

To whom do sites answer?

W3C Permissions Workshop

September 2018

Martin Thomson

Accountability

n., liability to be called on to render an account

Conditions for Accountability

1. Actions need to be reported
2. Misconduct requires punishment

“A is accountable to B when A is obliged to inform B about A's (past or future) actions and decisions, to justify them, and to suffer punishment in the case of eventual misconduct” -- Andreas Schedler, "Conceptualizing Accountability", 1999.

The Web, Pre-Javascript

Static pages

Pages could use network resources

Limited to displaying what fit in the window

(Caveat: `<blink>` and animated .gif)

Loading a site

Condition: visit site, click link

Accounting: loading spinner, ~~status bar~~

Redress: stop button

Rendering a site

Condition: visit site

Accounting: site is visible

Redress: close the window

Javascript

Condition: visit site

Accounting: slow script dialog (?), OS (?)

Redress: close window

window.open

Condition: ~~none~~ --> engagement gesture

Accounting: a new window is visible

caveat: popunders

Redress: close the window

<input type=file>

Condition: selection of file to upload

Accounting: none

Redress: none

geolocation

First real attempt to gain permission for access to sensitive information

First API to use the doorhanger

Accountability: varies

Redress: none

getUserMedia

Condition: doorhanger

Accounting: on-screen indicators, camera light

Redress: varies

Notifications

Condition: doorhanger

Accounting: popup message

Redress: action menu on notification, or
byzantine mess of hard-to-find,
browser-specific menus

Push

Condition: (see Notifications)

Accounting: (mostly Notifications)

Redress: (see Notifications)

For Discussion

Browsers often infer permission

Design language and expectations both evolve

Calling sites to account requires more than
passing the problem on to users