

# Public resources and privacy opt-ins

Yoav Weiss - TPAC '22

# Public resources

- Never change based on credentials
- All requests get the same response
- May vary based on content-negotiation (e.g. `Accept`)

# Opaque by default

- Can't read response body
- Can't read headers
- Can't get timing information
- Can't embed in cross-origin-isolated documents

# Response opt-in

- `Access-Control-Allow-Origin: *`
- `Access-Control-Allow-Headers: *`
- `Timing-Allow-Origin: *`
- `Cross-Origin-Resource-Policy: cross-origin`

# Request opt-in

- `Access-Control-Allow-Origin: *`
- `Access-Control-Allow-Headers: *`
- Both require e.g. `<img src=foorbar crossorigin>`
- Background images have no opt in



AEROTWIST

[HOME](#) [BLOG](#) [TUTORIALS](#)

# CORS for Concern

# Use cases

- Developer ergonomics
- Image CDNs
  - Get image timing
  - Canvas manipulation
  - Reduce complexity for
    - CORS requirements changes
    - New observable features
- Timing info for static embedded resources
- Others?

# Proposal

- <https://github.com/whatwg/html/issues/8143>
- A single (new) response opt in for public resources
  - Can't override ACAO:\* semantics to support origins with cookies
  - Can't add new ACAO value due to parsing rules
- No request opt-in required



## HOW STANDARDS PROLIFERATE: (SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC.)



# Access-Control: public ??

- Exposes response body
- Exposes timing and size metadata
- Passes CORP checks
- *Even for credentialed no-cors requests*

# Open questions

- Bikeshedding: Is the name clear/scary enough?
- Expose headers?
- `fetch(request, { mode: "no-cors" })`
- Footgun mitigations?