



# Insights from a 1-million-site Measurement of Online Tracking

#### Steven Englehardt @s\_englehardt

PRIVACYCON

Dillon Reisman @dillonthehuman Arvind Narayanan @random\_walker

This research was funded by NSF award CNS 1526353, a grant from the Data Transparency Lab, and by a cloud credits for research grant from Amazon Web Services.

#### Visiting 2 websites results in 84 third parties contacted



#### Open Web Privacy Measurement (OpenWPM)

citp / Op	enWPM							O Unwatch -	49	★ Unstar	435	<b>%</b> Fork	67
<> Code	() Issues	15 (')	Pull requests 0	🔲 Projec	ts O	🔳 Wiki	- Pulse	III Graph	s (¢	∦ Settings			
web priva	cy measurer	nent fram	nework https://we	ebtap.prince	ton.edu/	/— Edit							
0.40			0-01			20					+ CD	0.0	
· 48	0 commits		p 4 branches		🏷 <b>12</b> r	releases		LE 13 CONTRIDU	tors		sja GP	L-3.0	
Branch: mast	0 commits ter - New p	ull request	p 4 branches		© 12 r	releases	Create	new file Uplo	ad files	Find file	Clone	or downloa	ad <del>•</del>
T 48 Branch: mast	U commits ter → New p irdt Merge bran	<b>ull request</b> ch 'master'	p 4 branches	OpenWPM	© 12 r	releases	Create	new file Uplo	ad files	Find file	Clone	or downloa	ad <del>•</del> ago
T 48 Branch: mast	ter - New p rdt Merge bran	<b>ull request</b> ch 'master' Add	p 4 branches	OpenWPM out new comm	© 12 r nands	releases	Create	new file Uplo	ad files	Find file	Clone	or downlos 16 7 hours 15 days	ad <del>-</del> ago ago
Branch: mast	ter - New p rdt Merge bran	<b>ull request</b> ch 'master' Add disa	p 4 branches	OpenWPM but new comm t test for travi	to 12 mands s Cl	releases	Create	new file Uplo	ad files	Find file	Clone	or downlow 16 7 hours 15 days 15 days	ad - ago ago ago

#### https://github.com/citp/OpenWPM

### **The Princeton Web Census**

#### Monthly 1 Million Site Crawl



- All javascript files
- HTTP Requests and Responses
- Storage (cookies, Flash, etc)

Collecting:

#### **Results of the Princeton Web Census**



News site have the most trackers

PRIVACYCON



https://webtransparency.cs.princeton.edu/webcensus/

## Insights from the Princeton Web Census





#### Consolidation of top trackers



#### Only 6 organizations are present on >10% of sites





#### Takeaways of consolidation

- (1) Enforcement efforts can target large players, proactively set tracking norms.
- (2) Large trackers can quickly deploy technique to a massive number of sites.
- (3) Acquisitions can quickly shift tracking capability











Mixed content downgrades security indicator!





Of sites with mixed content:

half is caused solely by third parties (10% by trackers)







Of sites with mixed content:

half is caused solely by third parties (10% by trackers)

Half of all third-parties are HTTP-only



# Takeaway: Tracking may have second-order privacy impacts

- (1) Slow the adoption of encryption
- (2) Identifier leakage in requests to
- (3) Can aid network surveillance efforts





#### New Browser Features Used for Fingerprinting



https://webtransparency.cs.princeton.edu/webcensus/

#### Browsers remove BatteryStatus API citing privacy

•••	1313580 – Remove web content ac × +						
🗲 🧻 🔒 Mozilla Found	dation (US)   https://bugzilla.mozilla.org/show_bug.cgi?	id C Search	☆ 自 🔸 🧧	👳 🔶 🔳 🚍			
Bugzilla@Mozil	a	New Account   Log	In   Forgot Password	mozilla			
Home New Br	Product Dashboard						
	Persona is no longer an option for authentication on BMO. For more details see Persona Deprecated.						
Bug 1313580 -	Remove web content access to Ba	attery API		Last Comment			
Status: Whiteboard:	VERIFIED FIXED	Reported:	2016-10-27 23:28 PDT by Chris [:cpeterson]	Peterson			
Keywords:	addon-compat, dev-doc-needed, privacy, site-compat	Modified: CC List:	2016-11-02 09:53 PDT (History) 7 users (show)				
Product: Component:	Core (show info) DOM: Device Interfaces (show other bugs) (show info)	Flags: See Also:	ryanvm: in-testsuite-				
Version: Platform:	unspecified Unspecified Unspecified	Crash Signature:	(edit)				
Importance:	normal (vote)	QA Whiteboard: Iteration:					
Assigned To:	Chris Peterson [:cpeterson]	Points: Has Regression Range:					

#### Browsers remove BatteryStatus API citing privacy

•••	1313580 – Remove web content ac 🗴	+									
🗲 🛈 🔒 Mozilla Foundation (US)   https://bugzilla.mozilla.org/show_bug.cgi?id 🛛 C C Search 🏠 🖨 🦊 🧕 🧔 🤨 🔶 📧 🚍											
Bugzilla@Mozill	a		Signal Bug 164213 – Remove Battery Si	tat × +							
		€ → 0 ₽ 0	https://bugs. <b>webkit.org</b> /show_bug.c	gi?id=164213	C Sear	ch	☆ 自 🔸	5			
Home New Bro	owse Search		WebKit Bugzilla								
	Persona is no longer an option for a	a	Bug 164213: Remove Battery Status API from the tree								
Dug 1212500	Demous web content acco	Home   New   Bro	wse   Search	Search [?]	Reports   Requests	Help   New Accou	Int   Log In   Fe	orgot Pass	word		
Bug 1313580 -	Bug 1313580 - Remove web content acces		Le First Last N. & Brow Novt n. This hug is not is your last sourch results								
Status:	VERIFIED FIXED	Bug 164213	Pemove Battery Statu	ADT from the tr	<b>r</b> 00						
Whiteboard:		<u>Dug 104215</u>	Remove Dattery Status	SAFI Hom the t	CC						
Keywords:	Keywords: addon-compat, dev-doc-needed, privacy		Cus: RESOLVED FIXED		Reported: 201 Modified: 201	.6-10-30 20:26 P	DT by Brady I	idson			
	site-compat	Componen	t: WebKit WebKit Misc.		CC List: 8 u	sers ( <u>show</u> )	DT ( <u>THStory</u> )				
Product:	Core (show info)	Vers	on: WebKit Nightly Build								
Component:	DOM: Device Interfaces (show other bug (show info)	ge Platfo	rm: Unspecified Unspecified		See Also: 129	<del>-040</del>					
Version:	unspecified	Importar Assigned	ice: P2 Normal To: Alex Christensen								
Platform:	Unspecified Unspecified	<u>nooigneu</u>									
Importance:	normal (vote)	Keywo	: <u>ds</u> :								
Target Milestone:	mozilla52	Depends of	n:								
Assigned To:	Chris Peterson [:cpeterson]	Block	Show dependency tree / a	ranh							
			Show dependency tree / g	Ιαριι							
	A CVCON										

Takeaway: Expect any new API to be analyzed for its fingerprintability

- 1. Early detection of abuse can stem adoption
- 2. Browsers view fingerprinting as abuse
  - a. Mitigate fingerprinting during standardization
  - b. Remove APIs due to fingerprinting use



#### Our data is available!

The data is available as bzipped PostgreSQL dumps. The schema file used in all of the datasets is available here.

Dataset	Comments
1 Million Site Stateless	Parallel Stateless Crawl
100k Site Stateful	Parallel Stateful Crawl 10,000 site seed profile
10k Site ID Detection (1)	Sequential Stateful Crawl Flash enabled Synced with ID Detection (2)
10k Site ID Detection (2)	Sequential Stateful Crawl Flash enabled Synced with ID Detection (1)
55k Site Stateless with cookie blocking	Parallel Stateless Crawl Firefox set to block all third-party cookies
55k Site Stateless with Ghostery	Parallel Stateless Crawl Ghostery extension installed and set to block all possible trackers
55k Site Stateless with HTTPS Everywhere	Parallel Stateless Crawl HTTPS Everywhere installed

https://webtransparency.cs.princeton.edu/webcensus/index.html#data



#### Getting third-party responses from our data

```
tp guery = "SELECT r.url, h.value FROM http responses view AS r " \
           "LEFT JOIN http response headers view as h ON h.response id = r.id " \
                                                                                                   def get host plus ps(url):
           " WHERE r.top url LIKE %s AND " \
                                                                                                        """Strip the URL down to just a hostname+publicsuffix.
           "url not LIKE %s and h.name = 'Content-Type'"
cur = connection.cursor()
                                                                                                       If the provided url contains an IP address, the IP address is returned.
cur, itersize = 100000
                                                                                                       .....
try:
    top ps = utils.get host plus ps(top url) -
except AttributeError:
                                                                                                       hostname = urlparse(url).hostname
   print("Error while finding public suffix of %s" % top url)
                                                                                                       try:
    return None
                                                                                                            ip address(hostname)
                                                                                                            return hostname
cur.execute(tp query, (top url, top ps))
                                                                                                       except ValueError:
                                                                                                            return psl.get public suffix(hostname)
                                                     def is js(url, content type):
el parser = BlockListParser('easylist.txt')
                                                         if get top level type(content type) == 'script':
ep parser = BlockListParser('easyprivacy.txt')
                                                            return True
response data = defaultdict(dict)
                                                         if urlparse(url).path.split('.')[-1].lower() == 'js':
                                                            return True
for url, content type in cur:
                                                         return False
   if utils.should ignore(url):
       continue
                                                                 def is img(url, content type):
                                                                     if get top level type(content type) == 'image':
   url data = dict()
                                                                        return True
                                                                     extension = urlparse(url).path.split('.')[-1]
                                                                     if extension.lower() in IMAGE TYPES:
   url ps = utils.get host plus ps(url)
                                                                        return True
   if url ps == top ps:
                                                                     return False
       continue
   url data['url ps'] = url ps
                                                                                                    def get trackers(url list, first party, blocklist parser=None, blocklist="easylist.txt"):
                                                                                                         """Identify domains that are identified as trackers from list of URLs.
   is is = utils.is is(url, content type)
   is img = utils.is img(url, content type) -
                                                                                                         Returns set of domains/IPs filtered by the given blocklist parser.
   is el tracker = utils.is tracker(url,
                                                                                                         TODO: Better to return set of domains/IPs, or list of filtered urls?
                                   is is=is is,
                                                                                                         .....
                                   is img=is img,
                                   first party=top url,
                                                                                                         if not blocklist parser:
                                   blocklist parser=el parser)
                                                                                                             blocklist parser = BlockListParser(blocklist)
   is ep tracker = utils.is tracker(url,
                                   is js=is js,
                                                                                                         filtered domains = set()
                                   is img=is img,
                                                                                                         for url in url list:
                                   first party=top url,
                                   blocklist parser=ep parser)
                                                                                                             if is tracker(url, first party, blocklist parser):
   is tracker = is el tracker or is ep tracker
                                                                                                                 filtered domains.add(get host plus ps(url))
   url data['is js'] = is js
                                                                                                         return filtered domains
   url data['is img'] = is img
   url data['is tracker'] = is tracker
   response data[url] = url data
```

#### Getting third-party responses from our data



#### Getting third-party responses with Census.py

# census.get\_third\_party\_responses\_by\_domain( database\_connection, "http://nytimes.com" )

![](_page_21_Picture_2.jpeg)

![](_page_21_Picture_3.jpeg)

#### Getting third-party responses with Census.py

- get\_third\_party\_responses\_by\_domain
- get\_third\_party\_responses\_by\_domain
- get\_cookie\_syncs\_on\_domain
- is\_tracker
- get\_trackers

![](_page_22_Picture_6.jpeg)

#### Getting third-party responses with Census.py

- get\_third\_party\_responses\_by\_domain
- get\_third\_party\_responses\_by\_domain
- get\_cookie\_syncs\_on\_domain
- is\_tracker

PRIVACYCON

• get\_trackers

Contact us for access to "alpha" analysis server and library!

# Thanks for listening!

**Full Paper:** 

senglehardt.com/papers/ccs16\_online\_tracking.pdf

#### **Data and Analysis:**

PRIVACYCON

webtransparency.cs.princeton.edu/webcensus/

#### **Collaborate:**

webtap.princeton.edu/research/

#### Contact Me

**Email:** ste@cs.princeton.edu

Twitter: @s\_englehardt

Web: senglehardt.com

Image Assets from the Noun Project: Browser Network and Browser Battery by Aybige