

## vulnerability

Line 50 , column 50

### Target blank vulnerability

```
<a id="tribute-link"
href="https://www.biography.com/people/robin-williams-9532797" target="_blank">Robin Williams<a>
```

#### [MDN documentation:](#)

Note: When using target, consider adding rel="noopener noreferrer" to avoid exploitation of the window.opener API.

#### <https://mathiasbynens.github.io/rel-noopener/>

TL;DR If window.opener is set, a page can trigger a navigation in the opener regardless of security origin.

#### [targetblank---the-most-underestimated-vulnerability-ever](#)

People using target='\_blank' links usually have no idea about this curious fact:  
The page we're linking to gains partial access to the linking page via the window.opener object.

The newly opened tab can, say, change the window.opener.location to some phishing page. Or execute some JavaScript on the opener-page on your behalf... Users trust the page that is already opened, they won't get suspicious.

How to fix, Add this to your outgoing links.  
rel="noopener"

Update: FF does not support "noopener" so add this:  
rel="noopener noreferrer"

Remember, that every time you open a new window via window.open(); you're also "vulnerable" to this, so always reset the "opener" property  
var newWnd = window.open();  
newWnd.opener = null;

## error

Line 2, column 6:

**Bad value "https://fonts.googleapis.com/css?family=Gudea|KoHo" for attribute "href" on element "link": Illegal character in query: "|" is not allowed.**

```
<link href="https://fonts.googleapis.com/css?family=Gudea|KoHo" rel="stylesheet">
```

From [Stack Overflow](#):

The offending character here is “|” U+007C VERTICAL LINE, used by Google as a separator between font names; that’s a poor choice by them, since “|” is a reserved character, both by the “URL Living Standard” (which is what the HTML5 CR cites) and by the Internet-standard STD 66 (RFC 3986).

## CSS

---

## error

Line 66, column 11:

**Parse Error** ("<"). Maybe you can use a "Child combinator"

```
div < img {  
  align-self: center;  
}
```

[MDN documentation](#):

The child combinator (>) is placed between two CSS selectors. It matches only those elements matched by the second selector that are the children of elements matched by the first.