

**Finding Name: Uncontrolled Resource Consumption**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Team** | **Role** | **Project** | **Quality Assurance** | **Is this a re-tested Finding?** |
| Payas Paul | SCR | Team Lead | Ontrack |  |  |
|  |  |  |  |  |  |

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

**Finding Description**

A medium severity Uncontrolled Resource Consumption vulnerability has been identified in **rexml@3.2.5**. This vulnerability can lead to excessive resource usage, such as CPU and memory, potentially causing performance degradation or denial of service (DoS) in applications using the affected library.

**Risk Rating**  
Impact: moderate

Likelihood: Moderate

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **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 business impact of this vulnerability can be significant:

* **Performance Degradation**: Applications using the affected library may experience slow response times or become unresponsive due to excessive resource consumption.
* **Service Disruption**: In severe cases, the vulnerability can lead to application crashes or require manual restarts, causing downtime.
* **Increased Operational Costs**: Excessive resource usage can lead to higher operational costs, especially in cloud environments where resource usage is billed.
* **Security Risks**: While primarily a performance issue, resource exhaustion can be exploited as part of a broader attack strategy, potentially leading to further security breaches.

**Affected Assets**

* **Applications**: Any application using the **rexml** library to parse XML documents is directly affected.
* **Servers and Infrastructure**: Servers hosting the affected applications can experience increased CPU, memory, and I/O usage, impacting other services on the same infrastructure.
* **Network Resources**: In some cases, excessive resource consumption can affect network bandwidth if the application's behavior leads to increased data transmission.
* **User Experience**: End users of the affected applications may experience slow response times, unresponsive interfaces, or complete outages.
* **Operational Tools**: Monitoring and logging tools may become overwhelmed by the high volume of logs and alerts generated by the resource exhaustion.

.

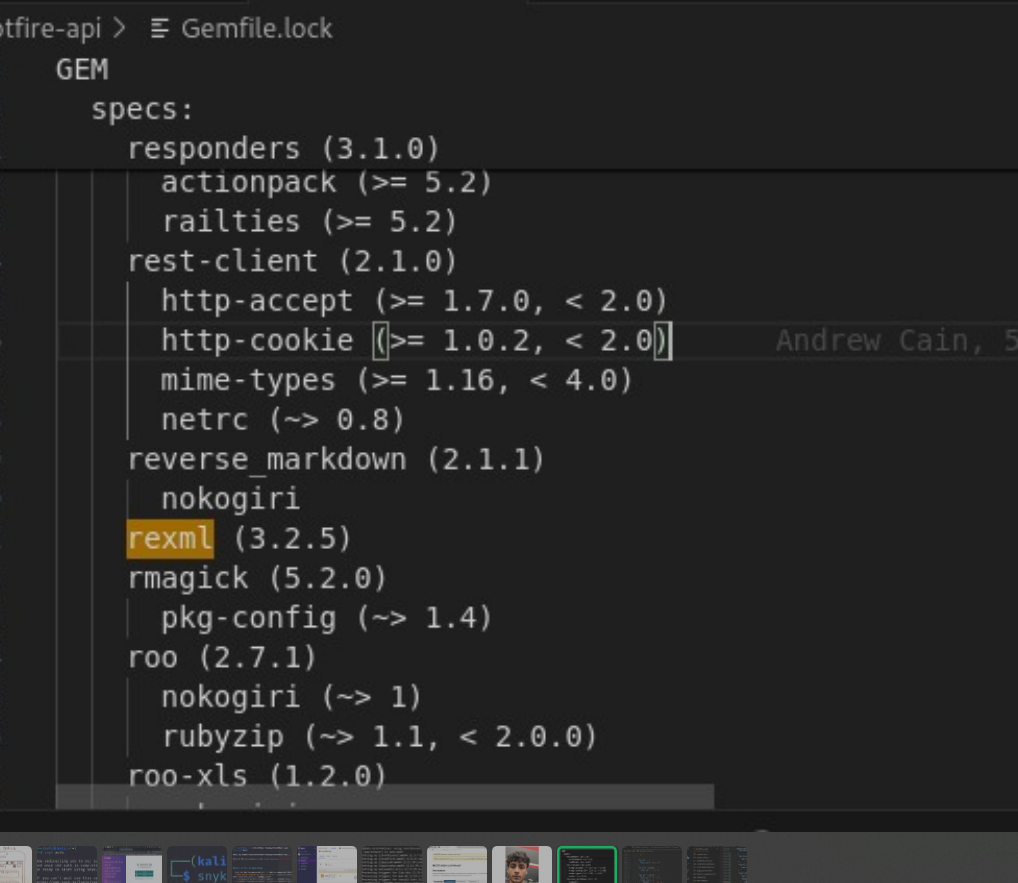
.

**Evidence**

Provide a step by step guide on how to reproduce the vulnerability with screenshots

Step 1. Go to doubtfire-deploy/doubtfire-api/Gemfile.Lock

Step 2.



Step 3.

Etc..

**Remediation Advice**

To mitigate the risk associated with this vulnerability, the following steps should be taken:

1. **Upgrade**: Update to a version of **rexml** without the vulnerability, if available.
2. **Resource Limiting**: Implement resource usage limits and quotas within the application to prevent excessive consumption.
3. **Input Validation**: Strengthen input validation to ensure that XML documents being processed are of reasonable size and complexity.
4. **Monitoring and Alerts**: Set up monitoring for resource usage and alerts for unusual patterns to respond proactively.
5. **Load Testing**: Perform load testing to identify potential bottlenecks and optimize resource usage.

**References**

Used Synk .

* OWASP Foundation. (2021). Code Injection. [online] Available at: https://owasp.org/www-community/attacks/Code\_Injection [Accessed 19 May 2024].
* Snyk. (2024). Arbitrary Code Injection (High Severity) in pdfjs-dist@2.14.305. [online] Available at: <https://security.snyk.io/vuln/SNYK-JS-PDFJSDIST-6810403> [Accessed 19 May 2024].
* Symantec Corporation. (2022). Internet Security Threat Report. [online] Available at: https://symantec.com/security-center/threat-report [Accessed 19 May 2024].

**Contact Details**

Payas Paul

2110994780

**Pentest Leader Feedback.**

The lead will provide feedback to enact on