Cross-Origin-Opener-Policy Explainer

Overview

The `Cross-Origin-Opener-Policy` response header provides a way for a document to request a new browsing context group / agent cluster to better isolate itself from other untrustworthy origins.

Resources:

- Spec discussion
- Draft definition
- Tracking bugs: Firefox, Chrome

Motivation

At least two types of attacks are possible when a document shares a browsing context group and possibly an operating system process with cross-origin documents:

- **Cross-window attacks**. For example, a malicious document may open a victim document in a new window and later navigate the window to a look-alike document to trick the user, or attempt to exploit postMessage vulnerabilities in the victim document.
- **Process-wide attacks**. Transient execution attacks like Spectre also pose a risk of the malicious document leaking data from the victim document, if they share an OS process.

In browsers that enable full Site Isolation and out-of-process iframes (e.g., Chrome Desktop), it is possible to mitigate process-wide attacks (at least at the site granularity) by putting documents from different sites in different processes. Cross-window attacks are less severe than process-wide attacks, but still remain possible in browsers with Site Isolation.

In browsers without out-of-process iframes, it is difficult to put cross-origin documents in a different process if they are in the same browsing context group. Consider a page with a top-level document on a.com and an iframe on b.com. If this page opens a popup to b.com, it is difficult to put the popup in a new process without breaking script interactions with the b.com iframe in the opener window.

We want to give sites the ability to sever all references to other browsing contexts to mitigate cross-window attacks, and to make it easier for browsers without out-of-process iframes to load the victim document in a new OS process to mitigate process-wide attacks like Spectre.

Goals

A web site should be able to include a response header on a top-level document that ensures it does not share a browsing context group with cross-origin documents. Under the hood, this should make it possible for browsers without out-of-process iframes to load this document in a different process from cross-origin documents in other windows, although this is not guaranteed.

Non-Goals

We will not require browsers to put certain documents in different OS processes.

Proposed Design

A `Cross-Origin-Opener-Policy` response header can be added to a document to ensure it does not share a browsing context group with cross-origin documents (or cross-site, if so desired), nor with same-origin documents with a non-matching policy header. This provides a greater degree of control over references to a window than '`noopener`,' which only affects outgoing navigations.

For a document with this header, browsers have the option to put the document in a different process than documents in other windows, even if the original opener window has same-origin documents in its frame tree.

See https://gist.github.com/annevk/6f2dd8c79c77123f39797f6bdac43f3e for a semi-formal draft description of the proposed header syntax and behavior.

Use Cases

Suppose Site A has sensitive data in a.com/sensitive, but not on a.com. It would like to ensure this data does not leak to any other origin, and that other origins cannot perform cross-window attacks on this document. It can add the following response header to a.com/sensitive:
`Cross-Origin-Opener-Policy: same-origin`

Any attempt to navigate to or open a popup to a.com/sensitive from a cross-origin document (or from a same-origin document lacking a matching header) will result in the creation of a new browsing context group. Under the hood, browsers without out-of-process iframes can load a.com/sensitive in a new process without worrying about breaking cross-window script interactions.

The document at a.com/sensitive can still include cross-origin iframes that it deems trustworthy, which will remain in its process in browsers without out-of-process iframes. Any popups it opens that are cross-origin or do not have a matching policy header (per the definition of how to

compare policies) will end up in a new browsing context group, unless the `same-origin-allow-popups` value is present in its `Cross-Origin-Opener-Policy` header.

Note that it may be important for Site A to ensure that certain subresources loaded by a.com/sensitive do not leak to other processes or documents. It can label such resources with the complementary `Cross-Origin-Resource-Policy` response header to prevent them from being delivered to cross-origin pages.

Alternatives Considered

See discussion at https://github.com/whatwg/html/issues/3740.