00-00-0000 and the form accepts alphanumeric string
   Mitigation: Firstly, server-side validation should be enforced to verify the format and validity of all submitted data, including the date of birth field. This validation should ensure that the date conforms to a specific format (e.g., YYYY-MM-DD) and falls within a reasonable range. Additionally, client-side validation can be employed to provide immediate feedback to users and prevent invalid submissions before they are sent to the server. Input masks or date picker controls in the user interface can guide users to enter dates in the correct format and reduce the likelihood of alphanumeric inputs. Furthermore, enforcing strict data validation rules on the server side, such as checking for the presence of valid digits in the date fields, can help prevent invalid or nonsensical submissions.
4. **Test Case 4 (True Positive)**
   Ruleset: Buffer Overflow
   Actual Result: POST Request successfully accepts an alphanumeric string of any length
   Mitigation: Implement strict server-side validation to enforce a specific format and length limit for the payment reference number. This could involve regular expressions to ensure correct alphanumeric patterns and a defined maximum character count. Additionally, we can consider using field types specifically designed for reference numbers, if available in server-side framework. It often makes sense to include client-side validation (e.g., with HTML5 input patterns) for immediate user feedback, but it shouldn't replace the essential server-side checks.

5. **Test Case 5 (False Positive)**
   Ruleset: SQL Injection
   Actual Result: The login page does not accept any SQL injection statements
   Reasoning: The developers may have taken protective measures such as input validation (filtering out special characters), sanitization (removing dangerous code elements), prepared statements (separating code and data), or stored procedures (pre-compiled code on the database). Alternatively, attacks might have incorrect syntax, or the specific login functionality might not be vulnerable to broad SQL injections.The database user account used by the application may have limited privileges, preventing it from executing certain types of SQL commands. A WAF can be configured to block requests that contain suspicious SQL injection patterns.

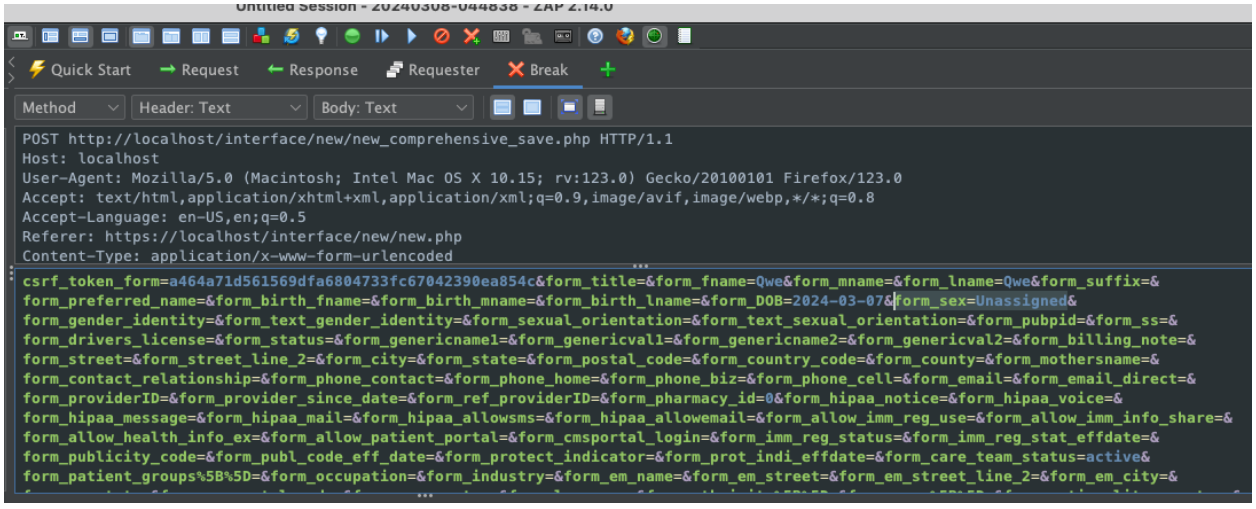
Screenshots corresponding to the actual test cases

Test Case 1

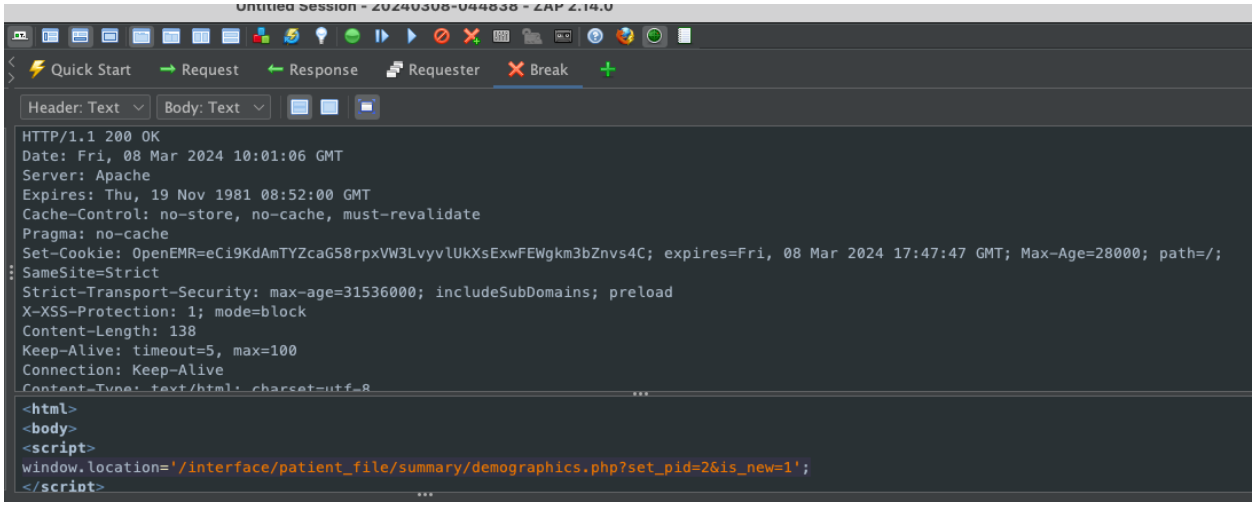
Break at URL

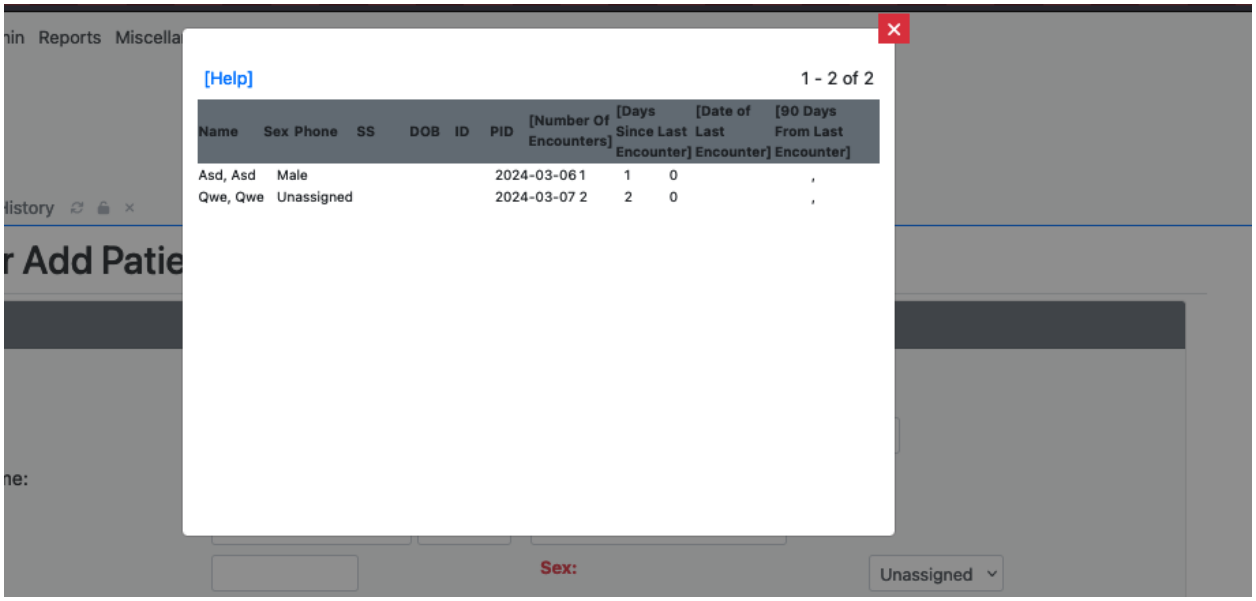


Bypass using malicious input

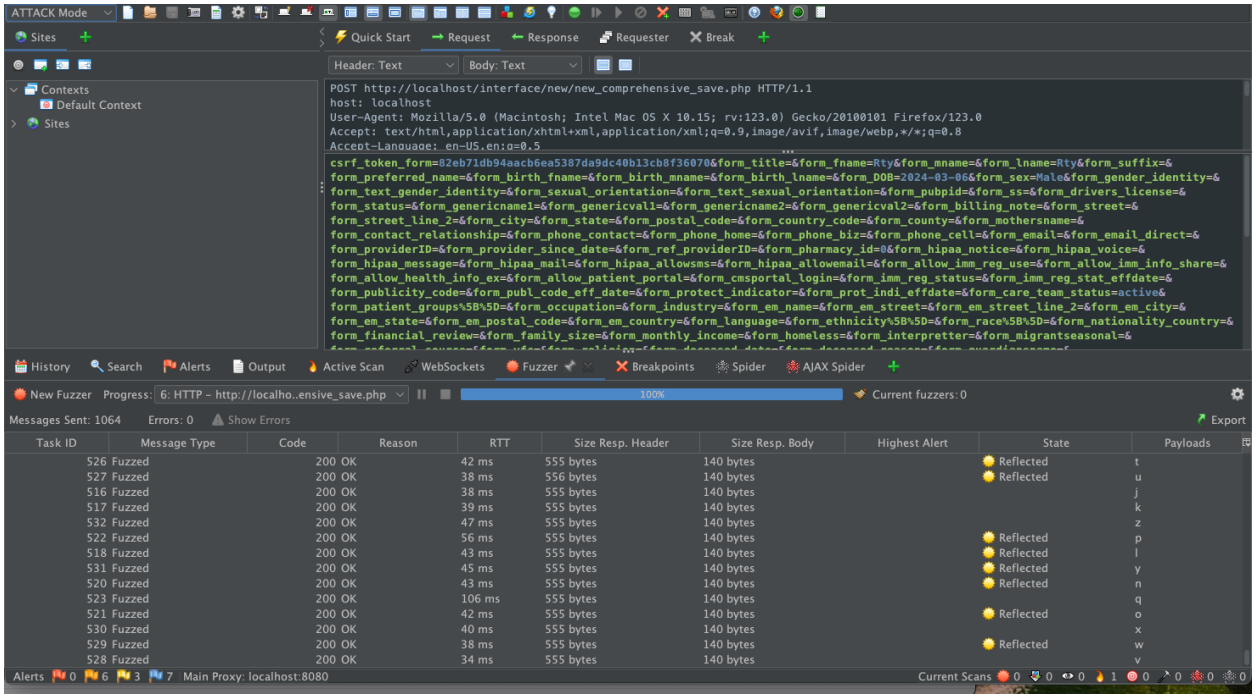


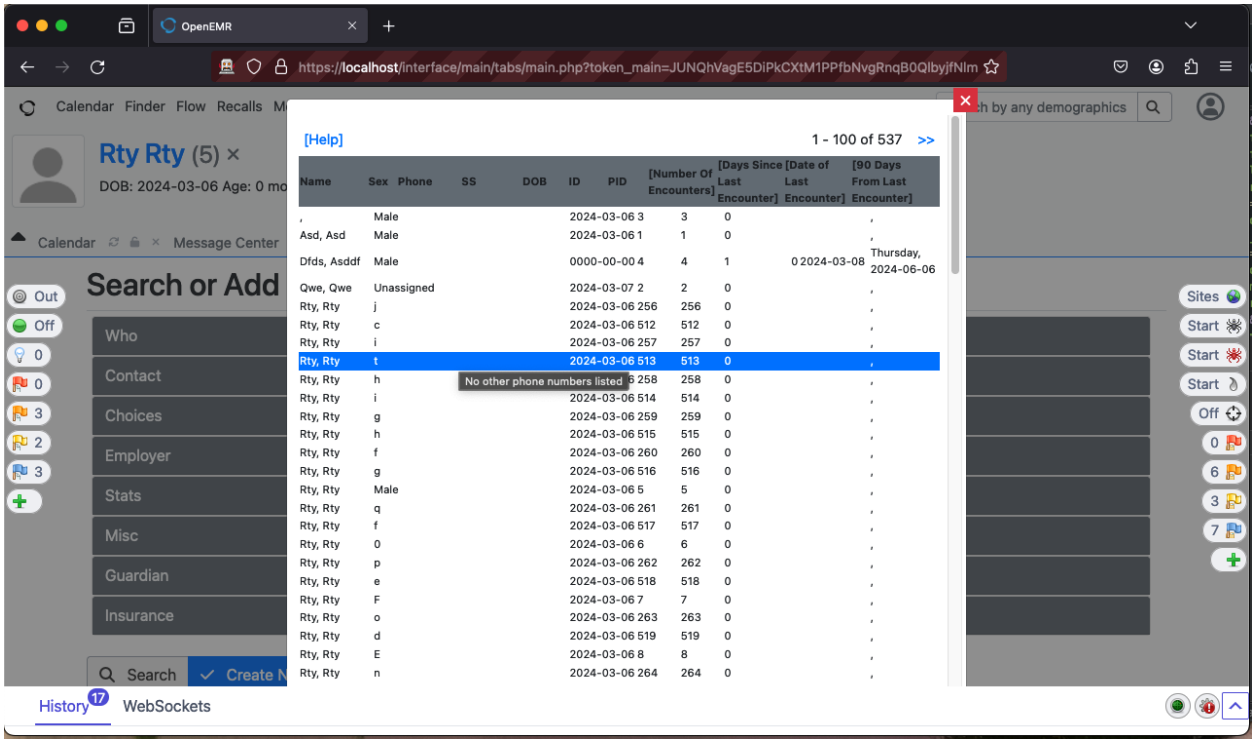
Result


Quick Start
Request
Response
Requester
Break
Header: Text
Body: Text
HTTP/1.1 200 OK
Date: Fri, 08 Mar 2024 10:01:06 GMT
Server: Apache
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Set-Cookie: OpenEMR=eCi9KdAmTYZcaG58rpxVW3LvyvlUkXsExwFEWgkm3bZnvs4C; expires=Fri, 08 Mar 2024 17:47:47 GMT; Max-Age=28000; path=/;
SameSite=Strict
Strict-Transport-Security: max-age=31536000; includeSubDomains; preload
X-XSS-Protection: 1; mode=block
Content-Length: 138
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
<html>
<body>
<script>
window.location='/interface/patient_file/summary/demographics.php?set_pid=2&is_new=1';
</script>


Reports
[Help]
1 - 2 of 2
Name Sex Phone SS DOB ID PID [Number Of Encounters] [Days Since Last Encounter] [Date of Last Encounter] [90 Days From Last Encounter]
Asd, Asd Male 2024-03-06 1 1 0 ,
Qwe, Qwe Unassigned 2024-03-07 2 2 0 ,
Sex:
Unassigned

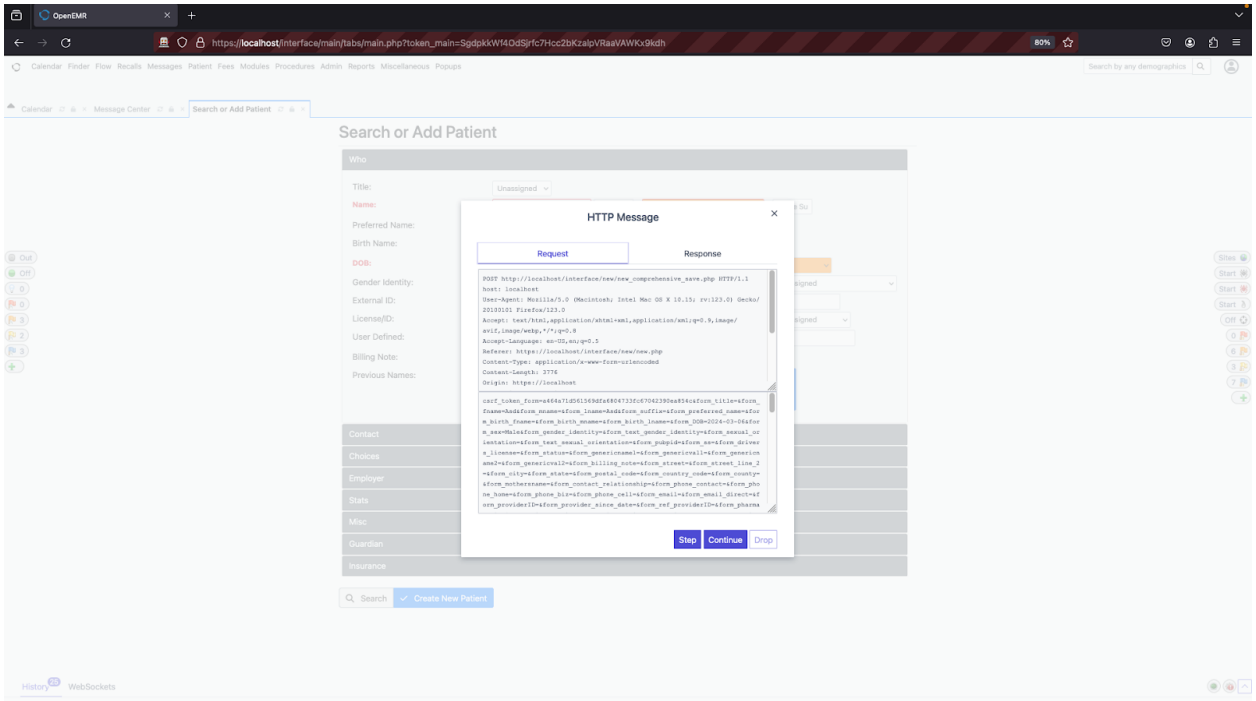
Fuzzing:



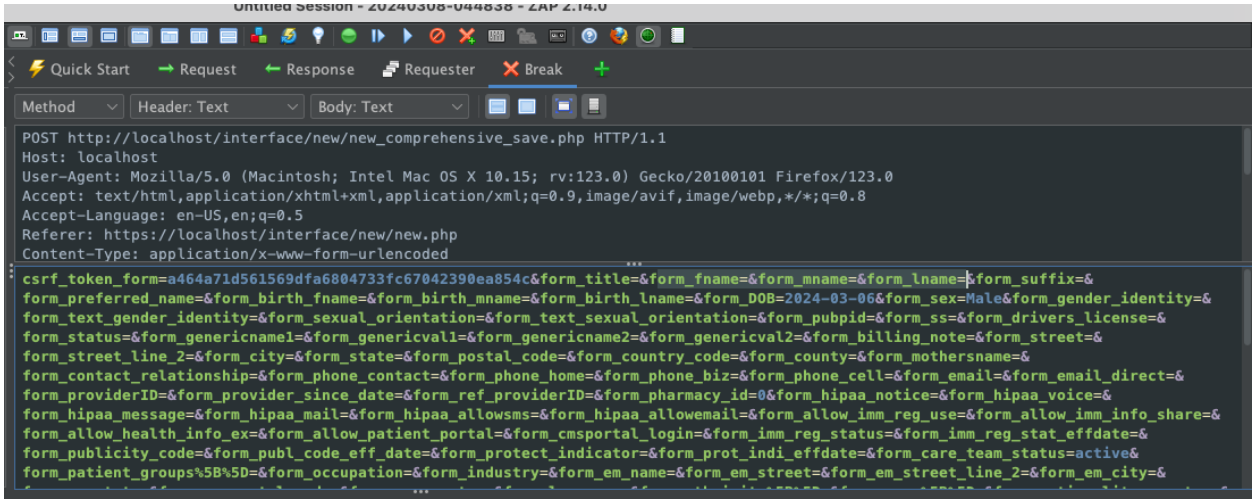


Test Case 2:

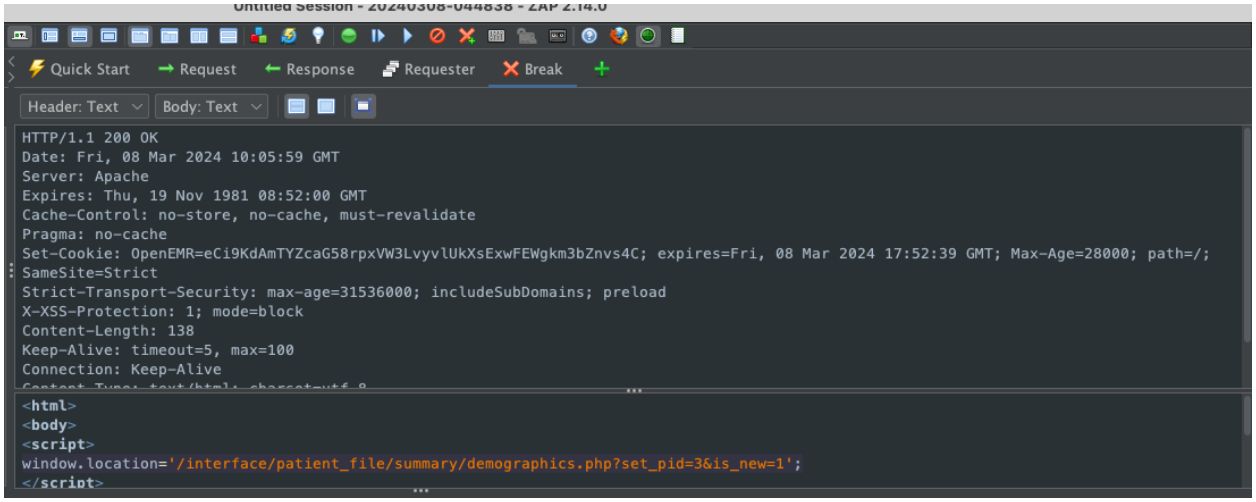
Break at the URL

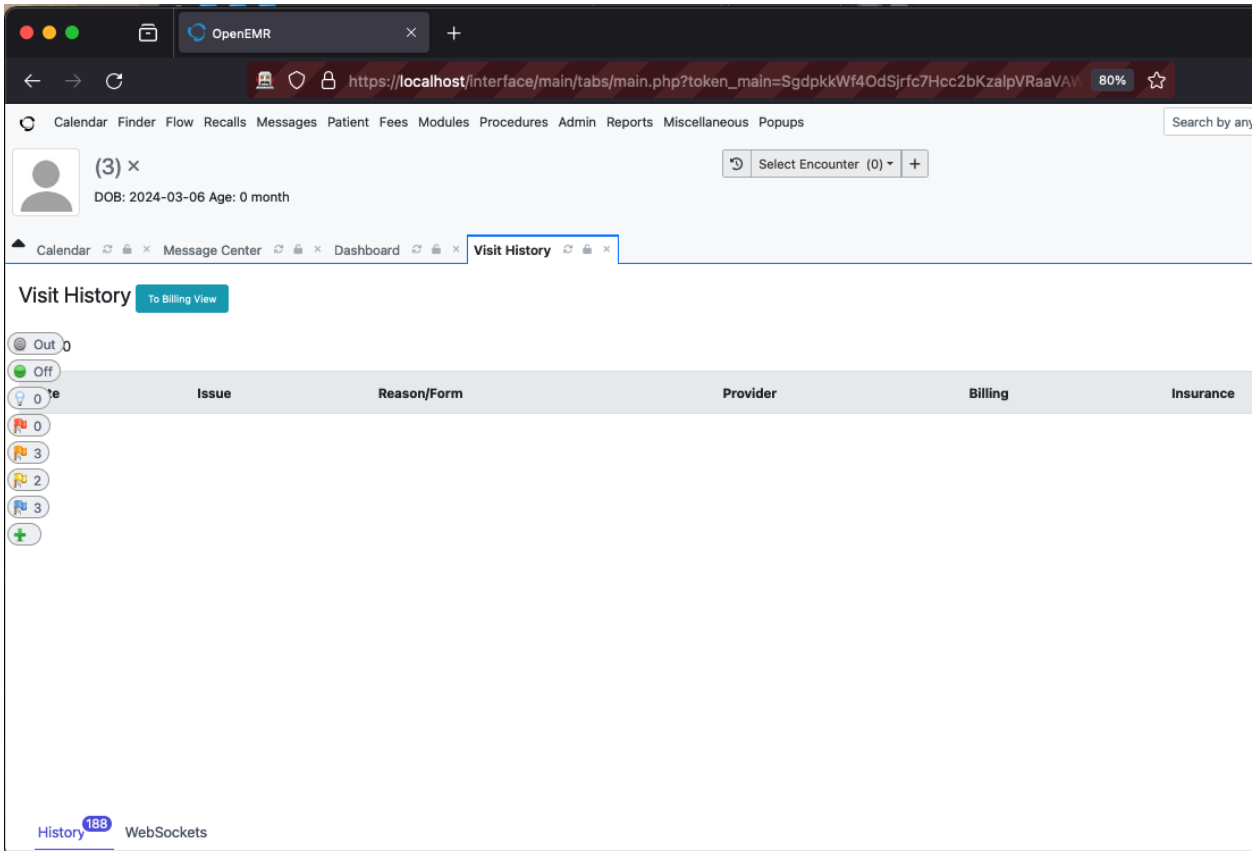


Bypass using malicious input:

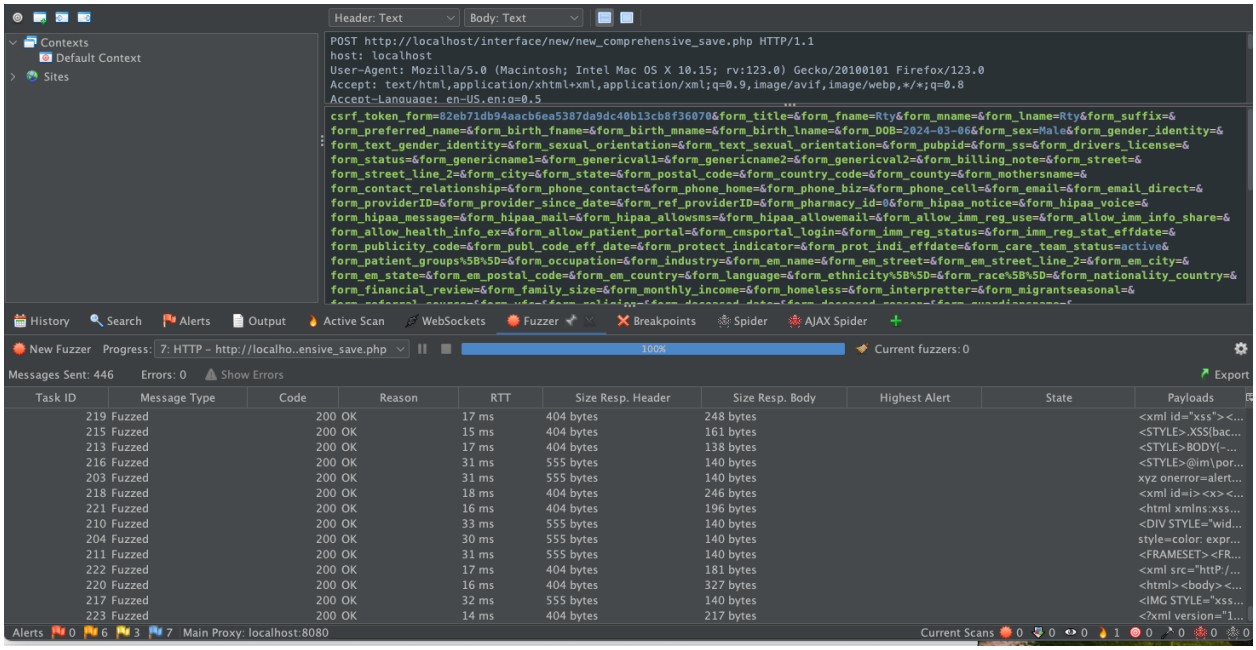


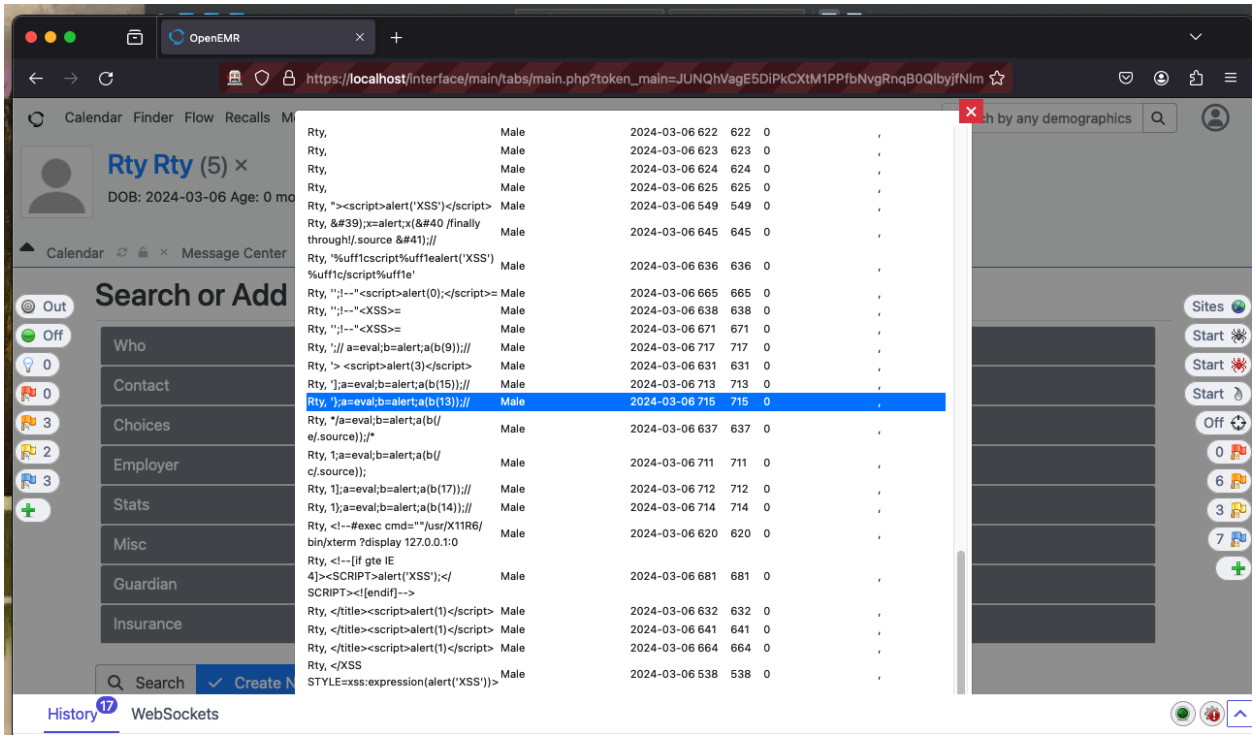
Result:


Quick Start
Request
Response
Requester
Break
Header: Text
Body: Text
HTTP/1.1 200 OK
Date: Fri, 08 Mar 2024 10:05:59 GMT
Server: Apache
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Set-Cookie: OpenEMR=eCi9KdAmTYZcaG58rpxVW3LvyvlUkXsExwFEWgkm3bZnvs4C; expires=Fri, 08 Mar 2024 17:52:39 GMT; Max-Age=28000; path=/;
SameSite=Strict
Strict-Transport-Security: max-age=31536000; includeSubDomains; preload
X-XSS-Protection: 1; mode=block
Content-Length: 138
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
<html>
<body>
<script>
window.location='/interface/patient_file/summary/demographics.php?set_pid=3&is_new=1';
</script>


OpenEMR
https://localhost/interface/main/tabs/main.php?token_main=SgdpkkWf4OdSjrfc7Hcc2bKzalpVRaaVAW
80%
Calendar
Finder
Flow
Recalls
Messages
Patient
Fees
Modules
Procedures
Admin
Reports
Miscellaneous
Popups
Search by any
(3) ×
DOB: 2024-03-06 Age: 0 month
Select Encounter (0)
Calendar
Message Center
Dashboard
Visit History
Visit History
To Billing View
Out
Off
Issue
Reason/Form
Provider
Billing
Insurance
History 188
WebSockets

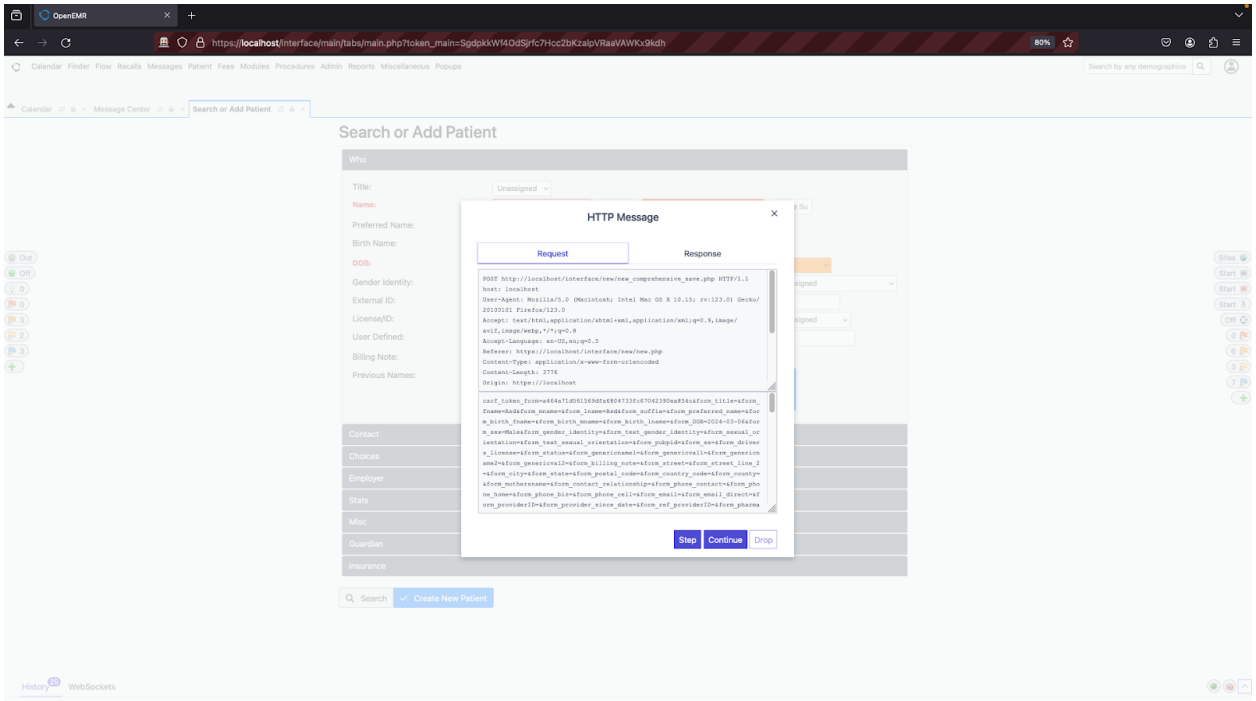
Fuzzing:



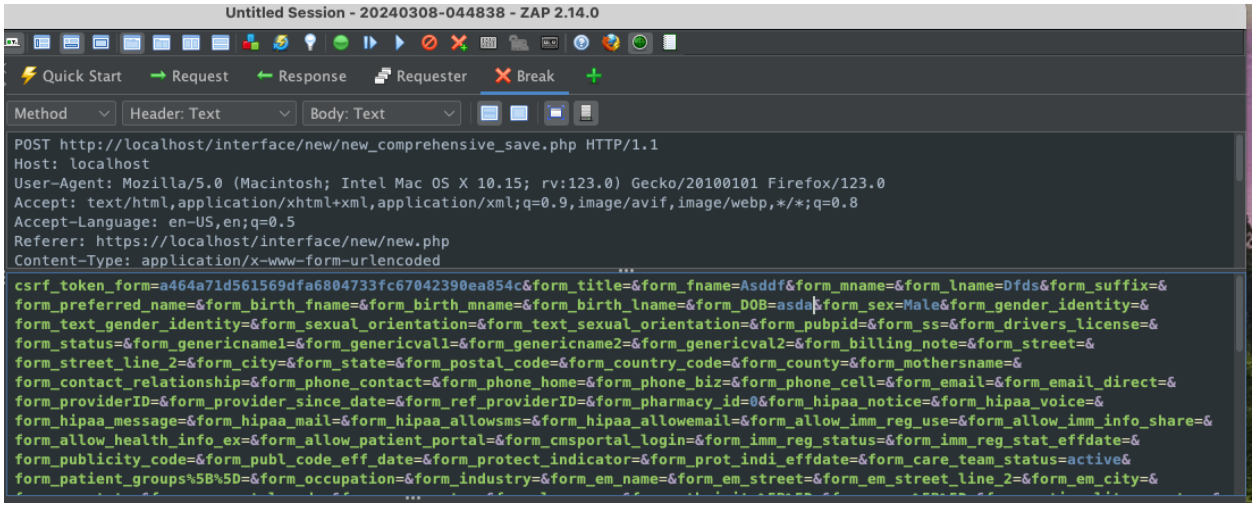


Test Case 3:

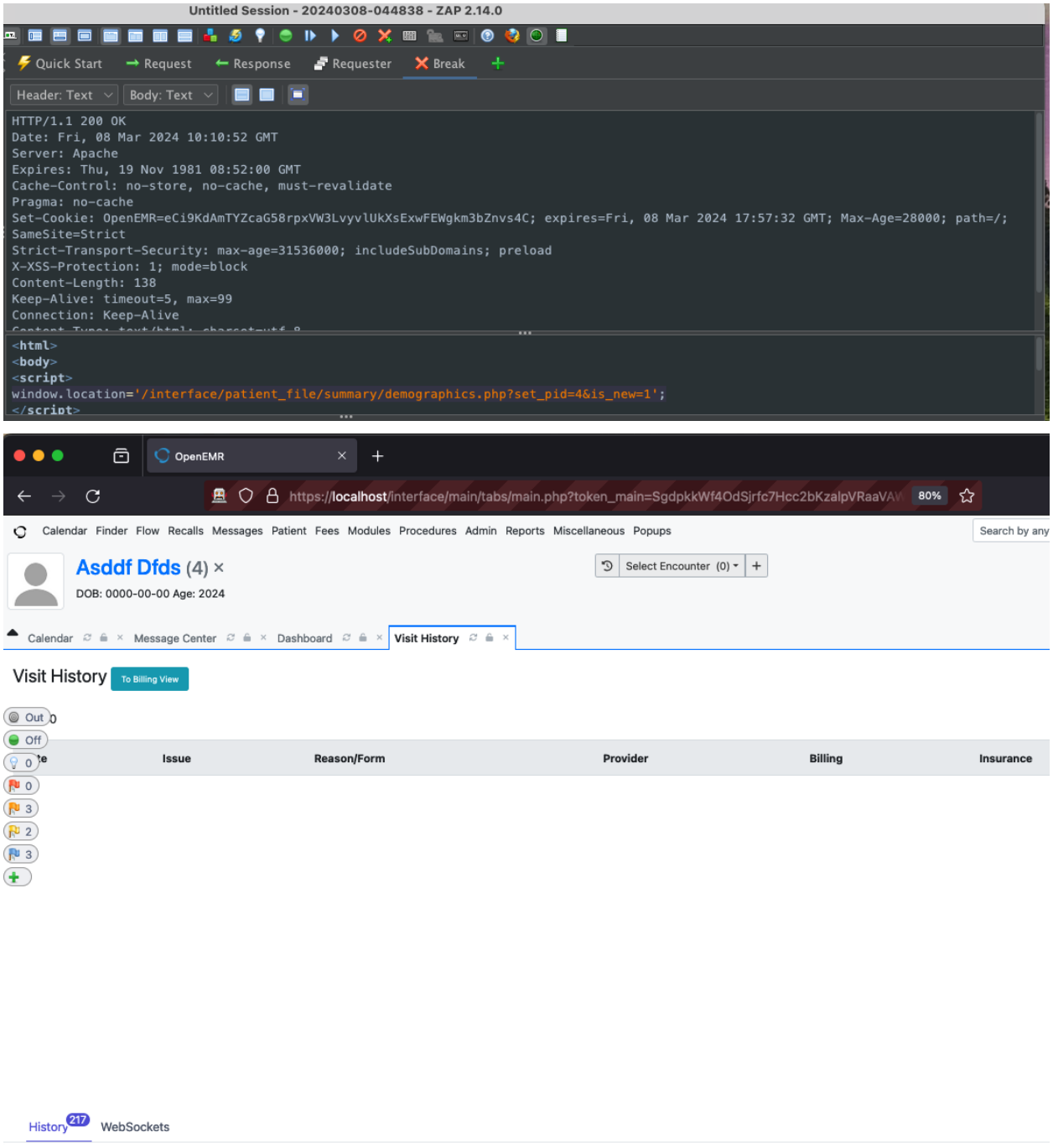
Break the URL



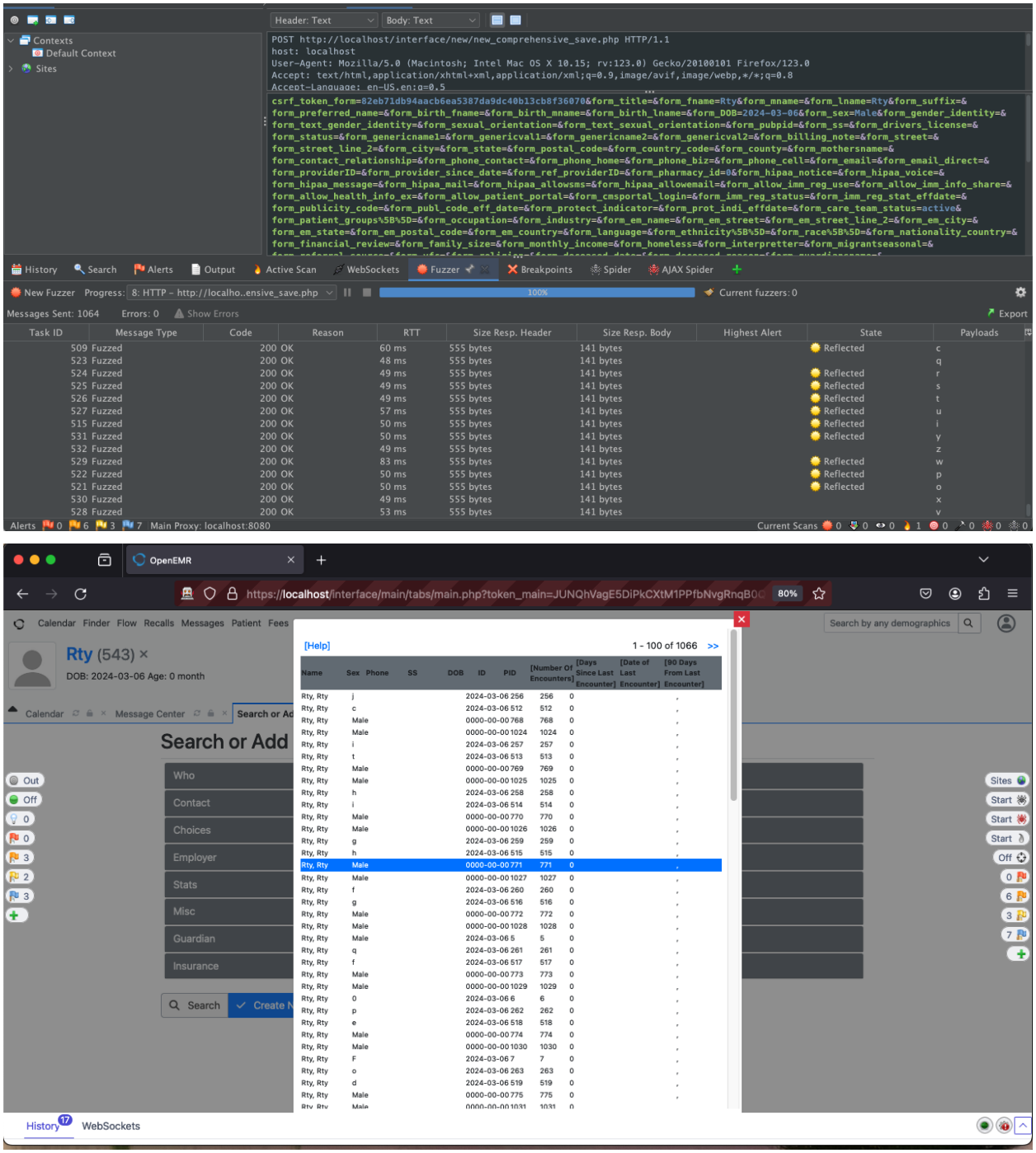
Bypass using malicious Input:



Result:


Untitled Session - 20240308-044838 - ZAP 2.14.0
Quick Start
Request
Response
Requester
Break
Header: Text
Body: Text
HTTP/1.1 200 OK
Date: Fri, 08 Mar 2024 10:10:52 GMT
Server: Apache
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Set-Cookie: OpenEMR=eCi9KdAmTYZcaG58rpxVW3LvyvlUkXsExwFEWgkm3bZnvs4C; expires=Fri, 08 Mar 2024 17:57:32 GMT; Max-Age=28000; path=/;
SameSite=Strict
Strict-Transport-Security: max-age=31536000; includeSubDomains; preload
X-XSS-Protection: 1; mode=block
Content-Length: 138
Keep-Alive: timeout=5, max=99
Connection: Keep-Alive
<html>
<body>
<script>
window.location='/interface/patient_file/summary/demographics.php?set_pid=4&is_new=1';
</script>
OpenEMR
https://localhost/interface/main/tabs/main.php?token_main=SgdpkkWf4OdSjrfc7Hcc2bKzalpVRaaVAV
80%
Calendar Finder Flow Recalls Messages Patient Fees Modules Procedures Admin Reports Miscellaneous Popups
Search by any
Asddf Dfds (4)
DOB: 0000-00-00 Age: 2024
Select Encounter (0)
Calendar
Message Center
Dashboard
Visit History
Visit History
To Billing View
Out
Off
Issue
Reason/Form
Provider
Billing
Insurance
History 217
WebSockets

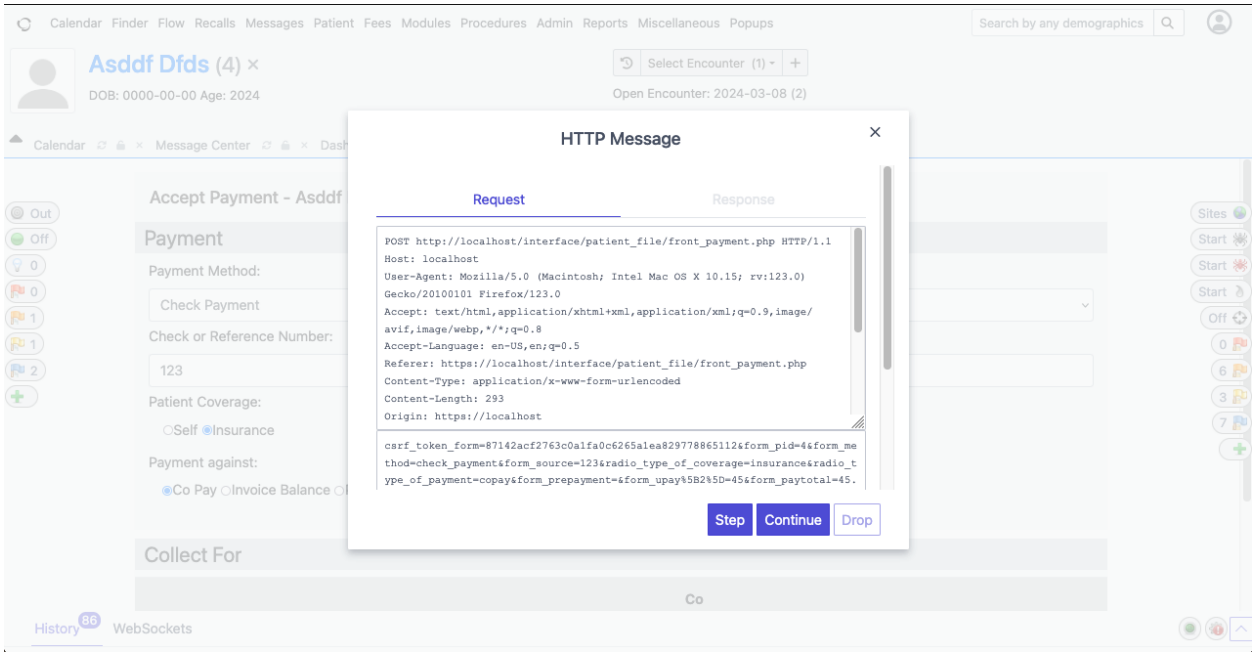
Fuzzing:



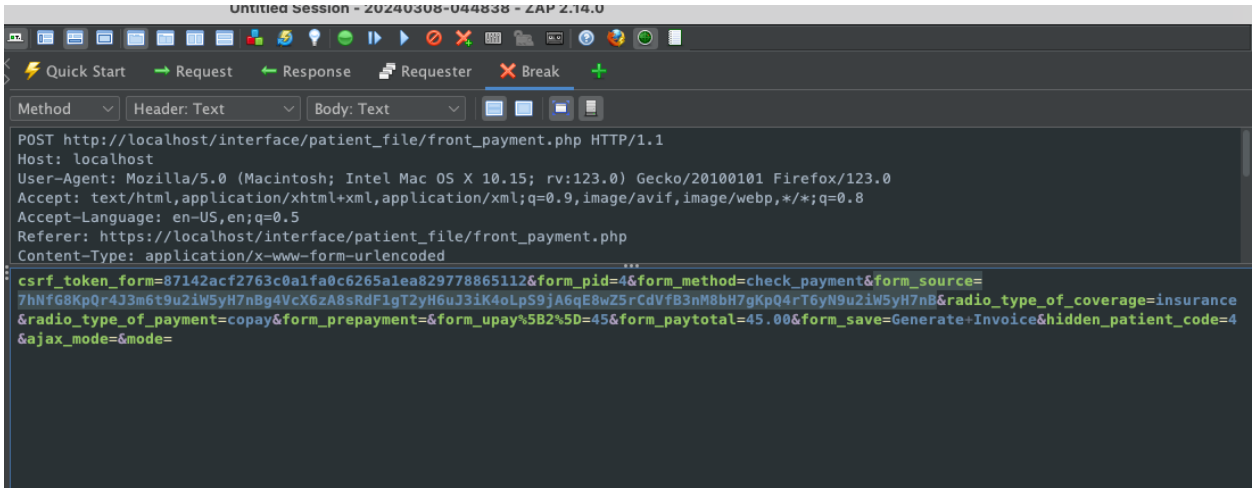
Note: This includes results from previous fuzzings as well

Test Case 4:

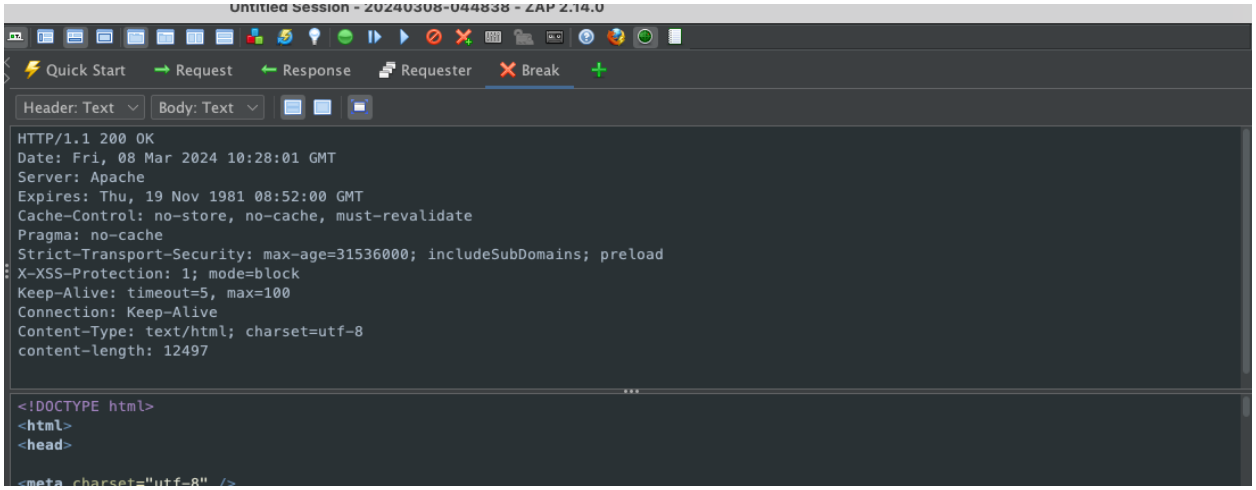
Break in the URL

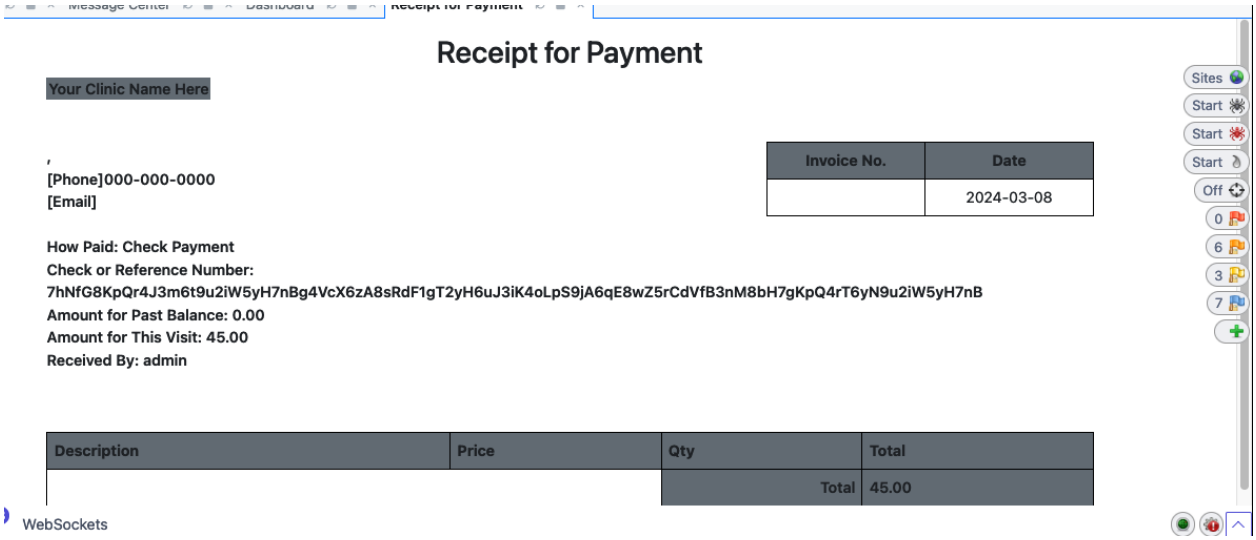


Bypass using malicious input

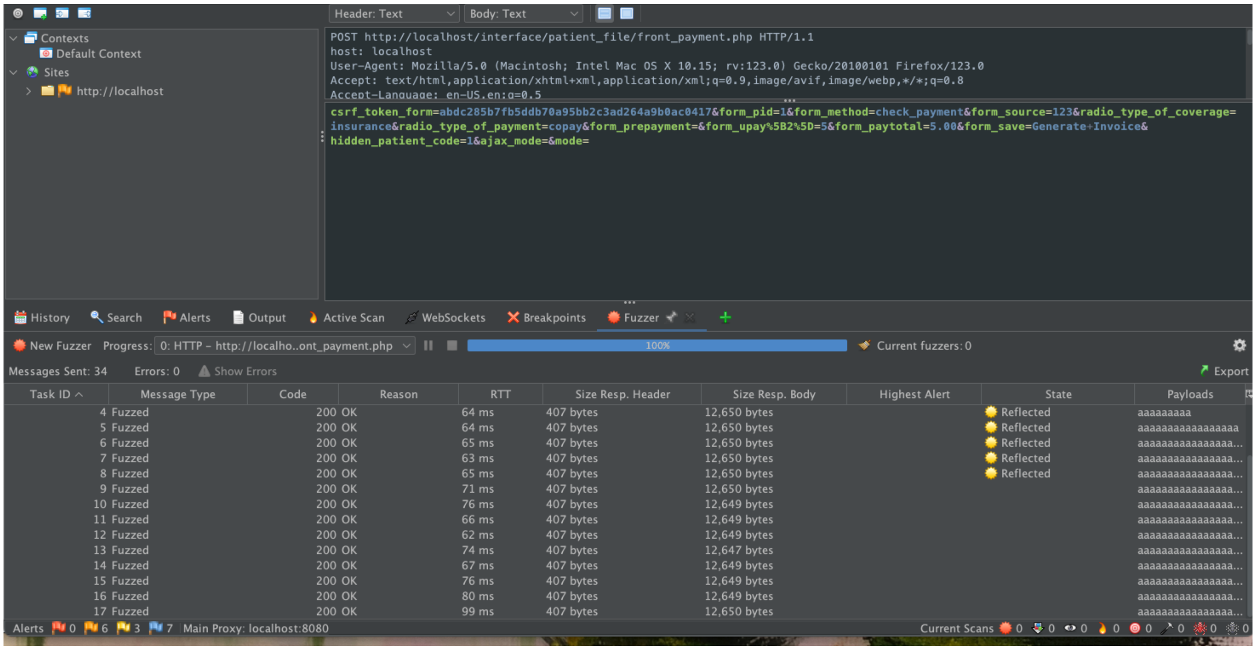


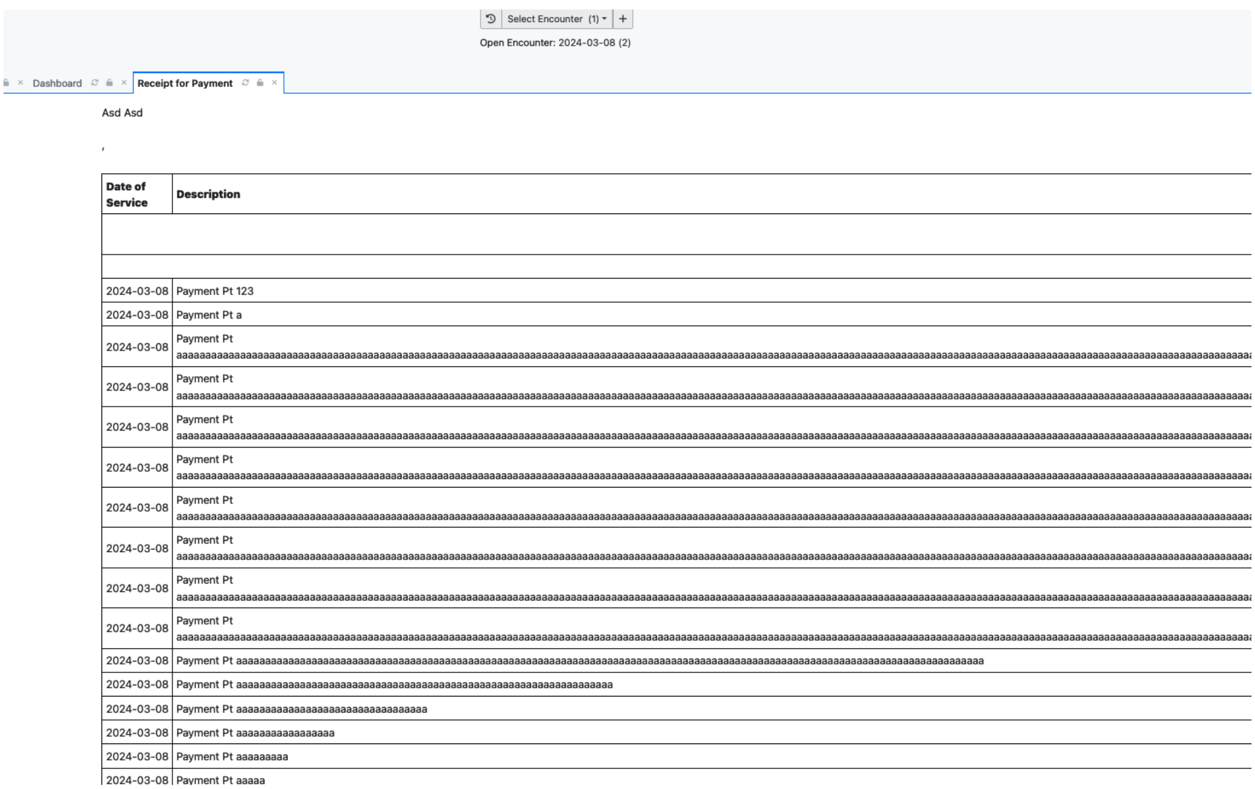
Results:


Untitled Session - 20240308-044838 - ZAP 2.14.0
Quick Start
Request
Response
Requester
Break
Header: Text
Body: Text
HTTP/1.1 200 OK
Date: Fri, 08 Mar 2024 10:28:01 GMT
Server: Apache
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Strict-Transport-Security: max-age=31536000; includeSubDomains; preload
X-XSS-Protection: 1; mode=block
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=utf-8
content-length: 12497
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8" />
WebSockets


Receipt for Payment
Your Clinic Name Here
[Phone]000-000-0000
[Email]
Invoice No.
Date
2024-03-08
How Paid: Check Payment
Check or Reference Number:
7hNfG8KpQr4J3m6t9u2iW5yH7nBg4VcX6zA8sRdF1gT2yH6uJ3iK4oLpS9jA6qE8wZ5rCdVfB3nM8bH7gKpQ4rT6yN9u2iW5yH7nB
Amount for Past Balance: 0.00
Amount for This Visit: 45.00
Received By: admin
Description
Price
Qty
Total
Total
45.00
Sites
Start
Start
Start
Off
0
6
3
7

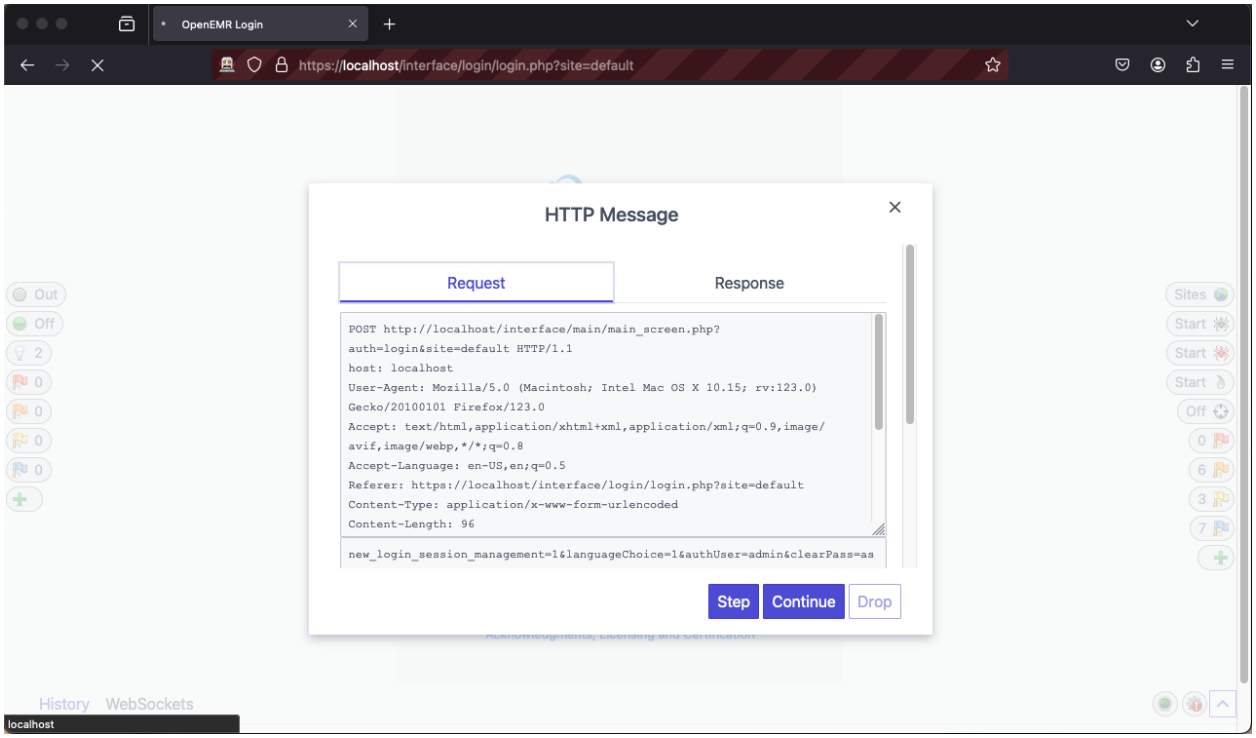
Fuzzing:



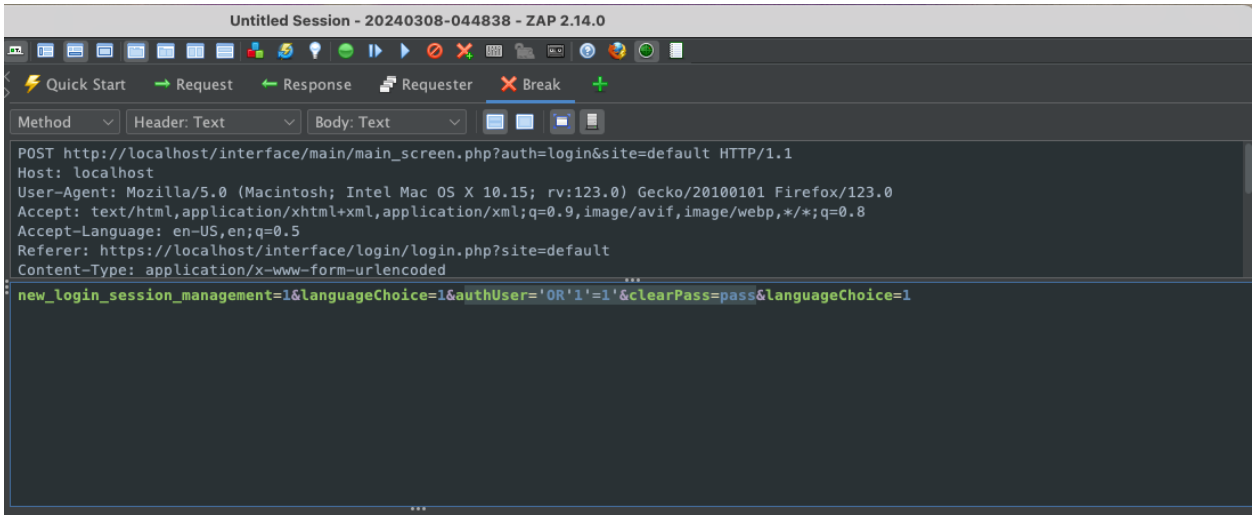


Test Case 5:

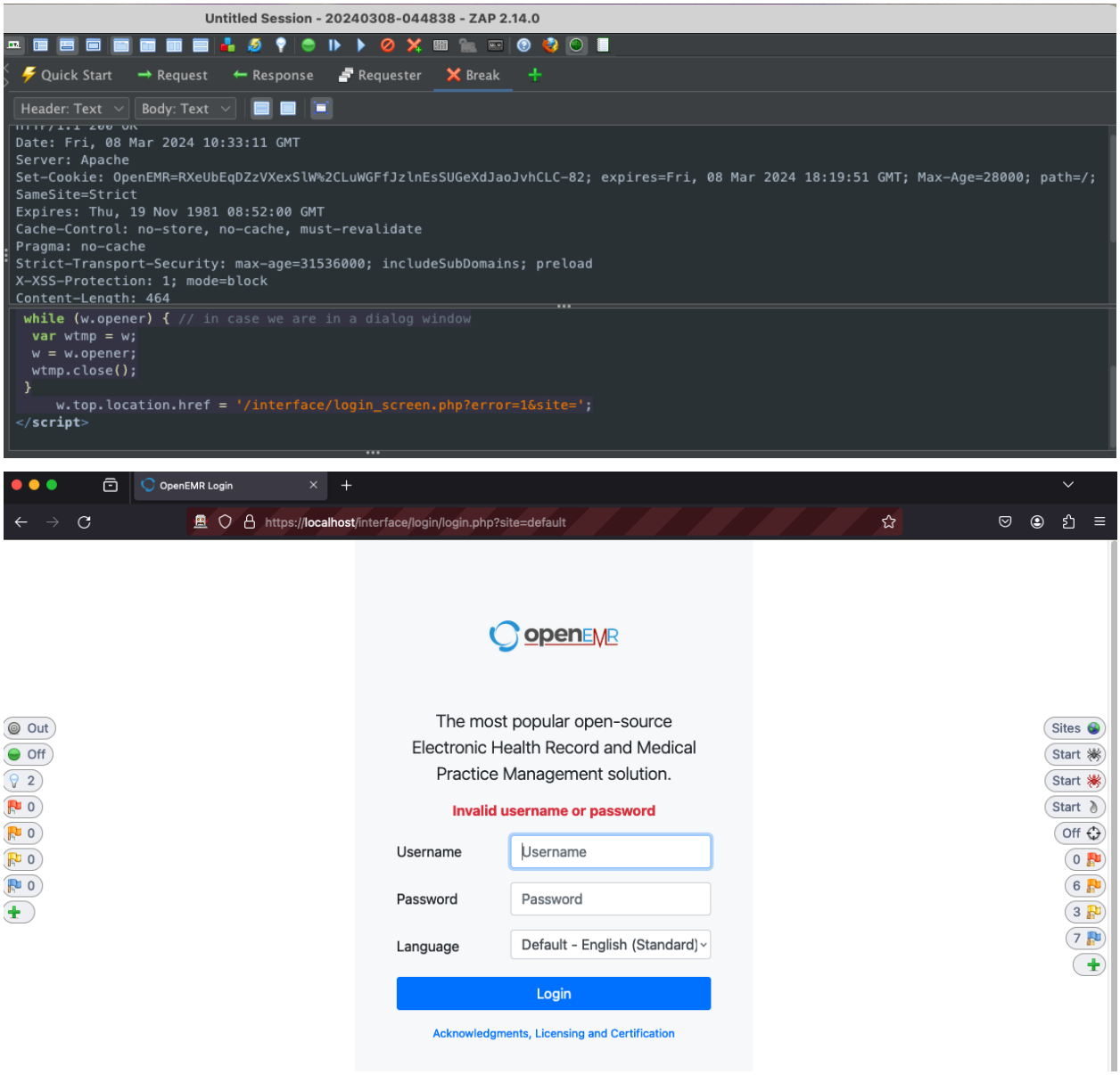
Break in the URL



Bypass using malicious input



Result:


Untitled Session - 20240308-044838 - ZAP 2.14.0
Quick Start
Request
Response
Requester
Break
Header: Text
Body: Text
Date: Fri, 08 Mar 2024 10:33:11 GMT
Server: Apache
Set-Cookie: OpenEMR=RXeUbEqDZzVXexSlW%2CLuWGFfJzlnEsSUGeXdJaoJvhCLC-82; expires=Fri, 08 Mar 2024 18:19:51 GMT; Max-Age=28000; path=/;
SameSite=Strict
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Strict-Transport-Security: max-age=31536000; includeSubDomains; preload
X-XSS-Protection: 1; mode=block
Content-Length: 464
while (w.opener) { // in case we are in a dialog window
var wtmp = w;
w = w.opener;
wtmp.close();
}
w.top.location.href = '/interface/login_screen.php?error=1&site=';
</script>

OpenEMR Login
https://localhost/interface/login/login.php?site=default
openEMR
The most popular open-source Electronic Health Record and Medical Practice Management solution.
Invalid username or password
Username
Password
Language
Default - English (Standard)
Login
Acknowledgments, Licensing and Certification
Out
Off
2
0
0
0
0
Sites
Start
Start
Start
Off
0
6
3
7

Fuzzing: