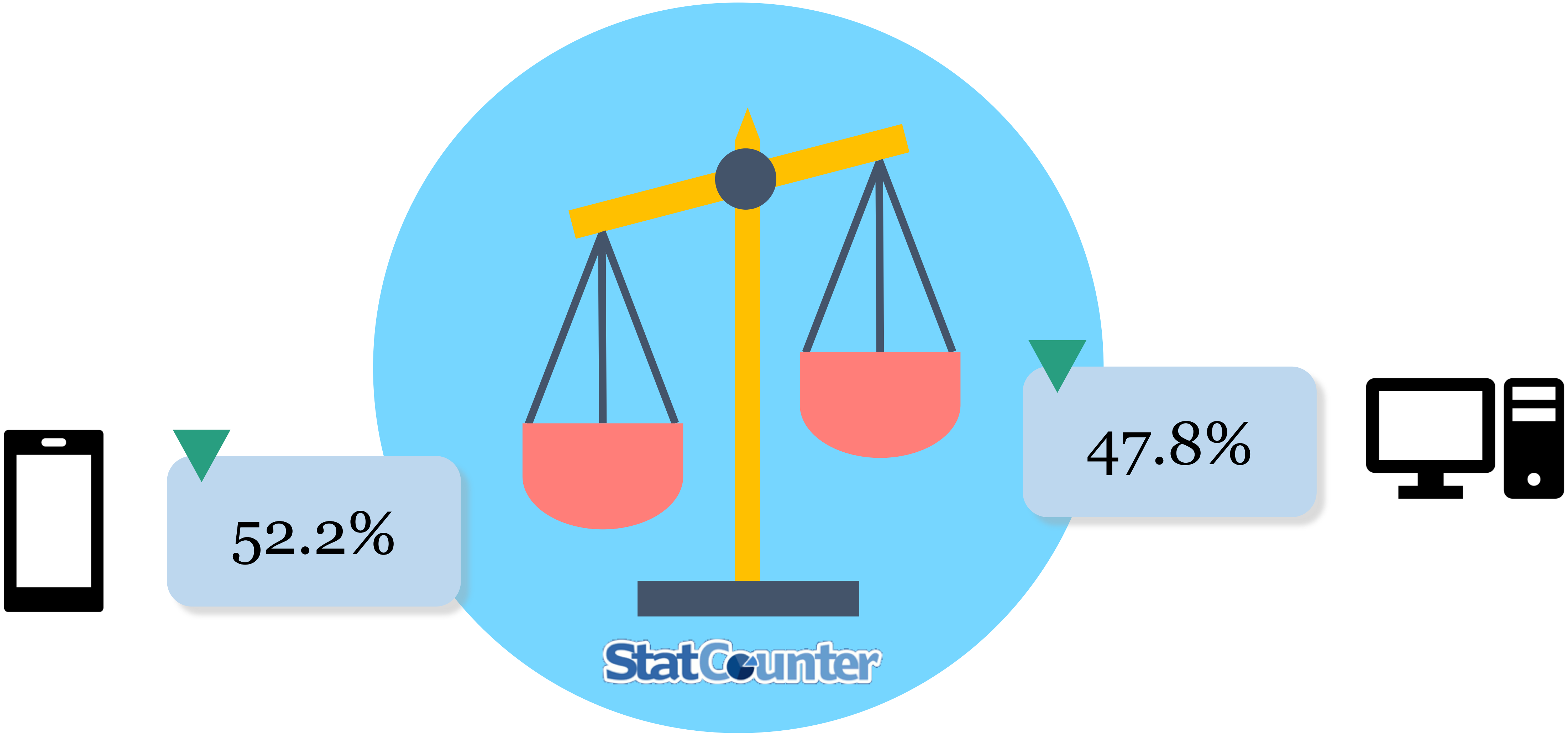
Time does not heal all wounds:
A longitudinal analysis of security-mechanism support in mobile browsers

Meng Luo, Pierre Laperdrix, Nima Honarmand, Nick Nikiforakis

NDSS Symposium
San Diego, February 25th, 2019

Stony Brook University
PragSec Lab

Traffic from mobile vs. desktop



Client-side attacks/preventions

ATTACKS

Cross-site scripting (XSS), cross-site request forgery (CSRF), SSL stripping, clickjacking, ...

DEFENSES

Isolating different origins:
Same-origin policy (SOP), Content Security Policy (CSP), X-Frame-Options, iframe sandboxing

Defending against man-in-the-middle attacks:
HTTP Strict-Transport-Security (HSTS), block-all-mixed-content, upgrade-insecure-requests

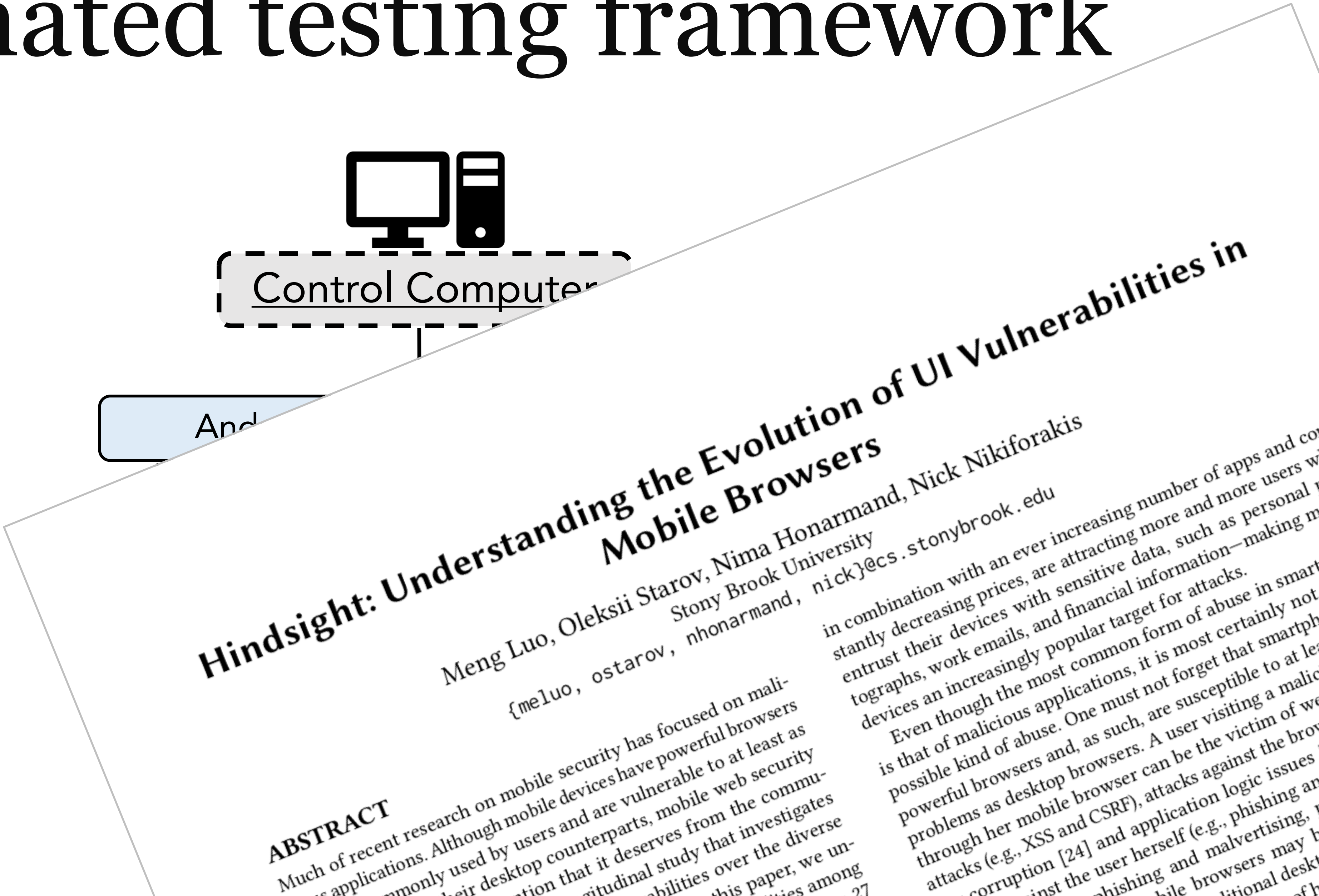
Protecting cookies and enhancing privacy:
Security-related flags for HTTP cookies, referrer policy

Motivation

- Hundreds of mobile browsers are available in the market – each advertising unique features
  - Built-in anti-tracking capabilities
  - Voice control
  - Increased performance
- Two users may get substantially different security guarantees depending on the browser they utilize
- No prior work on the adoption of security mechanisms in mobile browsers

Automated testing framework

Top 20
mobile browsers



Hindsight: Understanding the Evolution of UI Vulnerabilities in Mobile Browsers

Meng Luo, Oleksii Starov, Nima Honarmand, Nick Nikiforakis
Stony Brook University
{meluo, ostarov, nhonarmand, nick}@cs.stonybrook.edu

ABSTRACT
Much of recent research on mobile security has focused on mali-
... applications. Although mobile devices have powerful browsers
... monly used by users and are vulnerable to at least as
... their desktop counterparts, mobile web security
... tion that it deserves from the commu-
... gitudinal study that investigates
... abilities over the diverse
... this paper, we un-

in combination with an ever increasing number of apps and co...
stantly decreasing prices, are attracting more and more users w...
entrust their devices with sensitive data, such as personal ...
tographs, work emails, and financial information—making m...
devices an increasingly popular target for attacks.
Even though the most common form of abuse in smart...
is that of malicious applications, it is most certainly not...
possible kind of abuse. One must not forget that smartph...
powerful browsers and, as such, are susceptible to at lea...
problems as desktop browsers. A user visiting a malic...
through her mobile browser can be the victim of we...
attacks (e.g., XSS and CSRF), attacks against the brow...
corruption [24] and application logic issues...
against the user herself (e.g., phishing an...
phishing and malvertising...
mobile browsers may...

Security mechanisms

Category	Content	# tests
Same-origin Policy	DOM access, cookie scope, XMLHttpRequest and worker	33
Content Security Policy	Fetch (e.g. *script-src*) and other directives (e.g. *form-action*, *frame-ancestors* and *upgrade-insecure-requests*)	253
Cookie	Secure, HttpOnly and SameSite flags	11
Referrer policy	*no-referrer-when-downgrade* (default) and other values (e.g. *no-referrer*, *origin*, *same-origin* and *strict-origin*)	62
Iframe sandbox	JavaScript execution, form submission and top-level navigation	3
X-Frame-Options	Deny, SameOrigin and Allow-From values	30
Strict-Transport-Security	Basic and includeSubDomains value	2
X-Content-Type-Options	Script sniffing opt-out	1
total		**395**

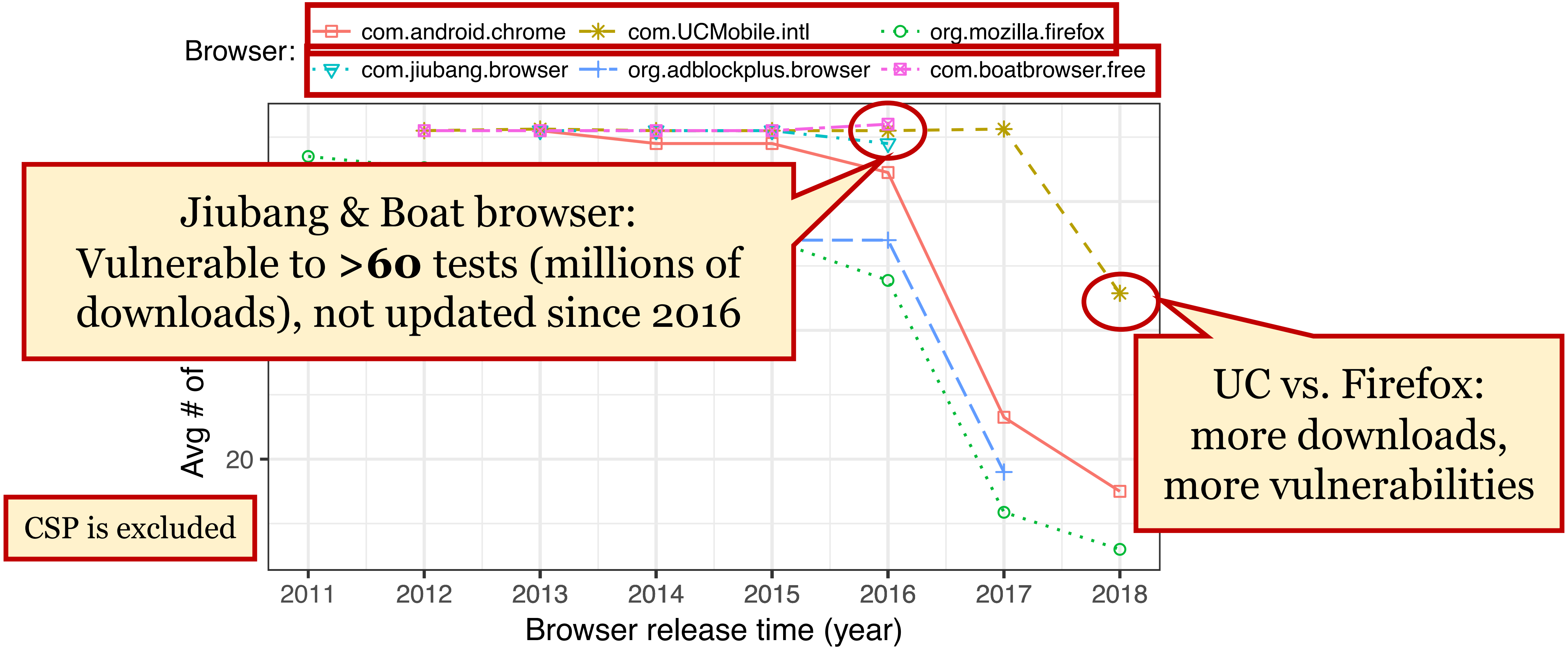
Evaluation

- 395 tests ✕ 351 browsers → >138K vulnerability reports
- Gauging the success/failure of a test,
  - Support -> "secure"
  - Lack of support -> "vulnerable"
- Analyzing vulnerability reports:
  - Longitudinal analysis capturing evolution of security-mechanism support
  - Dependencies between security-mechanism support and underlying Android system
  - Case studies of common vulnerabilities

Longitudinal analysis

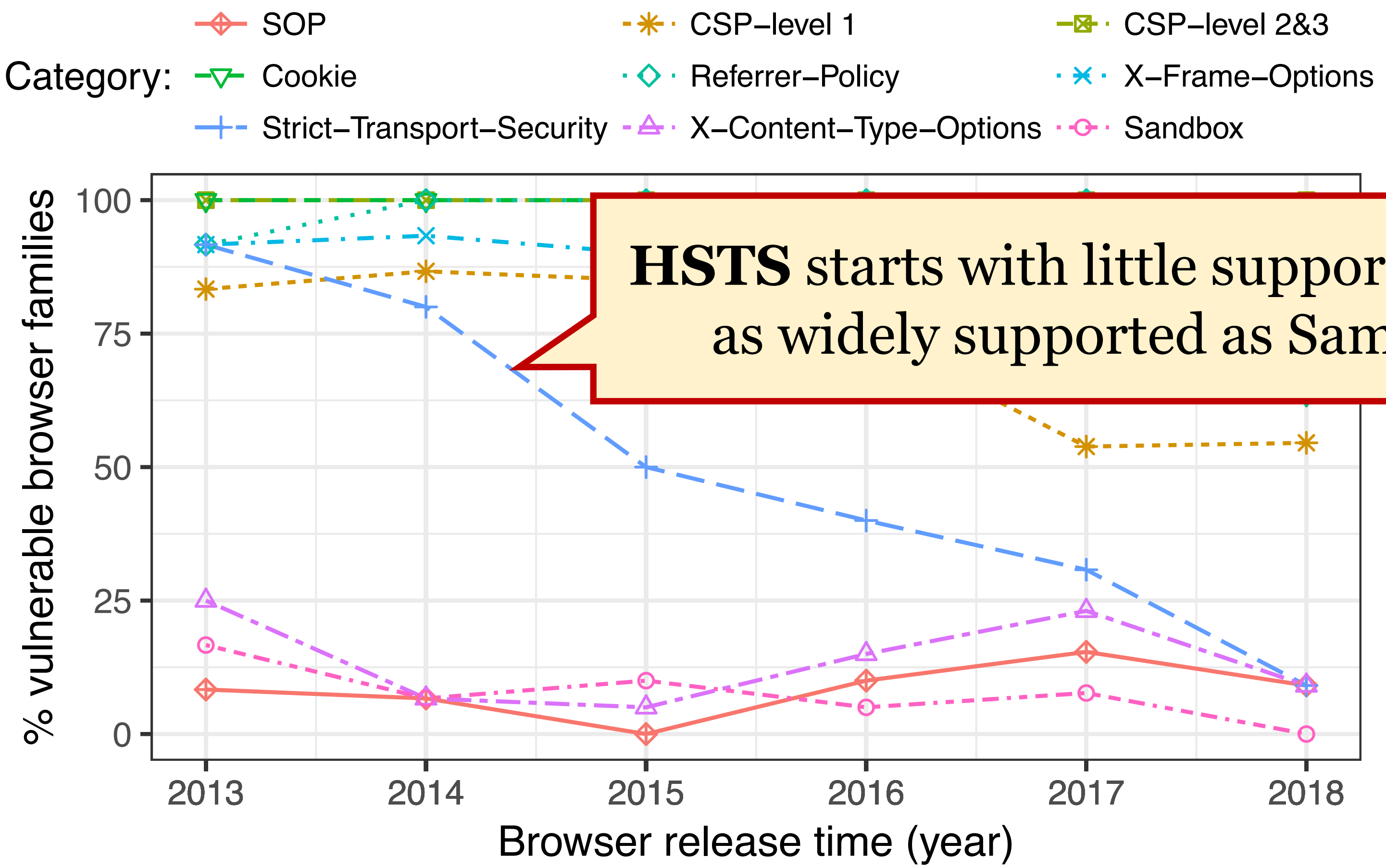
Adoption trend (1)

- Do browsers support more security mechanisms over time?



Adoption trend (2)

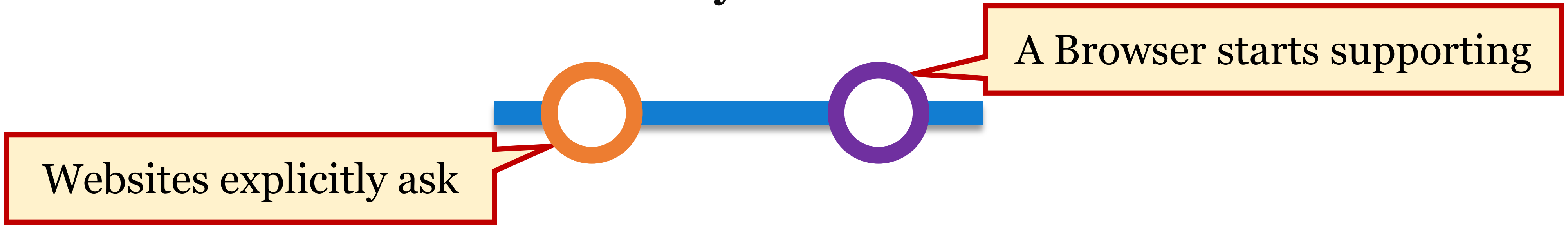
- Is there any difference in the support rate of security mechanisms?



HSTS starts with little support, but is eventually as widely supported as Same-origin Policy

Window of vulnerability (1)

- Time window of vulnerability



INTERNET ARCHIVE WaybackMachine

- Crawl snapshots of Alexa top 5K websites
- Obtain the earliest time when any website first utilizes a given security mechanism

Window of vulnerability (2)

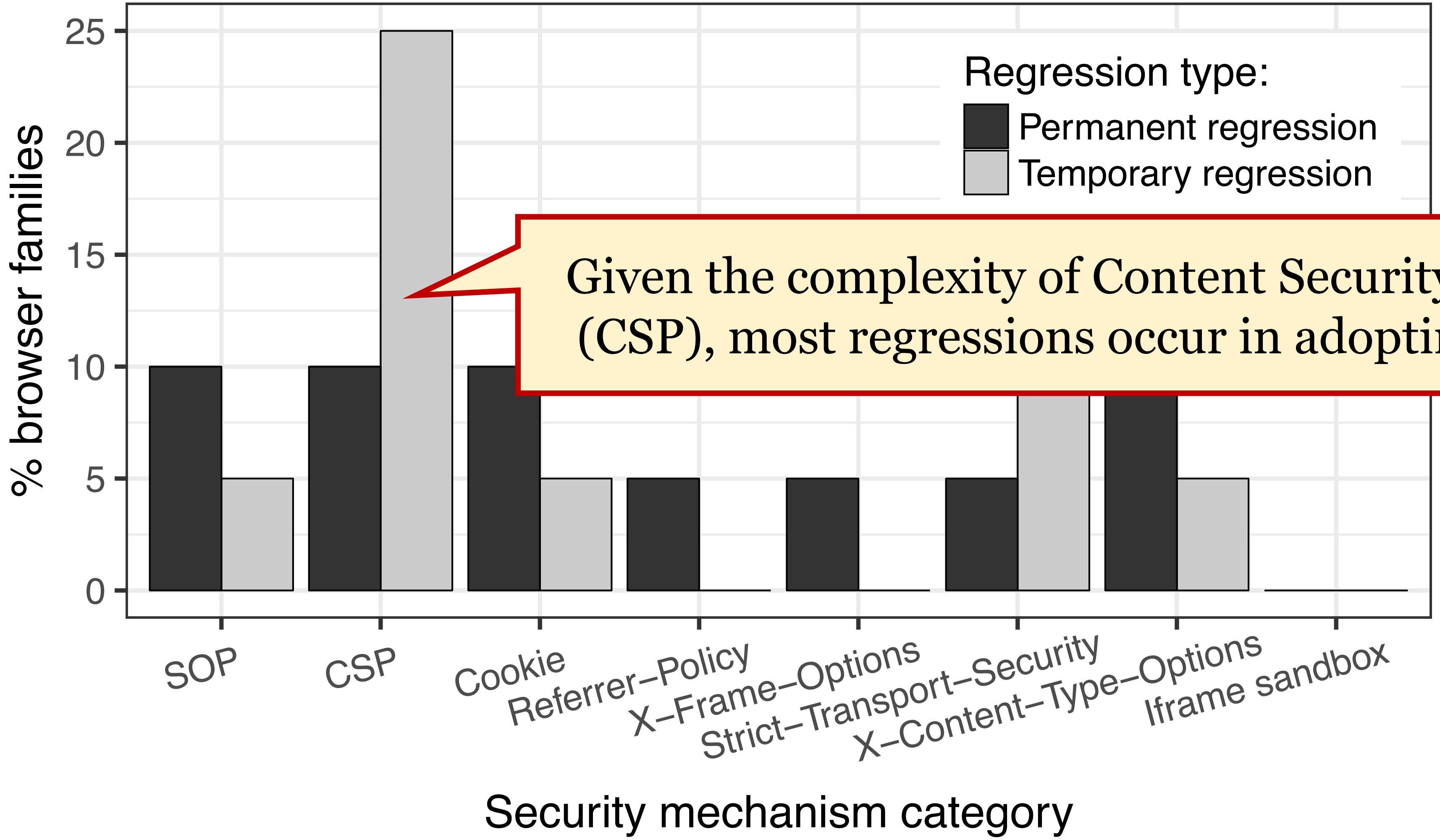
- The adoption of security mechanisms on mobile browsers is significantly slower than on desktop browsers

Security mechanisms	Website request	Chrome desktop	Firefox desktop	First mobile support	50% mobile support △	75% mobile support △
CSP	2011	2011	2011	2011	2014 (+3)	2015 (+4)
Cookie	<2011	2011	2009	2011	2013 (+2)	2014 (+3)
Referrer-Policy	2016	2012	2015	2015	2018 (+3)	Not yet
X-Frame-Options	<2011	2010	2010	2011	2013 (+2)	2014 (+3)
HSTS	<2011	2010	2011	2011	2015 (+4)	2016 (+5)
X-Content-Type-Options	<2011	2008	2011	2011	2013 (+2)	2015 (+4)

Security regression (1)

- Definition: a browser version stops supporting a security mechanism that was supported by an earlier version
- Types of security regressions:
  - temporary regression
  - permanent regression
- 55% browser families show regressions

Security regression (2)



Given the complexity of Content Security Policy (CSP), most regressions occur in adopting CSP

Security regression (3)

- Security regressions in top-5 browsers
- The red symbol indicates the presence of security regression

Mechanism	Chrome	UC	Firefox	Opera	Opera mini
[illegible]	[illegible]	✗	✓	✓	✓
[illegible]	[illegible]	✗	✗	✓	✓
Cookie	✓	✓	✓	✓	✓
Referrer-Policy	✓	✓	✓	✓	✓
X-Frame-Options	✓	✓	✓	✓	✗
[illegible]	[illegible]	[illegible]	[illegible]	✗	✓
[illegible]	[illegible]	[illegible]	[illegible]	✓	✓
Sandbox	✓	✓	✓	✓	✓

UC browser shows frequent security regressions

Opera stops supporting HSTS from v.15 to v.26 (1 year), thereby exposing users to MITM attacks

Studying latest mobile browsers

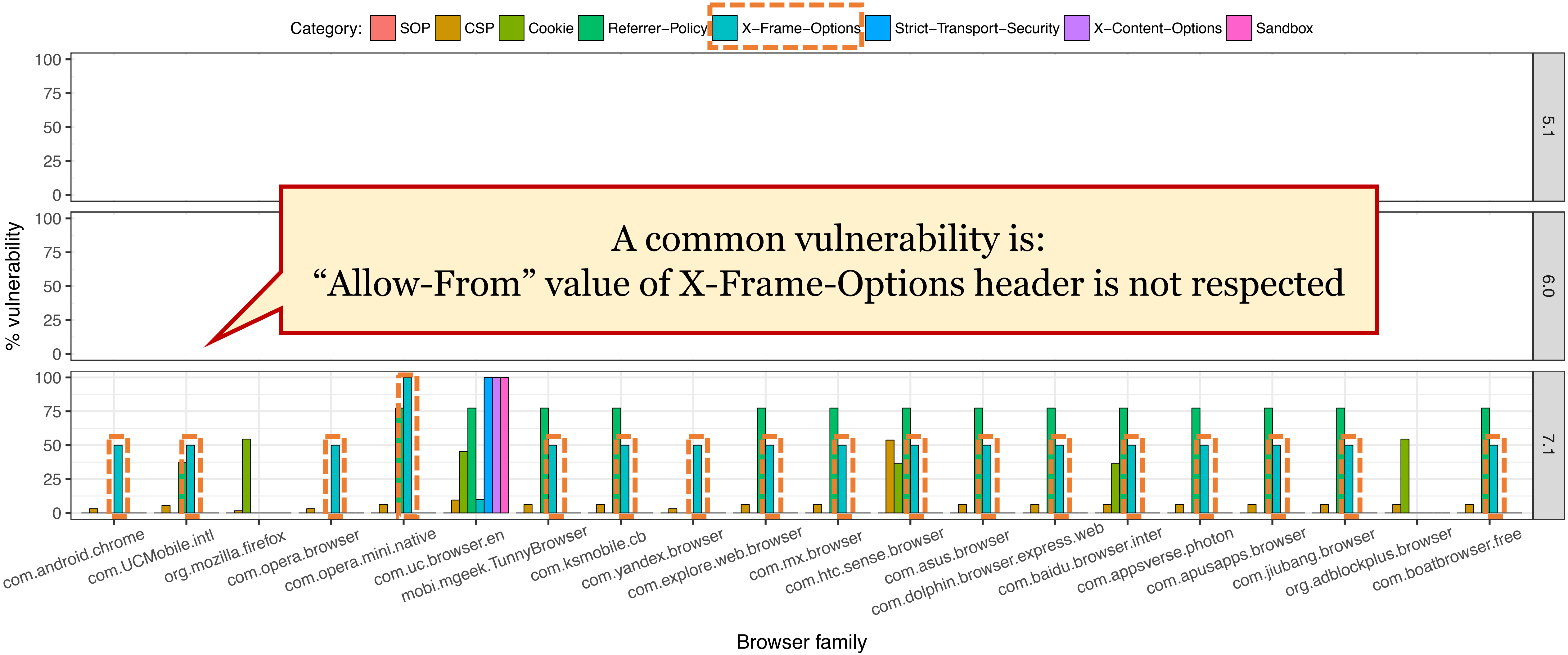
Relationship between security and underlying OS

- Many mobile browsers are developed using WebView
- The version of WebView depends on the underlying Android system
  - More recent versions support more security mechanisms
- Two users who utilize the latest version of the same browser, can experience vastly different levels of security.



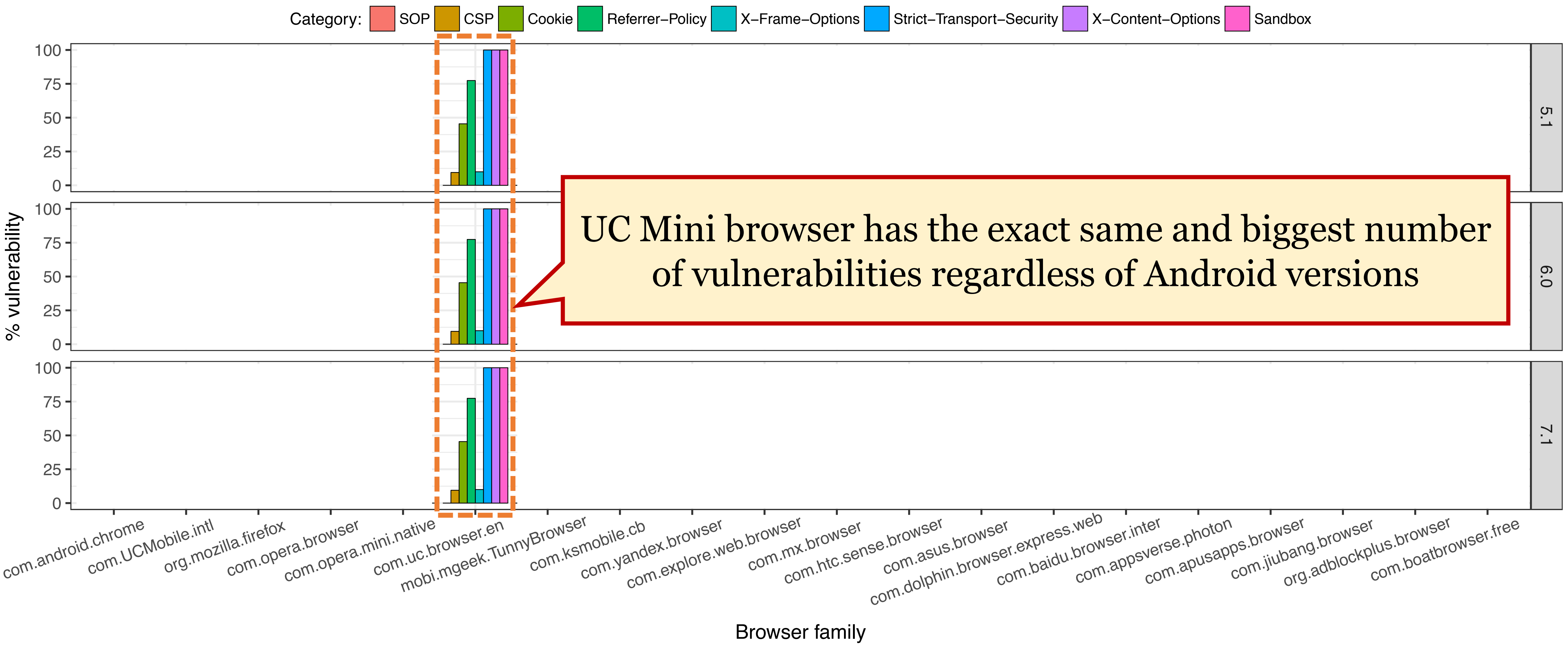
Vulnerability vs. Android version

Vulnerability vs. Android version



A common vulnerability is:
"Allow-From" value of X-Frame-Options header is not respected

Vulnerability vs. Android version



Case studies: the impact of browser vulnerabilities on web apps

Anti-clickjacking

- X-Frame-Options is an anti-clickjacking mechanism
- Chrome and WebView-based browsers discard the entire header when the "allow-from" directive is used
- 231/10,752 websites (from Alexa top 50K) that make use of the X-Frame-Options, are using "allow-from"
- 175/231 websites do not utilize CSP's "frame-ancestors"
  - Most of these websites have user accounts that could be abused through a clickjacking attack
  - American and Russian banking sites
  - Government sites of US, China, Brazil and India
  - Cloud instrumentation services

CSRF and SameSite

- CSRF occurs when a malicious website forges requests to perform unwanted actions on a vulnerable web app on behalf of an authenticated user
- The SameSite mechanism prevent browsers from attaching cookies to cross-site requests
- 93 websites (from Alexa top 50K) utilize SameSite, including an Italian bank and the biggest streaming platform
- Users of the above web apps may not be protected against CSRF if they use mobile browsers that do not support this mechanism
  - UC browser, Opera Mini

Conclusion

- Developed 395 tests to evaluate security-mechanism support
- 395 tests ✕ 351 mobile browsers ➜ 138K tests
- Analyzed the evolution of security-mechanism support
  - Quantified trends in the support of different security mechanisms
  - Multi-year window of vulnerability for all security mechanisms
  - Observed security regressions
- Security-mechanism support may depend on the underlying OS
- We need tools that can adapt to a user's mobile environment and employ different security mechanisms
  - E.g., upon detecting that 'allow-from' is not supported, emit 'frame-ancestors' CSP directive

THANK YOU

meluo@cs.stonybrook.edu