**REDBACK OPERTAIONS**

**VULNERABILITY FOUNDED - Cross-Site Scripting (XSS)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Team** | **Role** | **Is this a re-tested Finding?** |
| Pranav Sharma | Cybersecurity team | Secure Code Review | No |

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

**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**

This vulnerability allows attackers to inject malicious scripts into web pages viewed by other users. It could lead to the theft of sensitive information such as session cookies or login credentials, potentially resulting in unauthorized access to user accounts and data.

**Location of vulnerability**

**Src/components/IdleTimerContainer.js**

**Evidence**

****

**Remediation Advice**

Implement input validation and sanitization to prevent user input from being interpreted as executable code.

Utilize frameworks or libraries with built-in protection against XSS, such as React's dangerouslySetInnerHTML cautiously and only when absolutely necessary.

Educate developers on secure coding practices to mitigate the risk of introducing similar vulnerabilities in the future.

**References**

<https://owasp.org/www-project-cheat-sheets/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet>

<https://reactjs.org/docs/security.html>

**Contact Details**

Pranav Sharma

222208296