

The same origin concern



Secure realms towards safer composability driven web applications

Background

- Embracing composability
- Security concerns
 - Build time
 - e.g. packages, dependencies, etc
 - Runtime
 - e.g. ads, 3rd party scripts, etc

Motivation

- Focusing on runtime
- Runtime security tools
 - Include in web app
 - e.g. observe, block, limit, etc
- An important security layer

Manifestation

- Protecting a realm (ideally)

JavaScript

```
window.localStorage.getItem = function(key, secret = '') {  
    if (secret !== 'AGREED_UPON_SECRET') {  
        return null // protect access to localStorage items!  
    }  
    return localStorage[key]  
}
```

Problem (Introduction)

- The “Same Origin Concern”
 - Great protection against cross origin realms
 - Needed protection for same origin realms
 - Composability isn’t compatible with this
- Undermines runtime security tools

Problem (Demonstration)

- Protecting a realm (reality)


JavaScript

```
function getLocalStorageNaively() {  
    return window.localStorage  
}  
  
function getLocalStorageBypass() {  
    const ifr = document.createElement('iframe')  
    return document.body.appendChild(ifr).contentWindow.localStorage  
}  
  
getLocalStorageNaively().getItem('sensitive_pii') // null  
getLocalStorageBypass().getItem('sensitive_pii') // +972-5555-333
```

Problem (conclusion)

- Example easily extends
- Makes the “Same Origin Concern” matter

Solution (Present)

- JS shim
 - Snow JS 
 - Far from adequate
 - Security
 - Performance

Solution (Future)

- We're not sure
 - Provide security against same origin realms

Solution (Proposal)

- Leverage already existing APIs
- CSP - a great candidate
 - Good at enforcing rules on realms and delegating those recursively
 - adequate mechanism for enforcing security policies

Solution (Proposal)

- New directive
- Path to remote JS file
 - Same origin only (like Service Workers)
 - Forbidden by `<meta>` tag

Solution (Proposal)

```
// /scripts/realm.js
window.localStorage.getItem = function(key, secret = '') {
  if (secret !== 'AGREED_UPON_SECRET') {
    return null // protect access to localStorage items!
  }
  return localStorage[key]
}
```

```
1      <!-- CSP: "run-on-same-origin-realm /scripts/realm.js" -->
2      <html lang="">
3          <head><title></title></head>
4          <body>
5              <script>
6                  localStorage.getItem('sensitive_pii') // null
7              </script>
8              <iframe id="xyz" src="about:blank"></iframe>
9              <script>
10                 const ifr = document.getElementById('xyz');
11                 ifr.contentWindow.localStorage.getItem('sensitive_pii') // null
12             </script>
13         </body>
14     </html>
```

Sum Up