

**Finding Name:** Open Redirect

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Team** | **Role** | **Project** | **Quality Assurance** | **Is this a re-tested Finding?** |
| Payas Paul | SCR | Senior Lead | Ontrack | Jaspriya k, Natalua K |  |
|  |  |  |  |  |  |

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

**Finding Description**

Unsanitized input from data from a remote resource flows into assign, where it is used as an URL to redirect the user. This may result in an Open Redirect vulnerability.

.

**Risk Rating**  
Impact: Major  
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**

The following economic consequences are possible due to an open redirect vulnerability:

1. Phishing and Social Engineering Attack: By taking advantage of the open redirect, attackers can craft phishing links that seem authentic and originate from a reliable domain. Users may be duped into submitting personal information or downloading malicious software as a result of this.   
2. Session Hijacking: If an SSO-based application has an open redirect, attackers may be able to obtain unauthorized access and carry out session hijacking attacks by stealing user tokens.   
3. Getting Around Security Filters: As seen in the attack against Office 365 subscribers, attackers can utilize the open redirect vulnerability to get around email security filters and send malicious information directly to users' inboxes.

4. Reputational Damage: If an open redirect vulnerability on a company's website is discovered, it may be exploited, which could harm the company's reputation and reduce customer trust.   
5. Financial Losses: Successful attacks made possible by open redirect vulnerabilities may result in monetary losses from things like confidential information theft, interrupted business operations, and possible legal ramifications.

**Affected Assets**

the key assets affected by an open redirect vulnerability are:

1. User Data and Accounts: Unauthorized access to user accounts and data breaches may result from attackers using open redirects to launch phishing attacks and steal user credentials, session tokens, and other sensitive information.

2. Organizational Reputation: • A company's reputation and customer trust may suffer if an open redirect vulnerability on its website is discovered.

3. corporate Operations: Effective assaults made possible by open redirect vulnerabilities have the potential to impede corporate operations, resulting in monetary damages as well as possible legal ramifications.

4. Security Posture: • Open redirection can be exploited to get over security filters and infect users with malware or other harmful content, making the organization's overall security weaker.

1. Third-Party Integrations: Open redirection can be exploited to acquire user tokens and carry out session hijacking attacks in SSO-based applications, jeopardizing the security of interconnected systems.

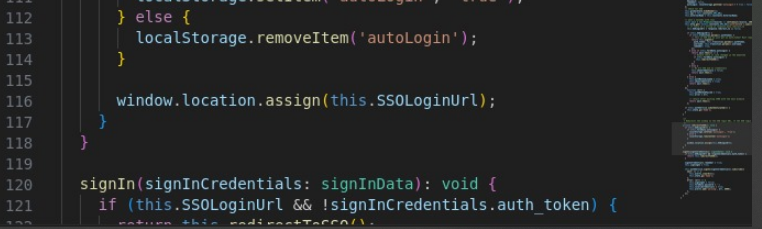
**Evidence**

**Step 1.**

In this case, the vulnerability is in the sign-in.components.ts file on line 45,

Path: src/app/sessions/states/sign-in/sign-in.component.ts, line 116

**Step 2.**

****

**Step 3.**

**Remediation Advice**

To remediate an open redirect vulnerability, follow these steps:

Verify and Sanitize Input: Make sure that all user input that is utilized to create redirect URLs has been thoroughly verified and cleaned. Verify that the input is a legitimate URL and is free of obscene characters or sequences.   
2. Use a Whitelist: Whenever possible, make use of a list of permitted redirect URLs or domains. Permit only known and reliable URLs to be redirected.   
3. Encode Output: To make sure the output URL is secure to use in a redirect, encode it. To encode special characters in the URL, use a secure encoding technique like URL encoding.   
4. Use a Safe Redirect technique: Make use of a safe redirect technique that prevents user-controlled input from being utilized as the destination and is offered by your framework or language.

5. Add Confirmation Step: If the redirect is going to be sensitive, think about adding a confirmation step beforehand. By requesting user confirmation, this can aid in preventing unwanted redirections.   
6. Security Headers: When a user clicks on a link that reroutes them, you can manage the amount of information that appears in the Referer header by using security headers like the Referrer-Policy header.   
RL.

**References**

1. **OWASP Open Redirect: The Open Web Application Security Project (OWASP) provides detailed information about open redirect vulnerabilities and how to prevent them. OWASP Open Redirect** [**https://cheatsheetseries.owasp.org/cheatsheets/Unvalidated\_Redirects\_and\_Forwards\_Cheat\_Sheet.html**](https://cheatsheetseries.owasp.org/cheatsheets/Unvalidated_Redirects_and_Forwards_Cheat_Sheet.html)
2. **CWE-601: URL Redirection to Untrusted Site ('Open Redirect'): Common Weakness Enumeration (CWE) provides a detailed description of the open redirect vulnerability and mitigation strategies. CWE-601**
3. **Microsoft Secure Coding Practices: Microsoft offers guidance on secure coding practices, including input validation and redirection security.** [**Microsoft Secure Coding Practices**](https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ff648641(v=pandp.10))

**Contact Details**

Payas Paul (Senor Lead)

[612payaspaul@gmail.com](mailto:612payaspaul@gmail.com)

**Pentest Leader Feedback.**

The lead will provide feedback to enact on.