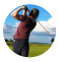
A forensic examination of web browser privacy-modes

Article *in* Forensic Science International Reports · October 2019

DOI: 10.1016/j.fsir.2019.100036



Graeme Horsman
Teesside University
69 PUBLICATIONS **406** CITATIONS

Ben Findlay
Teesside University
4 PUBLICATIONS **9** CITATIONS

Josh Edwick
Teesside University
1 PUBLICATION **3** CITATIONS

Forensic Science International: Reports 1 (2019) 100036

Contents lists available at ScienceDirect

Forensic Science International: Reports

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

A forensic examination of web browser privacy-modes

Graeme Horsman*, Ben Findlay, Josh Edwick, Alisha Asquith, Katherine Swannell, Dean Fisher, Alexander Grieves, Jack Guthrie, Dylan Stobbs, Peter McKain

Teesside University, Middlesbrough, Tees Valley, TS1 3BX, United Kingdom

ARTICLE INFO

Keywords:
Private browsing
Internet
Digital forensics
Internet history
Investigation

ABSTRACT

Private browsing facilities are part of many mainstream Internet browsing applications and arguably, there is now more awareness of their function and purpose by the average Internet user. As a result the potential for those engaging in malicious and/or illegal browsing behaviours, to do so in a ‘privatised’ way is increased. Many private browsing modes are designed to be ‘locally private’, preventing data denoting a user’s browsing actions from being stored on their device. Such actions, potentially compromise the availability of any evidential data, provide an investigatory headache. This work documents the examination of 30 web browsers to determine the presence of a ‘private mode’, and where available, the ‘privateness’ of said mode. Our test methodology is documented and results and limitations described for the purpose of open, transparent scrutiny and evaluation from those operating in this area.

1. Introduction

‘Private browsing’ (PB) is a generalised term utilised to reference mechanisms which are designed to prevent a user from having evidence of their web-browsing behaviour stored on their local device. From the outset, it is key to emphasise that in this context, private browsing refers only to those platforms which offer local privacy, and these should be distinguished from applications such as Tor (see https://www.torproject.org/) which also focus on online-privacy, and facilities which prevent remote tracking and monitoring, such as the W3C’s Tracking Preference Expression (aka “Do Not Track”). Dependant on the browser in user, an associated PB facility is referred to in different terminology; ‘incognito mode’ in Chrome, ‘InPrivate’ in Edge and the now unsupported Internet Explorer browser and a ‘private window’ in Firefox.

Arguably, through the increased sensitivity and publicity around privacy protection and the regulation of one’s digital footprint when online, PB technologies are likely to be in more frequent operation on a user’s device. Whilst it remains difficult to attribute definitive usage statistics to such actions, consensus surrounding online privacy provides an insight. In 2016, the use of a PB window was identified as the most popular form of online privacy measure globally [1]. In the United States alone, around 33% of users are reported to utilise PB, where over 70% admit to deleting their Internet History [2]. Whilst media coverage and increased notability of PB services has resulted in both widespread knowledge of it and understanding of its functionality, there remains an assumed assertion that substantial assessments of its local privacy have been undertaken, specifically from the context of a forensic examination. Yet this is not the case, and there is a limited set of academic commentaries which directly address the findability of PB session information following its utilisation. Whilst informal, forensic tool vendors and private organisations often pass comment via blog posts or corporate newsletters (see IntaForensics’s [3] discussion on mobile PB and comments from Magnet Forensics [4]). As a result, there is a gap in formalised knowledge with regards to definitively establishing how truly private PB facilities are.

While this may seem trivial, this lack of clarity has a significant impact on law enforcement forensic investigations and their approaches. Many investigations focus on locally resident data, ranging from traditional ‘dead-analysis’ of devices to Sexual Harm Prevention Orders (SHPO) in England and Wales (replacing previously implemented Sexual Offences Prevention Orders (SOPOs)) under the Sexual Offences Act 2003 (SOA), the latter posing an investigatory challenge with potential significant consequences. This paper provides an analysis of 30 available web browsers to determine their potential PB capabilities. The implemented PB test methodology is discussed in detail and results are presented highlighting those applications which offer a PB function and in turn whether or not it is in fact private, following digital forensic analysis. Finally, discussions and limitations are offered.

2. Private browsing

PB is a feature which has long since been on the radar of digital forensic practitioners. The risk it poses is arguably straightforward; any process which operates in a way which is designed to prevent potentially

* Corresponding author.
E-mail address: g.horsman@tees.ac.uk (G. Horsman).

http://doi.org/10.1016/j.fsir.2019.100036
Received 20 June 2019; Received in revised form 5 September 2019; Accepted 11 September 2019
Available online 14 October 2019

evidential content being stored on a local device (and therefore findable through examination techniques) raises investigatory concerns. Whilst PB itself has many legitimate uses and is not anti-forensic *per se* [5], it can be used with anti-forensic intent. Where Internet evidence forms the crux of an investigation, the absence of this content will pose regulatory issues. As a result, determining the extent and success of PB technology supports law enforcement in their approach to digital examinations of Internet content by helping to address the following points.

1 Where PB is suspected of occurring, knowing the success of a particular browser's PB facility helps to prevent unnecessary data processing (and time wastage) where browsing data does not actually exist on a device.
2 Knowing where PB may 'leak' browsing session information improves examination efficiency and prevents this content from being overlooked. This is particularly important where on-scene triage takes place, seen in some cases where a SHPO has been imposed.
3 Effective PB facilities require the acquisition and examination of alternative sources of browser information such as Internet Service Provider logged content.

Private browsing modes have been the focus of much informal commentary and experimentation since their mainstream marketing and implementation. Whilst many academic studies have assessed the 'privateness' of these modes, there are arguably less studies which provide a definitive decision, backed with a documented transparent test methodology designed to assess a service's ability to prevent private data being stored from a browsing session.

Research into PB must be continuous as web browser technology continues to develop at a pace as vendors seek to enhance the user experience and functionality for those operating their product. In addition, browser vendors are often reactive to any reported issues present in their software and seek to rectify this with the release of frequent updates. Therefore, both minor and major software updates may lead to PB data leakage if subsequent implementations have compromised its function and gone untested. Furthermore, development of the operating system(s) in which PB are usable may lead to the passive capturing of PB data. As a result, both differing versions of the browser itself and the underlying platforms and operating systems should be continually tested in combination with each other in order to maintain knowledge of the 'privateness' of a particular PB application.

2.1. Some existing studies

Satvat et al. [6] provide an insight into the vulnerabilities of private browsing sessions across Firefox, Chrome, Internet Explorer and Safari. The potential for plugin (also termed extension) vulnerabilities are noted, whilst limitations with residual data being held in physical memory are noted. In addition, program crashes and manually initiated bookmarking are noted as methods which may cause privacy leaks. Whilst the work did 'not observe any timestamp change of files under the profile directory after a private browsing session' it is difficult to infer from this statement alone the effectiveness of the local privacy afforded by these browsers. Testing took place on Mozilla Firefox (19.0), Apple Safari (5.1.7), Google Chrome (25.0.1364.97) and IE (10.0.9200.16521). Chivers's [7] analysis of Internet Explorer version 10 indicated 'that InPrivate browsing records can be reliably identified' on a local machine, particularly where a machine has been powered down during an InPrivate session. Whilst the study provides some insight into the recoverability of private session data, it is confined to a single browser vendor and version. Work by Gabet et al. [8] compared 'three enhanced privacy web browsers (Dooble, Comodo Dragon and Epic) and three commonly used web browsers in anonymous browsing mode (Chrome, Edge and Firefox)' with inconclusive results as which performed better from a privacy perspective. Muir et al. [9] indicate that records of session activity following use of the Tor Browser Bundle can be recovered with a focus noted for the NTUSER.DAT.log transaction log. Yet Jadoon et al's [10]. study of Tor makes no reference to such potential for recovered artefacts. More bespoke browsers have been targeted in recent work with Wang et al. [11] providing an analysis of the 'Browsar' application and [12] tackling 'Epic Privacy Browser'.

The use of volatile memory is often cited as a location of private browsing history recovery [6,9,13–17], however it is necessary to note that this work does not cover physical memory acquisition and analysis for PB content. Physical memory acquisition is still not common practice at all scenes and as physical memory must be collected before power is removed, in most cases this information may not be available to those investigating PB behaviours. Therefore as previous works have noted PB content is often in physical memory, this work opts to focus on examining hard disk drive content.

3. Methodology

Whilst studies of singular or small subsets of PB modes have been carried out, this work offers a review of 30 browsers. We have opted for a test platform of Windows 10 due to its wide-spread popularity, with a reported almost 70% market share [18]. All 30 browsers were located using the Google search engine, demonstrating accessibility to those who have a device and Internet connection. Regarding the work carried out, this article offers the following contributions:

1 A defined transparent methodology documenting test actions, the test platform and procedural tasks undertaken as part of the analysis. In doing so, effective scrutiny and evaluation of the work by peers is facilitated, allowing known or unknown constraints to be identified.
2 A benchmark test to determine the privacy of 30 browsers within a set of documented **known** documented conditions. It is important to define the circumstances of the tests in order to determine the boundaries of applicability of presented results, and where further testing may be required.

3.1. Context

Whilst the need to determine how effective PB services are, it is also necessary to offer context regarding the importance of knowing this information. The two main contexts to consider during a PB investigation are on-scene and in-lab. On-scene triage is often constrained by factors such as limited time and tool-type, which can mean only a targeted (and subsequently limited) approach to finding any potential evidential data is taken [19,20]. In comparison, in-lab processes may provide for the use of more comprehensive examination processes where time and resource constraints may be less (or indeed not relevant). Therefore in the presented experimentation, consideration has been given to the processes which have been implemented as part of digital forensic analysis of PB data in order to replicate both triage and comprehensive procedures.

3.2. Configuration

Table 1 documents the five test search terms and subsequently visited URLs utilised as part of our experimentation process. Prior to testing, our test platform was confirmed as having no instances of these strings present, following preliminary keyword searching to prevent contamination and false positives.

All testing took place using a stock Windows 10 virtual machine (VM) which was installed based on a standard Windows 10 ISO file acquired from the academic software licence portal (https://onthehub.com/). A Windows 10 V M was subsequently prepared in which to perform testing. Once prepared, this VM was forensically imaged and an elimination hash database was produced. The VM was subsequently exported as an appliance (OVA file) in order to deploy elsewhere. The decision to carry out our testing via this method was to ensure a consistent, stock environment across all the browsers being tested

The stock VM appliance was deployed to each individual laboratory machine and each was assigned a respective web browser to investigate.

Table 1
Documented test web browser data (Table submitted as separate file).

Search Term	URL Visited
"blackbag mobilyze"	https://www.blackbagtech.com/software-products/mobilyze.html
"griffeye"	https://www.griffeye.com/
"lunastar comic cast"	http://lunastar.thecomicseries.com/
"TDFCon"	http://www.tdfcon.com/
"pintofscience"	https://pintofscience.co.uk/

Prior to interacting with the VM itself, the web address of each browser was identified on an external machine to limit the searching required within the VM itself in order to find and download the relevant browser installer files. Once installed, each test browser within each VM was used to execute the same test browsing actions (described in Table 1). A series of prescribed browsing tasks were carried out in each VM. By performing exactly the same tasks in each VM, this allowed for consistent and reliable searching and investigation of the evidence subsequently. The VM was shutdown following the standard method. The decision to perform a standard, shutdown as opposed to a hard power-off was taken in order to mimic what was considered to be normal behaviour by a person utilising private browsing over a protracted period of time.

Each VM hard drive (.vmdk) was then forensically imaged into Expert Witness Format (.E01) for subsequent examination and loaded into X-Ways Forensics 19.7 to perform a Simultaneous Search (aka keyword search covering both standard Ansi, Unicode-UTF8 and Unicode-UTF16 formatted text). This process was done, in order to mimic a triage process which could be carried out on-scene offender processing, such as those instances where an offender is a managed sex offender. Typically a Simultaneous Search will provide faster results regarding keyword hits, but will not effectively handle content which for example has been compressed/encoded etc. Following a basic Simultaneous Search a Refine Volume Snapshot (aka evidence processing) was completed, followed by a Simultaneous Search. This process takes longer but provides for a more comprehensive examination where compressed volumes for example are uncompressed making them searchable, thereby mimicking a more comprehensive, in-lab examination.

The digital forensic image was also loaded into Griffeye Analyze DI Pro 18.5. The standard processing options, along with the LACE Carver v.12.8.56, were selected. The previously discussed elimination-hash database was then used to eliminate irrelevant files to allow for more efficient and accurate identification of any images of pertinence (Fig. 1).

4. Results

Table 2 offers an overview of the performance of those browsers tested. Of the 30 browsers tested, one was paywalled (Puffin), four encountered runtime issues (Lynx, Links, Falkon, Konqueror) and three did not have PB modes (GreenBrowser, Netsurf and Sleipnir). This left 22

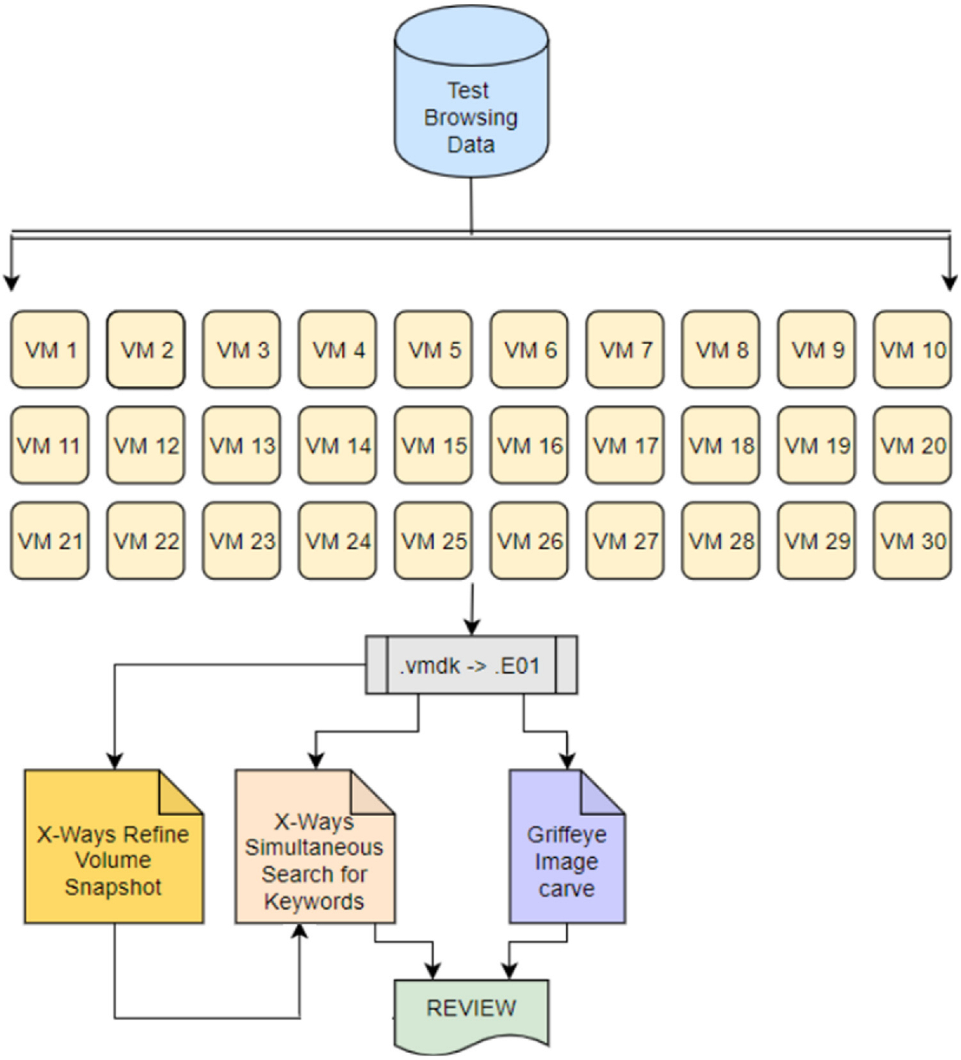


Fig. 1. Methodology used for PB study.

Table 2
A breakdown of the results for the 30 chosen browser platforms (Table submitted as separate file).

Browser	Version	Release date	Active Development?	Download ink	Private Function	Is it Private?
Avant	*12.5.0.0*	*2019-05-18*	*Yes*	http://www.avantbrowser.com/download.aspx?uil=en	Yes	No
Brave	*0.65.118*	*No Date*	*Yes*	https://brave.com/	Yes	Yes
Chrome	*76.0.3789.0*	*Frequent*	*Yes*	https://www.google.co.uk/chrome/	Yes	Yes
Chromium	*76.0.3805*	*No Date*	*Unknown*	https://www.chromium.org/getting-involved/download-chromium	Yes	Yes
Comodo IceDragon	*64.0.4.15*	*January 2019*	*Yes*	https://browser.comodo.com/	Yes	Yes
Comodo Dragon	*73.0.3683.75*	*No Date*	*Yes*	https://browser.comodo.com/	Yes	No
Dooble	*1.56e*	*November 2017*	*April 2019 Github*	https://textbrowser.github.io/dooble/	Yes	Yes
Edge	*44.17763.1.0*	*No Date*	*Yes*	Proprietary	Yes	No
Epic	*62.0.3202.94*	*No Date*	*Unknown*	https://www.epicbrowser.com/	Private when proxy is on	No
Falkon	*3.1.0 × 64*	*No Date*	*No (both x64 and 32 bit will not run)*	https://www.falkon.org/download/	N/A	N/A
FireFox	*67*	*Frequent*	*Yes*	https://www.mozilla.org/en-GB/firefox/new/	Yes	Yes
GreenBrowser	*6.9.1223*	*Dec 2016*	*No*	https://greenbrowser.en.softonic.com/	No	N/A
IE	*11.55.17763.0*	*No Date*	*No*	Proprietary	Yes	No
Konqueror	*N/A*	*N/A*	*N/A*	https://konqueror.org/	N/A	N/A
Links	*Flagged as malicious*	*Flagged as malicious*	*Flagged as malicious*	http://links.twibright.com/	N/A	N/A
Lynx	*2.8.9*	*2018*	*Will not run on Windows 10*	https://lynx.invisible-island.net/current/#major_docs	N/A	N/A
Maxthon	*5.2.7.2000*	*2018*	*Yes*	http://www.maxthon.com/mx5/	Yes	Yes
Midori	*0.5.11*	*No Date*	*Yes*	https://www.midori-browser.org/download/	Yes	Yes
Netsurf	*3.8*	*2017*	*no (last reported update in 2018)*	https://www.netsurf-browser.org/	No	N/A
Opera	*15*	*4-1-2019*	*Yes*	https://www.opera.com/download	Yes	Yes
Pale Moon	*28.5.0*	*April 2019*	*Yes*	https://www.palemoon.org/	Yes	Yes
Puffin	*Paywalled*	*Paywalled*	*Paywalled*	https://www.puffin.com/	N/A	N/A
Seamonkey	*2.49.4*	*July 2018*	*no (last reported update in 2018)*	https://www.seamonkey-project.org/releases/	Yes	Yes
Sleipnir	*6.3.7*	*Unknown*	*Unknown*	https://sleipnir.en.softonic.com/	No	N/A
SlimJet	*22.0.4.0*	*No Date*	*last reported update in March 2019*	https://www.slimjet.com/	Yes	Yes
Tor Browser	*60.7.0esr (64bit)*	*No Date*	*Yes*	https://www.torproject.org/download/	Yes	Yes
Torch	*65.0.0.1617 (32 bit)*	*No Date*	*Yes*	https://torchbrowser.com/tour	Yes	Yes
UC Browser	*7.0.185.1002*	*2018*	*Yes*	https://www.ucweb.com/	Yes	Yes
Vivaldi	*2.5.1525.46*	*No Date*	*Yes*	https://vivaldi.com/features/	Yes	Yes
WaterFox	*56.2.10*	*No Date*	*Yes*	https://www.waterfox.net/releases/	Yes	Yes

browsers with an operation 22 PB mode for testing. Of these 22 browsers, following testing, five browsers were found to have ‘leaked’ PB session data.

From the five browsers seen to have leaked PB data; Avant, Comodo Dragon, Edge, Epic and Internet Explorer, a breakdown of keyword hit locations for URL information is offered in Table 3 and the number of hits offered in Table 4. It was found that a triage-style keyword search (i.e. a simultaneous search of the evidence with no processing) was successful in recovering positive keyword hits in all cases where the performance of a volume snapshot followed by a keyword search was also successful. Whilst the more comprehensive examination and keyword search often resulted in larger numbers of keyword hits (see Table 4), there were effectively no occasions where evidence was missed by just performing a simultaneous search with no prior processing of the evidence.

4.1. Picture review

In addition to keyword string matching for Internet history records, each case has been carved for the presence of any cached imagery deriving from any of the test browsed websites using Griffeye’s DI Analyze Pro with LACE plug-in. All images were manually reviewed and those relevant highlighted with originating system locations noted in Table 5.

It should be noted that whilst five browser tests indicated PB website string data was recoverable, only three browsers (Avant, Epic and Internet Explorer) cached images to the local machine during testing.

5. Analysis and concluding thoughts

We note that the 30 targeted browsers performed as documented within the confines documented our methodology. As a caveat to the results offered, we feel that they must not be overstated and we cannot go as far as to say that those browsers which performed privately during our tests are confirmed and completely private in all circumstances. The reason for such statements lie with the following points:

1 Our chosen virtual machine platform ‘Virtual Box’ reports limited support for platform hibernation. As a result, it is possible that the browsers may leak PB content to the Hiberfil.sys on non-virtual platforms.
2 The length of time a browsing session takes place for may also be a factor, where both the Hiberfil.sys (as noted above) and system paging via the Pagefile.sys may be forensically valuable. Varying the length of browsing sessions and examining the impact of prolonged PB sessions on potential data leakage is an under-researched area and requires further work within the digital forensic field.
3 The impact of different hardware configurations should also be taken into account where for example, different amounts of system RAM may result in different memory caching processes and subsequent volumes of leakage.

Table 3
A breakdown of the keyword hit locations for URL information for the Avant, Comodo Dragon, Edge, Epic and Internet Explorer browsers (Table submitted as separate file).

Basic:- Simulataneous Search	
Browser Name	*Location*
Avant	$MFT $Logfile \Users\<USERNAME>\AppData\Roaming\AvantProfiles\.temp\sessions\132208\webkit\Default \Users\<USERNAME>\AppData\Roaming\AvantProfiles\.temp\sessions\132208\webkit\Default\Cache \Users\<USERNAME>\AppData\Roaming\AvantProfiles\.temp\sessions\132208\webkit\Default\Code Cache\js \Users\<USERNAME>\AppData\Roaming\AvantProfiles\.temp\sessions\132208\webkit\Default\Session Storage \Users\<USERNAME>\AppData\Roaming\AvantProfiles\.temp\sessions\132208\webkit\Default\Local Storage\leveldb Freespace
Comodo Dragon	\Users\<USERNAME>\AppData\Local\Temp\7ZipSfx.001\ccav_installer.msi
Edge	\Users\<USERNAME>\AppData\Local\Packages\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\AC\MicrosoftEdge\User\Default\Recovery\Active\{58D38B7A-81AB-4A8E-ACED-5A32599E789B}.dat
Epic	Freespace \Users\<USERNAME>\AppData\Local\Epic Privacy Browser\User Data\Default \Users\<USERNAME>\AppData\Local\EpicPrivacyBrowser\UserData\Default\Cache \Users\<USERNAME>\AppData\Local\EpicPrivacyBrowser\UserData\Default\Media Cache \Users\<USERNAME>\AppData\Local\EpicPrivacyBrowser\UserData\Default\Session Storage \Users\<USERNAME>\AppData\Local\EpicPrivacyBrowser\UserData\Default\Local Storage\leveldb \Windows\System32\sru\SRUDB.dat
Internet Explorer	$MFT, $Logfile $Extend\$UsnJournal \Users\<USERNAME>\AppData\Local\Microsoft\Windows\INetCache\Low\IE\ \Users\<USERNAME>\AppData\Local\Microsoft\Internet Explorer\Recovery\Active \Users\<USERNAME>\AppData\Local\Microsoft\Windows\WebCache \Users\<USERNAME>\AppData\Local\Packages\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\AC\Temp \Windows\System32\LogFiles\WMI \Windows\Temp Freespace
Advanced:- Refine Volume Snapshot	
Browser Name	*Location*
Avant	$MFT $Logfile \Users\<USERNAME>\AppData\Roaming\AvantProfiles\.temp\sessions\132208\webkit\Default \Users\<USERNAME>\AppData\Roaming\AvantProfiles\.temp\sessions\132208\webkit\Default\ Web Data \Users\<USERNAME>\AppData\Roaming\AvantProfiles\.temp\sessions\132208\webkit\Default\Cache \Users\<USERNAME>\AppData\Roaming\AvantProfiles\.temp\sessions\132208\webkit\Default\Code Cache\js \Users\<USERNAME>\AppData\Roaming\AvantProfiles\.temp\sessions\132208\webkit\Default\History \Users\<USERNAME>\AppData\Roaming\AvantProfiles\.temp\sessions\132208\webkit\Default\Session Storage \Users\<USERNAME>\AppData\Roaming\AvantProfiles\.temp\sessions\132208\webkit\Default\Local Storage\leveldb Freespace
Comodo Dragon	\Users\<USERNAME>\AppData\Local\Temp\7ZipSfx.001\ccav_installer.msi
Edge	\Users\<USERNAME>\AppData\Local\Packages\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\AC\MicrosoftEdge\User\Default\Recovery\Active\{58D38B7A-81AB-4A8E-ACED-5A32599E789B}.dat
Epic	Freespace \Users\<USERNAME>\AppData\Local\Epic Privacy Browser\User Data\Default \Users\<USERNAME>\AppData\Local\EpicPrivacyBrowser\UserData\Default\Cache \Users\<USERNAME>\AppData\Local\EpicPrivacyBrowser\UserData\Default\Media Cache \Users\<USERNAME>\AppData\Local\EpicPrivacyBrowser\UserData\Default\Session Storage \Users\<USERNAME>\AppData\Local\EpicPrivacyBrowser\UserData\Default\Local Storage\leveldb \Windows\System32\sru\SRUDB.dat
Internet Explorer	$MFT $Logfile $Extend\$UsnJournal \Users\<USERNAME>\AppData\Local\Microsoft\Internet Explorer\Recovery\Active \Users\<USERNAME>\AppData\Local\Microsoft\Windows\INetCache\Low\IE\ \Users\<USERNAME>\AppData\Local\Microsoft\Windows\WebCache \Users\<USERNAME>\AppData\Local\Packages\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\AC\Temp \Windows\System32\LogFiles\WMI \Windows\Temp Freespace

4 Virtualisation as a comparable platform raises some questions as whilst it is frequently adopted as a testing platform to combat the difficulty of testing on physical equipment, there remains a gap in research regarding the accuracy of its implementation.

Notwithstanding that there may be external factors such as those described in Muir et al. [9] which result in leakage of PB data to the disc which have not been investigated here, the research conducted here highlights that this leakage does not appear to occur in a virtual environment which does not support virtual memory. The precise cause of the leakage documented in Muir et al. has not been clearly established, however it is immediately apparent that there are 2 common-sense, obvious causes:

- A flaw in the browser design and development leading to data being leaked outwards from within, i.e. the browser is to blame

Table 4
A breakdown of the number of keyword hits for test URL information for the Avant, Comodo Dragon, Edge, Epic and Internet Explorer browsers (Table submitted as separate file).

Browser	Browser Processing and Subsequent Number of Keyword Hits									
	Avant:- simultaneous search	Avant:- refine volume snapshot + simultaneous search	Internet Explorer:- simultaneous search	Internet Explorer:- refine volume snapshot + simultaneous search	Comodo Dragon:- simultaneous search	Comodo Dragon:- refine volume snapshot + simultaneous search	Edge:- simultaneous search	Edge:- refine volume snapshot + simultaneous search	Epic:- simultaneous search	Epic:- refine volume snapshot + simultaneous search
blackbag	1024	1462	1383	1395	8	107	27	27	721	725
mobilyze	276	483	916	944	0	0	28	28	250	250
griffeye	618	936	879	886	1	26	22	22	445	450
lunastar	143	172	349	351	0	0	20	20	88	100
tdfcon	144	182	490	490	0	0	22	22	96	100
pintofscience	642	1192	319	377	0	0	29	29	900	900
blackbagtech.com	642	954	482	482	1	28	11	11	420	424
griffeye.com	487	720	268	268	0	0	8	8	337	341
comicseries.com	96	117	128	130	0	0	8	8	71	88
tdfcon.com	96	107	156	156	0	0	8	8	58	62
pintofscience.co.uk	121	178	200	232	0	0	5	5	72	72

Table 5
A breakdown of the cached image locations for the Avant, Epic and Internet Explorer browsers (Table submitted as separate file).

Browser Name	Location
Avant	Freespace \Users\<USERNAME>\AppData\Roaming\AvantProfiles\.temp\sessions\132208\webkit\Default\Cache Freespace
Epic	Freespace \Users\<USERNAME>\AppData\Local\EpicPrivacyBrowser\UserData\Default\Cache \Users\<USERNAME>\AppData\Local\EpicPrivacyBrowser\UserData\Default\Media Cache
Internet Explorer	\Users\<USERNAME>\AppData\Local\Microsoft\Windows\INetCache\Low\IE\ Freespace

- The operating system taking more control over the browser than it should, leading to data being extracted from without, i.e. the operating system is to blame

Either way, the results of this research have assessed and clearly demonstrate the effectiveness of the PB function itself within each browser.

Declaration of Competing Interest

There are no conflicts of interest to report.

Acknowledgement

We would like to acknowledge the School of Science, Engineering & Design for their support in running this event at Teesside University.

References

[1] Statista, Online Privacy Measures of Global Internet Users As of 2nd Quarter 2016 Available at: https://www.statista.com/statistics/617422/online-privacy-measures-worldwide/ (Accessed: 26 May 2019), (2016) .

[2] Statista, Which of These Measures Do You Take With Regard to Your Online Privacy? Available at: https://www.statista.com/statistics/714130/us-online-usage-privacy-measures-usage/ (Accessed: 26 May 2019), (2017) .

[3] IntaForensics, IOS 10 Private Browsing: How Private Is It? Available at: https://www.intaforensics.com/2016/09/30/ios-10-private-browsing-how-private-is-it/ (Accessed: 26 January 2018), (2016) .

[4] Magnet Forensics, Forensic Implications of a Person Using Firefox's "Private Browsing" Available at: https://www.magnetforensics.com/computer-forensics/forensic-implications-of-a-person-using-firefoxs-private-browsing/ (Accessed: 26 May 2019), (2013) .

[5] G. Horsman, D. Errickson, When finding nothing may be evidence of something: anti-forensic and digital tool marks, (Sci. Justice (2019) .

[6] K. Satvat, M. Forshaw, F. Hao, E. Toreini, On the privacy of private browsing–a forensic approach, Data Privacy Management and Autonomous Spontaneous Security, Springer, Berlin, Heidelberg, 2014, pp. 380–389.

[7] H. Chivers, Private browsing: a window of forensic opportunity, (Digit. Investig. 11 (1) (2014) 20–29.

[8] R.M. Gabet, K.C. Seigfried-Spellar, M.K. Rogers, A comparative forensic analysis of privacy enhanced web browsers and private browsing modes of common web browsers, (Int. J. Electron. Secur. Digit. Forensics 10 (4) (2018) 356–371.

[9] M. Muir, P. Leimich, W.J. Buchanan, A Forensic Audit of the Tor browser Bundle, Digital Investigation, 2019.

[10] A.K. Jadoon, W. Iqbal, M.F. Amjad, H. Afzal, Y.A. Bangash, Forensic Analysis of Tor Browser: A Case Study for Privacy and Anonymity on the Web, Forensic Science International, 2019.

[11] F. Wang, J. Mickens, N. Zeldovich, Veil: private browsing semantics without browser-side assistance, (Proceedings of Network and Distributed System Security Symposium (NDSS) (2018) .

[12] A. Reed, M. Scanlon, N.A. Le-Khac, Private web browser forensics: a case study on epic privacy browser, (Journal of Information Warfare 17 (1) (2018) .

[13] R. Dave, N.R. Mistry, M.S. Dahiya, Volatile memory based forensic artifacts and analysis, (Int. J. Res. Appl. Sci. Eng. Technol. 2 (1) (2014) 120–124.

[14] M.J.C. Huang, Y.L. Wan, C.P. Chiang, S.J. Wang, October. Tor browser forensics in exploring invisible evidence, 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), IEEE, 2018, pp. 3909–3914.

[15] D.J. Ohana, N. Shashidhar, Do private and portable web browsers leave incriminating evidence?: a forensic analysis of residual artifacts from private and portable web browsing sessions, (EURASIP J. Inf. Secur. 2013 (1) (2013) 6.

[16] A. Ghafarian, S.A.H. Seno, Analysis of privacy of private browsing mode through memory forensics, (Int. J. Comput. Appl. 132 (16) (2015) .

[17] A. Case, G.G. Richard III, Memory forensics: the path forward, (Digit. Investig. 20 (2017) 23–33.

[18] Statista, Market Share Held by the Leading Computer (desktop/tablet/console) Operating Systems Worldwide From January 2012 to February 2019, (2019) . (Accessed: 26 May 2019) https://www.statista.com/statistics/268237/global-market-share-held-by-operating-systems-since-2009/.

[19] S.L. Garfinkel, Digital media triage with bulk data analysis and bulk extractor, (Comput. Secur. 32 (2013) 56–72.

[20] G. Horsman, C. Laing, P. Vickers, A case-based reasoning method for locating evidence during digital forensic device triage, (Decis. Support Syst. 61 (2014) 69–78.