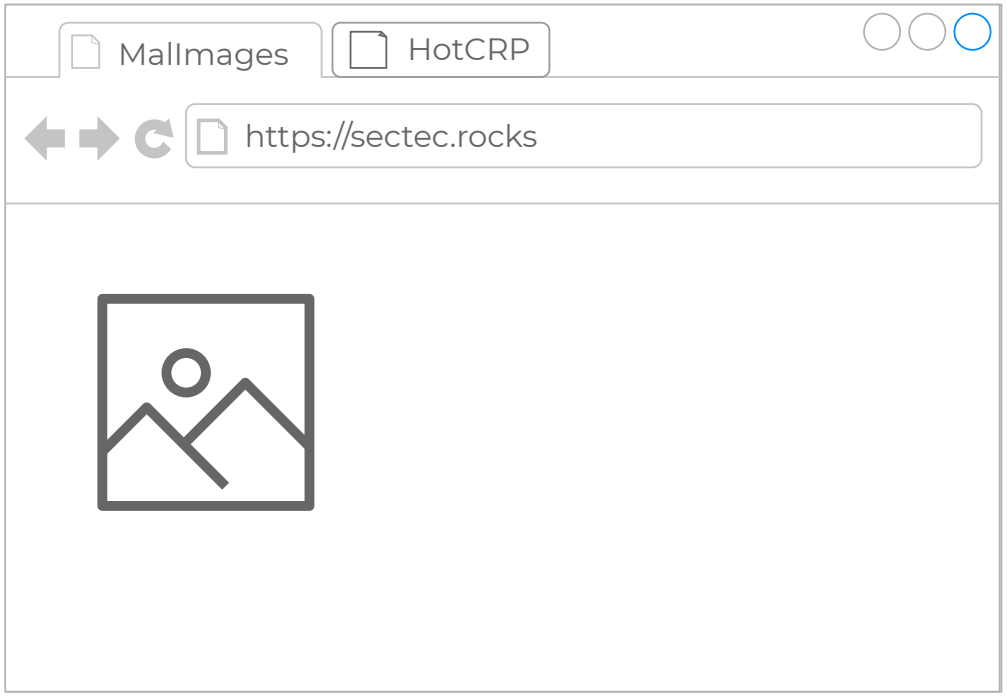
CISPA
HELMHOLTZ CENTER FOR INFORMATION SECURITY

The Leaky Web: Automated Discovery of Cross-Site Information Leaks in Browsers and the Web

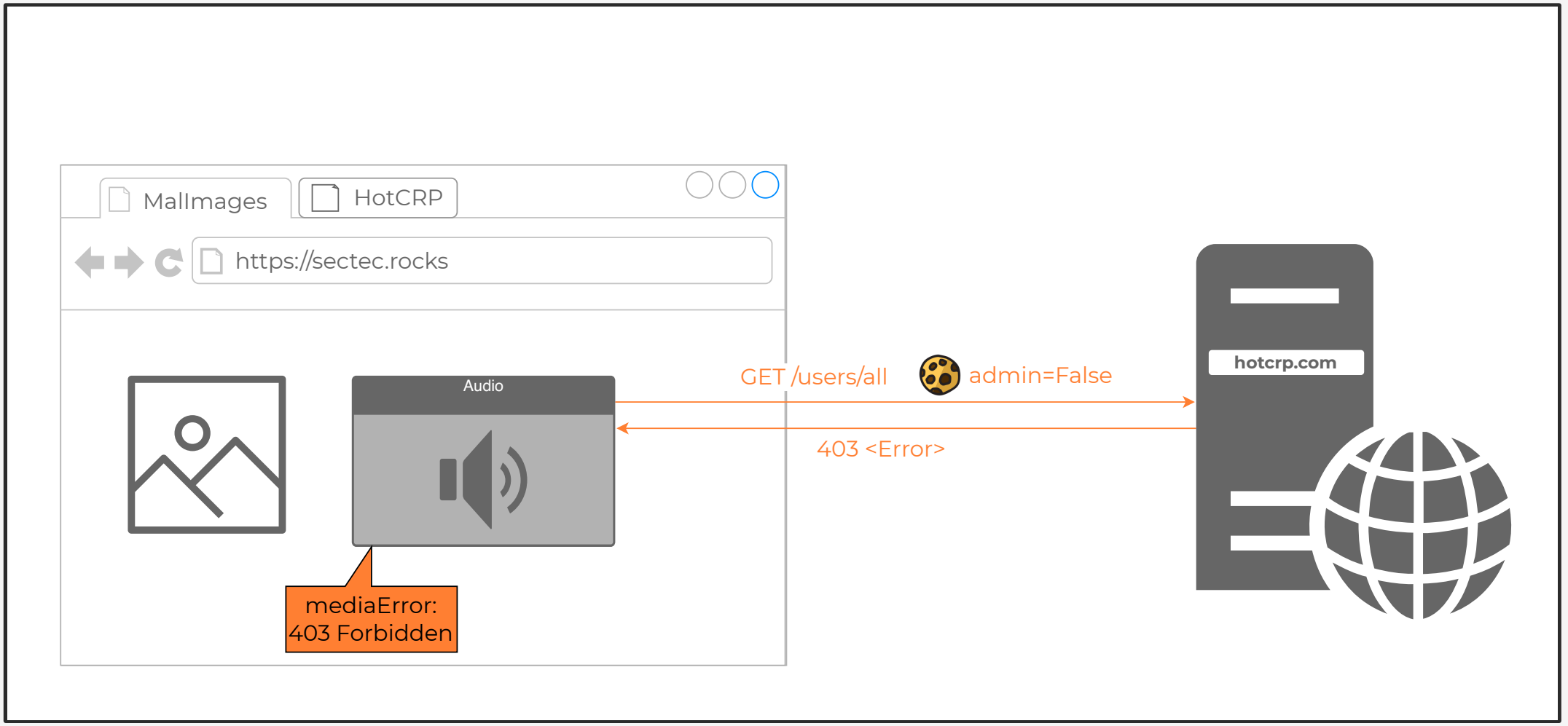
Jannis Rautenstrauch, *Giancarlo Pellegrino, Ben Stock*
CISPA Helmholtz Center for Information Security
jannis.rautenstrauch@cispa.de, *@jannis_r*

44th IEEE Symposium on Security and Privacy 2023

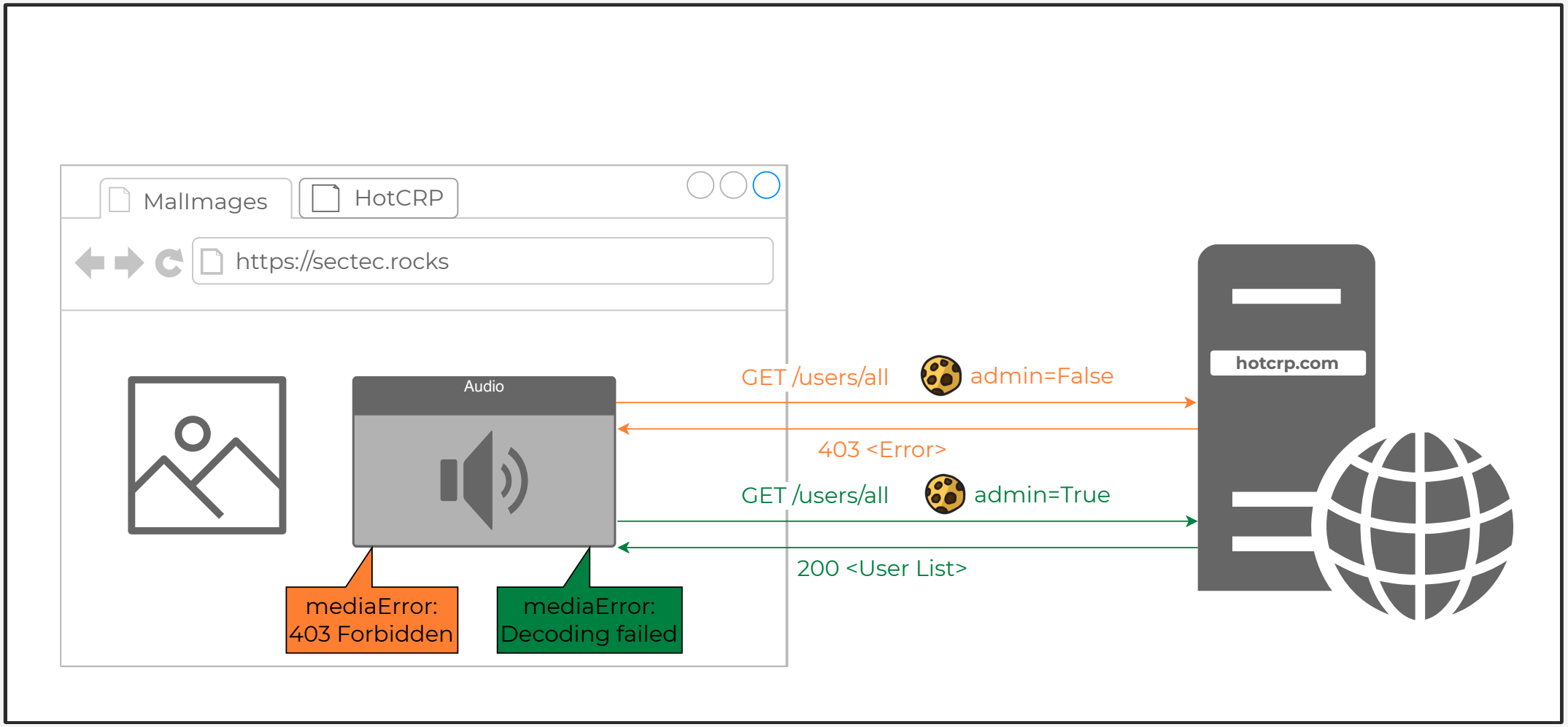
Browsers Leak Information Cross-Site



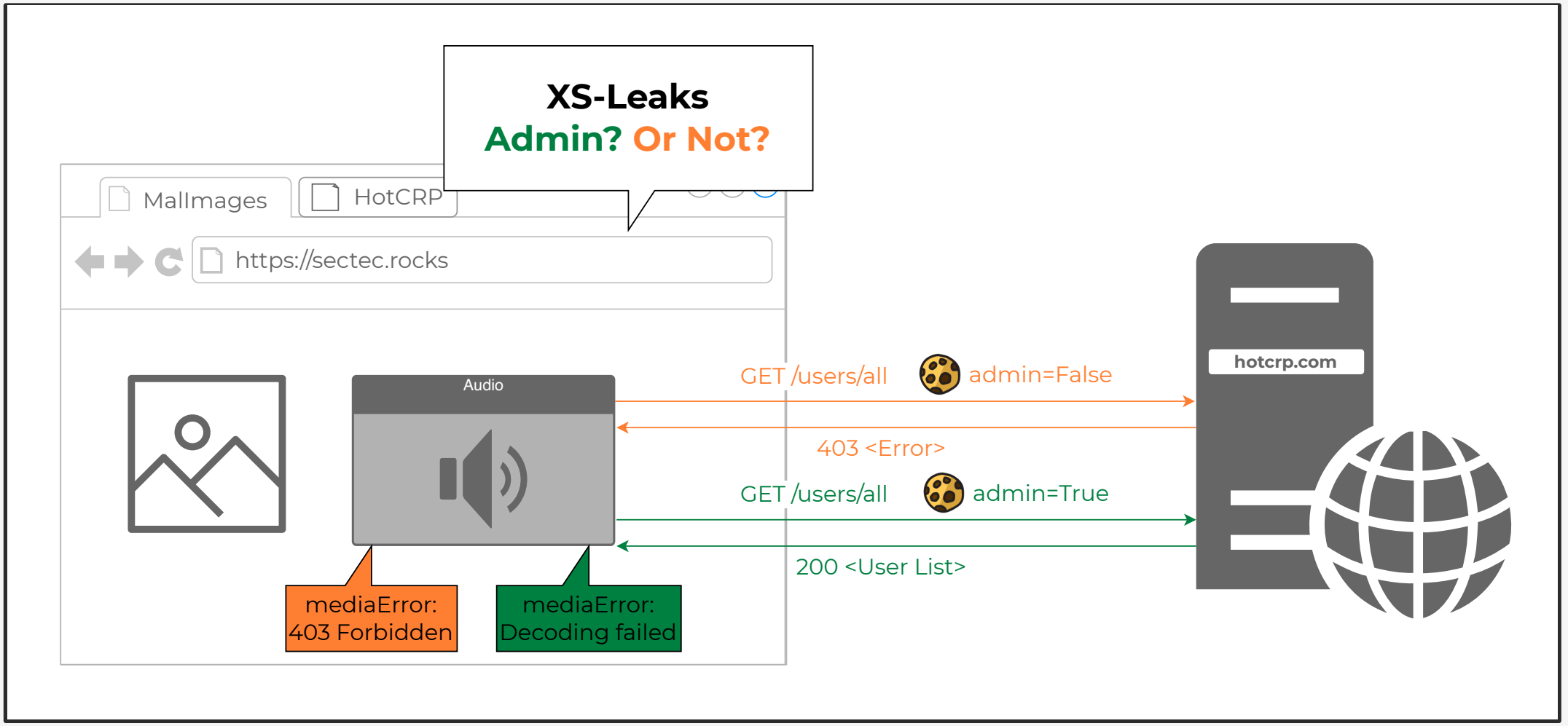
Browsers Leak Information Cross-Site



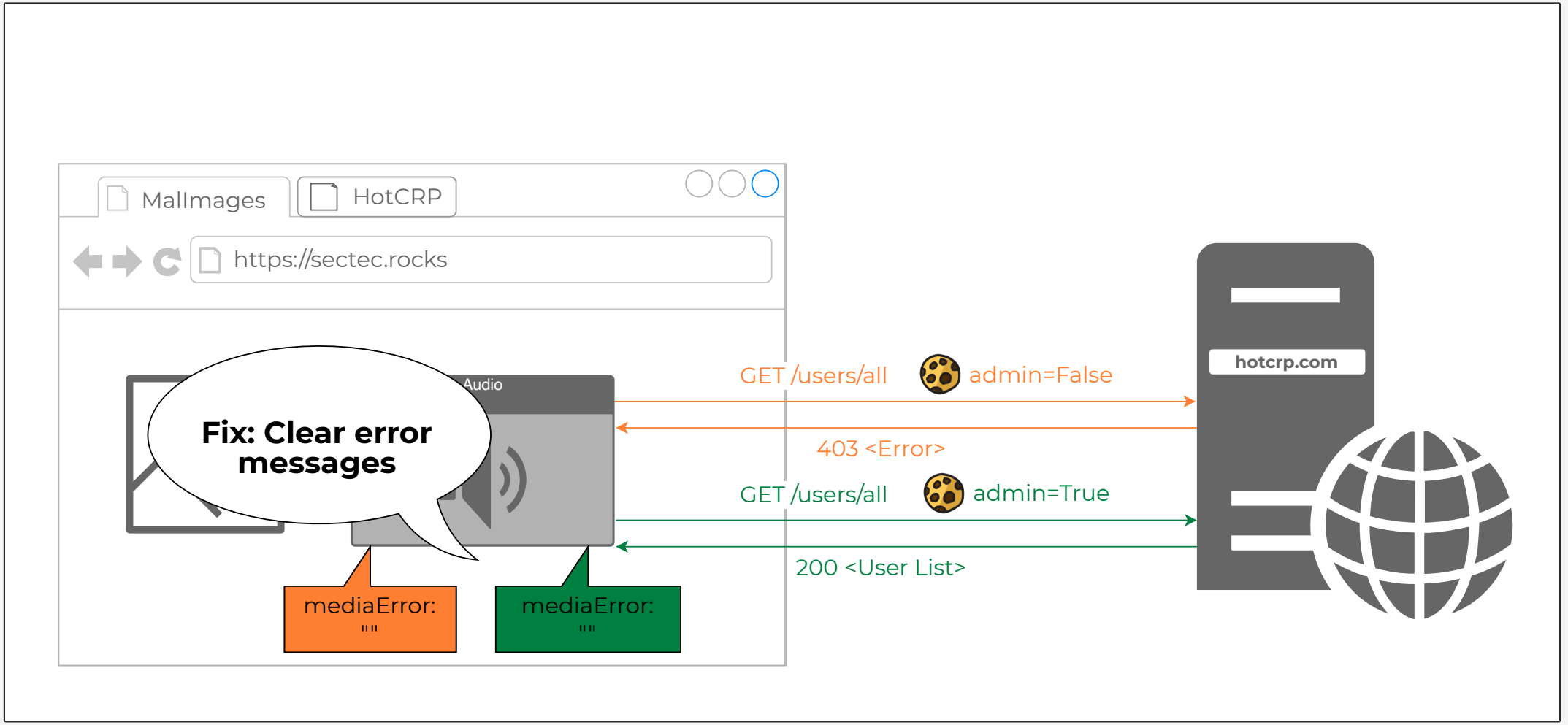
Browsers Leak Information Cross-Site



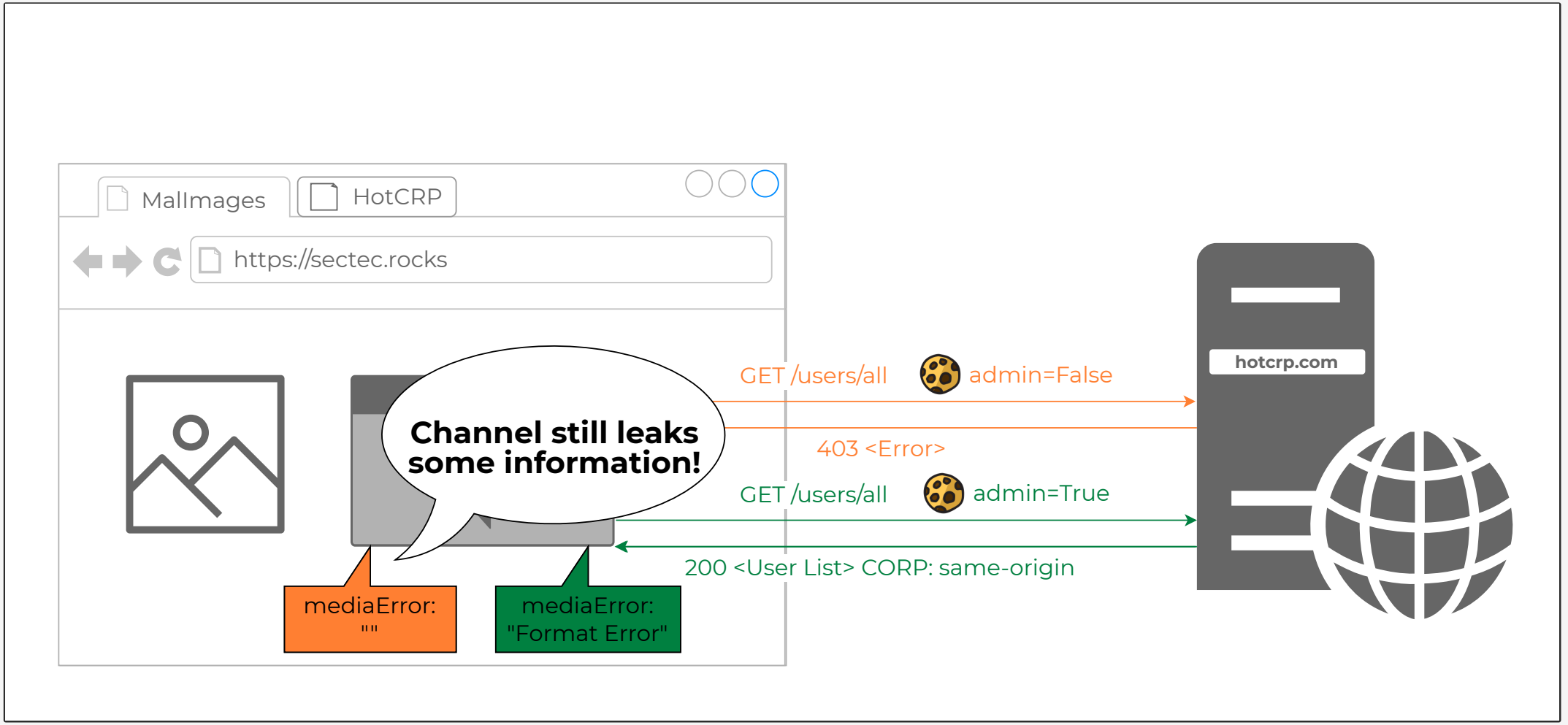
Browsers Leak Information Cross-Site



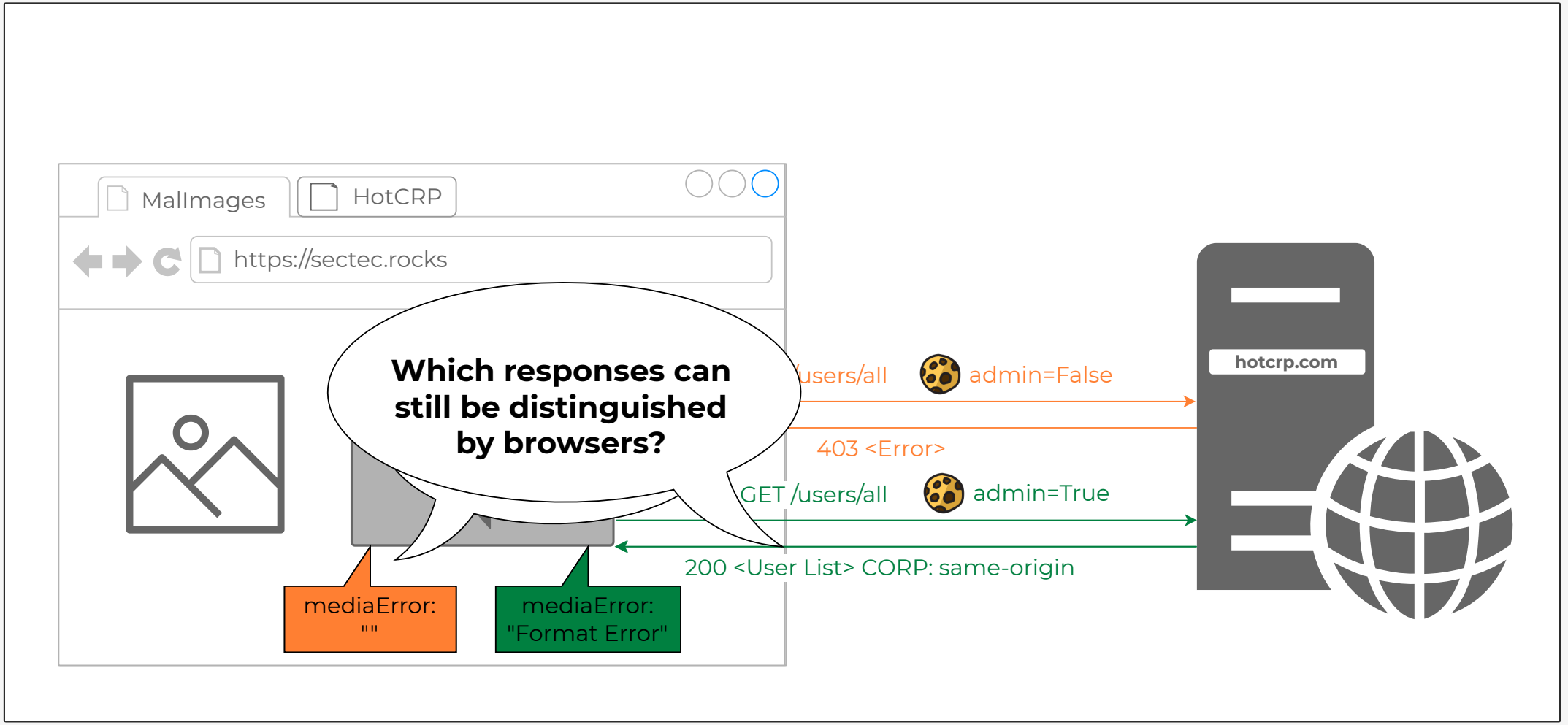
Systematic Analysis of XS-Leaks Problem Space



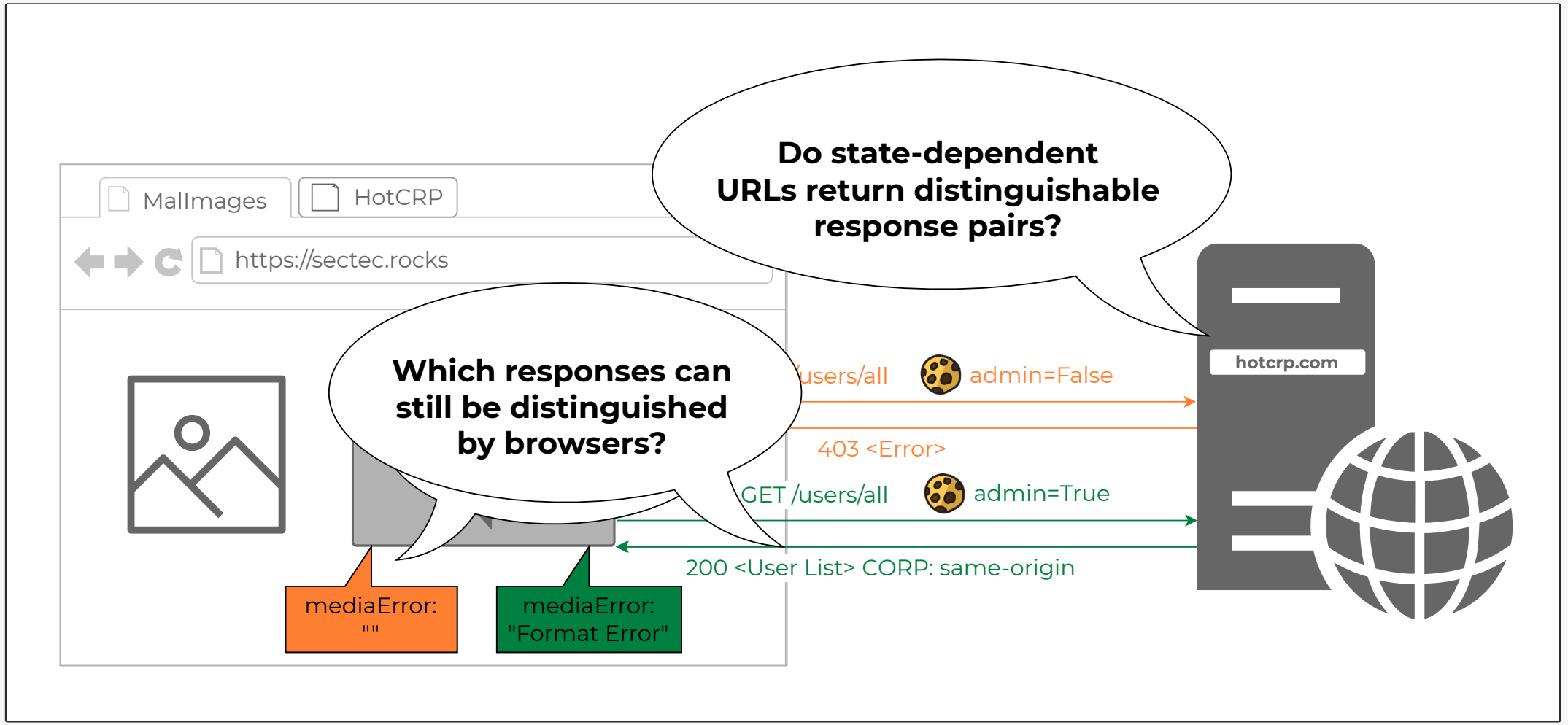
Systematic Analysis of XS-Leaks Problem Space



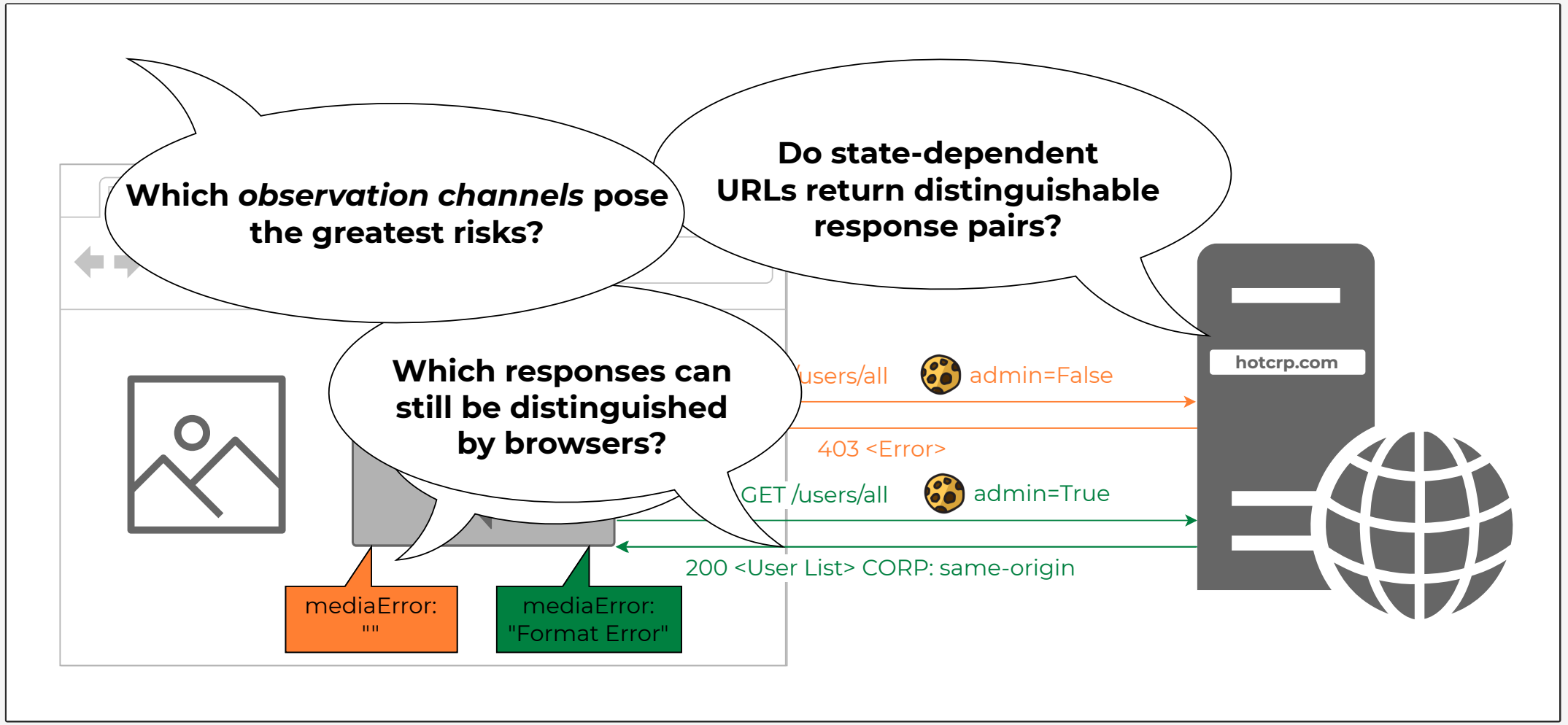
Systematic Analysis of XS-Leaks Problem Space



Systematic Analysis of XS-Leaks Problem Space



Systematic Analysis of XS-Leaks Problem Space



How to Find Leaky Observation Channels

- Collect data for channels and responses

How to Find Leaky Observation Channels

- Collect data for channels and responses
  - 2,040 potential channels

How to Find Leaky Observation Channels

- Collect data for channels and responses
  - 2,040 potential channels
    - 3 browsers (Chromium, Firefox, WebKit)

How to Find Leaky Observation Channels

- Collect data for channels and responses
  - 2,040 potential channels
    - 3 browsers (Chromium, Firefox, WebKit)
    - 20 inclusion methods (e.g., fetch, window.open)

How to Find Leaky Observation Channels

- Collect data for channels and responses
  - 2,040 potential channels
    - 3 browsers (Chromium, Firefox, WebKit)
    - 20 inclusion methods (e.g., fetch, window.open)
    - 34 observation methods (e.g., mediaError, frames.length)

How to Find Leaky Observation Channels

- Collect data for channels and responses
  - 2,040 potential channels
    - 3 browsers (Chromium, Firefox, WebKit)
    - 20 inclusion methods (e.g., fetch, window.open)
    - 34 observation methods (e.g., mediaError, frames.length)
  - 359,424 responses (63 status codes, 13 bodies, 8 headers)

How to Find Leaky Observation Channels

- Collect data for channels and responses
  - 2,040 potential channels
    - 3 browsers (Chromium, Firefox, WebKit)
    - 20 inclusion methods (e.g., fetch, window.open)
    - 34 observation methods (e.g., mediaError, frames.length)
  - 359,424 responses (63 status codes, 13 bodies, 8 headers)
- Create human-understandable summaries

Many Channels Leak Information

- 280 leaking channels

Many Channels Leak Information

- 280 leaking channels
  - audio-MediaError: Leaks whether a response is valid audio *and if CORP header is set*

Many Channels Leak Information

- 280 leaking channels
  - audio-MediaError: Leaks whether a response is valid audio *and if CORP header is set*
  - window.open-length: Leaks number of IFrames in an opened tab

Many Channels Leak Information

- 280 leaking channels
  - audio-MediaError: Leaks whether a response is valid audio *and if CORP header is set*
  - window.open-length: Leaks number of IFrames in an opened tab
- 11 bugs (3 CVEs)

Many Channels Leak Information

- 280 leaking channels
  - audio-MediaError: Leaks whether a response is valid audio *and if CORP header is set*
  - window.open-length: Leaks number of IFrames in an opened tab
- 11 bugs (3 CVEs)
- Many browser differences:

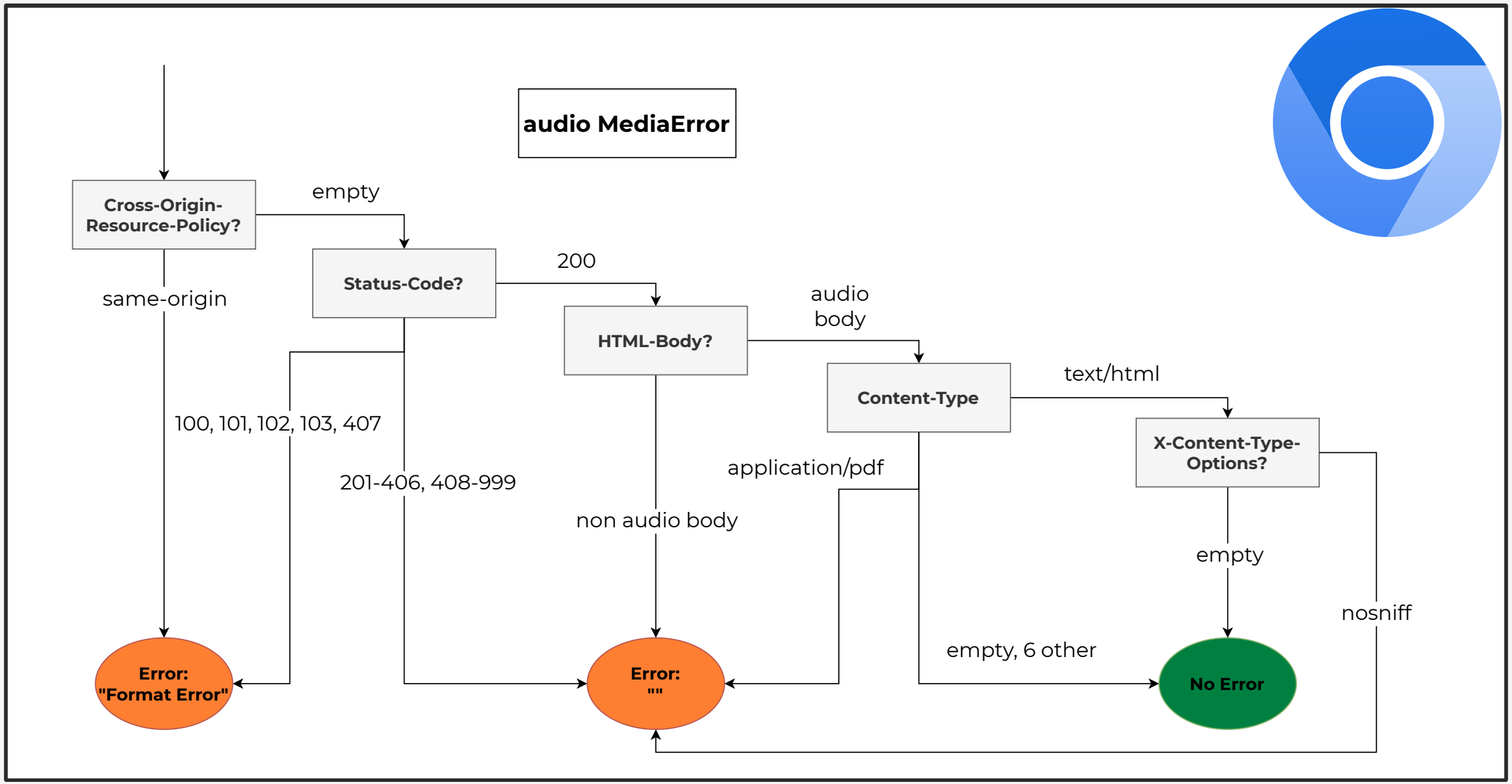
Many Channels Leak Information

- 280 leaking channels
  - audio-MediaError: Leaks whether a response is valid audio *and if CORP header is set*
  - window.open-length: Leaks number of IFrames in an opened tab
- 11 bugs (3 CVEs)
- Many browser differences:
  - Redirections: 300 only Firefox, 305 only WebKit

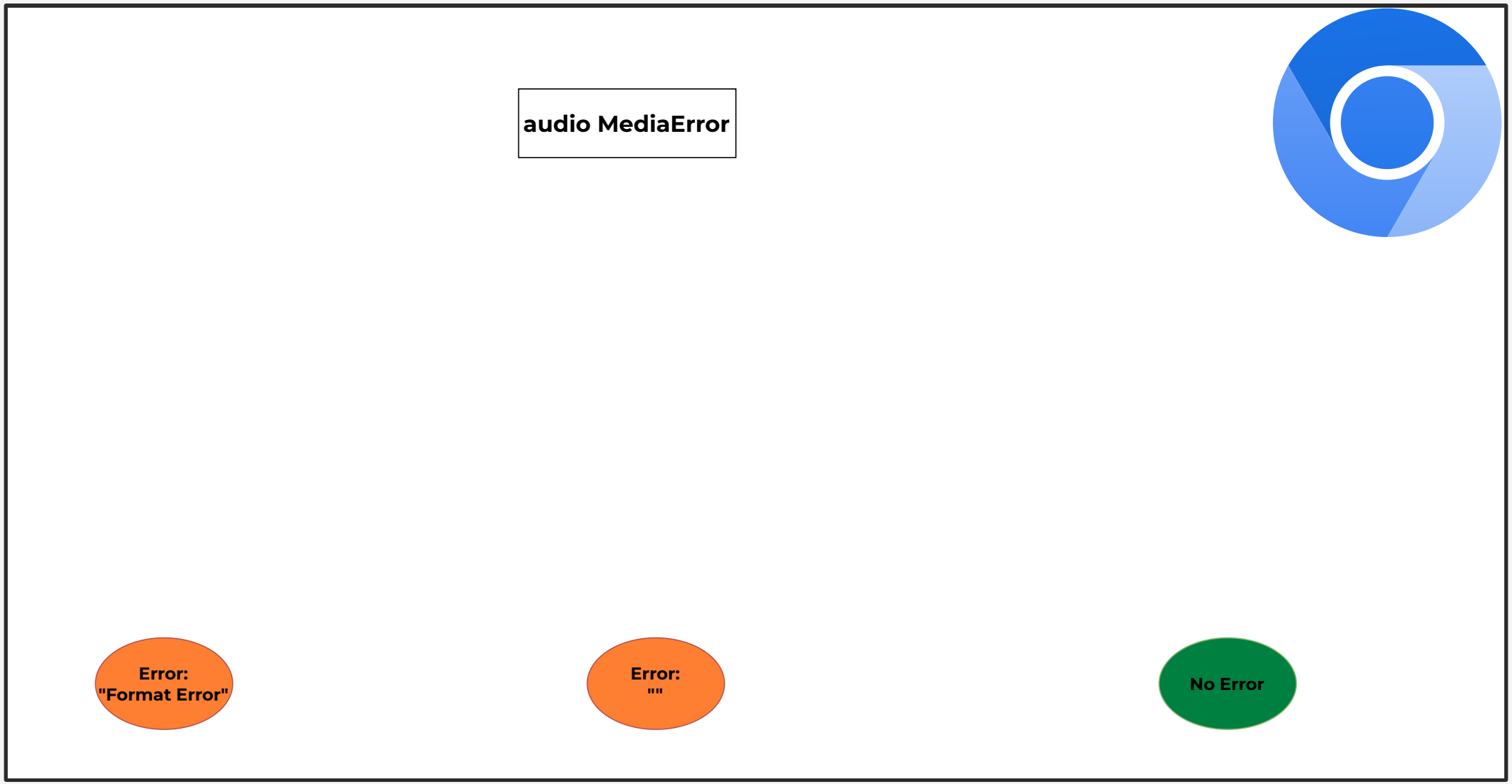
Many Channels Leak Information

- 280 leaking channels
  - audio-MediaError: Leaks whether a response is valid audio *and if CORP header is set*
  - window.open-length: Leaks number of IFrames in an opened tab
- 11 bugs (3 CVEs)
- Many browser differences:
  - Redirections: 300 only Firefox, 305 only WebKit
  - Cross-Origin-Read-Blocking (Chromium)

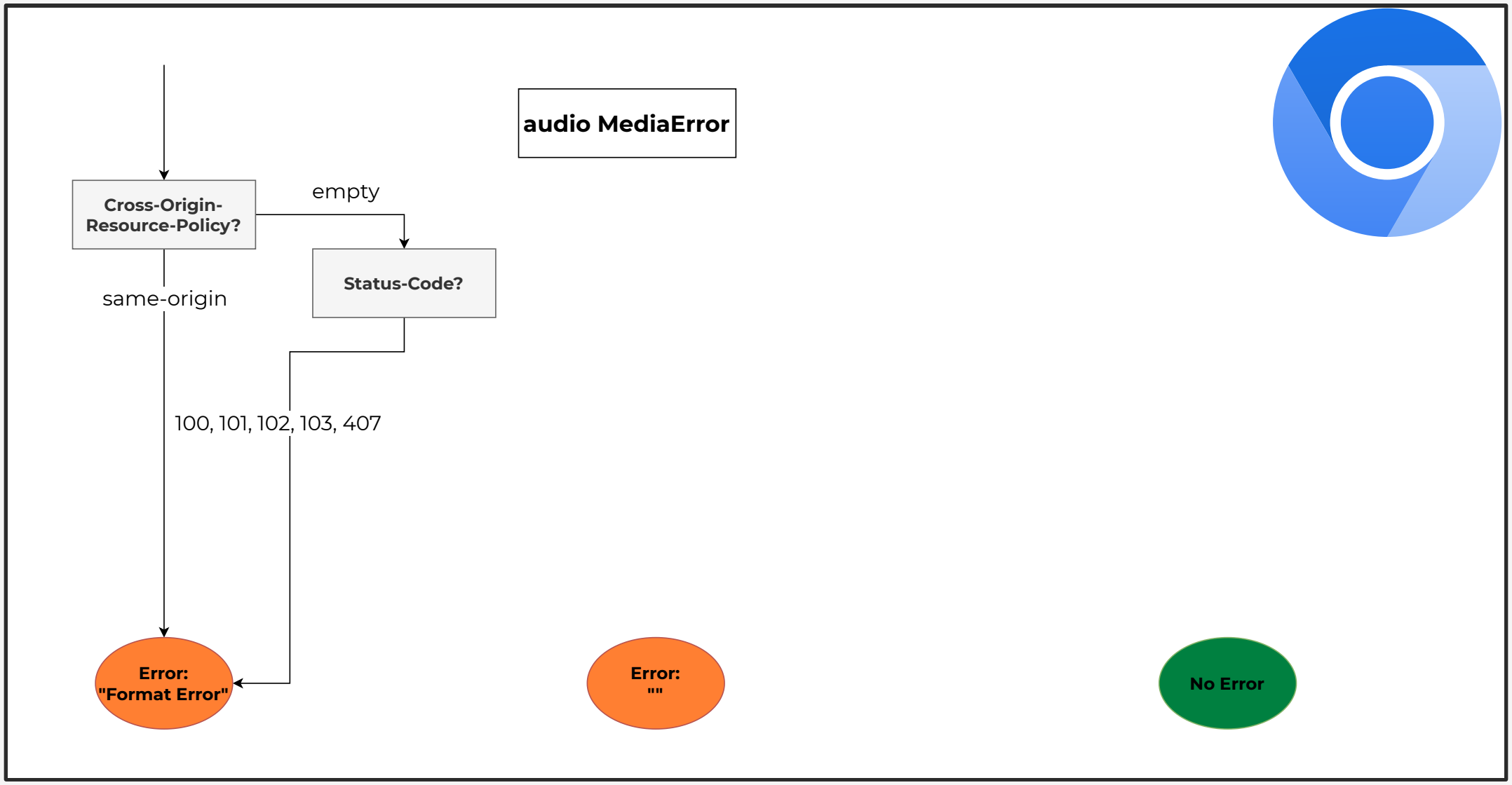
Decision Tree Summaries of Observation Channels



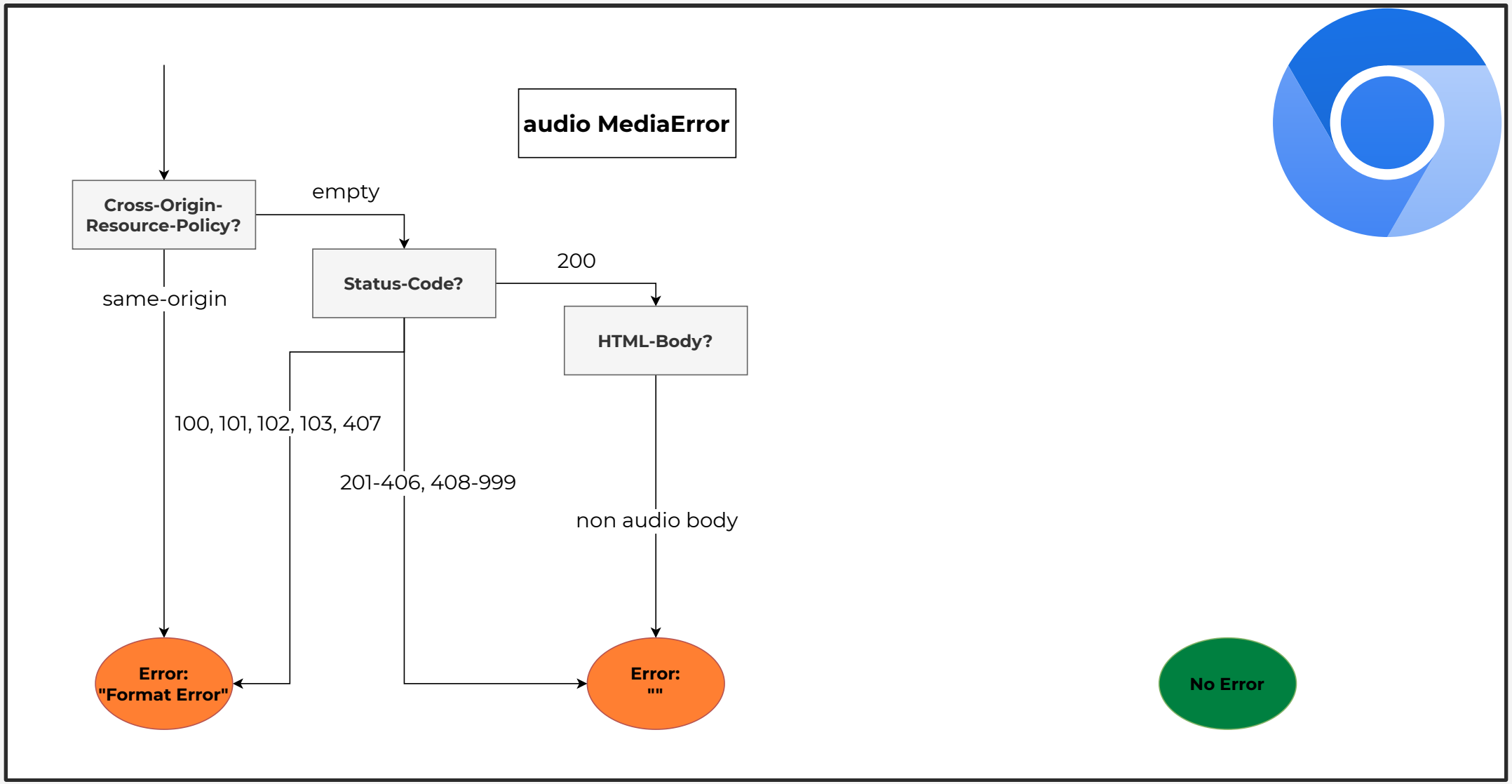
Decision Tree Summaries of Observation Channels



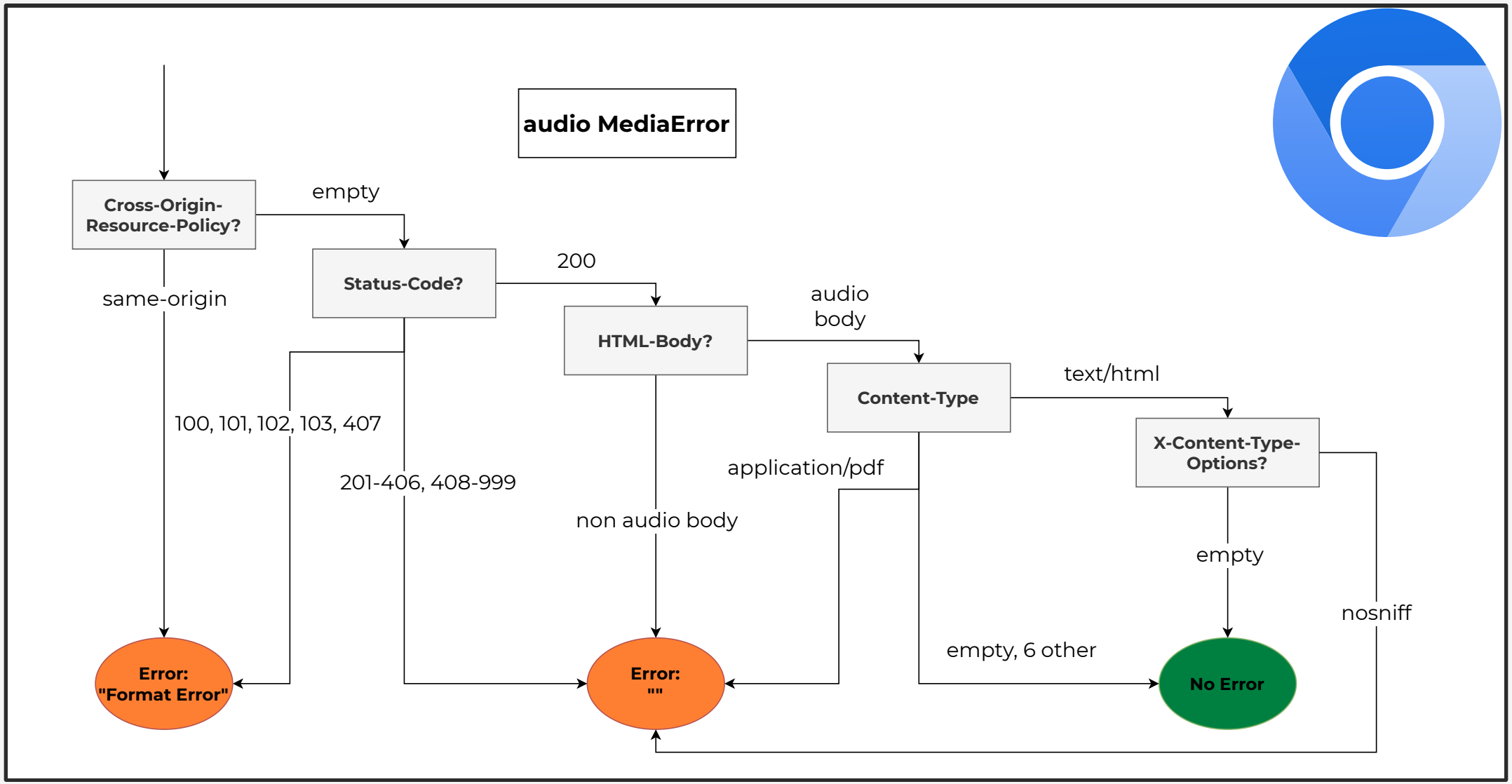
Decision Tree Summaries of Observation Channels



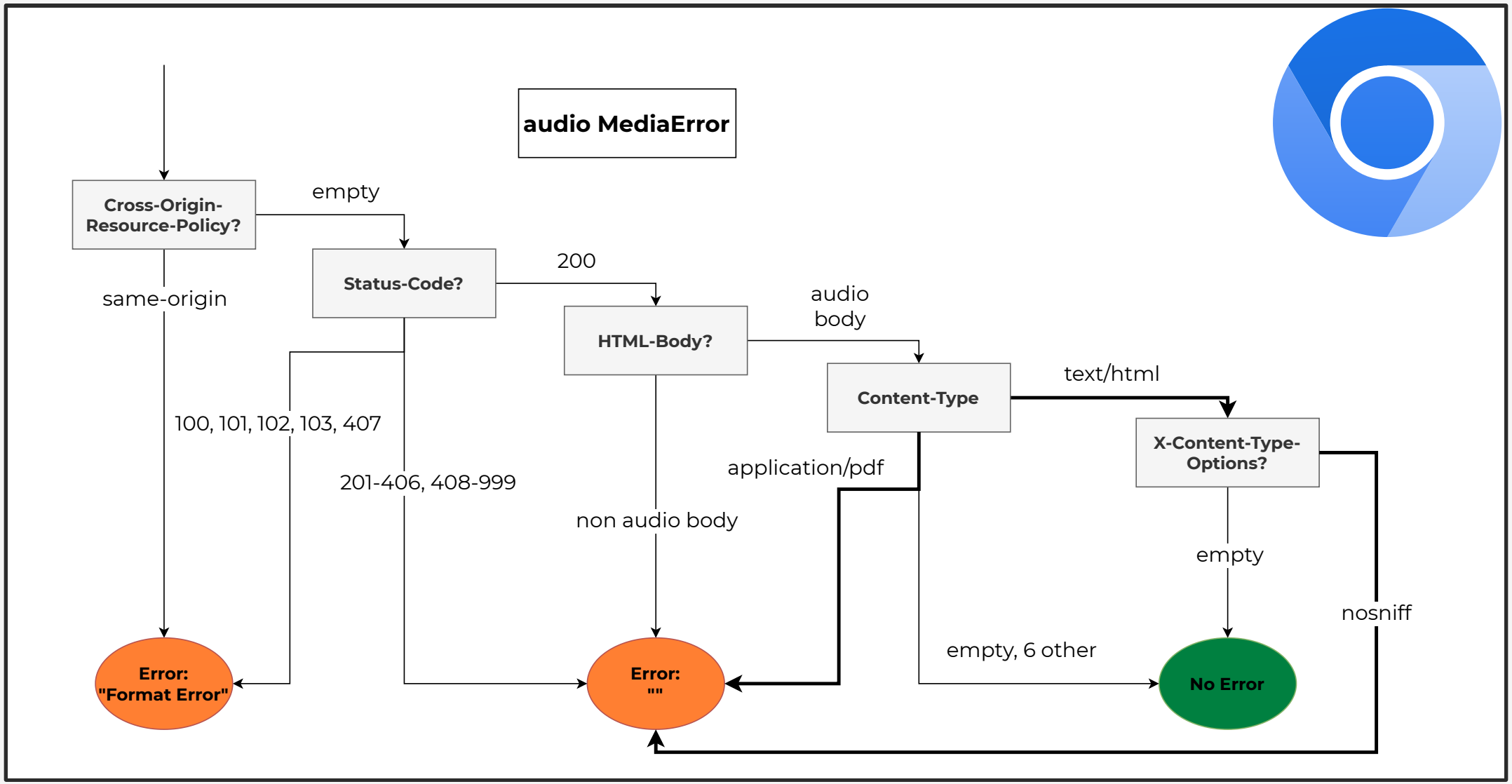
Decision Tree Summaries of Observation Channels



Decision Tree Summaries of Observation Channels



Decision Tree Summaries of Observation Channels



Which Channels Lead to XS-Leaks?

Which Channels Lead to XS-Leaks?

1. Create state

Which Channels Lead to XS-Leaks?

1. Create state
    - **Login detection**: automation problematic

Which Channels Lead to XS-Leaks?

1. Create state
   - **Login detection**: automation problematic
   - **Visit inference**: unclear if it makes a difference on sites

Which Channels Lead to XS-Leaks?

1. Create state
   - **Login detection**: automation problematic
   - **Visit inference**: unclear if it makes a difference on sites
   - **Cookie acceptance inference** (new): automation possible; changes more likely

Which Channels Lead to XS-Leaks?

1. Create state
   - **Login detection**: automation problematic
   - **Visit inference**: unclear if it makes a difference on sites
   - **Cookie acceptance inference** (new): automation possible; changes more likely
2. Collect responses in all states

Which Channels Lead to XS-Leaks?

1. Create state
   - **Login detection**: automation problematic
   - **Visit inference**: unclear if it makes a difference on sites
   - **Cookie acceptance inference** (new): automation possible; changes more likely
2. Collect responses in all states
3. Use trees to detect distinguishable response pairs

Which Channels Lead to XS-Leaks?

1. Create state
   - **Login detection**: automation problematic
   - **Visit inference**: unclear if it makes a difference on sites
   - **Cookie acceptance inference** (new): automation possible; changes more likely
2. Collect responses in all states
3. Use trees to detect distinguishable response pairs
4. Dynamic confirmation (necessary due to SameSite cookies, fetch metadata, …)

Many Sites are Vulnerable

State difference	Considered sites	Vulnerable sites	Percentage vulnerable

Many Sites are Vulnerable

State difference	Considered sites	Vulnerable sites	Percentage vulnerable
Login detection (Chromium or Firefox)	100	77	**77 %**

Many Sites are Vulnerable

State difference	Considered sites	Vulnerable sites	Percentage vulnerable
Login detection (Chromium or Firefox)	100	77	**77 %**
Visit inference (Chromium or Firefox)	8,355	1,291	**15,5 %**

Many Sites are Vulnerable

State difference	Considered sites	Vulnerable sites	Percentage vulnerable
Login detection (Chromium or Firefox)	100	77	**77 %**
Visit inference (Chromium or Firefox)	8,355	1,291	**15,5 %**
Cookie acceptance inference (Chromium)	3,160	1,097	**34,7 %**

Browsers and Channels Behave Differently

Observation Channel	Vulnerable only Chromium	Vulnerable only Firefox	Vulnerable either

Selected observation channels for visit inference

Browsers and Channels Behave Differently

Observation Channel	Vulnerable only Chromium	Vulnerable only Firefox	Vulnerable either
window.open-length (Top 1)	337	187	**745**

Selected observation channels for visit inference

Browsers and Channels Behave Differently

Observation Channel	Vulnerable only Chromium	Vulnerable only Firefox	Vulnerable either
window.open-length (Top 1)	337	187	**745**
fetch-creds-cors-performanceAPI (Top 5)	0	96	**96**

Selected observation channels for visit inference

Browsers and Channels Behave Differently

Observation Channel	Vulnerable only Chromium	Vulnerable only Firefox	Vulnerable either
window.open-length (Top 1)	337	187	**745**
fetch-creds-cors-performanceAPI (Top 5)	0	96	**96**
link-stylesheet-events-fired (Top 9)	49	8	**58**

Selected observation channels for visit inference

Response Distinguishing Oracle

	Response 1	Response 2
Status-Code	200	403
Body-Content	HTML body	HTML body
Content-Type	text/html	text/html
X-Content-Type-Options	empty	empty
X-Frame-Options	empty	empty
Content-Disposition	empty	empty
Location	empty	empty
Cross-Origin-Opener-Policy	empty	empty
Cross-Origin-Resource-Policy	empty	empty
Content-Security-Policy	empty	empty

Distinguish!

Response Distinguishing Oracle

	Response 1	Response 2
Status-Code	200	403
Body-Content	HTML body	HTML body
Content-Type	text/html	text/html
X-Content-Type-Options	empty	empty
X-Frame-Options	empty	empty
Content-Disposition	empty	empty
Location	empty	empty
Cross-Origin-Opener-Policy	empty	empty
Cross-Origin-Resource-Policy	empty	empty
Content-Security-Policy	empty	empty

Distinguish!

Inclusion method	Observation method	Browser	Observation 1	Observation 2
object	events-fired	chromium	load	error
object	events-fired	firefox	load	error
object	events-fired	webkit	load	uncalled
script	el-error	firefox	[object ErrorEvent]-undefined-undefined	uncalled
script	el-error	webkit	[object ErrorEvent]-undefined-undefined	uncalled
script	events-fired	chromium	load	error
script	events-fired	firefox	load	error
script	events-fired	webkit	load	error

Show more channels (8/23 shown)

Limitations

- Prototype implementation: lower bounds

Limitations

- Prototype implementation:
  lower bounds
- Window.open:
  not stealthy and require user interaction

Browsers Leak Information on Websites

- XS-Leaks prevalent

Browsers Leak Information on Websites

- XS-Leaks prevalent
- Focus on single responses problematic

Browsers Leak Information on Websites

- XS-Leaks prevalent
- Focus on single responses problematic
  - Incomplete fixes

Browsers Leak Information on Websites

- XS-Leaks prevalent
- Focus on single responses problematic
  - Incomplete fixes
  - Hidden browser differences

Browsers Leak Information on Websites

- XS-Leaks prevalent
- Focus on single responses problematic
  - Incomplete fixes
  - Hidden browser differences
- Tools open-sourced

Browsers Leak Information on Websites

- XS-Leaks prevalent
- Focus on single responses problematic
  - Incomplete fixes
  - Hidden browser differences
- Tools open-sourced

Thanks for your attention!

Questions?

https://github.com/cispa/xs-observations

jannis.rautenstrauch@cispa.de
@jannis_r