PrivacyTests.org

An open-source privacy audit of web browsers

Arthur Edelstein, February 25, 2022 Peterborough Linux User Group

In this talk

- My random walk in web privacy
- What is this all for, anyway?
- The high-level approach to PrivacyTests.org
- Privacy leaks and how to test them
- What I have learned
- Future work
- Discussion!

My random walk in web privacy

Developer for Tor Browser (2014-2018)

Senior Product Manager for Firefox Privacy and Security (2018-2021)

PrivacyTests.org (2021-Present)

Tor Project (Tor Browser developer)

First-Party Isolation

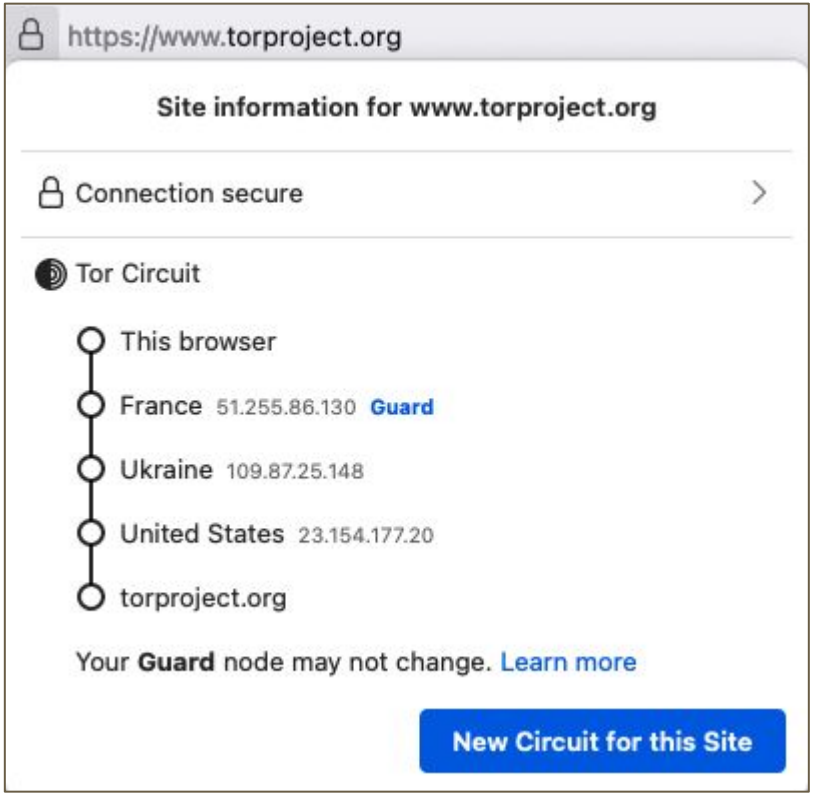
Fingerprinting Resistance

Stream Isolation

Tor Circuit Display

Tor Network Performance Improvements

Tor Uplift Project



Mozilla (PM for Firefox Privacy and Security)

2019: Enhanced Tracking Protection by Default, Fingerprinting Protections

2020: DNS-over-HTTPS rollout in the United States

Firefox 83: HTTPS-Only Mode

Firefox 85: Supercookie Protections

Firefox 86: Total Cookie Protection

Firefox 91: Enhanced Cookie Clearing

Wait, what is this all for?

- Some people care about privacy
- But many don’t know, don’t care or have given up hope
- Corporate politics is very challenging with lots of pushback!
- Is this effort worth it?
- Can we be more effective? How?

Let’s start with first principles!

Problem 0: Human suffering

War

Genocide

Tyranny

Oligarchy

Poverty

Discrimination

Corruption

Enabled by repression!

Problem 1: Repression prevents change

Commonly, oppressors tamper with information flows to preserve and enhance their own power:

- Deny access to information
- Deny freedom of public expression
- Violate privacy (freedom to operate)
- Restrict private communication

How? Mass surveillance!

Problem 2: Mass surveillance facilitates repression

Mass surveillance allows governments and corporations to watch, profile, discriminate against and manipulate virtually everyone...

...on the Web!

Problem 3: the Web is a major target of mass surveillance

- The Web is a primary means of modern reading, writing, communication and commerce.
- Most web browsers are heavily exposing most people to mass surveillance

Why? Bad incentives and lack of information

Problem 4: Hidden leaks and bad incentives

- Web browser privacy has not been a priority for most browsers, but people don’t know that
- Some major web browsers get their revenue from top trackers (Google, Bing), not from users
- Web browser privacy leaks are hidden, technical, and complex: meaning they are largely invisible to the public and, even invisible to engineers and managers in web companies

How web browsers facilitate surveillance

- Browsers allow websites you visit and the trackers embedded in them to gather your browsing history
- Browsers leak a lot of unnecessary data that can be used to track your browsing
- Browsers fail to encrypt your network connections, allowing your ISP or other network eavesdroppers to watch your browsing

How do we make browser makers accountable for fixing this?

PrivacyTests.org: an accountability effort

Try to hold web browsers accountable for protecting all web users from mass surveillance through:

- Detecting privacy leaks
- Monitoring those leaks over time
- Informing the public
- Informing and pressuring browser makers

Design of PrivacyTests.org

- Create open-source automated tests for known browser privacy leaks by mimicking surveillance/tracking techniques
- Run tests weekly across browsers and platforms
- Publish test results as a comparison between browsers
- Try to make tests and results easy to understand

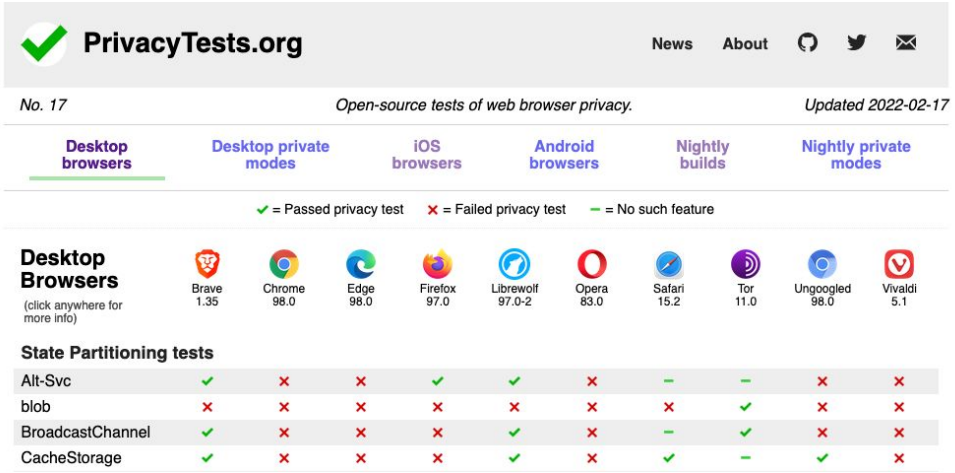
Building PrivacyTests.org

Proposed it at Tor 2018, and started (slowly) putting some tests together

Started working on it full time in August 2021

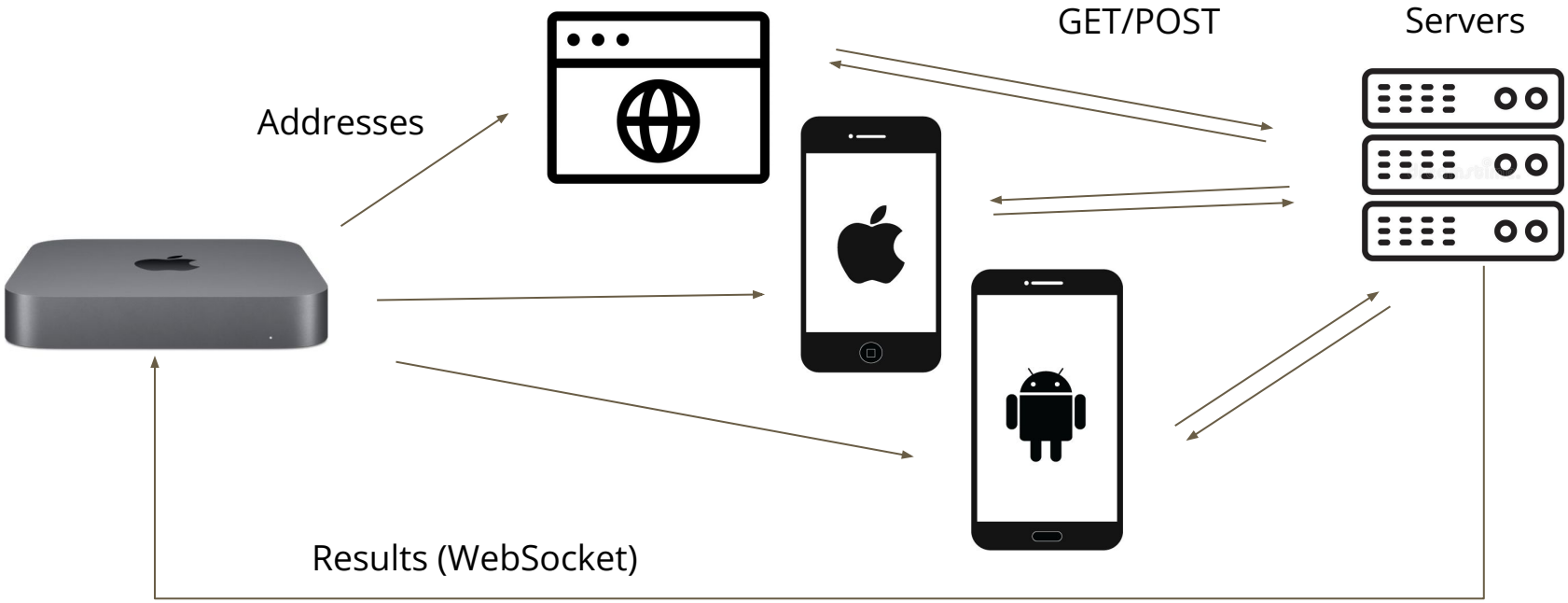
First launched in mid-October

Iterative – it's a work in progress!

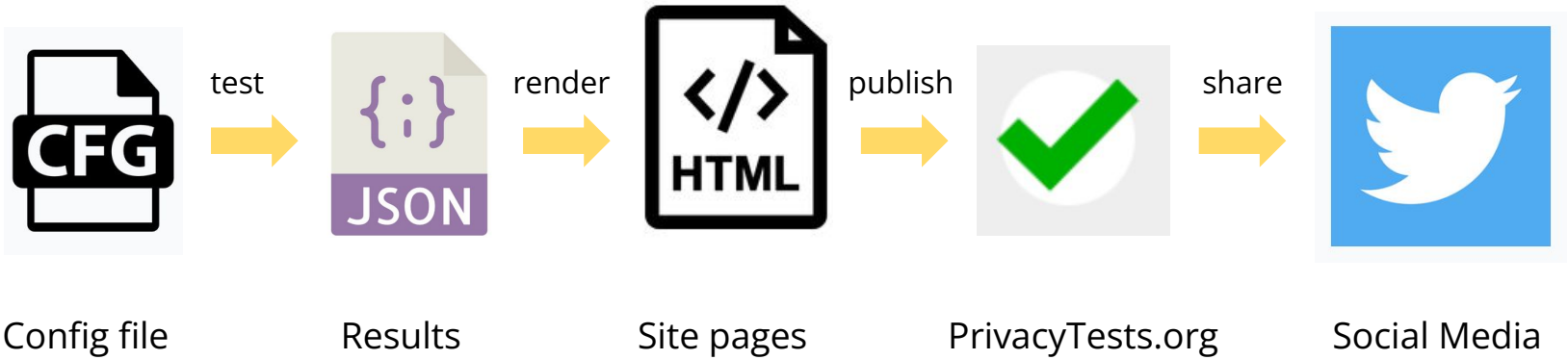


PrivacyTests.org browser testing framework

Almost all JavaScript (NodeJS and in-browser)



PrivacyTests.org data pipeline



PrivacyTests.org platform coverage

Platform	Device	Control	Browser windows
Desktop (MacOS)	Mac Mini	Browser command-line arguments	Regular, Private, Nightly, Nightly Private
Android	Samsung phone	Appium	Regular, Private
iOS	iPhone	Appium	Regular, Private

Kinds of browser privacy leaks

- ➡ Stateful tracking (e.g. Cookies)
- ➡ IP address (Server reflexive address)
- ➡ Network leaks (DNS, OCSP, SNI, HTTP)
- ➡ Fingerprinting (Fonts, Screen size, WebRTC)
- ➡ URL Query Parameters

  Forms/password stealing

  Telemetry

  Portal/Search engine history

Stateful tracking (1)

Stateful tracking (2)

User agent state

A likely inexhaustive enumeration of user agent state and ongoing standards activity:

- Cookies
- Network state:
  - HTTP cache (standardized in Fetch)
  - HTTP connections (standardized in Fetch)
    - Also consider speculative connections (unclear where these are created in standards, but if done through Fetch it would be correct)
  - WebSocket connections (whatwg/fetch #1243)
  - WebRTC connections (w3c/webrtc-pc #2613)
  - WebTransport connections (w3c/webtransport #128)
  - DNS
  - HTTP authentication
  - Alt-Svc
  - Fonts
  - HSTS
  - TLS client certificates
  - TLS session identifiers
  - HPKP
  - OCSP
  - Intermediate CA cache
  - Prefetch
  - Preconnect
  - CORS-preflight cache (standardized in Fetch)

Client-Side Storage Partitioning

A Work Item of the Privacy Community Group.

- Storage (whatwg/storage #88, whatwg/storage #90):
  - Indexed DB
  - Cache API
  - `localStorage`
  - `sessionStorage`
  - `BroadcastChannel` (whatwg/html #5803)
  - Shared workers
  - Service workers
  - Web Locks
- Web Authentication
- WebRTC's `deviceId` (w3c/mediacapture-main #675)
- Blob URL store (w3c/FileAPI #153)
- HTML Standard's list of available images
- `window.name` (standardized in HTML)
- Browsing context group's agent cluster map (only observable with popups)
- Permissions (Permissions Policy largely allows these to be disabled by default when the top-level site is not equal to the current site and require explicit delegation in such cases)
  - Persistent storage (whatwg/storage #87)
  - Notifications (whatwg/notifications #177)
- WebGL and WebGPU's cache of compiled shaders and pipelines (standardized by highlighting the risk in the security/privacy consideration section as the caches are only observable through timing)
- Non-standardized features:
  - Credentials (username and password storage)
  - Form autofill data storage
  - Per-site user preferences
  - Favicon cache
  - Page info media previews
  - Save Page As

Stateful tracking (3)

Desktop Browsers (click anywhere for more info)	Brave 1.35	Chrome 98.0	Edge 98.0	Firefox 97.0	Librewolf 97.0-2	Opera 83.0	Safari 15.2	Tor 11.0	Ungoogled 98.0	Vivaldi 5.1
State Partitioning tests										
Alt-Svc	✗	✗	✗	✓	✓	✗	–	–	✗	✗
blob	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗
BroadcastChannel	✓	✗	✗	✗	✓	✗	–	✓	✗	✗
CacheStorage	✓	✗	✗	✗	✓	✗	✓	–	✓	✗
cookie (HTTP)	✓	✗	✗	✗	✓	✗	✓	✓	✓	✗
cookie (JS)	✓	✗	✗	✗	✓	✗	✓	✓	✓	✗
CSS cache	✓	✓	✓	✓	✓	✗	✓	✓	✓	✗
favicon cache	✓	✗	✗	✓	✓	✗	✗	✓	✗	✗
fetch cache	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
font cache	✓	✗	✗	✓	✓	✗	✓	✓	✓	✗
H1 connection	✓	✗	✗	✓	✓	✗	✓	✓	✗	✗
H2 connection	✓	✗	✗	✓	✓	✗	✓	✓	✗	✗
H3 connection	✓	✗	✗	✓	✓	✗	–	–	✗	✗
HSTS cache	✗	✗	✗	✓	✓	✗	✓	✓	✗	✗
iframe cache	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
image cache	✓	✗	✗	✓	✓	✗	✓	✓	✓	✗
indexedDB	✓	✗	✗	✗	✓	✗	✓	–	✓	✗
localStorage	✓	✗	✗	✗	✓	✗	✓	✓	✓	✗
locks	✓	✗	✗	✗	✓	✗	–	–	✓	✗
prefetch cache	✓	✓	✗	✓	–	✗	–	✓	✓	✗
ServiceWorker	✓	✓	✓	✗	✓	✓	✓	–	✓	✓
SharedWorker	✓	✗	✗	✗	✓	✗	–	✓	✓	✗
TLS Session ID	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Web SQL Database	✓	✓	✓	–	–	✓	–	–	✓	✓
XMLHttpRequest cache	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

IP Address tracking (1)

A big problem!

Don't Count Me Out: On the Relevance of IP Address in the Tracking Ecosystem

Vikas Mishra
Inria / Univ. Lille
vikas.mishra@inria.fr

Pierre Laperdrix
CNRS / Univ. Lille / Inria
pierre.laperdrix@univ-lille.fr

Antoine Vastel
Univ. Lille / Inria
antoine.vastel@inria.fr

Walter Rudametkin
Univ. Lille / Inria
walter.rudametkin@univ-lille.fr

Romain Rouvoy
Univ. Lille / Inria / IUF
romain.rouvoy@univ-lille.fr

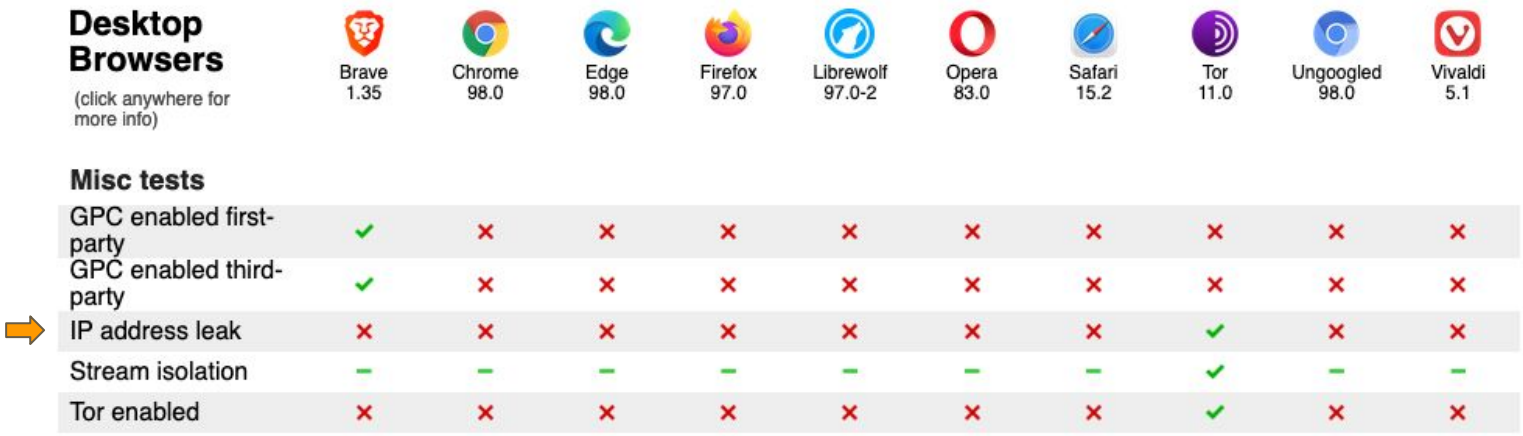
Martin Lopatka
Mozilla
mlopatka@mozilla.com

We present an analysis of 34,488 unique public IP addresses collected from 2,230 users over a period of 111 days and we show that IP addresses remain a prime vector for online tracking. 87 % of participants retain at least one IP address for more than a month and 45 % of ISPs in our dataset allow keeping the same IP address for more than 30 days. Furthermore, we also detect the presence of cycles of IP addresses in a user's history and highlight their potential to be abused to infer traits of the user behaviour, as well as mobility traces. Our findings paint a bleak picture of the current state of online tracking at a time where IP addresses are overlooked compared to other techniques like cookies or fingerprinting.

https://hal.inria.fr/hal-02435622/document

IP Address tracking (2)

- Tor
- VPNs/proxies



Desktop Browsers (click anywhere for more info)	Brave 1.35	Chrome 98.0	Edge 98.0	Firefox 97.0	Librewolf 97.0-2	Opera 83.0	Safari 15.2	Tor 11.0	Ungoogled 98.0	Vivaldi 5.1
Misc tests										
GPC enabled first-party	✔	✘	✘	✘	✘	✘	✘	✘	✘	✘
GPC enabled third-party	✔	✘	✘	✘	✘	✘	✘	✘	✘	✘
IP address leak	✘	✘	✘	✘	✘	✘	✘	✔	✘	✘
Stream isolation	–	–	–	–	–	–	–	✔	–	–
Tor enabled	✘	✘	✘	✘	✘	✘	✘	✔	✘	✘

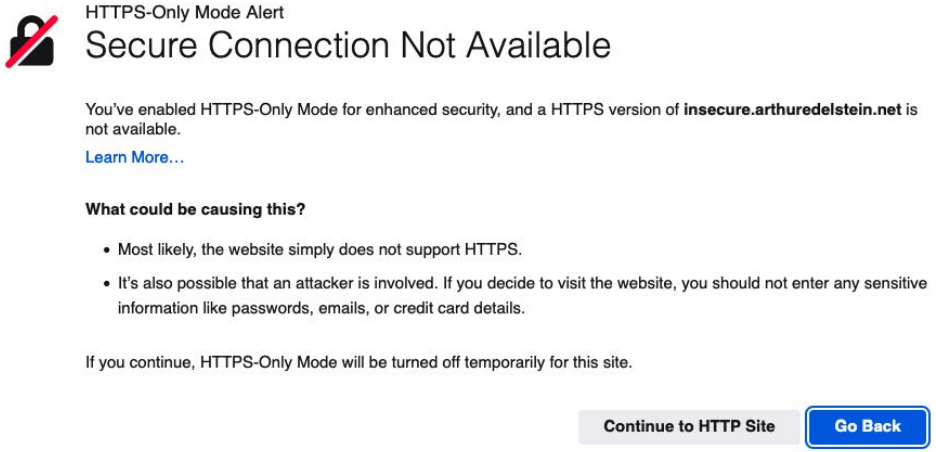
Network leaks (1)

- HTTP → HTTPS

HTTPS Is Actually Everywhere

BY ALEXIS HANCOCK | SEPTEMBER 21, 2021

https://www.eff.org/deeplinks/2021/09/https-actually-everywhere



Network leaks (2)

Desktop Browsers (click anywhere for more info)	Brave 1.35	Chrome 98.0	Edge 98.0	Firefox 97.0	Librewolf 97.0-2	Opera 83.0	Safari 15.2	Tor 11.0	Ungoogled 98.0	Vivaldi 5.1
HTTPS tests										
Insecure website	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗
Upgradable address	✗	✗	✗	✗	✓	✗	✗	✓	✗	✗
Upgradable hyperlink	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗
Upgradable image	✓	✓	✓	✗	✓	✓	✗	✗	✓	✓
Upgradable script	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Fingerprinting

Desktop Browsers (click anywhere for more info)	Brave 1.35	Chrome 98.0	Edge 98.0	Firefox 97.0	Librewolf 97.0-2	Opera 83.0	Safari 15.2	Tor 11.0	Ungoogled 98.0	Vivaldi 5.1
Fingerprinting resistance tests										
Media query screen height	✗	✗	✗	✗	✓	✗	✗	✓	✗	✗
Media query screen width	✗	✗	✗	✗	✓	✗	✗	✓	✗	✗
outerHeight	✗	✗	✗	✗	✓	✗	✗	✓	✗	✗
screen.height	✗	✗	✗	✗	✓	✗	✗	✓	✗	✗
screen.width	✗	✗	✗	✗	✓	✗	✗	✓	✗	✗
screenX	✗	✗	✗	✗	✓	✗	✗	✓	✗	✗
screenY	✗	✗	✗	✗	✓	✗	✗	✓	✗	✗
System font detection	✗	✗	✗	✗	✓	✗	✓	✓	✗	✗

Tracker query parameters (1)

https://www.vrbo.com/travel/staycation?utm_campaign=vrbo:prog:usa-en:t:g:xxx:iroas&utm_medium=display&utm_source=dbm&utm_content=a:ban:dbm:xxx:pro:xxx:lake:xxx&utm_term=20193083|252013460|133520644|448385033&dclid=CNrN5PDpm_YCFRQTfQodiRAJuA

Tracker query parameters (2)

Desktop Browsers (click anywhere for more info)	Brave 1.35	Chrome 98.0	Edge 98.0	Firefox 97.0	Librewolf 97.0-2	Opera 83.0	Safari 15.2	Tor 11.0	Ungoogled 98.0	Vivaldi 5.1
Tracking query parameter tests										
__hsfp	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
__hssc	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
__hstc	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
__s	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
_hsenc	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
_openstat	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
dclid	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
fbclid	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
gclid	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
hsCtaTracking	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
igshid	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
mc_eid	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
mkt_tok	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗
ml_subscriber	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
ml_subscriber_hash	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
msclkid	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
oly_anon_id	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
oly_enc_id	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
rb_clickid	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
s_cid	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
vero_conv	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
vero_id	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
wickedid	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
yclid	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗

Tracking cookie protection (new!)

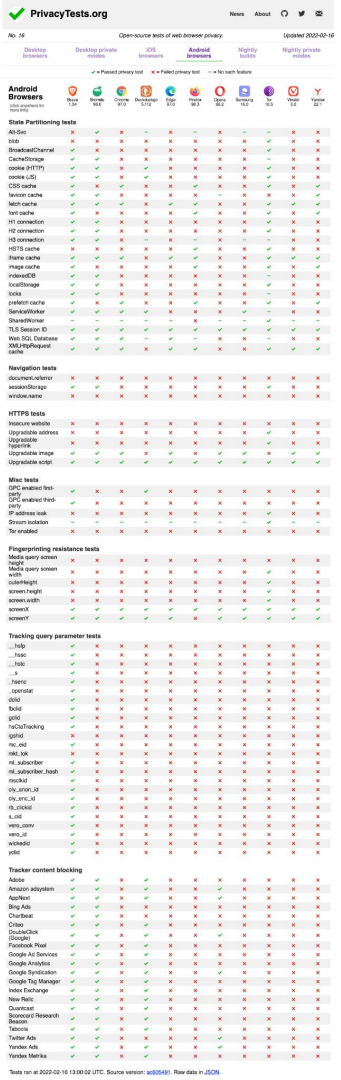
Desktop Browsers (click anywhere for more info)	Brave 1.35	Chrome 98.0	Edge 98.0	Firefox 97.0	Librewolf 97.0-2	Opera 83.0	Safari 15.2	Tor 11.0	Ungoogled 98.0	Vivaldi 5.1
Tracking cookie protection										
Adobe	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓
Adobe Audience Manager	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓
Amazon adsystem	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓
AppNexus	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓
Bing Ads	✓	✗	✗	✗	✓	✓	✓	✓	✓	✗
Chartbeat	✓	✗	✗	✓	✓	✓	✓	✓	✓	✓
Criteo	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓
DoubleClick (Google)	✓	✗	✓	✓	✓	✗	✓	✓	✓	✓
Facebook pixel	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓
Google (third-party ad pixel)	✓	✗	✓	✗	✓	✓	✓	✓	✓	✓
Google Analytics	✓	✗	✗	✓	✓	✗	✓	✓	✓	✓
Google Tag Manager	✓	✗	✗	✗	✓	✓	✓	✓	✓	✓
Index Exchange	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓
New Relic	✓	✗	✗	✓	✓	✓	✓	✓	✓	✓
Quantcast	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓
Scorecard Research Beacon	✓	✗	✗	✓	✓	✓	✓	✓	✓	✓
Taboola	✓	✗	✓	✓	✓	✗	✓	✓	✓	✓
Twitter pixel	✓	✗	✗	✗	✓	✓	✓	✓	✓	✗
Yandex Ads	✓	✗	✓	✗	✓	✓	✓	✓	✓	✓

Default search engines:

- Edge: Bing
- Firefox: Google
- Opera: Google
- Vivaldi: Microsoft Bing

What have I learned so far?

- All 3 browser engines (Chromium, WebKit, Gecko) have already been hardened for privacy in some browsers: no excuses!
- Some browser makers are making a good effort on privacy (Brave, DuckDuckGo, LibreWolf, Safari, Tor)
- Others are dragging their feet
- Browser testing is finicky, especially on mobile
- Continuous monitoring is hard
- Lots of people are really interested in browser privacy!



Future work

- More network leak tess (DoH, SNI, OCSP)
- More fingerprinting tests
- Telemetry tests
- Disk forensic tests
- “Test my Browser” feature
- Browser Extensions
- Mobile apps (based on browsers)
- More browsers

Acknowledgments

Steven Englehardt

Sukhbir Singh

Peter Snyder

John Wilander

Many people on github and twitter

Thank you!

Reach me at:
contact@privacytests.org

@privacytests (Twitter)

https://github.com/arthuredelstein/privacytests.org