

# Console Clutter: A Cross-Browser Measurement of Console Messages

*Thomas Helbrecht, Jannis Rautenstrauch*

26th International Conference of Web Engineering | 12.06.2026





# The Browser Console is Web Infrastructure

- Resides in modern web browsers' developer tools
- Developers can use it for monitoring application behavior on the client

```
DevTools - www.saarbruecker-zeitung.de/
Elements Console Sources Network Performance Memory Application Security >> 4 2 6
Filter Default levels 10 Issues: 6 2 2 5 hidden
Uncaught
TypeError: Cannot read properties of undefined (reading 'split')
    at e.decode (cmp.js?v=1716880186:1:40166)
    at e.update (cmp.js?v=1716880186:1:3946)
    at cmp.js?v=1716880186:1:92321
    at cmp.js?v=1716880186:1:92352
    at cmp.js?v=1716880186:1:92356
cleared: pwp-1075074401 VM168 main.js:642
Julep Webplayer version 2023-09-04 VM168 main.js:659
cleared: pwp3891887840 main.js:642
Julep Webplayer version 2023-09-04 main.js:659
init foot.js?v=1716880186:2
Header Ad Loaded. foot.js?v=1716880186:2
Third-party cookie will be blocked. Learn more in the Issues tab. www.saarbruecker-zeitung.de/:1
GET https://static.cleverpush.com/channel/loader/SJw40XKotdID7JwP2.js?nocache=1717186618969 net::ERR_CONNECTION_REFUSED foot.js?v=1716880186:2
GET https://cdn.cxense.com/cx.cce.js net::ERR_CONNECTION_REFUSED tinypass.min.js:1
widget ID {pianoSlider: '2276d97ee20544030b11b5b587f28427832e4b24'} VM232:8
Third-party cookie will be blocked. Learn more in the Issues tab. www.saarbruecker-zeitung.de/:1
POST https://api.cxense.com/public/widget/data net::ERR_CONNECTION_REFUSED foot.js?v=1716880186:2
```

```
DevTools - www.facebook.com/?locale=en_US
Elements Console Sources Network Performance Memory Application Security >> 1 5 1 1 hidden
Filter Default levels 2 Issues: 1 1 1 1 hidden
Error with Permissions-Policy header: Origin trial controlled feature not enabled: 'interest-conort'.
Error with Permissions-Policy header: Origin trial controlled feature not enabled: 'shared-storage'.
Error with Permissions-Policy header: Origin trial controlled feature not enabled: 'shared-storage-select-url'.
Error with Permissions-Policy header: Unrecognized feature: 'usb-unrestricted'.
ErrorUtils caught an error: minKfyHUGWH.js? nc_x=Ij3Wp8lg5Kz:69
JS::call("WebCookieUseSingleLevelManageDialogController", "init", ...) did not fire because it has missing dependencies.
WebCookieUseSingleLevelManageDialogController is ready
__call__WebCookieUseSingleLevelManageDialogController.init_31 is waiting for __elem_45d73b5d_0_1_ML
__elem_45d73b5d_0_1_ML is not defined
Subsequent non-fatal errors won't be logged; see https://fburl.com/debugjs.
Stop!
This is a browser feature intended for developers. If someone told you to copy-paste something here to enable a Facebook feature or "hack" someone's account, it is a scam and will give them access to your Facebook account.
See https://www.facebook.com/selfxss for more information.
```



# Why should we care?

🔒 https://localhost:3001/test.html

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6 </head>
7
8 <body>
9   <form action="http://localhost:3000/hello" method="post">
10     <input type="text">
11     <button type="submit">Submit form</button>
12   </form>
13 </body>
14
15 </html>
```

🌐 v130.0.6723.31

<empty />

🦊 v131.0

<empty />

🦋 v18.0

**warning** <URL> line\_number: 9 column: number: 12  
The page at https://localhost:3001/test.html contains a form which targets an insecure URL http://localhost:3000/hello.

- Console might help developers write more secure code
- W3C Mixed Content Candidate Recommendation Draft states that user agents "**MAY** choose to warn users of [...] not potentially trustworthy URLs."

see <https://www.w3.org/TR/mixed-content/#requirements-forms>



# Other Console Output: Page Errors

- Uncaught JavaScript exceptions during page visits
- Stack traces identify responsible code

The screenshot shows a browser's developer console with the 'Console' tab selected. The error message is: 'Uncaught TypeError: Cannot set properties of null (setting 'innerHTML')' at [script.js:1:16](#) (anonymous) @ [script.js:1](#). A blue arrow points from a grey callout box labeled 'Stack trace' to the stack trace text.

Example: TypeError with stack trace



# Console Message Origin

```
1 console.log("Hello World!")
2 // Output: Hello World!
3
4 console.warn("Hello World!")
5 // Output [severity=warn]: Hello World!
6
7 console.error("Hello World!")
8 // Output [severity=error]: Hello World!
9
10 console.table([1,2,3])
11 // Output [table representation]: (index) => value
12
13 console.count("Hello")
14 // Output: Hello: 1
```



example.js

## Developer-caused Console Messages

- Available through the Console API in JavaScript
- Controlled by web developers

```
1 // static
2 ...::CreateConsoleMessageAboutFetchAutoupgrade(
3     ... main_resource_url,
4     ... mixed_content_url) {
5     String message = String::Format(
6         "Mixed Content: The page at '%s' was loaded over HTTPS, but requested an "
7         "insecure element '%s'. This request was "
8         "automatically upgraded to HTTPS, For more information see "
9         "https://blog.chromium.org/2019/10/"
10        "no-more-mixed-messages-about-https.html", ...);
11     return MakeGarbageCollected<ConsoleMessage>(
12        mojom::ConsoleMessageSource::kSecurity,
13        mojom::ConsoleMessageLevel::kWarning, message);
14 }
15 return MakeGarbageCollected<ConsoleMessage>(
```



mixed-content-checker.cc

## Browser-caused Console Messages

- Signals issues detected during page visit
- Embedded in the browser code **as templates**



# Research Gap



## Message Volume

- Lack of insights about console output in the wild (at scale)



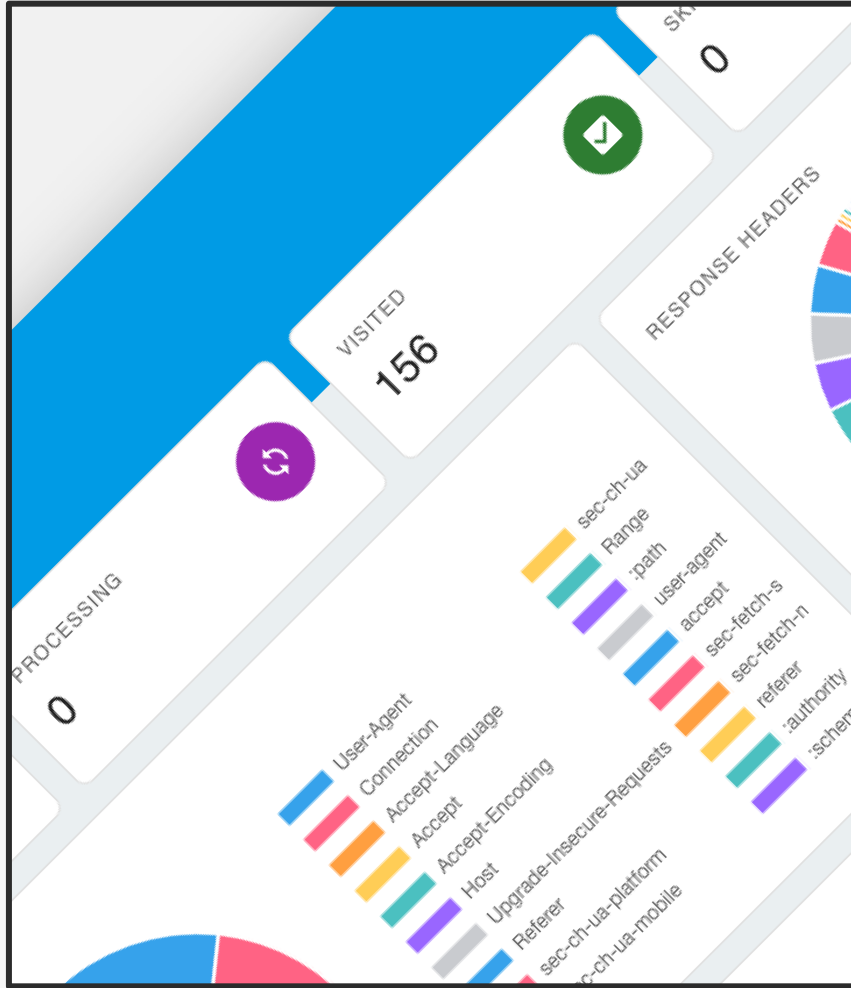
## Origin Attribution

- Missing separation in existing tooling



## Cross-Browser Consistency

- No data on consistency across browsers on public websites



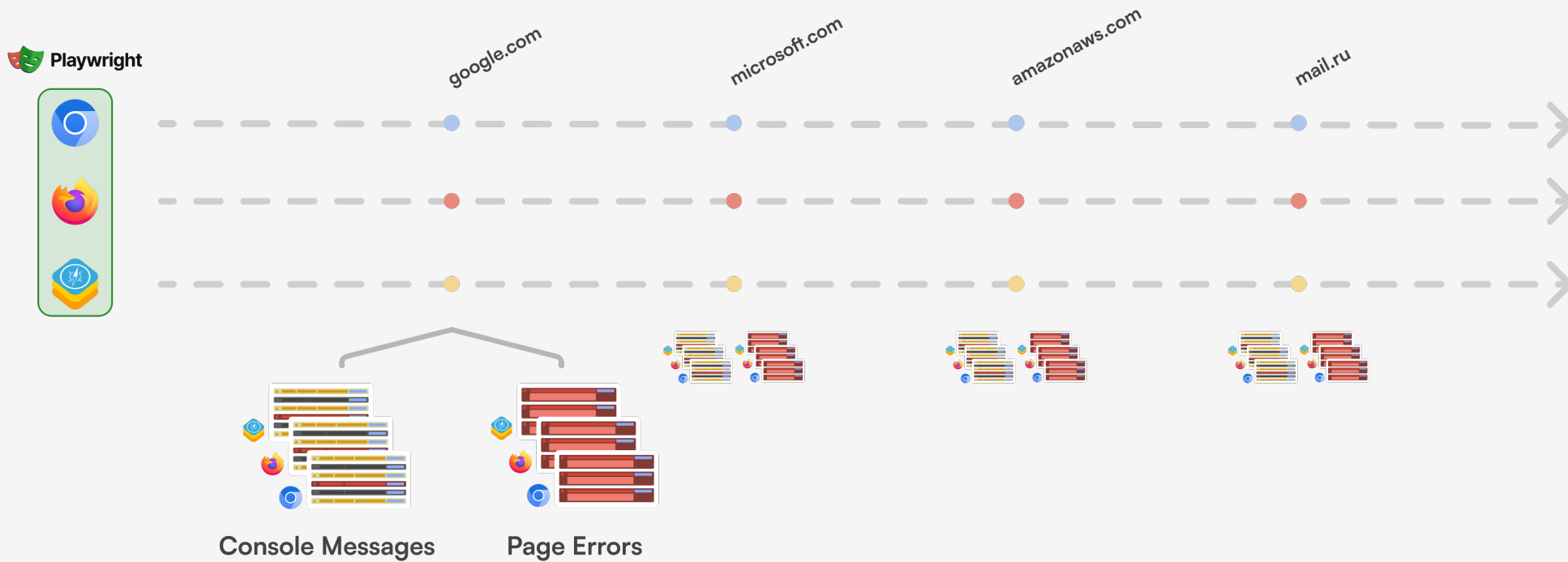
# Our Measurement Approach for Collecting Console Output

*Using Real Websites from Tranco*



# Data Collection

- Gather console output by visiting websites obtained from Tranco
- Same landing page for 30 seconds in Chromium, Firefox and WebKit





# What about the Message Origin?

```
warning <URL> line_number: - column: number: -
```

```
Mixed Content: The page at '<URL>' was loaded over HTTPS, but requested an insecure element '<URL>'. This request was automatically upgraded to HTTPS, For more information see https://blog.chromium.org/2019/10/no-more-mixed-messages-about-https.html
```

```
1 console.warn(`Mixed Content: The page at '<URL>' was loaded over HTTPS, but requested an
2 insecure element '<URL>'. This request was
3 automatically upgraded to HTTPS, For more information see
4 https://blog.chromium.org/2019/10/
5 no-more-mixed-messages-about-https.html`)
```

js test.js

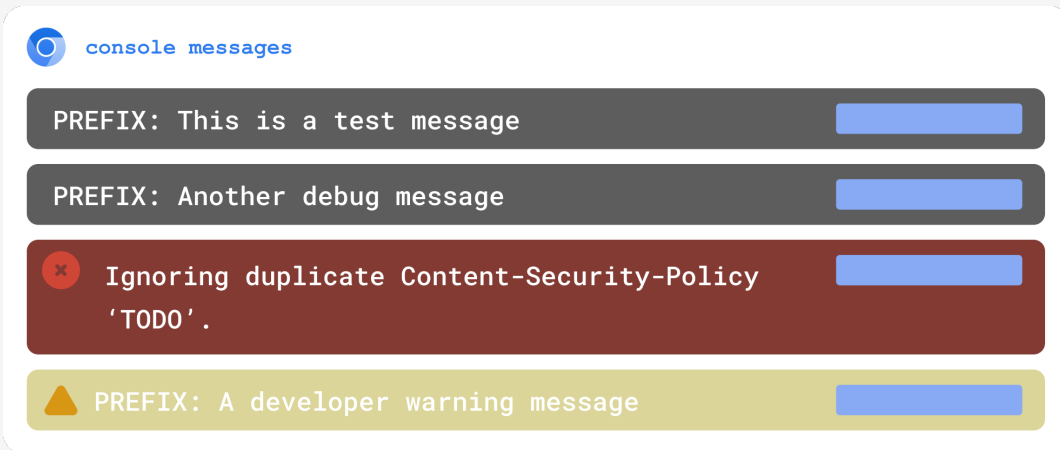
```
1 // static
2 ...::CreateConsoleMessageAboutFetchAutoupgrade(
3   ... main_resource_url,
4   ... mixed_content_url) {
5   String message = String::Format(
6     "Mixed Content: The page at '%s' was loaded over HTTPS, but requested an "
7     "insecure element '%s'. This request was "
8     "auto
9     "http
10    "no-more-mixed-messages-about-https.html".
```

- Lack of distinction of message origin in existing browser tooling
- Patched variants of Chromium, WebKit and regular expression for Firefox to prefix messages



# Obtaining Actual Templates

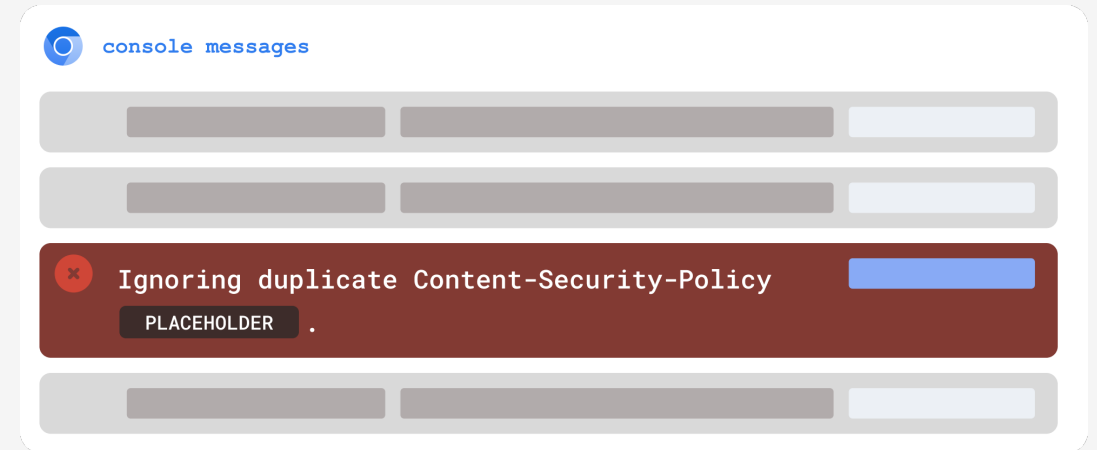
Filtering for browser-caused console messages through prefix/message structure



Capturing of prefixed and non-prefixed messages



Replace dynamic data with placeholders to obtain **template candidates**



Replace dynamic data in browser-caused console output



# Obtaining Actual Templates

- Using template candidates, query browser source code repositories
- Label colocated templates with high-level concept

The screenshot shows a web browser window displaying the Chromium Code Search interface. The search query is "1g duplicate Content-Security-Policy directive". The search results show a file named `content_security_policy.cc` with a search result on line 996. The search result is highlighted in yellow and reads: `"Ignoring duplicate Content-Security-Policy directive '%s'.",`. The surrounding code is as follows:

```
991 // A directive with this name has already been parsed. Skip further
992 // directives per
993 // https://www.w3.org/TR/CSP3/#parse-serialized-policy.
994 if (out->raw_directives.count(directive_name)) {
995     out->parsing_errors.emplace_back(base::StringPrintf(
996         "Ignoring duplicate Content-Security-Policy directive '%s'.",
997         std::string(directive.first).c_str()));
998     continue;
999 }
1000 out->raw_directives[directive_name] = std::string(directive.second);
1001
1002 if (!base::ranges::all_of(directive.second, IsDirectiveValueCharacter)) {
1003     out->parsing_errors.emplace_back(base::StringPrintf(
1004         "The value for the Content-Security-Policy directive '%s' contains "
```



# High-level Concepts for Comparing Browser-caused Console Messages



## Security

CSP, Mixed content, X-Frame-Options, CORS, ...



## Privacy

Cookies, Tracking, Permissions-Policy



## Content

HTML, CSS, fonts, WebGL, MIME types



## Networking

Failed requests, preload, preconnect



## Other

Browser-specific or uncategorized diagnostics



```
Setup swifter
set_defaults(
    dask_threshold=1,
    scheduler="processes",
    progress_bar=True,
    progress_bar_desc=None,
    allow_dask_on_strings=False,
    force_parallel=True,
)
# Configure output options
pd.set_option('display.max_rows', 50)
pd.options.display.float_format = "{:,.2f}".format

plt.rcParams['figure.dpi'] = 300
plt.rcParams['figure.figsize'] = [15*cm, 8*cm]

# Load all templates from test application
template_dict = load_templates("/home/jovyan/notebooks/templ

335]
c1_visited_domains = get_data(visited_domains_query)

..
The autoreload extension is already loaded. To reload it, use:
%reload_ext autoreload
Connecting to the PostgreSQL database...
Connection successful

Console Messages
```

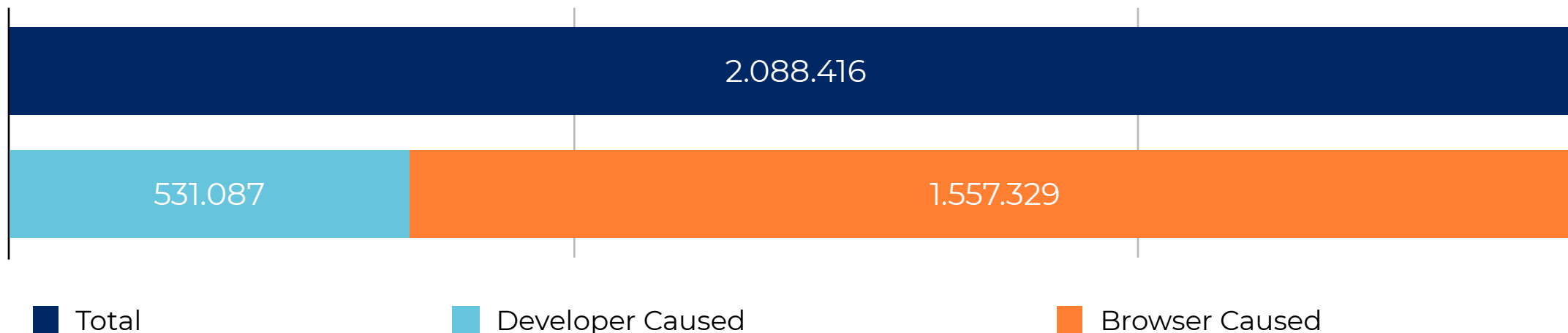
# Insights from our Data Collection

## *Results and Discussion*



# Overview

- Visited **51,984** websites in Chromium, Firefox and WebKit
- Practical relevance: More than **78%** of visited websites had at least one message (**13.39** on average)

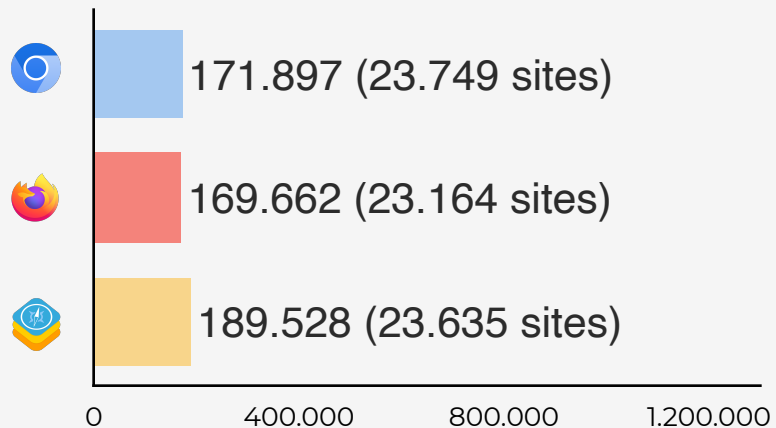


Total count of console messages



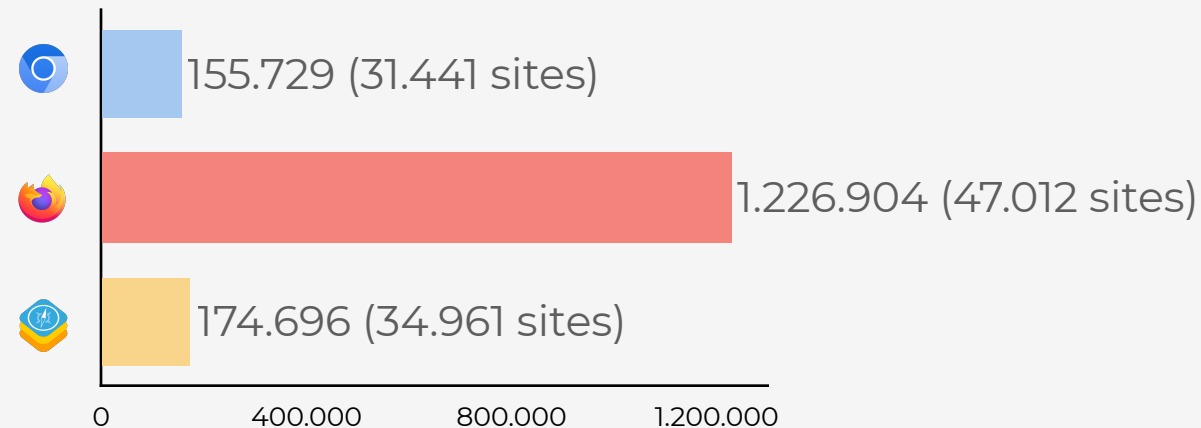
# Origin Attribution

Chromium Firefox WebKit



Distribution of developer-caused console messages (total and sites)

Similar quantities of developer-caused console output

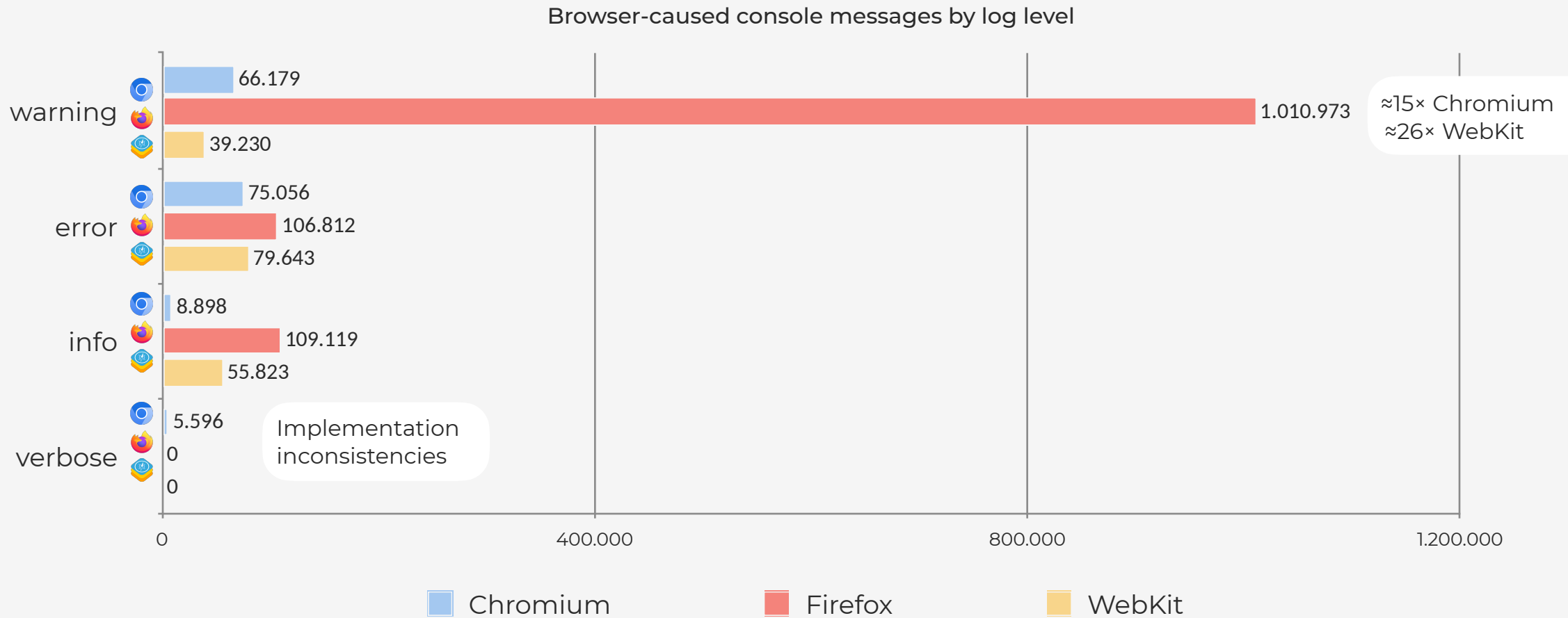


Distribution of browser-caused console messages (total and sites)

Firefox produces ~7× more than Chromium/WebKit

**Browser-Caused Messages Drive Cross-Browser Differences**

# Log Levels Differ Strongly Across Browsers



**i** The console is not only browser-specific in volume, but also in log level.



# Different Console Worlds

Chromium		Firefox		WebKit	
Failed resource load		SameSite cookie warning		Failed resource load	
request-failures (N)	18,664	cookies (P)	35,147	request-failures (N)	19,168
WebGL software fallback		Cookie attribute overwritten		Preconnect success	
webgl (C)	5,995	cookies (P)	19,251	preconnect (N)	18,266
Unused preload		Non-standard CSS zoom		Unused preload	
preload (N)	4,493	styles (C)	17,717	preload (N)	5,497

Top 3 browser-caused templates by number of affected sites



# Different Console Worlds

Chromium		Firefox		WebKit	
Failed resource load		SameSite cookie warning		Failed resource load	
request-failures (N)	18,664	cookies (P)	35,147	request-failures (N)	19,168
WebGL software fallback		Cookie attribute overwritten		Preconnect success	
webgl (C)	5,995	cookies (P)	19,251	preconnect (N)	18,266
Unused preload		Non-standard CSS zoom		Unused preload	
preload (N)	4,493	styles (C)	17,717	preload (N)	5,497

Top 3 browser-caused templates by number of affected sites

- The most visible console warnings differ **across browser engines**



# Different Console Worlds

Chromium		Firefox		WebKit	
Failed resource load		SameSite cookie warning		Failed resource load	
request-failures (N)	18,664	cookies (P)	35,147	request-failures (N)	19,168
WebGL software fallback		Cookie attribute overwritten		Preconnect success	
webgl (C)	5,995	cookies (P)	19,251	preconnect (N)	18,266
Unused preload		Non-standard CSS zoom		Unused preload	
preload (N)	4,493	styles (C)	17,717	preload (N)	5,497

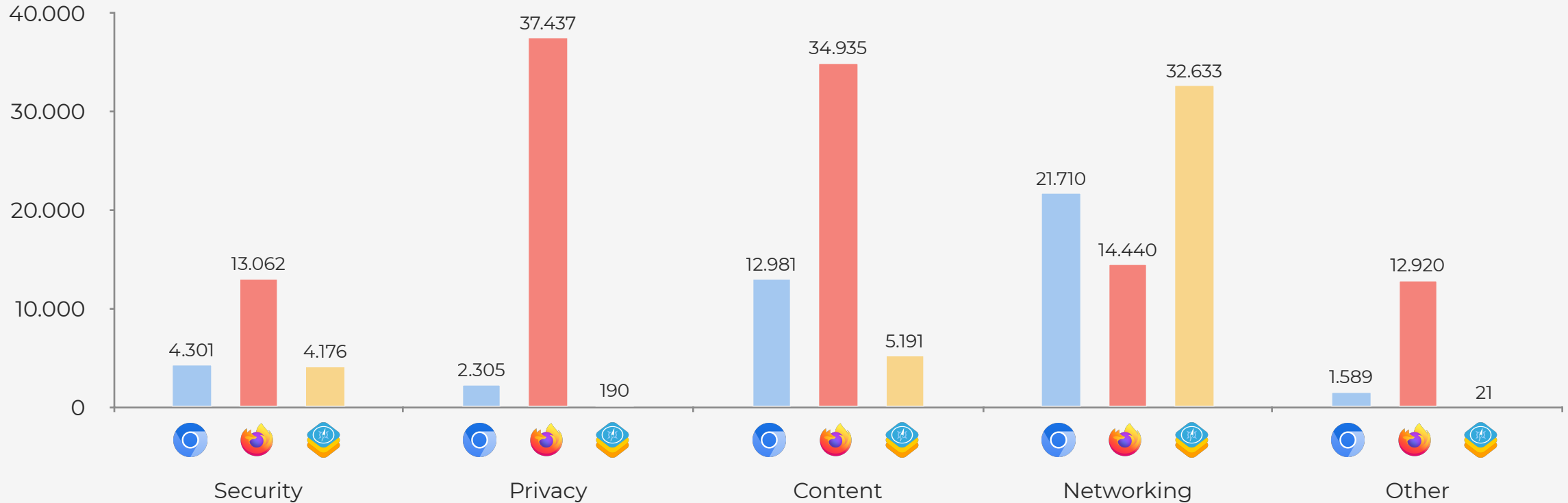
Top 3 browser-caused templates by number of affected sites

- The most visible console warnings differ **across browser engines**
- Reappearing categories may surface shared web engineering issues



# Templates Across High-Level Concepts

Sites with browser-caused messages from templates across high-level concepts



**i Significant differences in issue categories between browsers**



# Page Errors

- 92,518 page errors across 12,241 sites
- WebKit exposes 13k “ConsoleMessageLike” entries as page errors, revealing a browser/tooling classification artifact

Error Name	Chromium		Firefox		WebKit	
	Total	Sites	Total	Sites	Total	Sites
TypeError	17.726	3.514	11.006	3.565	5.614	3.918
ReferenceError	8.760	1.818	9,640	2.071	3.700	2.078
ConsoleMessageLike	0	0	0	0	13.288	4.884
Error (Generic)	4.545	889	6.270	832	1.532	1.082
Error (Custom Name)	1.822	924	1.466	740	1.934	1.142
SyntaxError	1.538	1.062	578	272	1.502	1.198
DOMException	591	351	748	120	147	133
RangeError	37	5	30	1	34	5
EvalError	3	3	2	2	2	2
URIError	1	1	1	1	1	1
Total	35.023	7.190	29.741	6.559	27.754	11.429

Count and unique sites per JavaScript error



# Page Errors

- 92,518 page errors across 12,241 sites
- WebKit exposes 13k “ConsoleMessageLike” entries as page errors, revealing a browser/tooling classification artifact

Error Name	Chromium		Firefox		WebKit	
	Total	Sites	Total	Sites	Total	Sites
TypeError	17.726	3.514	11.006	3.565	5.614	3.918
ReferenceError	8.760	1.818	9,640	2.071	3.700	2.078
ConsoleMessageLike	0	0	0	0	13.288	4.884
Error (Generic)	4.545	889	6.270	832	1.532	1.082
Error (Custom Name)	1.822	924	1.466	740	1.934	1.142
SyntaxError	1.538	1.062	578	272	1.502	1.198
DOMException	591	351	748	120	147	133
RangeError	37	5	30	1	34	5
EvalError	3	3	2	2	2	2
URIError	1	1	1	1	1	1
Total	35.023	7.190	29.741	6.559	27.754	11.429



# Case Study: Overwhelming Console Output

- 2,605 sites produced more than 100 messages in at least one browser
- Relevant warnings can become indistinguishable from background noise



How we built the Chrome DevTools Issues tab. <https://developer.chrome.com/blog/issues-tab>



# Case Study: Local File Inclusion On The Client

- Console may reveal deployment mistakes (e.g. unintended information leakage)

```
error    <URL> line_number: 9 column_number: 12  
Not allowed to load local resource: file:///C%7C/Users/sysw7/  
AppData/.../favicon.ico
```

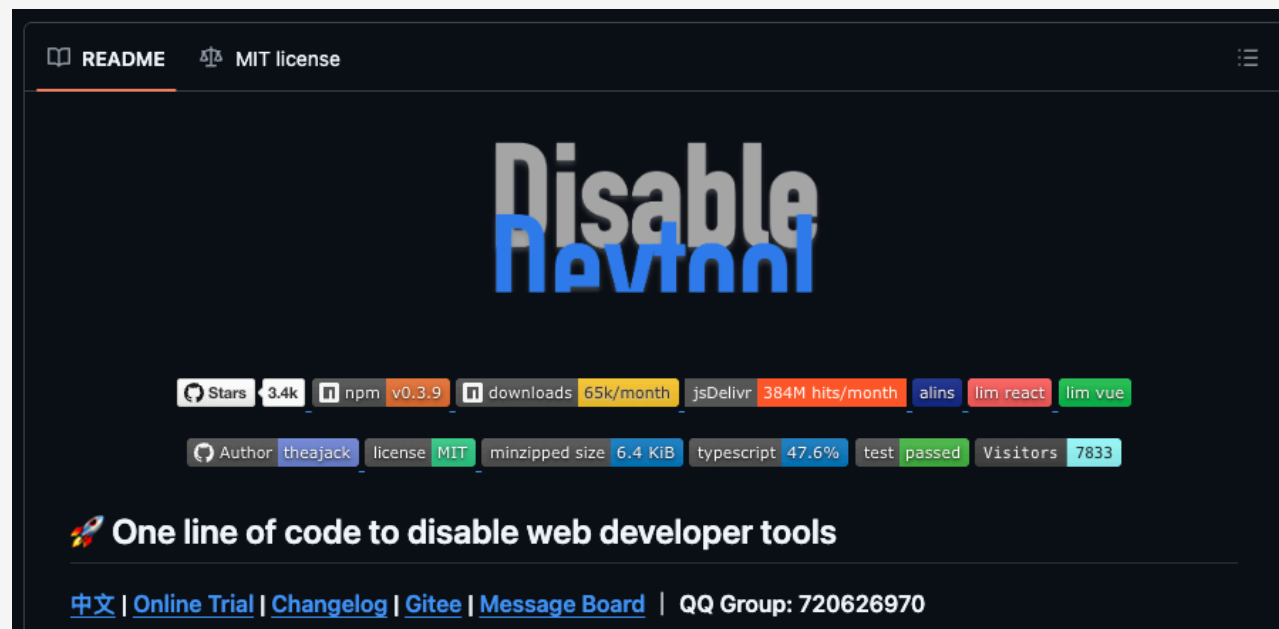
- Similar messages revealed probes for browser-extension resources

```
error    <URL> line_number: 9 column_number: 12  
Not allowed to load local resource:  
chrome://rumola/content/rumola48.png
```



# Case Study: Anti-Debugging Techniques

- High numbers of console.clear calls indicate anti-debugging behavior
- Use of libraries such as disable-devtool to inhibit usability of the console

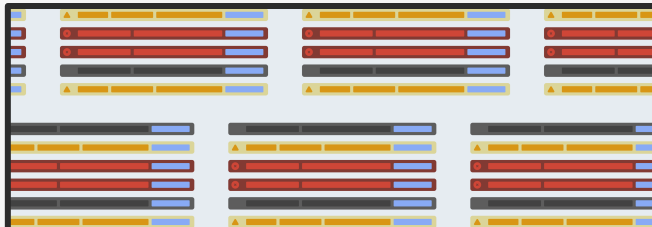


disable-devtool. <https://github.com/theajack/disable-devtool>

① Console can reveal attempts to interfere with inspection and debugging

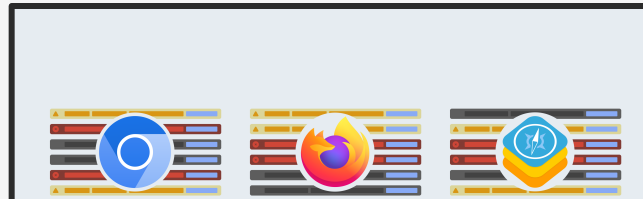


# Insights and Future Research Opportunities



## Flood of Messages

- High volume risks notification fatigue



## Browser Differences

- Browser-specific observability layer



Thank you. Questions?

## Usage in Measurements

- Leverage browser tooling for research

 Plethora of research opportunities with all involved stakeholders