

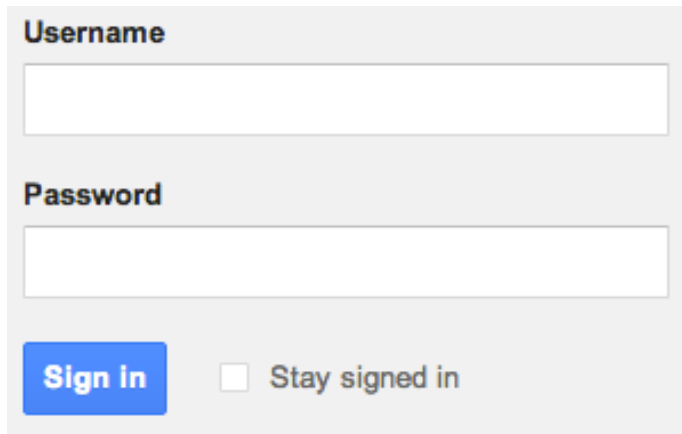
CSE 484 / CSE M 584
Computer Security:
Web Security

TA: Franz Roesner
franzi@cs.washington.edu

Logistics

- Homework #2 (crypto) due 2/22 5pm.
- Lab #2 (web security) due 2/27 5pm.
 - If you haven't signed up your group yet (email me UW NetIDs, groups name, password), **do it now!**
 - **Not just any SQL injection will work.**

SQL Injection



A login form with a light gray background. It contains two text input fields: the top one is labeled 'Username' and the bottom one is labeled 'Password'. Below the password field is a blue button labeled 'Sign in' and a checkbox labeled 'Stay signed in'.

What if this web app does something like this:

```
select * from users where  
user_name='$user' and  
user_password='$password'
```

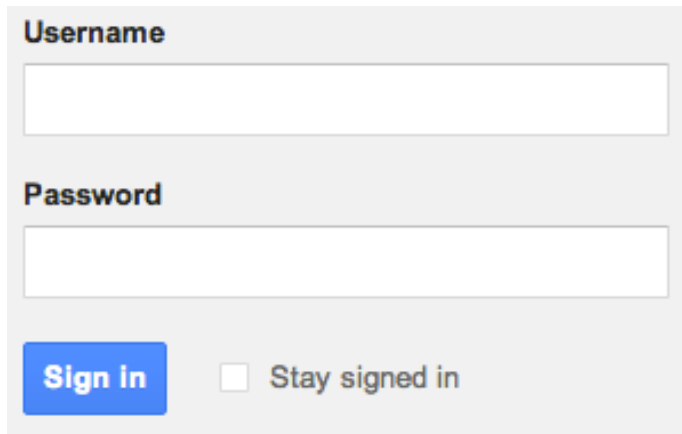
Attacker can log in by entering the username:

```
a' or '1'='1'; --
```

Why? SQL will execute:

```
select * from users where user_name = 'a' or '1'='1';
```

SQL Injection



A login form with the following elements:

- A text input field labeled "Username".
- A text input field labeled "Password".
- A blue button labeled "Sign in".
- A checkbox labeled "Stay signed in".

What if this web app does something like this:

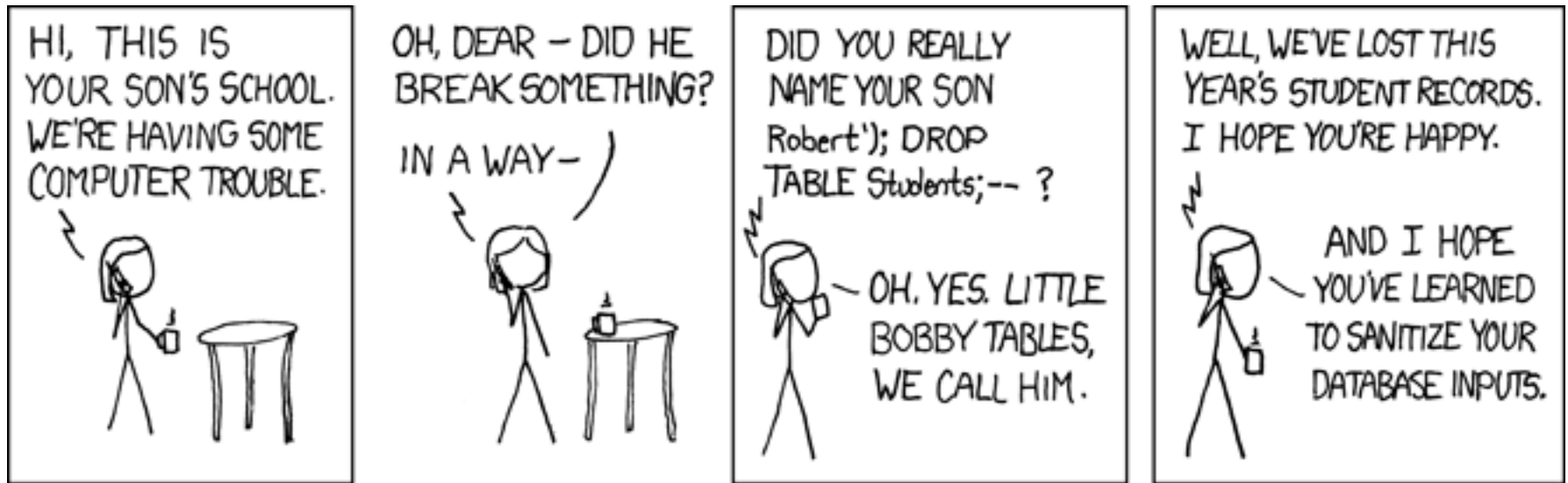
```
select * from users where  
user_name='$user' and  
user_password='$password'
```

Attacker can execute arbitrary SQL commands by entering the user name:

`a'; <Other Commands> --`

For example: `a'; DROP TABLE users; --`

SQL Injection



Clickjacking using the Cursor



Figure 1: Cursor spoofing attack page. The target Flash Player webcam settings dialog is at the bottom right of the page, with a “skip this ad” bait link remotely above it. Note there are two cursors displayed on the page: a fake cursor is drawn over the “skip this ad” link while the actual pointer hovers over the webcam access “Allow” button.

Other Web Security Resources

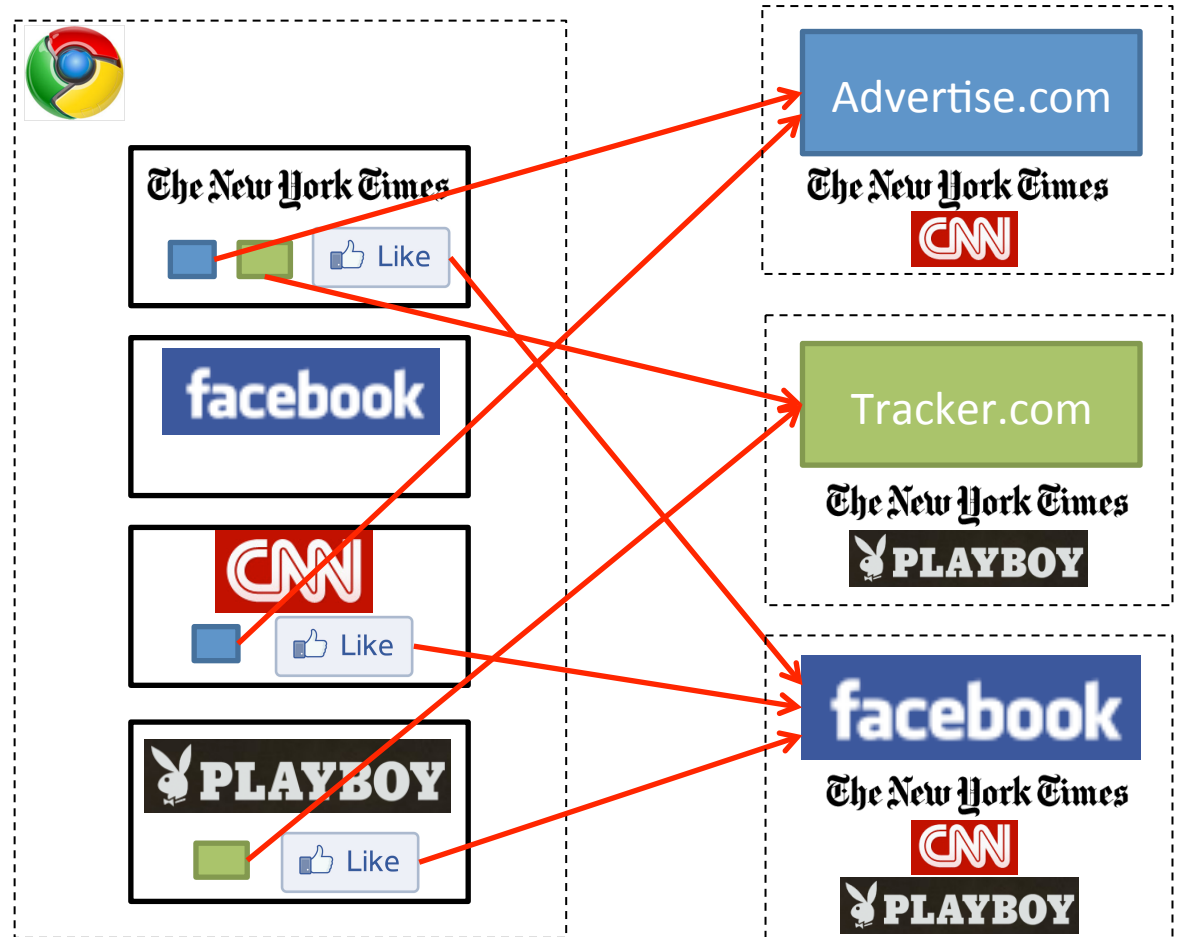
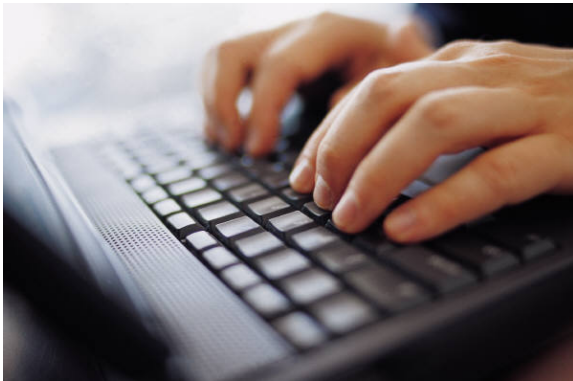
- Web Security Codelab: <http://google-gruyere.appspot.com/>
- <http://uwctf.cs.washington.edu/learntocook.php>
- Clickjacking: <http://www.grc.com/sn/notes-168.htm>
- SQL Injection: <http://sqlzoo.net/hack/>

Detecting and Defending Against Third-Party Tracking on the Web

Franziska Roesner, Tadayoshi Kohno, David Wetherall



Third-Party Web Tracking



(Hypothetical tracking relationships only.)

Bigger browsing profiles
= **increased value** for trackers
= **reduced privacy** for users

Tracking is Complicated

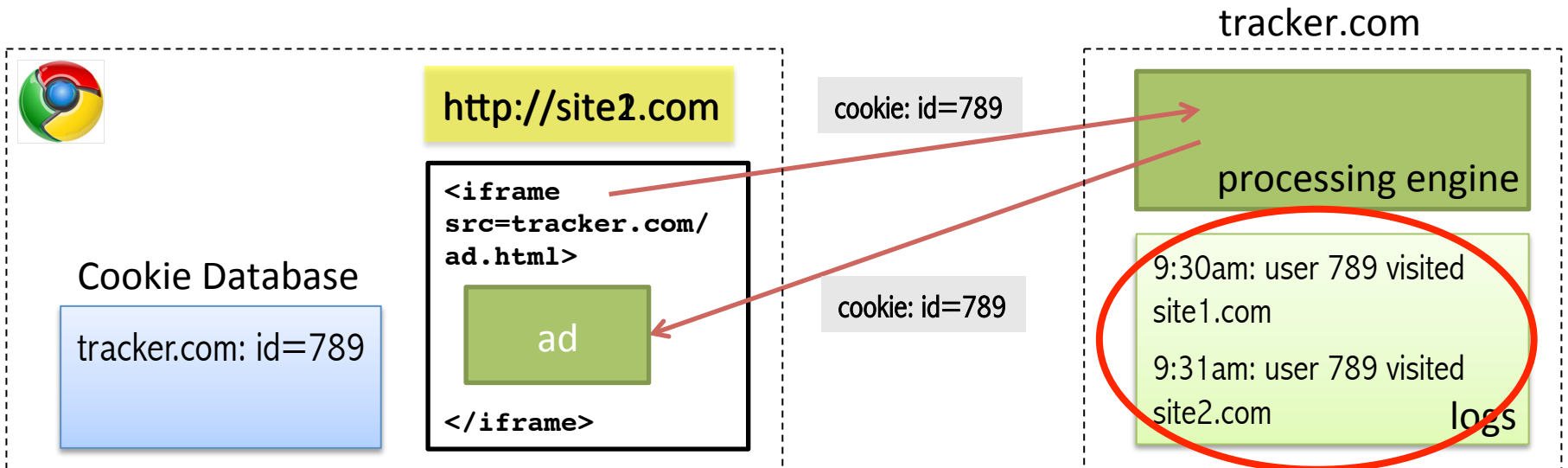
- Much discussion of tracking, but limited understanding of how it actually works.
- Our goals:
 - Understand the tracking ecosystem.
 - How is tracking actually done in the wild?
 - What kinds of browsing profiles do trackers compile?
 - How effective are defenses available to users?
 - Address gaps with new defense (ShareMeNot).

Mechanisms Required By Trackers

- **Ability to store user identity** in the browser
 - Browser cookies
 - HTML5 LocalStorage and Flash cookies (LSOs)
 - Not considering more exotic storage mechanisms or approximate fingerprinting
- **Ability to communicate** visited page and user identity **back to tracker**
 - Identity: Cookies attached to requests
 - Visited page: HTTP referrers
 - Both: scripts that embed information in URLs

Tracking: The Simple Version

- **Within-Site:** First-party cookies are used to track repeat visits to a site.
- **Cross-Site:** Third-party cookies are used by trackers included in other sites to create profiles.



Our Tracking Taxonomy

Name	Scope	User Visits Directly?	Overview
N/A	Within-Site	Yes	Site does its own on-site analytics.
Evolution: Embedding analytics libraries			
Analytics	Within-Site	No	Site uses third-party analytics engine (e.g., Google Analytics).
Vanilla	Cross-Site	No	Site embeds third-party tracker that uses third-party storage (e.g., Doubleclick).
Evolution: Third-party cookie blocking			
Forced	Cross-Site	Yes (forced)	Site embeds third-party tracker that forced the user to visit directly (e.g., via popup).
Referred	Cross-Site	No	Tracker relies on another cross-site tracker to leak unique identifier values.
Personal	Cross-Site	Yes	Site embeds third-party tracker that the user otherwise visits directly (e.g., Facebook).

Quirks of Third-Party Cookie Blocking

- Option blocks the **setting** of third-party cookies: all browsers
- Option blocks the **sending** of third-party cookies: **only Firefox**
- Result: Once a third-party cookie is somehow set, **it can be used** (in most browsers).

Our Tracking Taxonomy

Name	Scope	User Visits Directly?	Overview
N/A	Within-Site	Yes	Site does its own on-site analytics.
Evolution: Embedding analytics libraries			
Analytics	Within-Site	No	Site uses third-party analytics engine (e.g., Google Analytics).
Vanilla	Cross-Site	No	Site embeds third-party tracker that uses third-party storage (e.g., Doubleclick).
Evolution: Third-party cookie blocking			
Forced	Cross-Site	Yes (forced)	Site embeds third-party tracker that forced the user to visit directly (e.g., via popup).
Evolution: Complex ad networks			
Referred	Cross-Site	No	Tracker relies on another cross-site tracker to leak unique identifier values.
Personal	Cross-Site	Yes	Site embeds third-party tracker that the user otherwise visits directly (e.g., Facebook).

Referred Tracking



Cookie Database

tracker.com: id=522

http://site1.com

```
<iframe  
src=tracker.com/  
ad.html>
```

ad

```
</iframe>
```

tracker.com

processing engine

othertracker.com/track?
id=522&referrer=site1.com

http://site2.com

```
<iframe  
src=tracker.com/  
ad.html>
```

ad

```
</iframe>
```

processing engine

othertracker.com

logs

2:34pm:
site1.com: user 522
2:35pm:
site2.com: user 522

Our Tracking Taxonomy

Name	Scope	User Visits Directly?	Overview
N/A	Within-Site	Yes	Site does its own on-site analytics.
Evolution: Embedding analytics libraries			
Analytics	Within-Site	No	Site uses third-party analytics engine (e.g., Google Analytics).
Vanilla	Cross-Site	No	Site embeds third-party tracker that uses third-party storage (e.g., Doubleclick).
Evolution: Third-party cookie blocking			
Forced	Cross-Site	Yes (forced)	Site embeds third-party tracker that forced the user to visit directly (e.g., via popup).
Evolution: Complex ad networks			
Referred	Cross-Site	No	Tracker relies on another cross-site tracker to leak unique identifier values.
Personal	Cross-Site	Yes	Site embeds third-party tracker that the user otherwise visits directly (e.g., Facebook).
Evolution: Social networks			

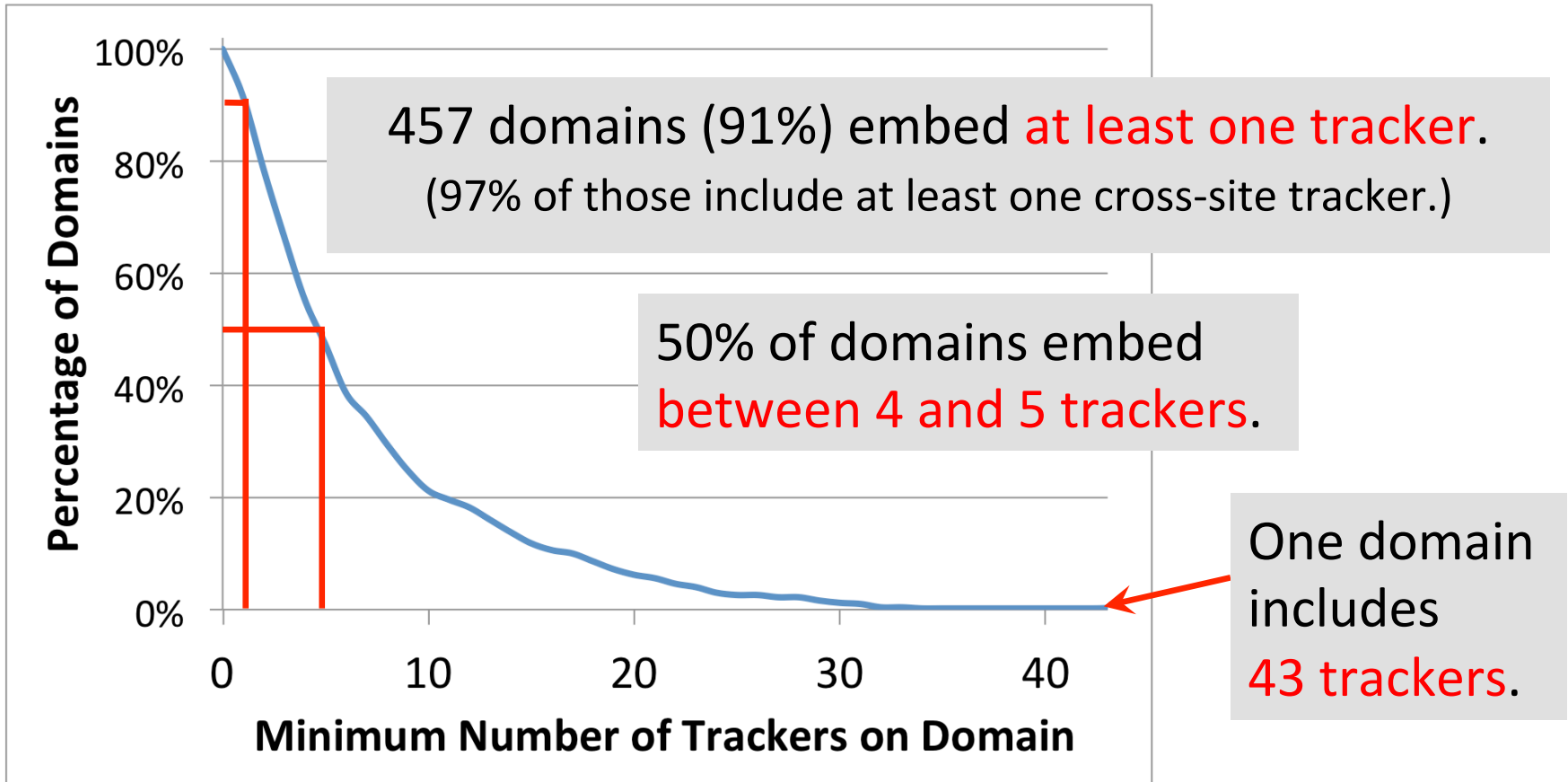
Anonymous

Measurement Study

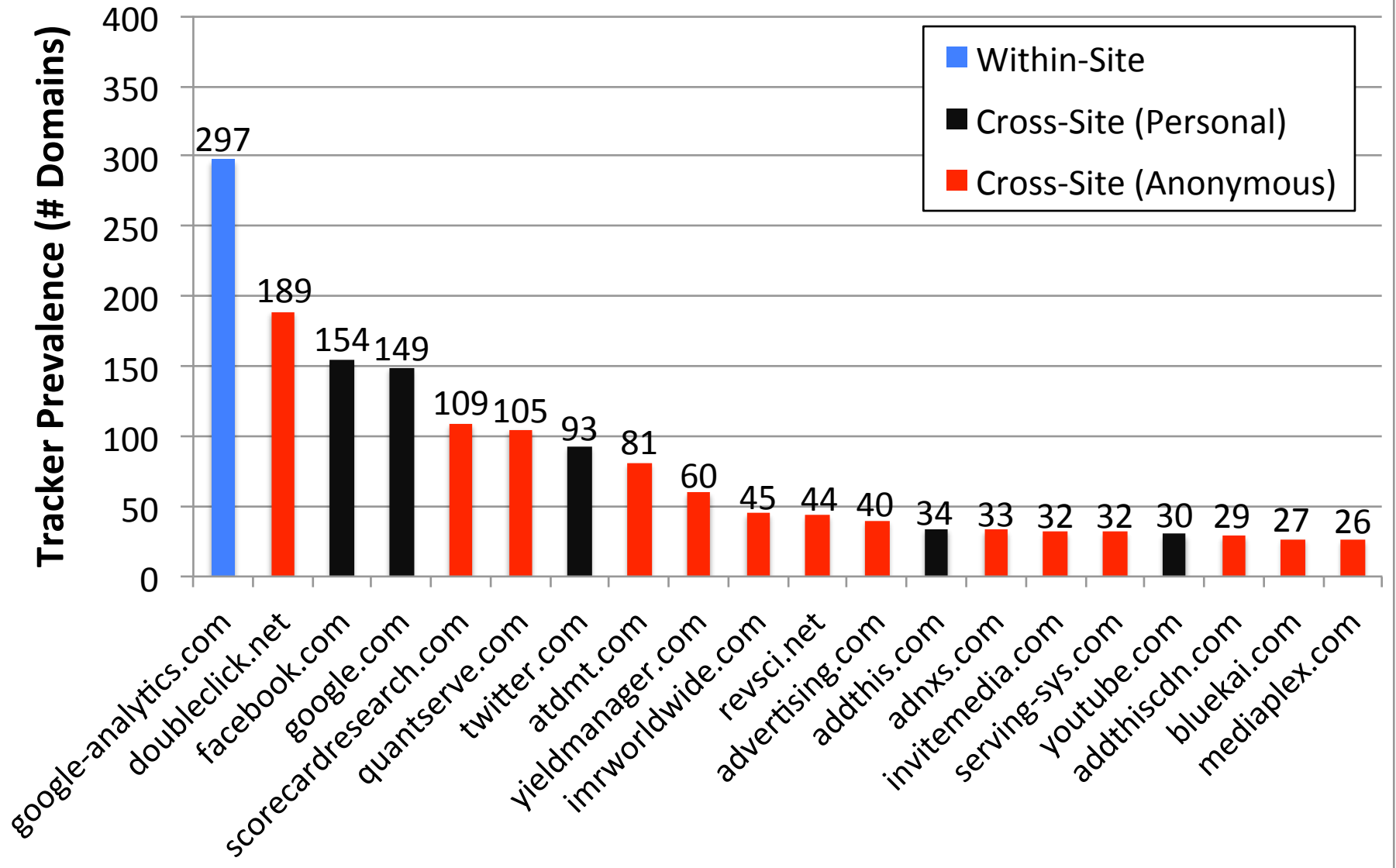
- Tool: [TrackingTracker](#) Firefox add-on that crawls the web and automatically categorizes trackers.
- 3 data sets
 - [Alexa Top 500](#)
 - 5 pages per domain: main page and up to 4 links
 - [Alexa Non-Top 500](#)
 - Sites ranked #501, #601, #701, etc.
 - 5 pages per domain: main page and up to 4 links
 - [AOL search logs](#)
 - 300 unique queries for 35 random users

Tracking Prevalence (Top 500)

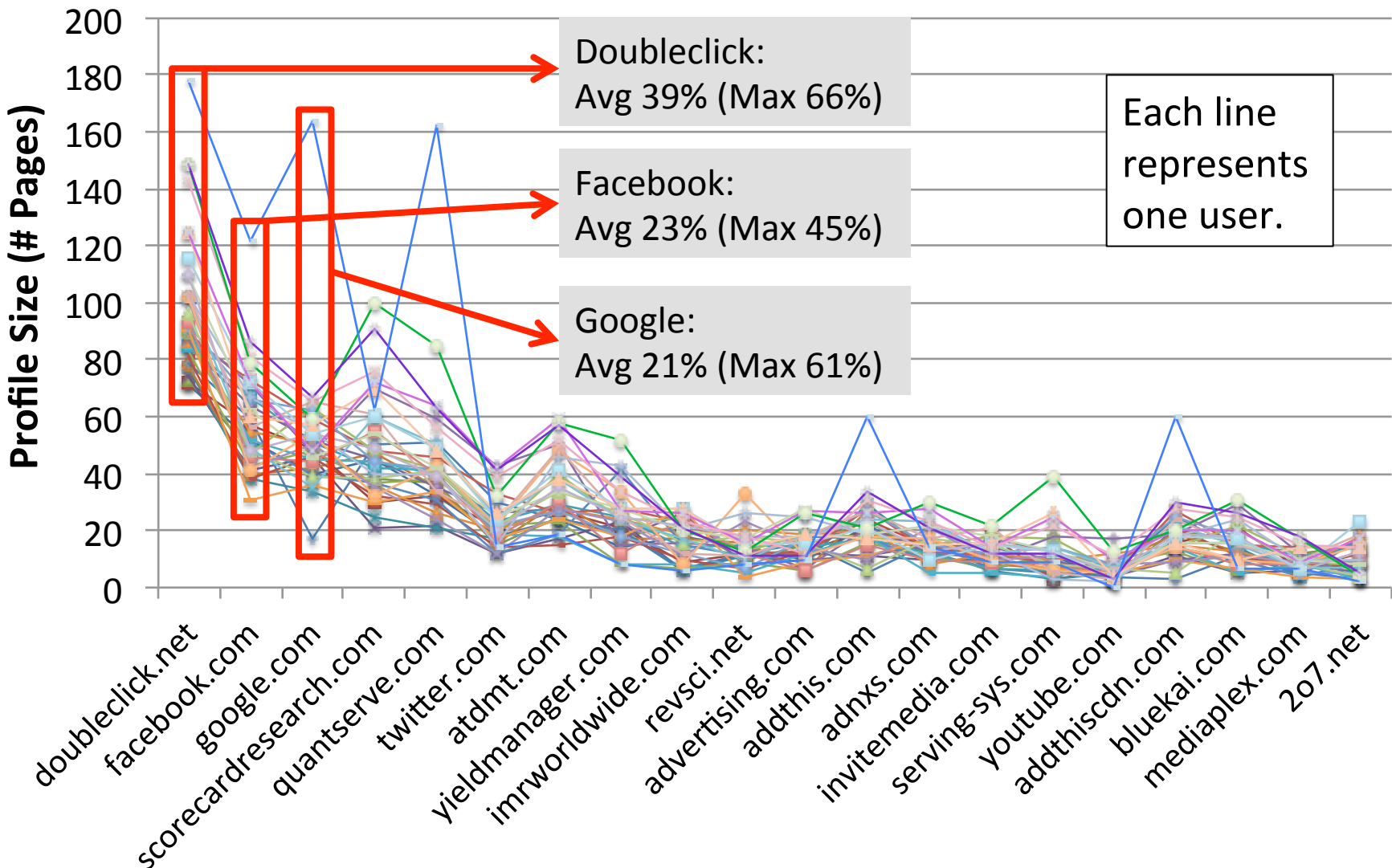
- 524 unique trackers on 500 domains



Top 20 Trackers on Top 500 Domains



AOL Users' Profile Sizes by Top 20 Cross-Site Trackers



LocalStorage and Flash Cookies

- Surprisingly little use of these mechanisms!
- Of 524 trackers on Alexa Top 500:
 - Only 5 set unique identifiers in LocalStorage
 - 35 set unique identifiers in Flash cookies
- Respawning:
 - LS → Cookie: 1 case; Cookie → LS: 3 cases
 - Flash → Cookie: 6 cases; Cookie → Flash: 7 cases