RETROCSP:
Retrofitting Universal Browser-Support for CSP

Moritz Wilhelm, Sebastian Roth, Ben Stock

CISPA Helmholtz Center for Information Security

SecWeb Workshop 2022

Content Security Policy (CSP)

CISPA HELMHOLTZ CENTER FOR INFORMATION SECURITY

```
<html>
 <body>
  <!-- ad.com includes company.com -->
  <script
      src="https://ad.com/someads.js">
  </script>
  <script>
    // ... meaningful inline script
  </script>
 </body>
</html>
```

'12

```
script-src
    https://ad.com
    https://company.com
    'unsafe-inline'
```

```
script-src
    https://company.com
    'nonce-d90e0153c074f6c3fcf53'
```

'14

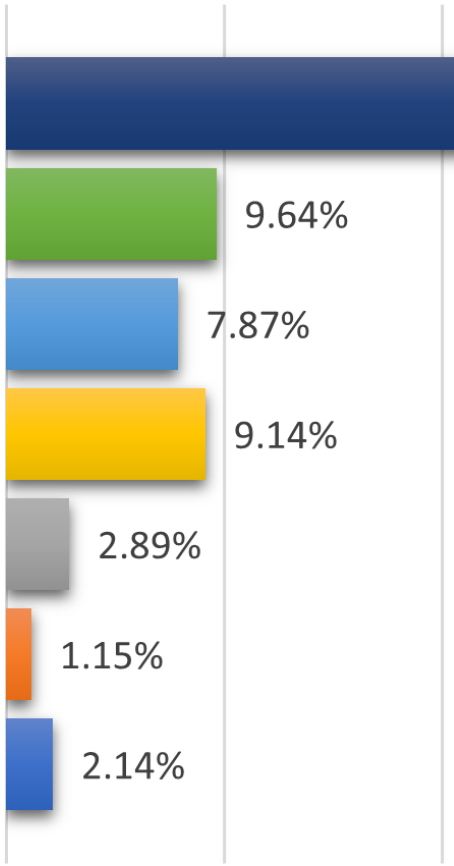
```
<html>
 <body>
  <!-- ad.com includes company.com -->
  <script nonce="d90e0153c074f6c3fcf53"
     src="https://ad.com/someads.js">
  </script>
  <script nonce="d90e0153c074f6c3fcf53">
    // ... meaningful inline script
  </script>
 </body>
</html>
```

```
<html>
 <body>
  <script nonce="d90e0153c074f6c3fcf53">
    let script =
       document.createElement("script");
    script.src = "https://ad.com/ad.js";
    document.body.appendChild(script);
  </script>
 </body>
</html>
```

'16

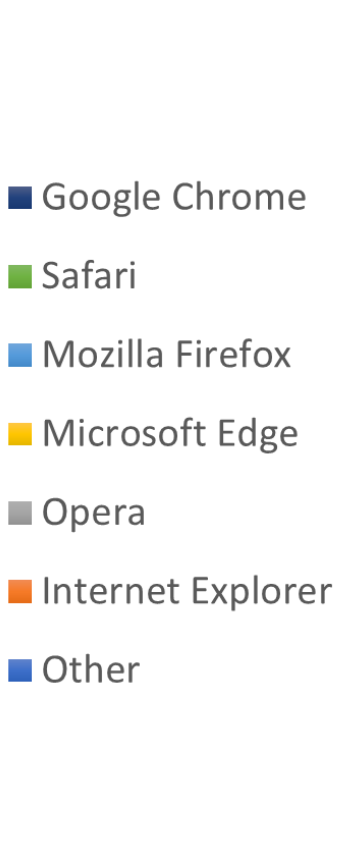
```
script-src
    'nonce-d90e0153c074f6c3fcf53'
    'strict-dynamic'
```

Current Challenges of CSP Deployment

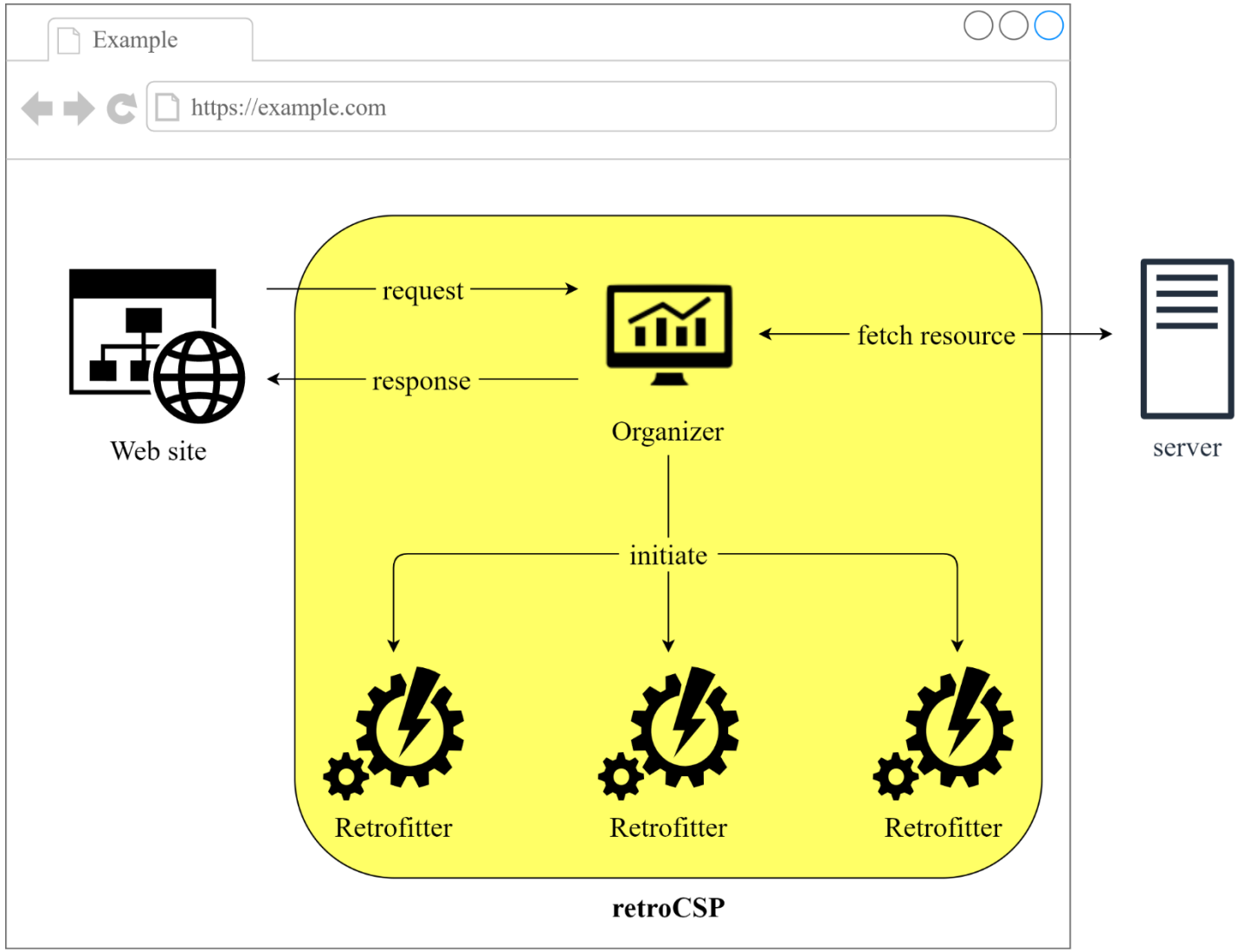


Browser Market Share Worldwide (Oct 2021)

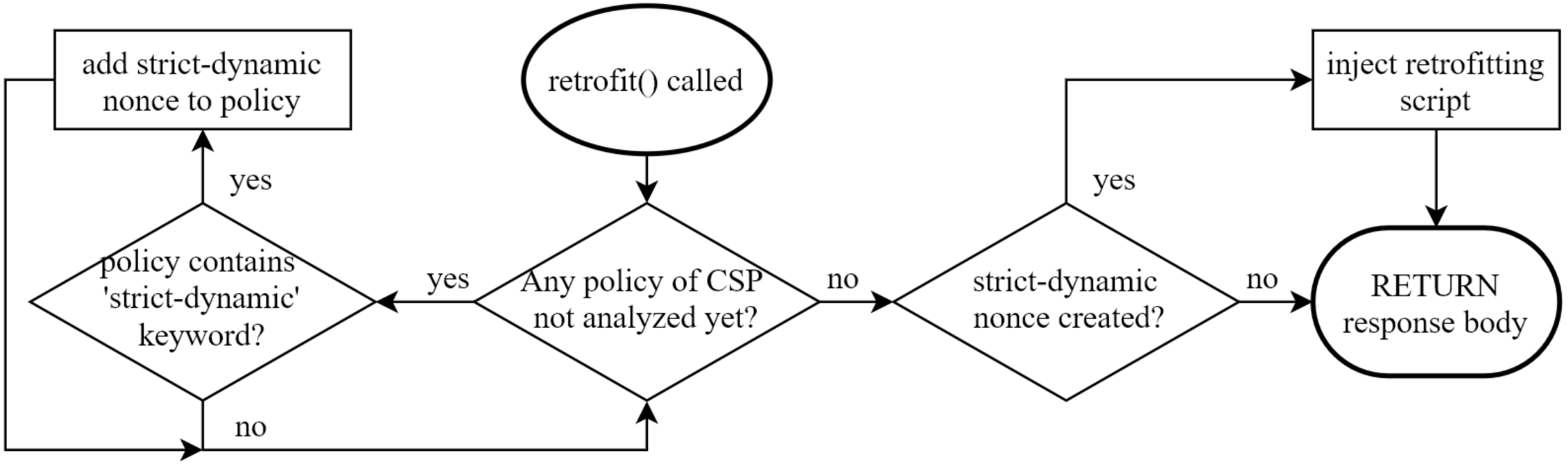
Feature	Chrome	Edge	Opera	Firefox	Safari
manifest-src	✓	✓	✓	✓	✗
plugin-types	✓	✓	✓	✗	✓
script-src-attr	✓	✓	✓	✗	✗
script-src-elem	✓	✓	✓	✗	✗
strict-dynamic	✓	✓	✓	✓	✗
unsafe-hashes	✓	✓	✓	✗	✗
style-src-attr	✓	✓	✓	✗	✗
style-src-elem	✓	✓	✓	✗	✗
navigate-to	✗	✗	✗	✗	✗
report-to	✓	✓	✗	✗	✗
prefetch-src	✗	✗	✗	✗	✗
trusted-types	✓	✓	✓	✗	✗
worker-src	✓	✓	✓	✓	✗



Retrofitting CSP Features on the Client Side


Example
https://example.com
request
response
Web site
Organizer
fetch resource
server
initiate
Retrofitter
Retrofitter
Retrofitter
retroCSP

StrictDynamicRetrofitter (Proxy Phase)



→ Establishes *strict-dynamic nonce* and deactivates any host-based allowlist

StrictDynamicRetrofitter (Client Phase)

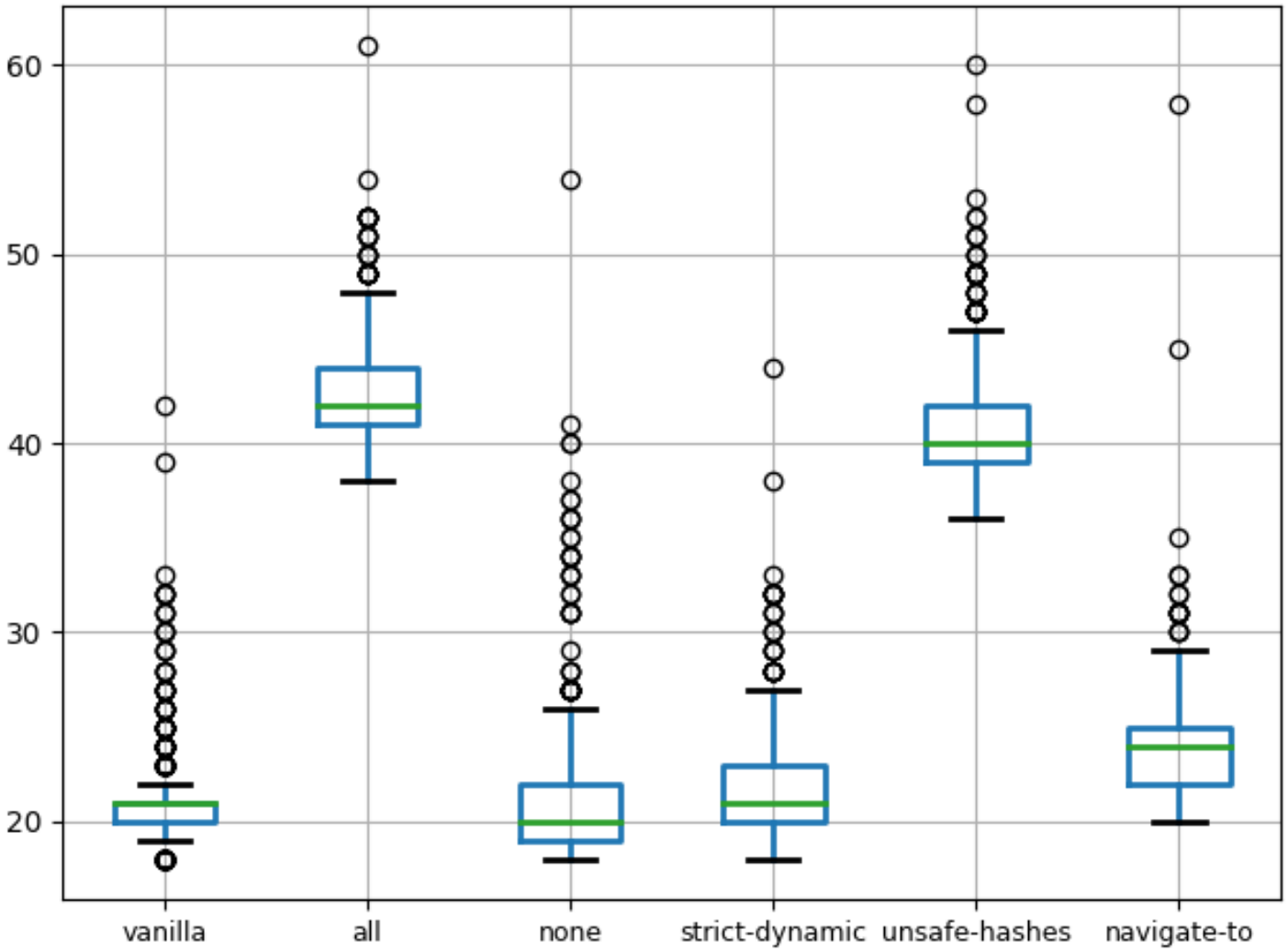
```
StrictDynamicRetrofitter.retrofittingScript = function (strictDynamicNonce) {
    let original_createElement = document.createElement;
    document.createElement = function () {
        let element = original_createElement.apply(this, arguments);
        if (element.tagName === 'SCRIPT')
            element.nonce = strictDynamicNonce;
        return element;
    };
```

- Two additional retrofitting modules:
  - UnsafeHashesRetrofitter
  - NavigateToRetrofitter

Functionality Evaluation

Browser	'strict-dynamic'
Google Chrome	✓
+ RETROCSP	✓
Microsoft Edge	✓
+ RETROCSP	✓
Opera	✓
+ RETROCSP	✓
Mozilla Firefox	(✓)
+ RETROCSP	✓
Safari	✗
+ RETROCSP	✓

Performance Evaluation



overhead equiv. to loading ~2KB from *localhost*

Limitations

- Bound to Service Worker capabilities
  - Bound to *https*
  - Requires *worker-src*

- *Window.location* object cannot be hooked
  - Prevents code execution control check in *UnsafeHashesRetrofitter* (javascript: URL)
  - Prevents navigation check in *NavigateToRetrofitter* (modifying *location.href*)

Standard Compliance & Browser Inconsistencies

Standard Violation	Chrome	Edge	Opera	Firefox	Safari	RETROCSP
'strict-dynamic' in default-src	⚡	⚡	⚡	-	(-)*	-
'strict-dynamic' in illegal directive	⚡	⚡	⚡	-	(-)*	-
CSP Level 2 & 3 in default-src	-	-	-	⚡	-	-
'unsafe-hashes' as default behavior	-	-	-	⚡	-	-
Hash value of javascript: URLs	(⚡)	(⚡)	(⚡)	(⚡)	(-)*	-
Non-Punycode-encoded host	⚡	⚡	⚡	-	⚡	-
Non-blocked window.open()	-	-	-	⚡	-	-

(*) Safari did not support the underlying functionality at all

Non-Punycode-encoded source expression

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="Content-Security-Policy" content="script-src
    ↪  https://kündigungsschutz.com/">
</head>
<body>
<script>alert(42)</script>
</body>
</html>
```

```
The value for Content Security Policy directive 'script-src' contains an invalid character: 'htt
ps://kündigungsschutz.com/'. Non-whitespace characters outside ASCII 0x21-0x7E must be percent-encoded, as described in RFC
3986, section 2.1: http://tools.ietf.org/html/rfc3986#section-2.1.
```

2. If |token| is an empty string, [=iteration/continue=].

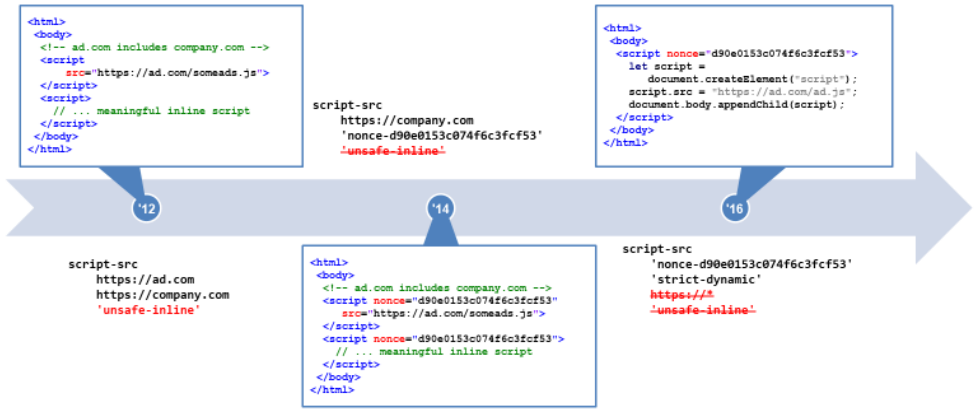
2. If |token| is an empty string, or if |token| is not an [=ASCII string=], [=iteration/continue=].

Extending the architecture

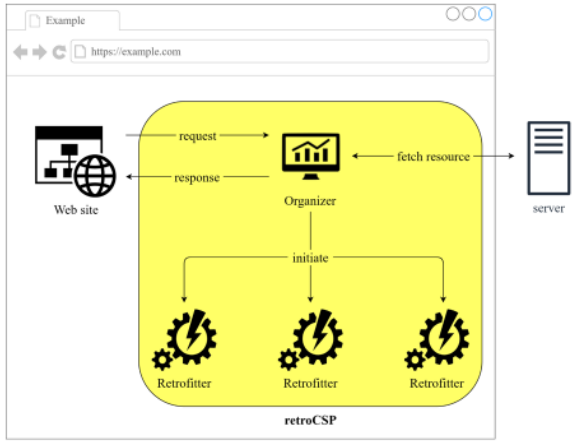
Feature	Chrome	Edge	Opera	Firefox	Safari
manifest-src	✓	✓	✓	✓	✗
plugin-types	✓	✓	✓	✗	✓
script-src-attr	✓	✓	✓	✗	✗
script-src-elem	✓	✓	✓	✗	✗
strict-dynamic	✓	✓	✓	✓	✗
unsafe-hashes	✓	✓	✓	✗	✗
style-src-attr	✓	✓	✓	✗	✗
style-src-elem	✓	✓	✓	✗	✗
navigate-to	✗	✗	✗	✗	✗
report-to	✓	✓	✗	✗	✗
prefetch-src	✗	✗	✗	✗	✗
trusted-types	✓	✓	✓	✗	✗
worker-src	✓	✓	✓	✓	✗

Conclusion

Content Security Policy (CSP)



RETROCSP



https://github.com/moritzwilhelm/retroCSP

Non-Punycode-encoded source expression

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="Content-Security-Policy" content="script-src
    ↪ https://kündigungsschutz.com/">
</head>
<body>
<script>alert(42)</script>
</body>
</html>
```

```
The value for Content Security Policy directive 'script-src' contains an invalid character: 'htt non-punycode-default-src.html:5
ps://kündigungsschutz.com/'. Non-whitespace characters outside ASCII 0x21-0x7E must be percent-encoded, as described in RFC
3986, section 2.1: http://tools.ietf.org/html/rfc3986#section-2.1.
```

```
2.  If |token| is an empty string, [=iteration/continue=].
2.  If |token| is an empty string, or if |token| is not an [=ASCII string=], [=iteration/continue=].
```


Extending the architecture

Feature	Chrome	Edge	Opera	Firefox	Safari
manifest-src	✓	✓	✓	✓	✗
plugin-types	✓	✓	✓	✗	✓
script-src-attr	✓	✓	✓	✗	✗
script-src-elem	✓	✓	✓	✗	✗
strict-dynamic	✓	✓	✓	✓	✗
unsafe-hashes	✓	✓	✓	✗	✗
style-src-attr	✓	✓	✓	✗	✗
style-src-elem	✓	✓	✓	✗	✗
navigate-to	✗	✗	✗	✗	✗
report-to	✓	✓	✗	✗	✗
prefetch-src	✗	✗	✗	✗	✗
trusted-types	✓	✓	✓	✗	✗
worker-src	✓	✓	✓	✓	✗

Adoption of CSP in the Wild

- Visited starting page of TRANCO top 10,000
- Collected CSP headers and meta-tags declaring policies
- 8,776 (87.7%) responded successfully of which 1824 (21%) presented a CSP

Category	Number of Policies	Fraction
Script Content Restriction	630	37.4%
Framing Control	881	52.3%
TLS Enforcement	575	34.1%

- 81.1% of first category trivially bypassable
- In total only 1.4% of top 1k might mitigate effects of XSS attacks using CSP

Retrofitted CSP feature	Number of Policies	Fraction
`'strict-dynamic'`	39	2.3%
`'unsafe-hashes'`	2	0.1%
`navigate-to` directive	0	0%