

**Finding Name: REST API Swagger** **\_rails Endpoint Detected**.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Team** | **Role** | **Project** | **Quality Assurance** | **Is this a re-tested Finding?** |
| Vishnu Madhusoodanan Nair | PT | Junior Team Member | Ontrack | Nabiha Masood | Yes |

|  |
| --- |
| **Was this Finding Successful?** |
| Yes |

**Finding Description**

The server software versions used by the application are revealed by the web server. Displaying. version information of software information could allow an attacker to determine the technology and type of the application used in the server, which vulnerabilities are present in the software, particularly if an outdated software version is in use with published vulnerabilities. Using a library with missing security patches can make your website exceptionally easy to exploit. Developers can delineate exactly who is eligible to access what.

**Risk Rating**  
Impact: Significant  
Likelihood: High

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Impact values** | | | | |
| **Very Minor** | **Minor** | **Significant** | **Major** | **Severe** |
| Risk that holds little to no impact. Will not cause damage and regular activity can continue. | Risk that holds minor form of impact, but not significant enough to be of threat. Can cause some damage but not enough to impede regular activity. | Risk that holds enough impact to be somewhat of a threat. Will cause damage that can impede regular activity but will be able to run normally. | Risk that holds major impact to be of threat. Will cause damage that will impede regular activity and will not be able to run normally. | Risk that holds severe impact and is a threat. Will cause critical damage that can cease activity to be run. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Likelihood** | | | | |
| **Rare** | **Unlikely** | **Moderate** | **High** | **Certain** |
| Event may occur and/or if it did, it happens in specific circumstances. | Event could occur occasionally and/or could happen (at some point) | Event may occur and/or happens. | Event occurs at times and/or probably happens a lot. | Event is occurring now and/or happens frequently. |

**Business Impact**

Exposed REST Swagger endpoint at /assets/grape\_swagger\_rails/application-d6cdcd50fdc84eaaad8b61d0ce3c96545683669f45c2ad087d44105d9c2a968a.js. This endpoint reveals a Swagger interface, which is an Interface Description Language used to describe RESTful APIs using JSON. Swagger is often employed alongside open-source tools to design, build, document, and utilize RESTful web services. This exposure poses a significant business risk due to its potential to disclose sensitive information about the underlying APIs, services and potentially revealing internal workings, data structures, and endpoints. Such exposure can aid attackers in understanding system architecture and identifying potential weaknesses for exploitation. The software associated with this endpoint has publicly disclosed vulnerabilities.

Discovered software is stated below and it has publicly disclosed vulnerabilities.

|  |  |
| --- | --- |
| Deployments | http://localhost:3000 |

|  |  |  |
| --- | --- | --- |
| # | Path | Version found |
| 1 | /assets/grape\_swagger\_rails/application-d6cdcd50fdc84eaaad8b61d0ce3c96545683669f45c2ad087d44105d9c2a968a.js | jquery version 1.8.0 |
| handlebars version 2.0.0 |
|  |  | jQuery hashchange event, headers - 1.2 |

**Vulnerabilities for *jquery version 1.8.0***.

* This version couldn't reliably tell if the input is a selector or HTML making it vulnerable to Cross-Site Scripting. ([CVE-2012-6708](https://nvd.nist.gov/vuln/detail/CVE-2012-6708))
* Problem with the load method. This method didn't properly handle HTML tags like "<script>" if they had a space before the closing ">" character, like "</script >". Because of this, any scripts within these tags could still run, even if they shouldn't. This made it possible for attackers to sneak harmful code into a webpage and have it executed without detection. ([CVE-2020-7656](https://nvd.nist.gov/vuln/detail/CVE-2020-7656))
* May execute CORS request. ([CVE-2015-9251](https://nvd.nist.gov/vuln/detail/CVE-2015-9251))
* No longer receiving security updates.(<https://github.com/jquery/jquery.com/issues/162>)
* Issue with how it handled the jQuery.extend function which merge objects together ([CVE-2019-11358](https://nvd.nist.gov/vuln/detail/CVE-2019-11358))
* Passing HTML containing <option> elements from untrusted sources even after sanitizing it - to one of jQuery's DOM manipulation methods (i.e. .html(), .append(), and others) may execute untrusted code. ([CVE-2020-11023](https://nvd.nist.gov/vuln/detail/CVE-2020-11023))

**Vulnerabilities for handlebars version 2.0.0**

* This version is vulnerable to Arbitrary Code Execution. The lookup helper fails to properly validate templates, allowing attackers to submit templates that execute arbitrary JavaScript. This can be used to run arbitrary code on a server processing Handlebars templates or in a victim's browser effectively serving as XSS. ([CVE-2019-20920](https://nvd.nist.gov/vuln/detail/CVE-2019-20920))
* Attributes without proper quotation mark in templates can lead to XSS. ([CVE-2015-8861](https://nvd.nist.gov/vuln/detail/CVE-2015-8861))
* Prototype Pollution in handlebars. Prototype pollution is when unexpected properties or behaviors are added to all objects of a certain type in JavaScript, potentially causing security vulnerabilities or bugs ( [CVE-2021-23383](https://nvd.nist.gov/vuln/detail/CVE-2021-23383))
* Remote code execution in handlebars when compiling templates ([CVE-2021-23369](https://nvd.nist.gov/vuln/detail/CVE-2021-23369))

**Vulnerabilities for *jQuery hashchange event, headers - 1.2***

* Exploit for Cross-site Scripting in Jquery. ([C0148A2C-75C9-5375-AE2F-DBEDCCD0999F](https://vulners.com/githubexploit/C0148A2C-75C9-5375-AE2F-DBEDCCD0999F))
* jQuery vulnerable to Cross-SiteScripting(XSS) Cross-site scripting (XSS) vulnerability in jQuery before 1.6.3, when using location.hash to select elements, allows remote attackers to inject arbitrary web script or HTML via a crafted tag. ([GHSA-579V-MP3V-RRW5](https://vulners.com/github/GHSA-579V-MP3V-RRW5))
* Linksys EA7500 2.0.8.194281 - Cross-Site Scripting Vulnerability ([1337DAY-ID-36032](https://vulners.com/zdt/1337DAY-ID-36032))
* Versions prior to 1.9.0 are vulnerable to Cross-Site Scripting. The load method fails to recognize and remove "<script>" HTML tags that contain a whitespace character, i.e: "</script>", which results in the enclosed script logic to be executed. ([NODEJS:1524](https://vulners.com/nodejs/NODEJS:1524))

**Affected Assets**

The vulnerability in the jQuery hashchange event,headers, handlebars and jquery could potentially impact various components of our system. This includes but may not be limited to:

* Frontend Components: Any frontend elements utilizing jQuery functionality may be affected.
* API Endpoints: Endpoints interacting with headers or relying on jQuery events may be vulnerable.
* Backend Services: Backend services handling requests and responses with specific headers could be affected.
* User Interfaces: User interfaces relying on dynamic content loading or navigation through hash changes may experience issue.

The use of third-party JavaScript libraries can introduce a range of DOM-based vulnerabilities, including some that can be used to hijack user accounts like DOM-XSS.

**Evidence**

**Step 1**:

Login as student and create a burp scan. Go to target tab. Right click on endpoint and go to engagement tools 🡪 Discover content

A screenshot of a computer

Description automatically generated

**Step 2**

After content discovery completed. Now Do a passive scan. After some time, we can see the issues pop up on the issues tab.

A screenshot of a computer

Description automatically generated

Exposed Swagger endpoint and the vulnerable software versions are as follows.

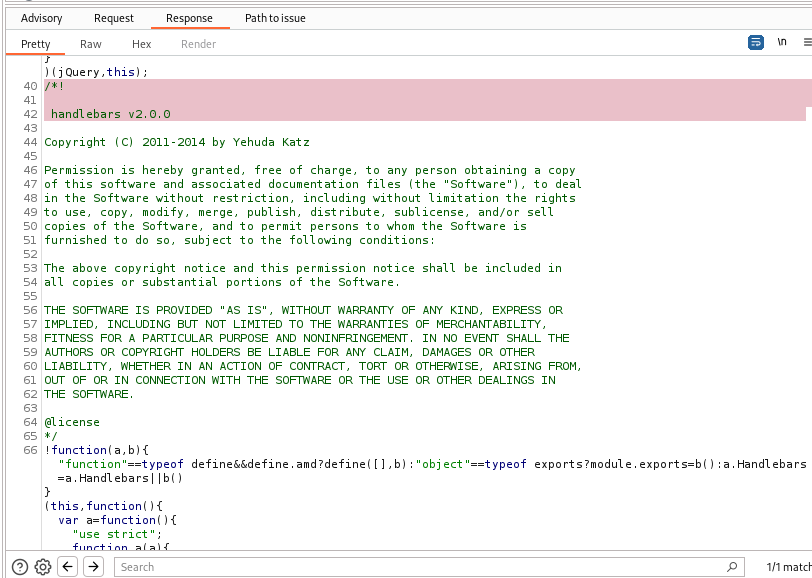
**Request :**

A screenshot of a computer

Description automatically generated

**Response :**

Screenshot shows handlebar version as displayed in burpsuite scan.



Screenshot shows hashchange event version as displayed in burpsuite scan.

A screenshot of a computer

Description automatically generated

Screenshot shows jquery version as displayed in burpsuite scan.

A screenshot of a computer

Description automatically generated

**Remediation Advice**

Develop a patch-management strategy to ensure that security updates are promptly applied to all third-party libraries in your application. Also, consider reducing your attack surface by removing any libraries that are no longer in use.

* Update jQuery Version: Ensure that you are using the latest version of jQuery. As of now, the latest version is jQuery 3.6.0. Update your jQuery dependency to this version or newer, as vulnerabilities may have been patched in these releases.
* Review Code Implementation: Conduct a thorough review of your codebase to identify any instances where the jQuery hashchange event is being used. Verify that it is implemented securely and does not expose any sensitive information.
* Sanitize Headers: If the application interacts with headers, ensure that input validation and sanitization mechanisms are in place to prevent header injection attacks. Avoid using user-controlled input directly in headers.
* Implement Content Security Policy (CSP): Consider implementing a Content Security Policy to mitigate risks associated with client-side attacks, including cross-site scripting (XSS) vulnerabilities.

**References**

* [**https://github.com/cowboy/jquery-hashchange/**](https://github.com/cowboy/jquery-hashchange/)
* <https://security.snyk.io/package/npm/jquery-hashchange>
* [**https://medium.com/@diablo0x/lab-dom-xss-in-jquery-selector-sink-using-a-hashchange-event-1a63f5ffab6a**](https://medium.com/@diablo0x/lab-dom-xss-in-jquery-selector-sink-using-a-hashchange-event-1a63f5ffab6a)

**Tools Used**

Burpsuite

**Contact Details**

Name : Vishnu Madhusoodanan Nair

Email : s223250182@deakin.edu.au

**Pentest Leader Feedback.**