How to create a worker in a sandboxed iframe?

Asked 8 years, 10 months ago Modified 8 years, 10 months ago Viewed 7k times

17

I am building a sandbox for running untrusted code. For this reason I create a sandboxed iframe (which only has the `allow-scripts` permission set in its `sandbox` attribute) in order to protect the origin, and then inside that iframe I create a web-worker to ensure a separate thread and prevent freezing the main application in case the untrusted code has an infinite loop for instance.

The problem is, if I try to load the sandbox over https, recent Google Chrome does not allow to create a worker. On other browsers it works, and it also works if I load the sandbox in Chrome via http.

Here is the code:

index.html:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Sandbox test</title>
    <script type="text/javascript" src="main.js"></script>
  </head>
  <body></body>
</html>
```

main.js:

```
// determining absolute path of iframe.html
var scripts = document.getElementsByTagName('script');
var url = scripts[scripts.length-1].src
    .split('/')
    .slice(0, -1)
    .join('/')+'/iframe.html';

window.addEventListener("load", function() {
    var iframe = document.createElement('iframe');
    iframe.src = url;
    iframe.sandbox = 'allow-scripts';
    iframe.style.display = 'none';
    document.body.appendChild(iframe);

    window.addEventListener('message', function(e) {
        if (e.origin=='null' && e.source == iframe.contentWindow) {
            document.write(e.data.text);
        }
    });
}, 0);
```

iframe.html:

```
<script src="iframe.js"></script>
```

iframe.js:

```
var code = 'self.postMessage({text: "sandbox created"});';
var url = window.URL.createObjectURL(
    new Blob([code], {type: 'text/javascript'})
);

var worker = new Worker(url);

// forwarding messages to parent
worker.addEventListener('message', function(m) {
    parent.postMessage(m.data, '*');
});
```

Demo:

http://asvd.github.io/sandbox/index.html - http demo (works everywhere)

https://asvd.github.io/sandbox/index.html - https demo (doesn't work in Chrome)

https://github.com/asvd/asvd.github.io/tree/master/sandbox - source (exactly as inlined in this question)

Google Chrome then complains:

Mixed Content: The page at 'https://asvd.github.io/sandbox/iframe.html' was loaded over HTTPS, but requested an insecure Worker script 'blob:null/a9f2af00-47b1-45c1-874e-be4003523794'. This request has been blocked; the content must be served over HTTPS.

I also tried to load the worker code by https from a file instead of a blob, but this is not permitted anywhere, since I cannot access the files of the same origin from an iframe.

I am wondering if there is an opportunity to make such a sandbox work in Chrome, without adding `allow-same-origin` permission to the iframe.

javascript iframe sandbox web-worker

Share Improve this question Follow

asked May 31, 2015 at 14:53
asvd
974 ●11 ●17

1 Answer

Sorted by:
Highest score (default)

4

As you have discovered, Chrome won't let you access non-https content (such as a data blob) from a https page, and also treats blob URLs as not being https. And without `allow-same-origin`, it then can't load any worker script files from any domain.

My only suggestion is to have the iframe served from a separate https-served domain (/subdomain), and then have both `allow-scripts` and `allow-same-origin`. Due to being on a separate domain, the code in the iframe still won't be able to access the DOM/data of the parent page.

Share Improve this answer Follow

answered Jun 1, 2015 at 7:31

Michal Charemza
26.4k ● 14 ● 102 ● 170

3 That would work for a one-time solution, but I maintain a library (github.com/asvd/jailed) - therefore I'm trying to find a solution without making users have a separate domain. – asvd Jun 3, 2015 at 19:21