On the Privacy of Private Browsing

Kiavash Satvat, Matt Forshaw, Feng Hao, Ehsan Toreini

Newcastle University

DPM'13

Introduction

- 2005, Safari first introduced private browsing
- Today, private browsing has become an integrated feature in all major browsers
- How many people use it in the real world?
  - 19% based on a survey (Aggarwal et al, 2010)
  - 2.4 billion Internet users (world stat, 2012)
  - Roughly, 450 millions users of private browsing
- How secure is private browsing?

Threat model

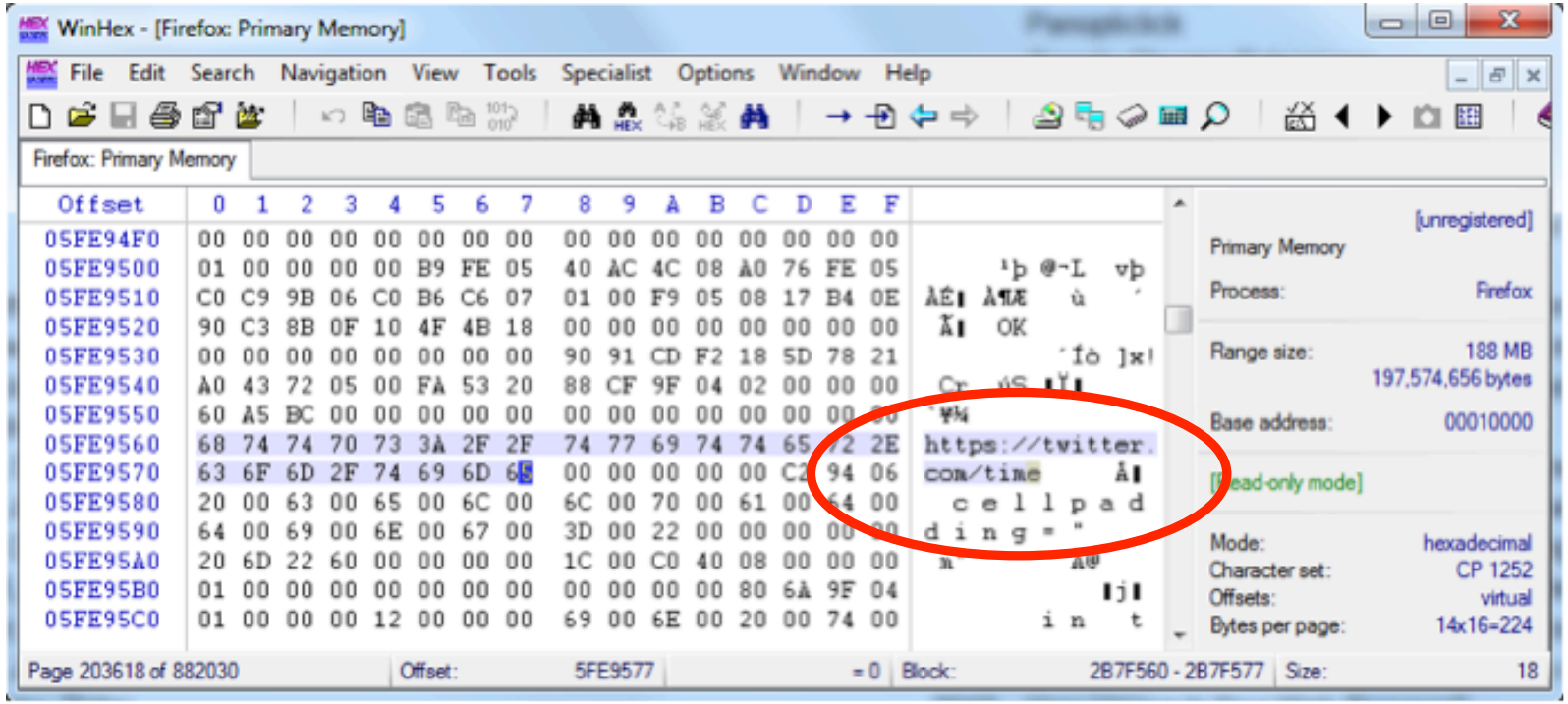
- First, need to define what is meant by “secure”
- Local attacker
  - Capability: full physical access to the computer after private session, but not before
  - Goal: discover any sensitive information related to the private session
- Remote attacker
  - Capability: able to engage with user through http (e.g., news website)
  - Goal: discover if the user is in the private session

Summary of attacks

	Firefox	Chrome	IE	Safari
Domain name system	√	√	√	√
Memory inspection	√	√	√	√
File timestamp	–	√	–	√
Index.dat *	N/A	N/A	√	N/A
SQLite database crash *	√	√	N/A	√
SQLite added bookmark *	√	√	N/A	√
Profile recovery *	√	√	N/A	N/A
Extension *	√	√	–	√
Cross-mode Interference *	N/A	√	N/A	N/A
Hyperlink attack	√	√	√	√
Timing attack *	√	√	–	√

- * new results discovered by our work
- We will select only a few attacks to present here

Local attack – memory inspection



- Artefacts about private browsing scattered in memory even after the browser is closed

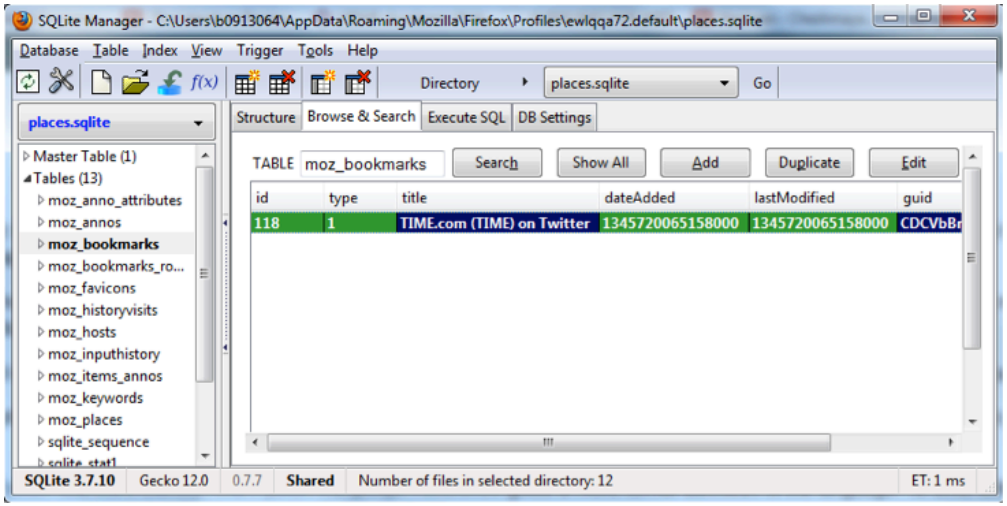
SQLite Database

- SQLite: an open source database used by Firefox, Chrome and Safari to store user profile
- In normal cases, it seems all browsers have removed private browsing records successfully
- However, it is essential to also test edge cases:
  - When the browser crashes
  - When the user adds a bookmark

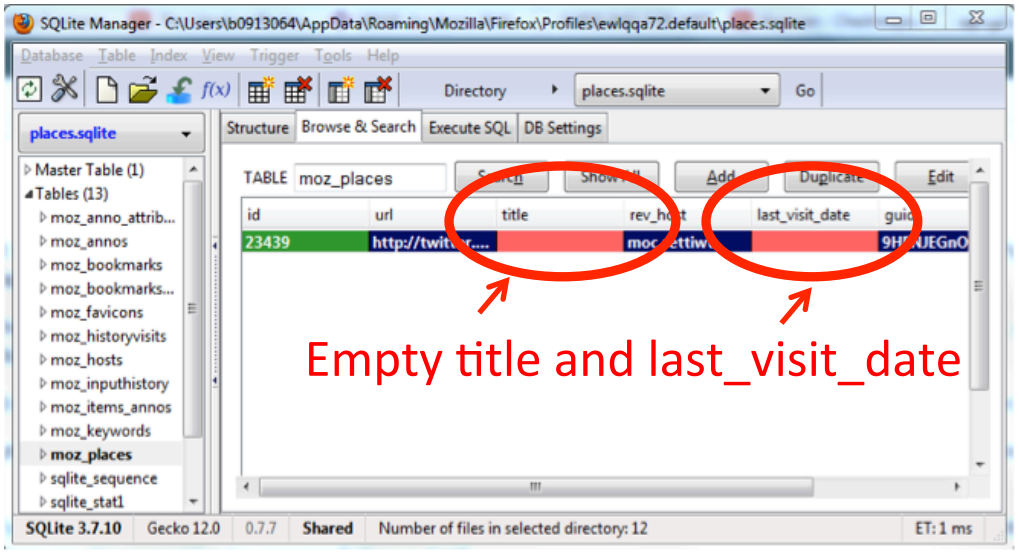
When the browser crashes

- May happen due to overload, manual termination etc
- Firefox (minor)
  - WAL files left on disk
  - Indicate occurrence of private browsing and times
- Chrome (minor)
  - Journal files left on disk
  - Indicate occurrence of private browsing and time
- Safari (serious)
  - Doesn't use in-memory SQLite
  - Inserts records of private browsing and deletes later
  - But in case of crash, private browsing records will persist

Adding a bookmark (Firefox)

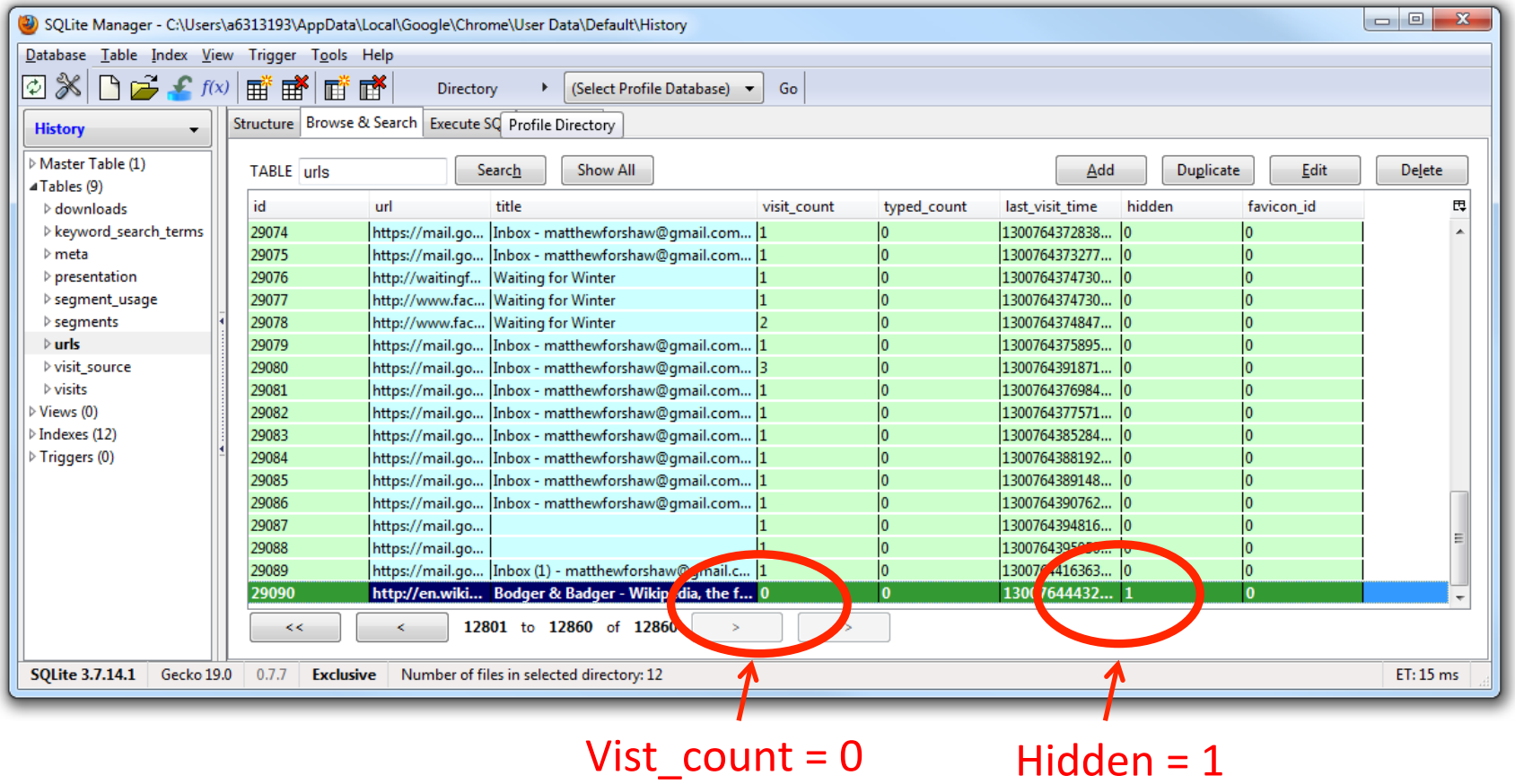


Moz_bookmarks (table)

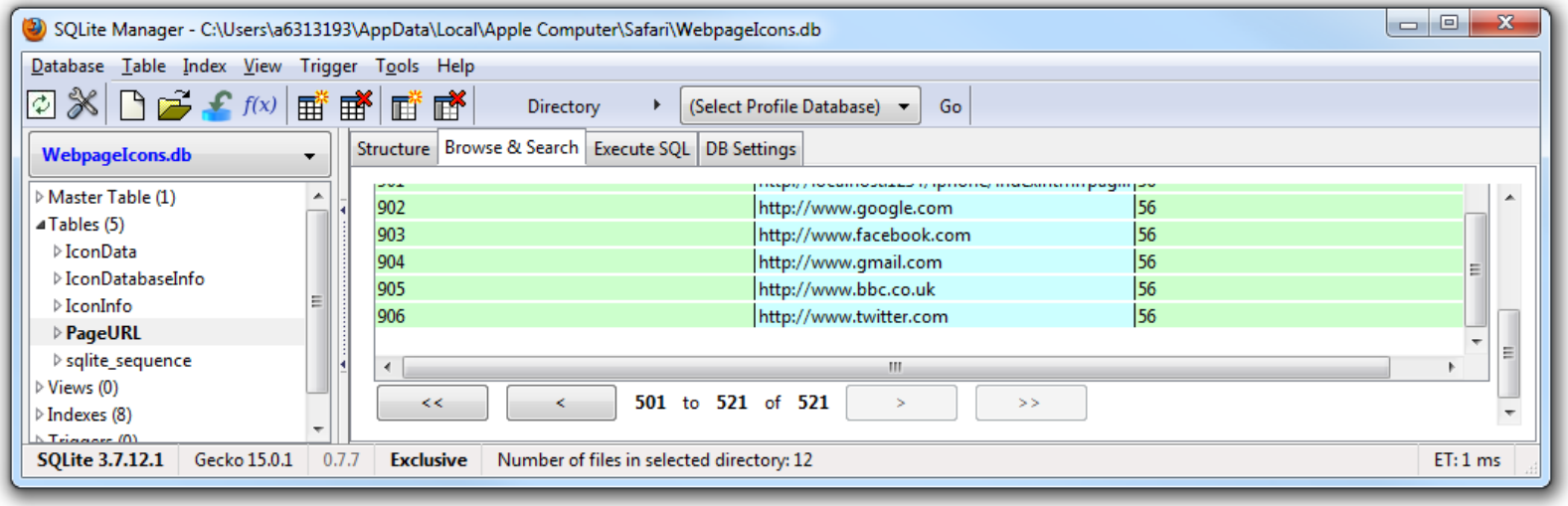


Moz_places (table)

Adding a bookmark (Chrome)



Adding a bookmark (Safari)

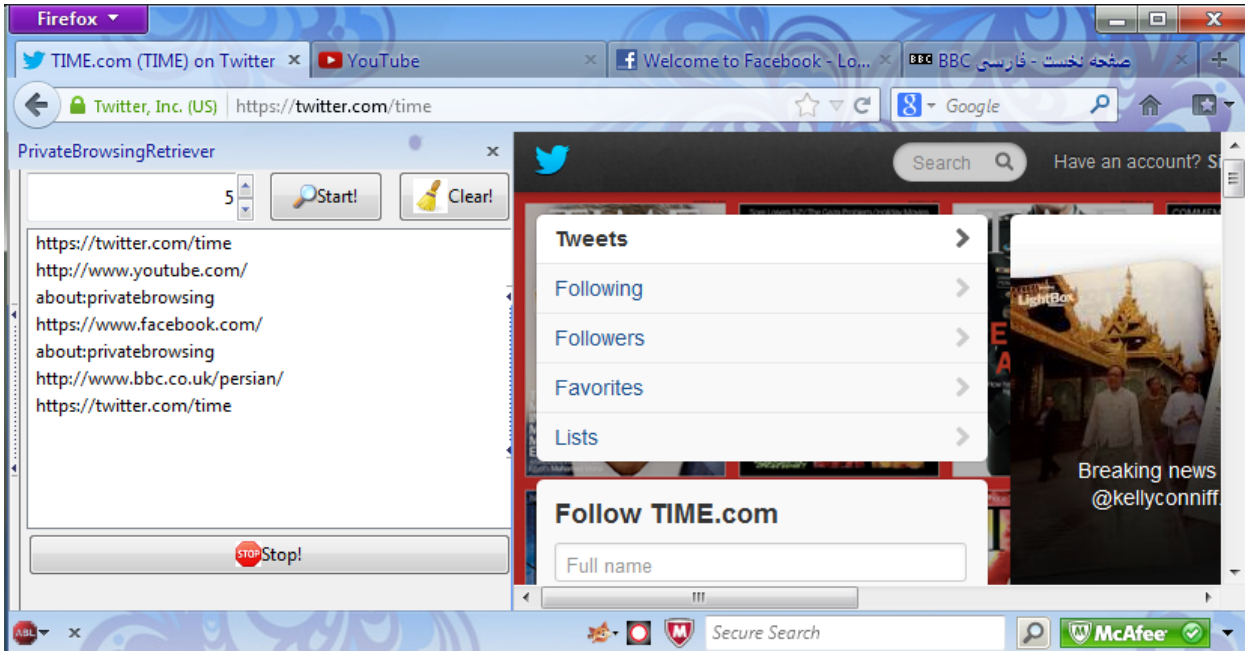


- (serious) Once the user adds one bookmark, all websites visited in private mode will persist in the database.
- We filed a bug report (#14685058)
  - 12/08 (Apple): “Engineering has determined that this is not to be fixed.”
  - 13/08, we asked Apple to clarify the decision.
  - 18/08 (Apple): “After much deliberation, engineering has removed this feature.”

Browser extensions

- Browser extensions pose a realistic threat to break privacy of private browsing.
- We tested four latest browsers in 2013
  - Firefox: extension enabled by default (vulnerable)
  - Safari: extension enabled by default (vulnerable)
  - Chrome: extension disabled by default (good)
  - IE: extensions disabled by default (good)

Firefox extension (proof of concept)



- Records all user activities in private session
- Then sends to a remote server

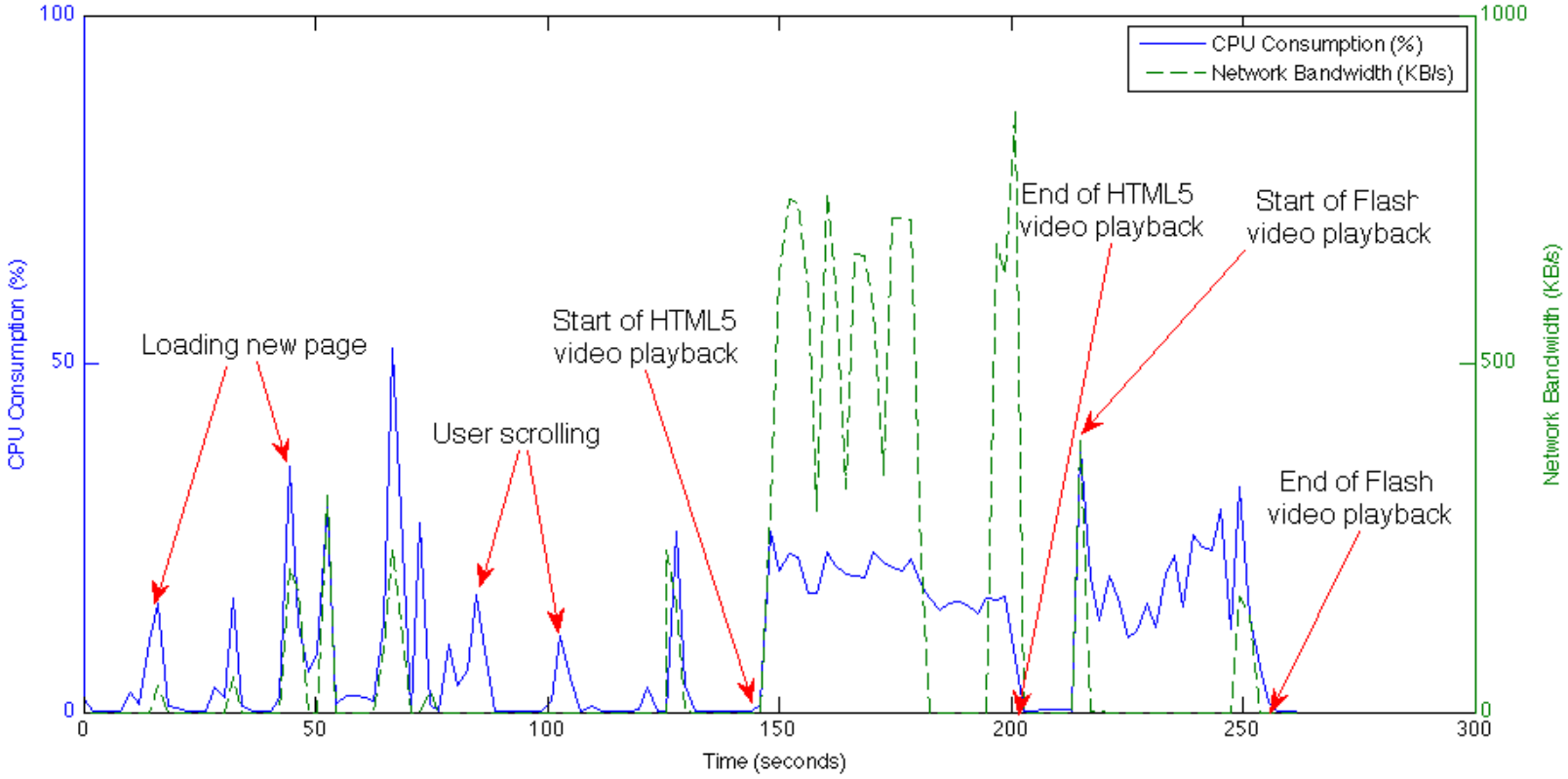
Addressing the threat of extensions

- One straightforward solution is to disable extensions by default in the private mode
- Adopted by Google Chrome and Microsoft IE
- However, we still need to be careful.

Cross mode interference

- Chrome allows two modes to run in parallel
  - Normal mode window: extension enabled
  - Private mode window: extension disabled
- However, since the two windows share some common resources
- Attacker may exploit cross mode interference

Example of cross mode interference



- Our suggested countermeasure: always run in a single mode

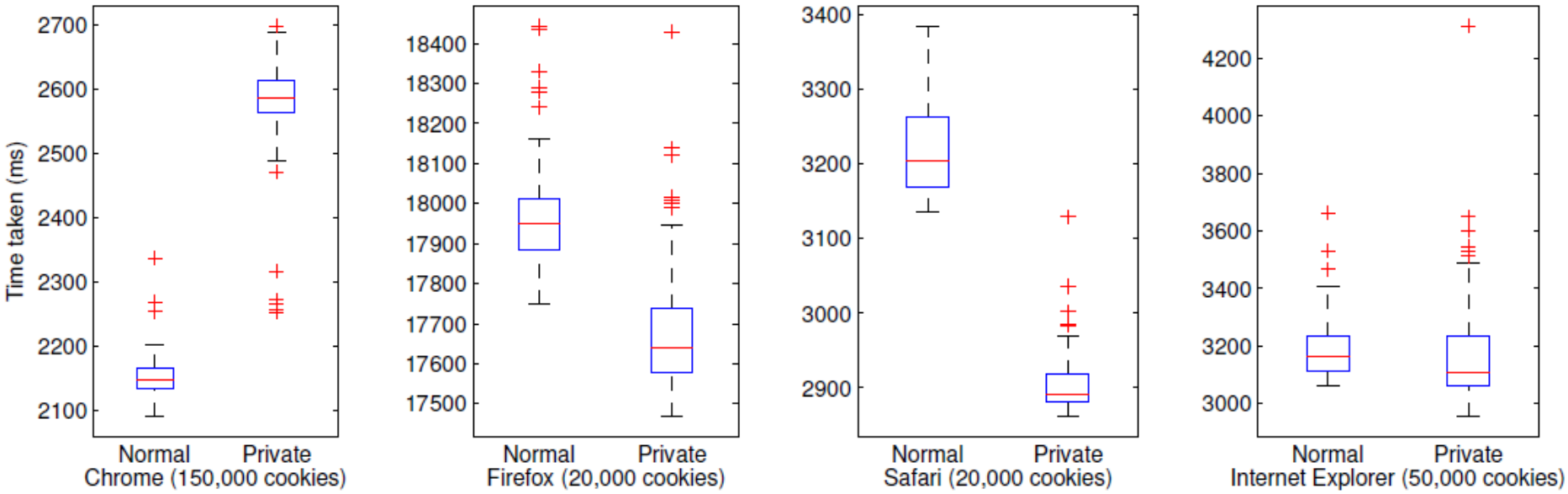
Remote attacks

- Goal of attack: remote website wishes to find out if the user is in the private mode.
- E.g., if the user is in the private mode, remote website may push more adult-oriented content or advertisement.
- Hence, we consider **the fact of using private browsing a privacy feature itself**.

Example: cookie timing attack

- The time it takes to write cookies is different between the usual and private modes.
- We conducted extensive experiments to collect data.

Results (box plots)



- With the exception of IE, the timing difference between the two modes is significant.

Conclusion

- Is private browsing private?
- We took a forensic approach
  - Defined a threat model to define “security”
  - Evaluated against local/remote attacks
  - Validated all previously known attacks
  - Discovered several new attacks
- For further details
  - See the paper and also open source code