Brave, Fingerprinting and Privacy on the Web
Me (the early years)

- Grew up in Chicago
  ...actual Chicago

- Law school, then freelance web design
  Started: Anchorage, AK
  Ended: Judge Judy Show invitation

- PhD in Computer Science
  University of Illinois at Chicago
Me, now

- **Privacy Researcher at Brave**
  Research to improve privacy in the browser

- **Co-Chair of PING**
  Privacy reviews of new web standards

- **Academic Collaborator**
  “Pure” research
Brave in a Slide

- Privacy focused
- Alternative funding model for the web
- Research and engineering focused
- Browsers and infrastructure now, more to come…
Overview

1. Why websites track (and how much)
2. “Classic” tracking
3. Fingerprinting / “passive tracking”
4. Fingerprinting counter measures
5. Anti-finger printing exercise
6. Privacy protections in Brave
7. Wrapping up
Overview

1. Why websites track (and how much)
2. “Classic” tracking
3. Fingerprinting / “passive tracking”
4. Fingerprinting counter measures
5. Anti-finger printing exercise
6. Privacy protections in Brave
7. Wrapping up
Why Does Tracking Exist?
Welcome the The "First" Banner Ad

Yes, this site is supposed to look this way. After all, this is what most web pages looked like back on October 27, 1994 -- the day that Wired Magazine flipped the switch on its first website, hotwired.com, starting a revolution in web content and advertising that still reverberates today.

This site is dedicated to showing off one of the ads that ran on that site. No, it wasn't the "first" as there were a handful of other ads that ran on various sections of hotwired.com. This site is also here to tell the story of how that ad came to be, how it succeeded beyond anything we had imagined, and how we tried to set an example for how corporations could communicate with their audiences.

This site launched on October 27, 2014. It is being constantly updated, so please check back again soon for more. In the meantime, get started by clicking your mouse in the banner ad above explore these other options:
Gratuitous use of frames is a common mistake of web designers.

Many older browsers do not support frames. They disrupt the flow of the website and can be difficult to anticipate where a page may appear when a link is clicked. Click here for an example of a frames page which is opening in the wrong window. Use your browser's 'Back' button to escape.

Check out these links to websites whose opinions about frames is self-evident:

The "I Hate Frames" Frames Page
Another I Hate Frames Page
The International I Hate Frames Club
Why Frames Suck (Most of the Time)
Identify “expensive” people here

Pay a little to advertise to them here
How Target Figured Out A Teen Girl Was Pregnant Before Her Father Did

Every time you go shopping, you share intimate details about your consumption patterns with retailers. And many of those retailers are studying those details to figure out what you like, what you need, and which coupons are most likely to make you happy. Target, for example, has figured out how to data-mine its way into your womb, to figure out whether you have a baby on the way long before you need to start buying diapers.
But how much…
Online Tracking:  
A 1-million-site Measurement and Analysis

Steven Englehardt  
Princeton University  
ste@cs.princeton.edu

Arvind Narayanan  
Princeton University  
arvindn@cs.princeton.edu

ABSTRACT
We present the largest and most detailed measurement of online tracking conducted to date, based on a crawl of the top 1 million websites. We make 15 types of measurements on each site, including stateful (cookie-based) and stateless (fingerprinting-based) tracking, the effect of browser privacy tools, and the exchange of tracking data between different sites (“cookie syncing”). Our findings include multiple sophisticated fingerprinting techniques never before measured in the wild.

This measurement is made possible by our open-source web privacy measurement tool, OpenWPM\(^1\), which uses an automated version of a full-fledged consumer browser. It supports parallelism for speed and scale, automatic recovery from failures of the underlying browser, and comprehensive browser instrumentation. We demonstrate our platform’s strength in enabling researchers to rapidly detect, quantify, and characterize emerging online tracking behaviors.

1. INTRODUCTION
Web privacy measurement — observing websites and services to detect, characterize and quantify privacy-impacting behaviors — has repeatedly forced companies to improve to resort to a stripped-down browser \(^{31}\) (a limitation we explore in detail in Section 3.3). (2) We provide comprehensive instrumentation by expanding on the rich browser extension instrumentation of FourthParty \(^{33}\), without requiring the researcher to write their own automation code. (3) We reduce duplication of work by providing a modular architecture to enable code re-use between studies.

Solving these problems is hard because the web is not designed for automation or instrumentation. Selenium\(^2\) the main tool for automated browsing through a full-fledged browser, is intended for developers to test their own websites. As a result it performs poorly on websites not controlled by the user and breaks frequently if used for large-scale measurements. Browsers themselves tend to suffer memory leaks over long sessions. In addition, instrumenting the browser to collect a variety of data for later analysis presents formidable challenges. For full coverage, we’ve found it necessary to have three separate measurement points: a network proxy, a browser extension, and a disk state monitor. Further, we must link data collected from these disparate points into a uniform schema, duplicating much of the browser’s own internal logic in parsing traffic.

A large-scale view of web tracking and privacy.
In this paper we report results from a January 2016 mea-
But how much... 
a lot / too much
Overview

1. Why websites track
2. “Classic” tracking
3. Fingerprinting / “passive tracking”
4. Fingerprinting counter measures
5. Anti-finger printing exercise
6. Privacy protections in Brave
7. Wrapping up
• Javascript
• DOM / Initial Web API
• Netscape
• Firefox
• Brave + BAT
Web 0.0

good-site.com
Web 0.0

GET /home.html

<html>…</html>

good-site.com
Web 0.0

GET /home.html
<html>…</html>

GET /other.html
<html>…</html>

good-site.com
Web 0.0
Birth of the Tracking

- **Problem**
  - Authentication?
  - Can’t log in every time
  - HTTP auth is terrible and limited

- **Solution**
  - Server gives token to user
  - User returns it on requests
  - Aka “cookies”
Web 0.0
good-site.com
Web 0.0

GET /home.html
<html> + id=XYZ
Web 0.0

GET /home.html
<html> + id=XYZ

GET /secret.html + id=XYZ
<html>...

good-site.com
Web 0.0

GET /home.html
/html> + id=XYZ

GET /secret.html + id=XYZ
/html>...

GET /secret.html

good-site.com
But in the meantime...
cat-cuties.com

kozy-kittens.com

cookies, cookies everywhere...
Cookies
+ 3p Resources

---

Tracking
Site A

Id=abc

Tracking Site
Site A

Site B

Tracker knows the same person visited A + B
Tracking Patient Zero

- The internet’s “original sin”
  - cross origin resources
  - 3p cookies
  - or both...

- “I invented Javascript and 3p script, and I’ve been making up for it ever sense...” (paraphrase)
“Ever-Cookies”

- Some browsers started fighting back
  Brave, Safari, Firefox, extensions...

- Trackers fought back
  Moving IDs information out of cookies, to other location

- Long list of locations
  - Local and Session Storage
  - HSTS
  - Cache (etags, Cache API, etc)
  - Plugins
  - many many many more…
Overview

1. Why websites track (and how much)
2. “Classic” tracking
3. Fingerprinting / “passive tracking”
4. Fingerprinting in web standards
5. Fingerprinting counter measures
6. Anti-finger printing exercise
7. Wrapping up
Fingerprinting, what’s diff?

• Classic tracking
  - Website stores an id on the client
  - The client returns the id to the server (cookie or JS)
  - The id is what allows re-identification

• Fingerprinting / passive tracking
  - Website finds things different about each visitor
  - That difference allows re-identification
Fingerprinting, how

- Large number of semi-identifiers
  - Browser size
  - Extra fonts
  - Audio hardware
  - Video hardware
  - Installed plugins
  - Color depth
  - etc etc etc…
All browser users
All browser users:
3 billion people

You
1 person in 3 billion
All browser users:
3 billion people
All browser users: 3 billion people

Windows users
All browser users:
3 billion people
All browser users: 3 billion people
All browser users: 3 billion people

Using ad blocker
All browser users: 3 billion people

You 1 person in 100
Succeeding at Fingerprinting

1. **Breath of fingerprints**
   Large number of semi-identifiers

2. **Depth of fingerprints**
   How uniquely each identifier can… identify
Breath (examples)

- User agent string
- Installed fonts
- Canvas / WebGL
- Hardware (many)
- Height / width
User Agent String

• History of the Browser user-agent string
  https://webaim.org/blog/user-agent-string-history/

• Katamari-Damacy of identifiers

• Brave / Chrome
  Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_0) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.50 Safari/537.36

• Easy to extract
  - navigator.userAgent
  - User-Agent:
Installed Fonts

- Three categories of fonts
  - System
  - Local
  - Web

- “Local” is the tricky part
  - Office
  - Photoshop
  - Goofery

- Easy to extract
  - plugins
  - css + span + width
['Andale Mono', 'Arial', 'Arial Black', 'Arial Hebrew', 'Arial MT', 'Arial Narrow', 'Arial Rounded MT Bold']...
['Andale Mono', 'Arial', 'Arial Black', 'Arial Hebrew', 'Arial MT', 'Arial Narrow', 'Arial Rounded MT Bold'...]

<span>Example</span>
For each font...

Fingerprinter

Example
Fingerprinter

For each font...

```javascript
for (const fontName of fonts) {
  // 1. Apply font to span
  // 2. Measure width of span
  // 3. If it changes, user has font...
}

<span>Example</span>
```
Canvas / WebGL

- **Pixel Perfect:** Fingerprinting Canvas in HTML5
  Keaton Mowery and Hovav Shacham

- **Drawling APIs**
  e.g. Drawing lines / shapes

- **Standardized, but subtle differences**

- **Easy to extract**
  - Create canvas
  - Do some drawing
  - toDataURL()
Hovav Shacham

The Geometry of Innocent Flesh on the Bone: Return-into-libc without Function Calls
Hardware Identifiers

• Many Web APIs leak capabilities
  - number of cores (HTML)
  - number of audio channels (Web Audio API)
  - num shaders and similar (WebGL API)
  - device memory (Device Memory API)
  - network (WebRTC, Network status API)

• Semi identifying

• Easy to extract
  - All browsers have subset of the above
  - Most platforms have no permissions
Height / Width

?
Height / Width

- What does it mean?
- How to extract w/ JavaScript?
- How to extract w/o JavaScript?
- Brb 5 min (Go go go go go go!)
Fingerprinting Depth

https://panopticlick.eff.org/
Fingerprinting in Practice

- Needs to be in a database...
- Hash each endpoint
- Hash each value into a single identifier...
- Nice implication: “poisionability”...
Exercise

- Read fingerprint2.js
- List as many finger-printing approaches as possible
- Understand how they’re carried out
- Predict which are most identifying
Overview

1. Why websites track (and how much)
2. “Classic” tracking
3. Fingerprinting / “passive tracking”
4. Fingerprinting counter measures
5. Anti-finger printing exercise
6. Privacy protections in Brave
7. Wrapping up
Fingerprinting Countermeasures

- Remove the functionality
- Make the functionality consistent
- Restrict access (permissions, 1p vs 3p, user gesture, etc)
- Noise
- “Privacy budget”
Remove the functionality

- Delete JS end point
- Remove the HTTP header
- Remove the runtime capability
Consistency

- Make every browser return the same value
- ... or, most?
- Not that diff in practice from “removing”
Restrict access

- Permission prompt
- User gesture
- 1p vs. 3p
- “Site engagement”
Noise

- Stenography
- Make different every time

Original  With Hidden Data

00110101  00110100
Privacy Budget

• Allow some identifiability

• After “identifiability budget” is exhausted do… something

• Google folks love it

• Everyone else is… skeptical
Overview

1. Why websites track (and how much)

2. "Classic" tracking

3. Fingerprinting / "passive tracking"

4. Fingerprinting counter measures

5. Anti-finger printing exercise

6. Privacy protections in Brave

7. Wrapping up
Fingerprint2 Again...

- Choose two fingerprinting vectors to combat
  - Propose counter measures
- Choose two fingerprinting vectors that are hard
  - Why are counter measures hard?
Fingerprint2 pt 3...

- Pretend you're the attacker
- How would you respond to those defenses...
Fingerprint2 pt 4...

- Pretend you're the defender again
- How would you modify your defenses given the previous round...
Overview

1. Why websites track (and how much)
2. “Classic” tracking
3. Fingerprinting / “passive tracking”
4. Fingerprinting counter measures
5. Anti-finger printing exercise
6. Privacy protections in Brave
7. Wrapping up
Brave Privacy Protections

Shields UP for this site
If a site appears broken, try shields down

japantoday.com
11 items blocked

- 11 Cross-site trackers blocked
- 0 Connections upgraded to HTTPS
- 0 Scripts blocked
- 0 Cross-site cookies blocked
- 0 Cross-site device recognition blocked

Simple View
Change global shield defaults
Brave Privacy Protections

- Shields
- Global protection from tracking
- On by default
- Can be disabled if needed
Brave Privacy Protections

- Block cross site trackers
- Lists of known tracking websites
- Refuse to load
- Both community and Brave generated
Blocking Cross-Site Trackers in Brave

- **EasyList and EasyPrivacy**
  Used by AdBlock Plus, etc.

- **Disconnect**
  Used by Firefox, extensions

- **uBlock Origin**
  Excellent blocking extension

- **Brave generated**
  Open source, shared with community
Brave Privacy Protections

- Don’t send identifiers to third party sites
- Send to “main” site
- Same with other storage methods
Brave Blocks Tracking Cookies

Site A

Id=abc

Tracking Site
Brave Blocks Tracking Cookies

Site A

Tracking Site

User Agent: Brave Blocks

Id=abc
Brave Blocks Tracking Cookies

Site A

Site B

Tracking Site
Brave Blocks Tracking Cookies

Site A

Tracking Site

Site B

Id=abc
Brave Blocks Tracking Cookies

Site A

Site B

Tracking Site

Tracker can’t link A and B
Brave Privacy Protections

- Reduce finger printing vectors

- Currently:
  - Hardware identifiers
  - Canvas
  - WebGL
  - Audio

- Planned:
  - Fonts
  - User agent
  - Screen size
Under Exploration Possible Privacy Protections

- Restrictions on third-party scripts
- Identifying tracking behaviors, not just scripts / URLs
- Query parameters filtering
- Bounce tracking
- Much more…
Overview

1. Why websites track (and how much)
2. “Classic” tracking
3. Fingerprinting / “passive tracking”
4. Fingerprinting counter measures
5. Anti-fingerprinting exercise
6. Privacy protections in Brave
7. Wrapping up
Unasked for Advice

• Brave is hiring, keep us in mind

• Privacy is more than just web, there’s lots to do

• Don’t accept privacy as a feature…

• Choose your employer with values in mind
Thanks!

- **Pete Snyder**
  - Privacy Researcher
  - pes@brave.com
  - @pes10k

- Questions?
  - Standards work?
  - Privacy jobs?
  - Brave business model
  - BAT / Block chain
  - Anything else?