

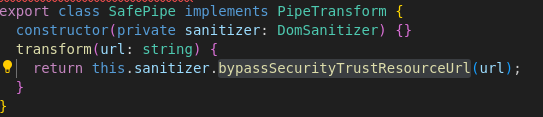
**Finding Name: Disabled Angular JS Sanitisation for Pipes**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Team** | **Role** | **Project** | **Quality Assurance** | **Is this a re-tested Finding?** |
| Pushkar Goel | SCR | SCR Team Member | Ontrack | Jaspriya Kaur and Payas Paul |  |
|  |  |  |  |  |  |

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

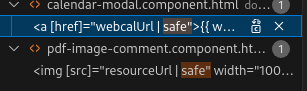
**Finding Description**

The values sent to transform function of SafePipe Class (**doubtfire-web/src/app/common/pipes/safe.pipe.ts**) has been deemed as trusted by developer and thus, they have overridden the default untrusted setting of Angular JS by using the function **bypassSecurityTrustResourceUrl.**



This method is used when special characters are expected in the input, and thus need to be accepted, which is in contrast to AngularJS setting of systematically sanitizing the value before they are inserted into the DOM. Accidentally trusting malicious data will introduce an XSS vulnerability in the application and enable a wide range of serious attacks like accessing/modifying sensitive information or impersonating other users.

This method is used for the urls in webCalendar and pdf-image-comment-component, both of which contain URL with HTTP protocol.



Thus, they will allow script tag within the URL, thus creating a XSS vulnerability

**Risk Rating**  
Impact: Major  
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 XSS vulnerability which arises due to the insecure code can have severe business implications. It allows attackers to inject malicious scripts into web applications, compromising the security and integrity of the system. The impact includes theft of sensitive data such as student data such as grades, personal details, and academic records as well as login credentials. Moreover, XSS attacks can disrupt the OnTrack operations by defacing websites, redirecting users to malicious sites, or spreading malware. Also, the stolen information can be used by attackers to conduct spear phising campaign using the leaked information

**Affected Assets**

* Data assets
* Web Application
* Server Infrastructure
* Regulatory Compliance

**Remediation Advice**

* The URL being used is created in conjunction with the resourceURL and thus it can be used to create a static template . This can then be used to validate the url passed into the function, thus making it secure from XSS attacker
* Use HTTPS protocol instead of HTTP protocol for the URL as HTTPS does not allow scripting inside the url, which can further reduce the possibility of XSS attack

**Contact Details**

Pushkar Goel – s222162584@deakin.edu.au

**Secure Code Review Leader Feedback.**

The lead will provide feedback to enact on.