COMPUTERS & SECURITY 67 (2017) 181–197

Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/cose

Computers & Security

Exploring the protection of private browsing in desktop browsers

Nikolaos Tsalis [a,*], *Alexios Mylonas* [b], *Antonia Nisioti* [b], *Dimitris Gritzalis* [a], *Vasilios Katos* [b]

[a] *Information Security & Critical Infrastructure Protection (INFOSEC) Laboratory, Dept. of Informatics, Athens University of Economics & Business, Greece*

[b] *Faculty of Science and Technology, Bournemouth University, United Kingdom*

ARTICLE INFO

Article history:
Received 24 March 2016
Received in revised form 4 March 2017
Accepted 7 March 2017
Available online 14 March 2017

Keywords:
Private browsing
Web browser
Web security
Browsing artefacts
Privacy

ABSTRACT

Desktop browsers have introduced *private browsing mode*, a security control which aims to protect users' data that are generated during a private browsing session by not storing them in the filesystem. As the Internet becomes ubiquitous, the existence of this security control is beneficial to users, since privacy violations are increasing, while users tend to be more concerned about their privacy when browsing the web in a post-Snowden era. In this context, this work examines the protection that is offered by the private browsing mode of the most popular desktop browsers in Windows (*i.e.*, Chrome, Firefox, IE and Opera). Our experiments uncover occasions in which even if users browse the web with a private session, privacy violations exist contrary to what is documented by the browser. To raise the bar of privacy protection that is offered by web browsers, we propose the use of a virtual filesystem as the storage medium of browsers' cache data. We demonstrate with a case study how this countermeasure protects users from the privacy violations, which are previously identified in this work.

1. Introduction

Since Internet penetration has risen in the last years (almost 3.4 million users by the end of 2015 (Internetworldstats, 2016)) it is important to preserve an adequate level of privacy to protect the average user while browsing the web. Average users, *i.e.*, those who are not technical, nor security savvy, rely on the default security countermeasures that are provided by the popular web browsers, such as protection from sites serving malware or hosting phishing attacks. However, previous works have revealed that the actual protection offered by these controls is rather limited (Virvilis et al., 2014; Tsalis et al., 2015a; Virvilis et al., 2015; Tsalis et al., 2015b; Mylonas et al., 2013).

Private browsing is a security control implemented by all popular web browsers in order to provide enhanced privacy to the end user while browsing the web (Google, 2016a; Google, 2016b; Mozilla, 2016; Microsoft, 2016a; Opera, 2016). Its primary goal is to protect the confidentiality of users' data, which are generated in a *private browsing session*, by avoiding to store them in the filesystem. In contrast, when the user is not under a private session (hereinafter this paper will refer to this mode as *normal mode*), the data generated while she is browsing the web are stored in the filesystem for usability (*e.g.*, facilitate

* *Corresponding author.*
E-mail addresses: ntsalis@aueb.gr (N. Tsalis), dgrit@aueb.gr (D. Gritzalis), amylonas@bournemouth.ac.uk (A. Mylonas), anisioti@bournemouth.ac.uk (A. Nisioti), vkatos@bournemouth.ac.uk (V. Katos).
http://dx.doi.org/10.1016/j.cose.2017.03.006
0167-4048/

authentication) and efficiency reasons (*e.g.*, caching). Thus, private browsing can aid users to protect their privacy against a local attacker who has access (temporal or permanent) to their device and attempts to uncover their online activities. After the revelations of state sponsored mass surveillance by Snowden (2016; BBC), average users are concerned, more than ever, about protecting their privacy. In a recent survey (Gao et al., 2014, November), 200 people were asked about the use of private browsing. Nearly half of them (39.5%) stated that they use private browsing, so as to prevent their browsing history and any cookies from being saved.

This paper examines the protection offered by private mode in popular web browsers, *i.e.*, *Chrome*, *Firefox*, *Internet Explorer* and *Opera*. A specific set of web artefacts was surveyed, which is typically created in a normal browsing session, to uncover if and where these are stored after the private session is terminated, contrary to the browser's documentation. Therefore, this work uncovers the deficiencies of the private browsing mode in web browsers and the respective privacy violations. In addition, to estimate the impact of the findings, a user survey was performed so as to note user opinion, based on the tested artefacts and their importance. Lastly, this work proposes the use of a virtual filesystem as a countermeasure against the privacy violations that have been uncovered.

The rest of the paper is structured as follows: Section 2 presents the related work. Section 3 includes our methodology. Section 4 contains the survey and test results. Section 5 presents our case study. Finally, Section 6 adds a discussion and concludes our work.

2. Related work

To the best of our knowledge, research regarding private mode and its effectiveness is still limited and in an early stage. To begin with, Aggarwal et al. (2010) was amongst the first to cope with the analysis of private browsing and the artefacts that were exposed after the private session. More specifically, Aggarwal et al. tested a subset of the artefacts that are discussed in this work, in earlier versions of Chrome, Firefox, Internet Explorer and Safari. Also, the authors expanded their analysis in both extensions and plugins, so as to identify any security weaknesses. They concluded to the inadequate implementation of private mode in those browsers, which exposed users' activities. Additionally, they proposed a mechanism for Firefox, which protects against extensions that expose browsing artefacts after private mode.

In 2011, Oh et al. (2011) focused on analysing the log files created by the browser, focusing on timeline analysis (*e.g.*, timestamps), search history, URL encoding, search keywords and the recovery of deleted data. The authors proposed WEFA, a tool for evidence collection and analysis. Their analysis was limited only on the normal browsing mode and also the browsers' versions used during the experiments are currently outdated. Said et al. (2011) examined if private browsing artefacts were available in the system's memory. The work of Ohana and Shashidhar (2013) focused on portable browsers (*e.g.*, stored on a USB flash drive) and whether artefacts are still available after the session terminates. The approach resembles the work of Said et al., 2011) in terms of capturing and analysing RAM, while the artefacts tested included history, credentials, images and videos.

Heule et al. (2015) provided a control for that purpose, which is based on mandatory access control and protects sensitive data that may be accessed and used by Chrome extensions. Similarly, Lerner et al. (2013) focused on JavaScript extensions on Firefox, while in private mode. The authors verified a number of extensions, from a safety, behavioural and debugging perspective that resulted in identifying which extensions could be malicious. Satvat et al. (2014) expanded the work of Aggarwal et al. (2010) by performing RAM, filesystem and network analysis, which revealed a notable amount of inconsistencies in the private browsing implementation. The authors created extensions for Chrome, Internet Explorer, Safari and Firefox to evaluate whether browser extensions leave artefacts that violate user's privacy. Opera and Chrome's guest modes were not evaluated and only a subset of the artefacts of Table 3 was considered.

Ruiz et al. (2015) focused on recovery techniques for page related data (*i.e.*, text and graphics) created during private browsing. The authors performed their tests within 4 individual phases: shutdown, freeze, kill process and power down, while each phase indicated the way the browser was terminated (*e.g.*, kill process – browser interruption). Their results showed that all phases included flaws regarding user's privacy in terms of acquiring browsing artefacts. In addition, Montasari and Peltola (2015) analysed both system's locations and RAM in all browsers except Opera. Although the selected operating system is not clarified, it is implied that the authors used Windows for their experiments. Their results showed that Chrome is the most secure browser, since there are no artefacts available after private browsing, while Firefox only included low risk artefacts.

In a parallel work, Xu et al. (2015) studied private browsing using the threat model defined in Aggarwal et al. (2010). They analysed the data flows that were generated by Firefox and Chrome with a system call tracer (for Linux) and detected the privacy violations that occurred, similar to our work. To mitigate the identified privacy threats, they implemented UCOGNITO for Firefox and Chrome only, which also sandboxes the browser in order to control and delete the files that are created by the browser. UCOGNITO uses MBOX to redirect (write) access to the filesystem by rewriting file paths in a static location, which can be deleted after the private session. However, as in UCOGNITO the browsing artefacts are stored in the filesystem, they can be recovered even if they are deleted unless secure deletion is used (Gutmann, 1996), which is time consuming. In our work all the browsing artefacts are stored in a virtual filesystem, instead of a long term storage medium (*e.g.*, hard disk). As a result, any browsing artefact cannot be recovered when the electromagnetic load of the RAM is lost. In addition, secure deletion in the RAM is quicker compared to hard disks. Finally, the proposed solution can be used with any browser irrespective of its technology.

In a similar approach, recent works focused on the forensic perspective of mobile versions of web browser. Marrington et al. (2012) dealt with Chrome's normal and incognito mode and the forensic traces left behind in comparison to the installed and the portable version of the browser. The results showed that both versions revealed the same amount of artefacts, thereby concluding that the portable version of the