Serial number	PF-API-cZoYe2
Date	2023-05-05 10:56

www.cherokeek12.net Platform API Fuzzy Detection Report

Version 1.2

2020/12

Poc-Sir、KpLi0rn、Liucy、RachesseHS、Lupin-III

■ Copyright

■ Statements of application

This report applies to all target systems tested with this tool, so please keep it safe and do not pass it on without the permission of the target system owner.

■ Disclaimer

This report is automatically generated by the tool based on the user's test results, and the contents of the report do not represent the position and opinion of our team. Any direct or indirect consequences and losses caused by the dissemination or use of the testing functions provided by this tool are the responsibility of the user, and the team does not assume any responsibility for them. Please comply with the local laws and regulations of the user and the country where the target system is located when using this tool, any unauthorized testing is not allowed.

■ Revision history

Number	Revision date	Revisor	Modifi	Ratifier
1.0	2020/8/9	RachesseHS	None	Poc-Sir
1.2	2020/12/18	RachesseHS	Add more	Poc-Sir

I. Summary of the report

Scanned platforms：www.cherokeek12.net

Input parameter values：https://www.cherokeek12.net/

This scan uses Simple Version Scan mode，use 168.9.19.233 As scan IP，total time spent 94sec。

Initiate scanning time：2023-05-05 10:54:54

Scan completion time：2023-05-05 10:56:28

The scan found 4 valid API interfaces.

19 related JS files found.，respectively：

- https://www.cherokeek12.net//§§§
- https://www.cherokeek12.net//§§§
- https://www.cherokeek12.net//§§§
- https://www.cherokeek12.net//§§§
- https://www.cherokeek12.net//§§§
- https://www.cherokeek12.net//§§§
- https://www.cherokeek12.net/
- https://static.cloudflareinsights.com/beacon.min.js/v52afc6f149f6479b8c77fa569edb0118168

1764108816

- https://www.cherokeek12.net/uploaded/themes/fs-theme-builder/main.js?1681808466
- https://translate.google.com/translate_a/element.js?cb=googleTranslateElementInit
- https://www.cherokeek12.net/assets/in_layout_head2-b5f3f7bb27b030c8a055a13df1d1b8478510f6b58a77eae357fe92b1bc6df1b6.js
- https://platform.twitter.com/widgets.js
- https://www.cherokeek12.net/assets/application-36ff6eac57ace980b0c5eaee1ce5543a09607a801103821c056dc233de38c61b.js
- https://platform.twitter.com/tweet.b81b6d7af2d75db873cff6099e4f433a.js
- https://platform.twitter.com/moment.cd19c6b67c2f5cf62643a0c94915ac9a.js
- https://platform.twitter.com/periscope_on_air.2a64e13e06c13d3a9f08ffe7272b04e3.js
- https://platform.twitter.com/timeline.16b53cc33aaa562f8f41a495bf720289.js
- https://platform.twitter.com/dm_button.443ebd443503664187a11c2bdea4b296.js
- https://platform.twitter.com/button.e7f9415a2e000feaab02c86dd5802747.js

A total of5 security vulnerabilities were identified，Of which 0 high-risk、5 medium-risk、0 low-risk。respectively：

- Unauthorized Access Vulnerability：3
- Sensitive Information Disclosure Vulnerability：2

Additional Cookies Information：Cookies are not enabled：

Additional transmission header information：The additional head feature is not enabled：

Security risk level of the target platform as analyzed by this tool：

Medium risk

II. Vulnerability details

2.1 groupsUnauthorized access vulnerability in interface (medium risk)

API address : https://www.cherokeek12.net/fs/elements/cf_endpoints/routes.cfm/media/groups

Associated JS address : https://www.cherokeek12.net/assets/application-36ff6eac57ace980b0c5eaee1ce5543a09607a801103821c056dc233de38c61b.js

Response content：

III. List of APIs

API address：

https://www.cherokeek12.net/fs/elements/cf_endpoints/routes.cfm/media/groups

Associated JS：

https://www.cherokeek12.net/assets/application-36ff6eac57ace980b0c5eaee1ce5543a09607a801103821c056dc233de38c61b.js

Response content：

```
Bad Request
```

API address：

https://www.cherokeek12.net/fs/elements/cf_endpoints/routes.cfm/news/posts

Associated JS：

https://www.cherokeek12.net/assets/application-36ff6eac57ace980b0c5eaee1ce5543a09607a801103821c056dc233de38c61b.js

Response content：

```
Bad Request
```

API address：

https://www.cherokeek12.net/fs/elements/cf_endpoints/routes.cfm/constituents/groups

Associated JS：

https://www.cherokeek12.net/assets/application-36ff6eac57ace980b0c5eaee1ce5543a09607a801103821c056dc233de38c61b.js

Response content：

```
Bad Request
```

API address：

https://www.cherokeek12.net/album

Associated JS：

https://www.cherokeek12.net/assets/application-

36ff6eac57ace980b0c5eaee1ce5543a09607a801103821c056dc233de38c61b.js

Response content：

```
" "
```

IV. Security recommendations

4.1For unauthorized access vulnerability.

◆ For backend interfaces, ensure that all API interfaces go through the login controller first.

◆ No data is interacted with until user identity privileges are verified.

4.2For sensitive information disclosure vulnerabilities.

◆ Do not store plaintext passwords, test data, and other information in JS, and promptly troubleshoot and delete the corresponding content.

4.3Comprehensive Security Reinforcement Recommendations.

◆ Equipped with professional web application security protection equipment to deal with mainstream web application security attacks from the Internet.

◆ Periodic professional security assessments to keep abreast of the security status of information systems.

◆ Establish an effective safety emergency process and conduct regular safety training for employees.

◆ Seeking a professional security service team or using a security crowdsourcing approach such as ThorSRC (Bountyteam China).

◆ Improve the security management system system and standardize the routine maintenance and use of information systems.

V. Extra Info

当前网站资源树如下：

```
www.cherokeek12.net
|----
        |----§§§
|----widgets.js
|----tweet.b81b6d7af2d75db873cff6099e4f433a.js
|----dm_button.443ebd443503664187a11c2bdea4b296.js
|----button.e7f9415a2e000feaab02c86dd5802747.js
|----fs
        |----elements
                |----cf_endpoints
                        |----routes.cfm
                                |----news
                                        |----posts
                                |----constituents
                                        |----groups
                                |----media
                                        |----groups
|----beacon.min.js
        |----v52afc6f149f6479b8c77fa569edb01181681764108816
|----uploaded
        |----themes
                |----fs-theme-builder
                        |----main.js?1681808466
|----moment.cd19c6b67c2f5cf62643a0c94915ac9a.js
|----album
|----translate_a
        |----element.js?cb=googleTranslateElementInit
|----timeline.16b53cc33aaa562f8f41a495bf720289.js
|----assets
        |----in_layout_head2-
b5f3f7bb27b030c8a055a13df1d1b8478510f6b58a77eae357fe92b1bc6df1b6.js
```

```
        |----application-36ff6eac57ace980b0c5eaee1ce5543a09607a801103821c056dc233de38c61b.js
|----periscope_on_air.2a64e13e06c13d3a9f08ffe7272b04e3.js
```

VI. Appendix

■ Description of vulnerabilities

CORS vulnerability： Cross-domain resource sharing can relax the browser's homology policy, which allows different websites and different servers to communicate with each other through the browser. Suppose a user logs on to vuln.com, a website with a CORS configuration, and also accesses evil.com, a link provided by the attacker. The evil.com website makes a request to vuln.com for sensitive data, and the browser's ability to receive the information depends on the configuration of vuln.com. If vuln.com is configured with the Access-Control-Allow-Origin header and is expected, then it is allowed to receive it, otherwise the browser will not receive it due to the same origin policy.

Unauthorized Access Vulnerability: Unauthorized access to the interface, as the name implies, can directly access and operate the corresponding business logic functions without requesting authorization. This is usually caused by a flawed or unauthenticated authentication page, improper security configuration, etc.

Sensitive Information Disclosure Vulnerability: Information disclosure refers to the disclosure of sensitive information in a website page or JS file. Through this sensitive information, an attacker can further compromise the server.

Horizontal Override Vulnerability: an override vulnerability is when an application does not strictly verify the identity permissions of the current user's operations, resulting in users being able to operate functions that are beyond their administrative privileges, thus operating some behaviors that are not available to that user. Level override can result in users between the same level having access to each other's sensitive information, such as name, phone number, contact address, personal data, order history, and so on. It may also be possible to perform a line of functions, such as delete, add, modify, etc., as other users with level override privileges.

SQL Injection Vulnerability: SQL injection vulnerability arises because the web application is not written for the user to submit data to the server to verify the legitimacy of the data (type, length, legitimacy of business parameters, etc.), and there is no effective special character filtering of user input data, making the user input directly into the database execution, beyond the expected results of the original design of the SQL statement, resulting in a SQL injection vulnerability.

Weak password vulnerability: Website management and operations personnel use very easy to remember passwords or directly adopt the system's default password due to insufficient security awareness, in order to facilitate and avoid forgetting passwords. Attackers can use this vulnerability to directly enter the application system or management system, so as to tamper with and delete the system, web pages, data, illegally access the system, user data, and may even lead to the fall of the server.

Arbitrary file upload vulnerability: The application system checks the legitimacy of the file type, format and content of the file uploaded by the user at the file upload function, which allows an attacker to upload a malicious Web shell script file or a file of non-expected format such as: HTML file, SHTML file, etc. At the same time, he can use the directory jump characters or control the upload. directory, uploading files directly to the web directory or any directory, which may result in the execution of arbitrary malicious script files on a remote server to gain direct access to the application system.

■ Vulnerability levels

This report has three built-in vulnerability levels, which are: low, medium and high risk. The high-risk vulnerability types are: weak password vulnerability, arbitrary file upload vulnerability, SQLi vulnerability; the medium-risk vulnerability types are: horizontal leapfrogging vulnerability, sensitive information disclosure vulnerability, unauthorized access vulnerability; the low-risk vulnerability types are: CORS vulnerability.

The confidence level for the "low" level of detection results will automatically reduce the vulnerability of a level of harm, if it is at the lowest vulnerability level is not a downgrade. For example, a vulnerability for: SQL injection vulnerability, should be a high-risk vulnerability, but the confidence level is "low", it is automatically downgraded to the risk of vulnerability.

■ Risk level

This report has four risk levels: no risk, low risk, medium risk, and high risk, with a scoring scale of 0, 5, 10, and 18, respectively. For example, in a certain scan, one high-risk, two medium-risk and five low-risk vulnerabilities are found, the score is calculated as 1 x 6 + 2 x 2 + 5 x 1 = 15 (points), the score is greater than 10 and less than 18, so the risk level is "medium-risk".

(Conclusion of the report, with the following blanks)