

**Finding Name: Lack Of Error Handling.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Team** | **Role** | **Project** | **Quality Assurance** | **Is this a re-tested Finding?** |
| Shamail Haider | SCR | Senior Team Member | Ontrack | Natalia Khobotova | Yes |
|  |  |  |  |  |  |

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

**Finding Description**

Promise that might get rejected because there is no catch method for the promise. It will result in unhandled promise rejection. Leading to attacks like denial-of-service attack as unhandled promises can lead to uncaught exception, which might crash our application.

**Risk Rating**  
Impact: Minor  
Likelihood: Unlikely

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **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**

Lack of error handling can lead to security bypass, also Unhandled promise rejections might result in uncaught exceptions, potentially crashing application or service. Attackers might actively cause these unhandled rejections by delivering malicious input or exploiting vulnerabilities, rendering your application inaccessible and resulting in a DoS assault.

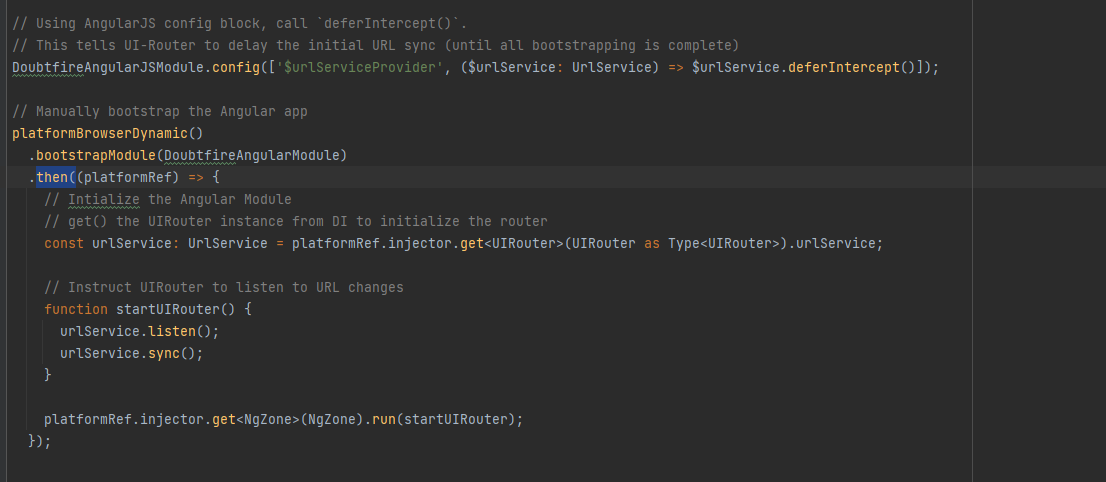
**Affected Assets**

**doubtfire-web\src\main.ts**

**doubtfire-web\src\app\sessions\service-worker-updater\check-for-update.service.ts**

**Evidence**

**Step 1.**

****

**doubtfire-web\src\main.ts**

**A screen shot of a computer program

Description automatically generated**

**doubtfire-web\src\app\sessions\service-worker-updater\check-for-update.service.ts**

**Remediation Advice**

resolve(module)

}).catch(err => {

reject(err)

})

**References**

[**https://owasp.org/www-community/Improper\_Error\_Handling**](https://owasp.org/www-community/Improper_Error_Handling)

[**https://www.veracode.com/security/error-handling-flaws-information-and-how-fix-tutorial**](https://www.veracode.com/security/error-handling-flaws-information-and-how-fix-tutorial)

**Contact Details**

**Shamail Haider**

**S222408241@deakin.edu.au**

**Pentest Leader Feedback.**

Great job!