A Forensic Analysis of Android Mobile Private Browsing Artifacts

GIAC (GCFA) Gold Certification

Author: Warren Thompson, warrentnt@gmail.com
Advisor: *Lee Crognale*

Accepted: 13 January 2022

Abstract

The adoption of mobile devices and functionality continues to grow daily. Simultaneously, there has been an increase in privacy awareness and the desire for anonymity while browsing the internet. This research paper aims to explore the effectiveness of private browsing at mitigating the generation and persistence of filesystem-based artifacts on Android-based mobile devices. Four popular browsers used by advocates of private browsing were studied: Chrome, Firefox, DuckDuckGo, and Tor Browser. In some cases, the results were in line with expectations. In others, the results were alarming. The research shows that gaining access to a full disk image, including unallocated disk space, could allow an individual to access private browsing history, as well as the screenshots of websites visited in specific cases. In all cases, persistent filesystem-based artifacts were generated. In summary, private browsing is not as private as popularly perceived.

1. Introduction

1.1. Mobile device utilization

Over the past ten years, smartphone utilization to access the internet increased from just over 6% to 54% (Figure 1.1-1). Simultaneously, the percentage of people worldwide utilizing desktop computers to access the internet decreased from 93% to 44%. The chart below depicts the shift to mobile internet access over the previous decade.

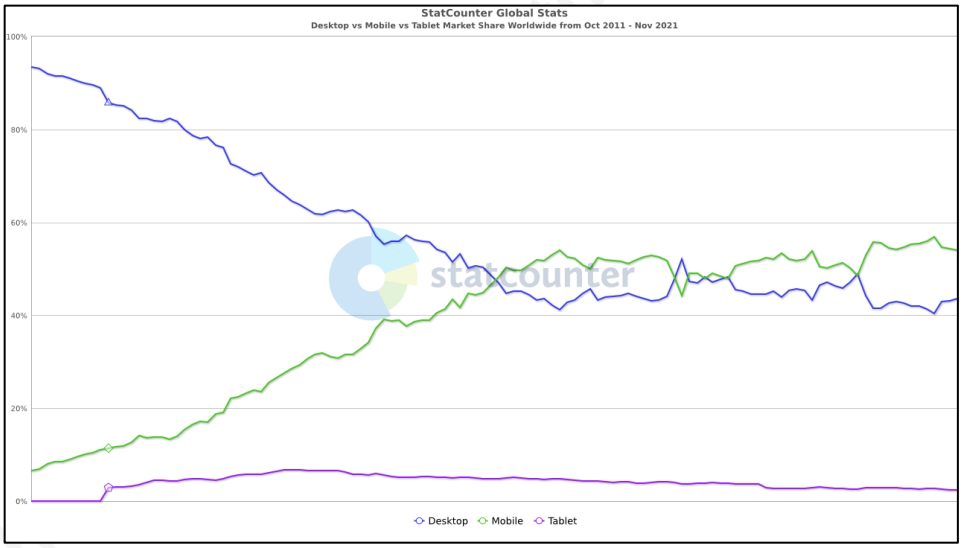


Figure 1.1-1 StatCounter Global Stats: Desktop vs Mobile vs Tablet Market Share

This shift would likely not have taken place if the mobile browsing experience was not comparable or better than the desktop experience. Mobile operating systems and browsers needed to provide the same functionality and ease of use that users had come to expect on their desktop counterparts.

By 2010, desktop browsers had evolved to include plugins, add-ons, internal cache, advanced bookmarking, and navigation capabilities, all to improve the browsing experience and entice users to switch from a competing browser. Mobile browsers followed the same path, resulting in web browsers, both desktop and mobile, that were increasingly coupled to the underlying operating system and that generated persistent browsing artifacts, thereby decreasing the privacy of users' online browsing behaviors.

Author Name, email@address

1.2. The Desire For Privacy

Recent statistics suggest that Google's Android OS accounts for over 70% of the worldwide smartphone market share (Figure 1.2-1). According to the popular statistic tracking website StatCounter, Google Android's market share has remained relatively consistent over the past two years (see below).

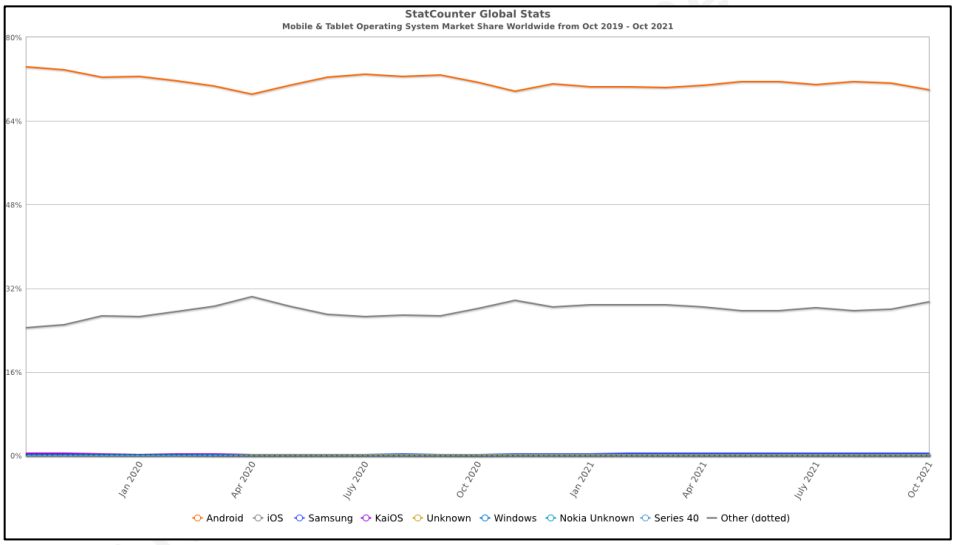


Figure 1.2-1 StatCounter Global Stats: Mobile and Tablet Operating System Market Share

Along with the general increase in mobile device and smartphone usage, there has been an increase in the desire for anonymity and privacy while connecting to the internet. Personal VPN services and private web browsing applications have helped to address these larger privacy concerns. At the same time, law enforcement officials need to be able to forensically extract data that may be used to prosecute and convict criminals based on their web browsing history and behavior. VPN services and private web browsing applications have hindered these processes. However, previous research has indicated that there may yet be artifacts that can be recovered from mobile devices.

The possible generation and persistence of web browsing artifacts associated with private browsing applications are of particular concern to those who place a high value on personal online privacy. Recent research, (DuckDuckGo, 2017) and (Habib et al., 2018), suggests that an increasing number of internet users are adopting private web browsing practices daily to secure their online browsing habits. The growing expectation that one's online activity should be secured from prying eyes could be compared to the expectation

of personal privacy within one's home. Individuals with this perspective may believe that private browsing applications completely mask their online activity. DuckDuckGo (2017) suggests that as much as 66% of private browser users overestimate the protection that private browsing provides. Admittedly, this overestimation is primarily driven by a misunderstanding of how much network traffic remains exposed during private browsing. However, users of private browsers will likely be surprised that there may well be artifacts that remain on their phones post private browsing.

1.3. The law enforcement perspective

While the public has displayed an increased desire to secure their online browsing behavior, the law enforcement community has long recognized the potential investigative value of convenient web browser features. As mentioned previously, such features were designed to improve the online browsing experience by tracking and retaining an abundance of user-related information. Benson (2018) indicated that due to the widespread adoption of HTTPS, relying solely on network traffic is no longer enough for investigators and incident responders. To build a more complete picture, investigators need to include a robust review of browser artifacts on endpoint devices and extract all possible data to effectively reconstruct the subject's online activities. The increased adoption of private browsing applications and features substantially decreases the ability to create a pattern of online behavior and build an effective case that would stand up to legal scrutiny in court.

1.4. Relevance to the cybersecurity community

Both privacy-conscious users and investigative officials have a vested interest in understanding whether persistent artifacts are generated during private browsing. Popular Android web-browsing applications include Chrome and Firefox, which both have a private browsing mode. Additionally, DuckDuckGo and Tor Browser are popular web browsers that are used specifically to browse the internet privately and anonymously. This project seeks to determine what, if any, filesystem-based artifacts can be retrieved from an Android phone that uses each of the four browsers mentioned and presents the findings. While similar research efforts have been conducted, they focused primarily on the Windows OS environment, or more recently on smartphone memory forensics. This

research focuses on four popular browsers used for private browsing: Chrome, Firefox, DuckDuckGo, and Tor Browser, running on Android OS. This research effort will help inform the larger cyber security and law-enforcement communities, as well as traditional users who have a particular interest in securing their mobile web browsing.

2. Literature Review

2.1. Private web browsing

According to research conducted by DuckDuckGo, most people misunderstand what private browsing entails (DuckDuckGo, 2017). The research defines Private Browsing as a system of web browsing that clears browsing history and file cache after use. According to Internet Security firm, Norton, when using a private browser, your browsing history, search records, and cookies are not retained (Johansen, 2020). There are numerous other definitions of "private browsing." However, the general expectation seems to be that any records associated with a user's browsing activity will be erased when the browser is closed. The following subsections provide an overview of the private browsing capabilities provided by each of the browsers in this research effort.

2.1.1. Chrome

Google refers to its private browsing mode in Chrome as "Incognito." Google clearly outlines what information is, and is not, recorded when using Chrome in Incognito mode. The following screenshot is taken from Google's support website:

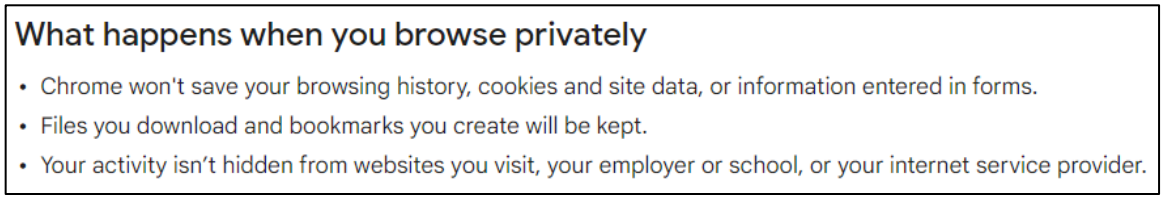


Figure 2.1-1 Screenshot from Google's website (https://support.google.com/chrome/answer/95464)

Additionally, Google states that cookies and site data are remembered while a user browses the internet privately. The data is then deleted when the user closes all incognito windows. According to Asim et al (2019), in normal browsing mode, Chrome stores the following forensically relevant data in the associated paths:

	Forensics Data	Path
Chrome	Web History	com.android.chrome/app˙chrome/Default/History
	Bookmarks	com.android.chrome/app˙chrome/Default/ bookmarks
	Cookies	com.android.chrome/app˙chrome/Default/cookies
	Local Storage	com.android.chrome/app˙chrome/Default/Local Storage
	User Credentials	com.android.chrome/app˙chrome/Default/Login˙Data
	Sync Data	com.android.chrome/app˙chrome/Default/ SyncData
	Saved Pages	com.android.chrome/app˙chrome/Default/Offline Pages/metadata/offlinepages.db com.android.chrome/app˙chrome/Default/Offline Pages/archives/
	Recent Tabs	com.android.chrome/app˙tabs/0
	Favourite icons	com.android.chrome/app˙chrome/Default/favicons
	Most Visted Websites	com.android.chrome/app˙chrome/Default/Top Sites.db
	Search Keywords	com.android.chrome/app˙chrome/Default/History
	User Preferences	com.android.chrome/app˙chrome/Default/Preferences
	Auto-Complete	com.android.chrome/app˙chrome/Default/Web Data
	Security Certificate Settings	com.android.chrome/app˙chrome/Default/Origin Bound Certs
	Cache	com.android.chrome/cache/
	Session	com.android.chrome/app˙chrome/Default/Session Storage/

Figure 2.1-2 Chrome browsing artifact locations on Android (Asim et al., 2019)

2.1.2. Firefox

Mozilla refers to its private browsing mode in Firefox as "Private." Like Google, Mozilla also clearly outlines what private browsing includes. The following screenshot is taken from Mozilla's website:

> Private Browsing does not save your browsing information, such as history and cookies, and leaves no trace after you end the session. Firefox also has Enhanced Tracking Protection, which prevents hidden trackers from collecting your data across multiple sites and slowing down your browsing.

Figure 2.1-3 Screenshot from Mozilla's website (https://support.mozilla.org/en-US/kb/private-browsing-use-firefox-without-history)

Mozilla is also very specific in stating that new passwords and bookmarks users create and save during Private browsing will in fact be saved, as well as any files that users chose to download to their computer. This is in line with Google's Incognito browsing mode. According to Asim et al (2019), in normal browsing mode Firefox stores the following forensically relevant data in the associated paths:

	Forensics Data	Path
Firefox	Web History	org.mozilla.firefox/files/mozilla/xxxxxxxx.default/ browser.db
	Bookmarks	org.mozilla.firefox/files/mozilla/xxxxxxxx.default/ browser.db
	Cookies	org.mozilla.firefox/files/mozilla/xxxx.default/cookies.sqlite
	Local Storage	org.mozilla.firefox/files/mozilla/yybtc8zi.default/ storage/
	User Credentials	org.mozilla.firefox/files/mozilla/xxxxxxxx.default/signons.sqlite
	Sync Data	
	Saved Pages	sdcard/Android/data/org.mozilla.firefox/files/Download
	Tabs Info	org.mozilla.firefox/files/mozilla/xxxxxxxx. default/sessionstore.js
	Favicons	org.mozilla.firefox/cache/icons
	User Preference	org.mozilla.firefox/files/mozilla/xxxxxxx.default/Prefs.js
	Search engines	org.mozilla.firefox/files/mozilla/xxxxxxx.default/search.json.mozlz4
	Auto-complete history	org.mozilla.firefox/files/mozilla/xxxxxxx.default/formhistory.sqlite
	DOM storage	org.mozilla.firefox/files/mozilla/xxxxxxx.default/webappsstore.sqlite
	Security certificate settings	org.mozilla.firefox/files/mozilla/xxxxxxx.default/cert9.db
	Cache	org.mozilla.firefox/cache/xxxxxxxx.default
	Session	org.mozilla.firefox/files/mozilla/yybtc8zi.default/sessionstore.js

Figure 2.1-4 Firefox browsing artifact locations on Android (Asim et al., 2019)

Author Name, email@address

2.1.3. DuckDuckGo

DuckDuckGo began as an alternative to commonly used search engines like Google, Yahoo, and Bing. The creators of DuckDuckGo were originally, and still are, very concerned about "search leakage", defined as the sharing of a user's search terms with the website that the user clicked on. According to the DuckDuckGo website (https://duckduckgo.com/privacy#s4), the company addresses the oversharing of personal search data by redirecting the request in a proprietary way that does not forward a user's search terms to other sites.

In 2018, DuckDuckGo released a mobile web browser application for Android, which was designed to be inherently private and included its proprietary private search engine technology, a key differentiator between traditional browsers and DuckDuckGo. The following screenshot is taken from DuckDuckGo's website:

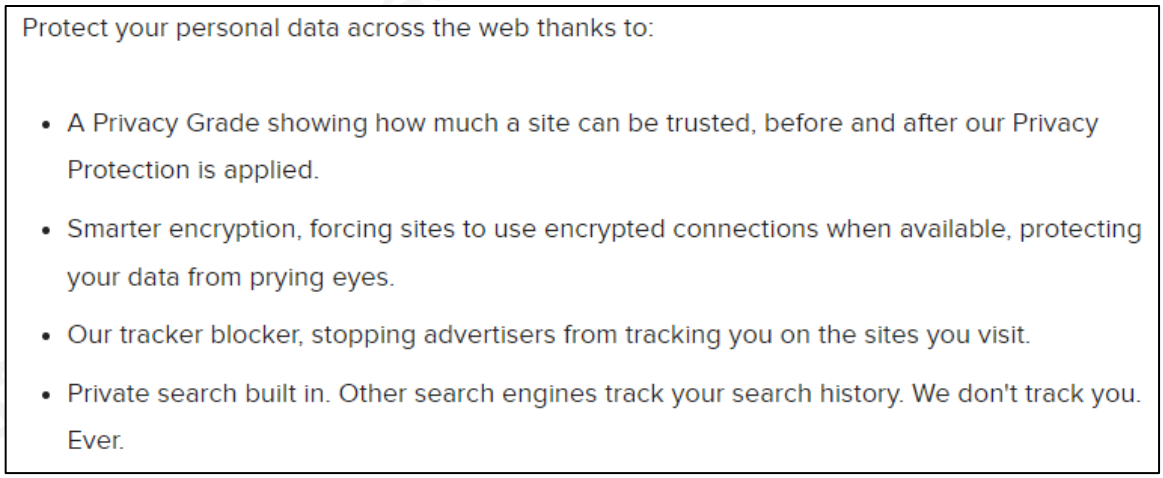


Figure 2.1-5 Screenshot from DuckDuckGo's website (https://help.duckduckgo.com/duckduckgo-help-pages/mobile/android/)

The description provided in the screenshot above is all that could be located at the DuckDuckGo website regarding the details of the privacy provided by the mobile browser application. One of the more popular features of the mobile application is the "Fire Button." The "Fire Button" is located at the top of the application to the right of the URL search bar and when clicked/pressed, immediately clears all open tabs and browsing data. Previous research focusing on the DuckDuckGo Android mobile application is difficult to find. However, based on the efforts of this research, DuckDuckGo artifacts on

Android can be found in the following location:

`/data/data/com.duckduckgo.mobile.android/`

2.1.4. Tor Browser

Like DuckDuckGo, the Tor Project began with privacy and anonymity in mind. The concept of "onion routing," the "or" in "Tor," was to encrypt traffic through multiple nodes to prevent anyone viewing the traffic from knowing where the traffic originated, and the intended destination. This idea has since grown in popularity, and the Tor project has expanded not only to include a robust international network of traffic-anonymizing nodes, but also a mature, multi-OS browser that incorporates the private browsing functionality of traditional browsers like Chrome and Firefox. The following screenshot is taken from the Tor Project website:

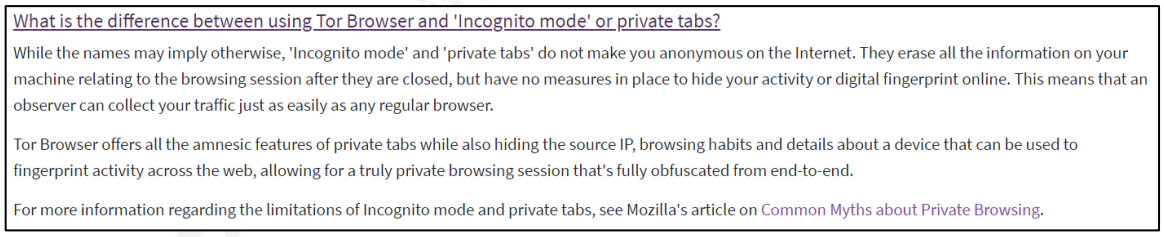


Figure 2.1-6 Screenshot taken from the Tor Project's website (https://support.torproject.org/)

The description above suggests that the Tor Browser provides all the functionality of traditional private browsing, as well as the Tor Project's ability to anonymize internet traffic in one convenient application. It should be noted that the version of the Tor Browser used in this research is based on Mozialla's Firefox and includes DuckDuckGo as the default search engine. According to the SANS Android Third-Party Apps Forensics Reference Guide (https://www.sans.org/posters/android-third-party-apps-forensics/) Tor Browser artifacts on Android can be found in the following locations:

`/data/org.torproject.torbrowser/app_torservice/.tor/`

`/data/org.torproject.torbrowser/files/mozilla/<ID>.default/`

2.2. Previous Research Efforts

Different browsers have different implementations of private browsing, and users should be aware of exactly what each browser provides during a "private browsing" session. To that end, there have been many research efforts revolving around the notion

of private browsing. Previous research efforts focused primarily on the Windows OS environment because of the commanding market share that Microsoft holds in personal computers, specifically desktops and laptops. Pretorius (2017) provides one of the more robust examples of such research in which the research team goes into great depth exploring and cataloging the various Windows OS locations where persistent browsing artifacts are found. Ohana (2013) provides an older, yet comprehensive view of the persistent browsing artifacts on Windows 7.

Yet, other research efforts have focused on the Android OS mobile space, but with the emphasis on extracting artifacts from memory. Younis et al. (2021) provide a very recent example of such research that is both comprehensive and complete. And Flowers et al. (2016) provide an example that combines the analysis of both memory and filesystem-based artifacts. This research relied heavily on ideas and methodologies from these and other previous efforts, all of which are included as references. The following section details the specific methodology used for this study.

3. Research Methodology

3.1. High-Level Approach

The research design was largely quantitative in nature and based on the methodology used in a previous research effort at Purdue University (Gabet, 2016). A virtual lab environment was established to create the conditions to collect artifacts from the four mobile browsers included in the experiment (Chrome, Firefox, DuckDuckGo, and Tor Browser). A modified Computer Forensics Field Triage Process (CFFTP) model (Rogers et al., 2006) served as the framework for the artifact collection process.

Six identical virtual machines (VMs) were created for VirtualBox. The first VM was used as the baseline and a control variable against which subsequent variables were compared. The second VM was used as an additional control variable identify artifacts that would be generated by a normal Chrome browsing session. The remaining four VMs were used to generate artifacts and data associated with the private web browsing functionality of the four tested browsers.

Author Name, email@address

A test browsing script was followed and executed on each VM, except for the first VM which was the control. Upon completion of the test script, FTK Imager was used to create an EnCase forensics image file (.E01) of the VM's virtual hard disk (VHD) for follow-on analysis. The resulting Encase image file was first analyzed using the recently released Bulk_extractor 2.0 tool, and the results were stored in a specific folder for follow-on analysis using Linux command-line tools. The results of the Linux command line analysis were then tabularized and compared to the results from the control variables. Finally, the Encase image file was analyzed using Autopsy, and the results were manually reviewed for the existence of artifacts related to the activities conducted in the test script.

3.2. Lab environment details

The table below contains a list of the software and version numbers used in the lab environment.

Table 3.2-1

Software	Version
Host OS – Windows 10 Professional	20H2 OS Build: 19042.1348 Windows Feature Experience Pack 120.2212.3920.0
Oracle Virtual Box	6.1
Android x86 *	9.0-r2
FTK Imager	4.5.0.3
Kali Linux (analysis machine)	2021.3
Autopsy	4.19.2
Android Google Chrome	96.0.4664.45
Android Mozilla Firefox	94.1.2
Android DuckDuckGo	5.102.3
Android Tor Browser	10.5.9 (91.2.0-Release)
Bulk_extractor	2.0

* Note: Android x86 version 9.0-r2 used because it was the most stable release available at the time of the research

The lab environment utilized Oracle Virtual Box to emulate the Android x86 environment. Although not the preferred scenario, Android x86 version 9.0-r2, a port of Android Pie, was used because it was the most stable release available for the study. It should be noted that at the time of the study the most widely used version of Android was version 11 and represents roughly 35% of the worldwide Android install base, compared

to roughly 14% for Pie (StatCounter Global Stats., 2021) However, there were no stable ports for Android 10, 11, or 12 available. All six Android VM configurations were as follows:

- Processor: 2 Processors
- Memory: 4 GB
- Storage: 14 GB
- VM File Format: Virtual Hard Disk (VHD)
- OS: Android x86 version 9.0-r2
- Installed browsers: Chrome, Firefox, DuckDuckGo, Tor Browser

3.3. The Test Script

Except for the first baseline VM, the following test script was followed:

1. Launch VM
2. Open the browser in private mode (for common browsers) or normal mode (for enhanced privacy browsers).
3. Open 7 tabs
4. Navigate to each URL from Table 3.3-1 on the first 4 tabs and bookmark the URL as indicated in the table
5. On the 5th tab navigate to hotmail.com, bookmark it and log in to the website using credentials from Table 3.3-2 (save credentials in the browser if possible)
6. On the 6th tab navigate to yahoo.com, bookmark it and log in to the website using credentials from Table 3.3-2 (save credentials in the browser if possible)
7. On the 7th tab navigate to twitter.com, bookmark it and log in to the website using credentials from Table 3.3-2 (save credentials in the browser if possible)
8. Send test email from HOTMAIL to YAHOO using Table 3.3-4

9. Send test email from YAHOO to HOTMAIL using Table 3.3-4
10. Read test email in YAHOO
11. Read test email in HOTMAIL
12. Search 10 terms using an online search engine as indicated in Table 3.3-3
13. Close the browser window and all tabs (For DuckDuckGo, use the "Fire Button" to wipe all browsing data and clear all tabs before closing the browser window)
14. Save and shut down VM

Table 3.3-1

Website to visit	Bookmark
https://checkboxolympics.com/	Yes
https://guardian.co.tt/	Yes
https://www.wikirecreation.com/	No
http://burymewithmymoney.com/	No

Table 3.3-2

Website	User Account	Password
www.hotmail.com	JoePandaRocks@hotmail.com	sfkjd@,Jrs5Gs$
www.yahoo.com	JoePandaRocks@yahoo.com	sfkjd@,Jrs5Gs$
www.twitter.com	JoePandaRocks@hotmail.com	sfkjd@,Jrs5Gs$

Table 3.3-3

Search Engine	Search String
www.google.com	Connection_search_research_test

Author Name, email@address

www.google.com	Convict_search_research_test
www.yahoo.com	Symptom_search_research_test
www.yahoo.com	Deprive_search_research_test
www.youtube.com	Flood_search_research_test
www.youtube.com	Nightmare_search_research_test
www.youtube.com	Craftsman_search_research_test
www.bing.com	Tolerate_search_research_test
www.bing.com	Flow_search_research_test
www.bing.com	Spill_search_research_test

Table 3.3-4

Email	Destination	Subject	Body
www.hotmail.com	JoePandaRocks@yahoo.com	unanimous2_email_research_test	This is a research email sent from Hotmail to Yahoo. Key word: unanimous2_email_research_test
www.yahoo.com	JoePandaRocks@hotmail.com	continental2_email_research_test	This is a research email sent from Yahoo to Hotmail. Key word: continental2_email_research_test

3.4. Data Extraction and Population

After completing the test script and shutting down the VM, the process of extracting the data began by using FTK Imager to create an EnCase image file (.E01) of the VM's virtual hard disk (VHD). FTK Imager is a free imaging tool by Exterro that creates perfect forensic images of computer data (Exterro, 2021) for follow-on analysis. The Encase image standard is part of a popular software suite of forensics investigation products provided by OpenText Security. The standard is broadly accepted in law enforcement circles and recognized for its ability to preserve data from a wide variety of devices (OpenText Security, n.d.).

Both Exterro and OpenText Security also provide follow-on forensics analysis tools in FTK® Forensic Toolkit and EnCase Forensic respectively at varying costs. Autopsy is a free and open-source alternative provided by the Autopsy Digital Forensics team, with much of the same functionality as its commercial-grade counterparts. Running Autopsy against a target EnCase image yields a wealth of aggregated data and reports. However, Autopsy did not provide a simple way to access the raw data for the quantitative analysis required for this research effort.

To provide a semi-aggregated summary of the raw data for quantitative analysis, the research effort utilized Bulk_extractor. Bulk_extractor is a high-performance digital forensics exploitation tool (https://github.com/simsong/bulk_extractor) written to analyze various types of input, extract structured information, and store the results in text files. These text files can then be used in follow-on analysis. According to its developers (Garfinkel, 2013), Bulk_extractor differentiates itself from other forensics tools by probing each byte of data to decode and recursively extract structured data into a specified output directory. The command used was as follows:

```
bulk_extractor -o output_directory target_image
```

where *output_directory* is the path of the directory where the results will be stored, and *target_image* is the path of the image to be processed.

Figure 3.4-1 below shows an example of the output directory.

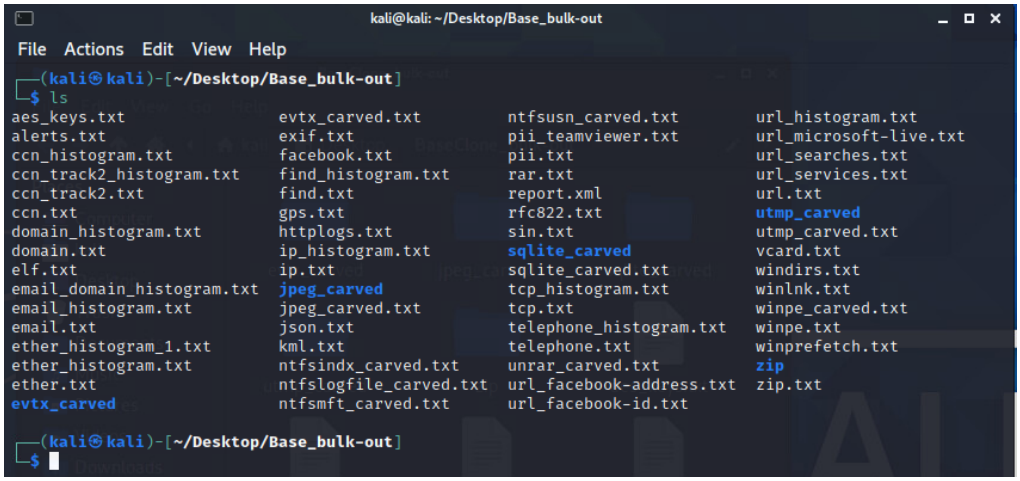


Figure 3.4-1 Sample Bulk_extractor output directory

This enabled the research effort to perform the analysis in a standard, repeatable manner to generate statistics for quantitative analysis. Table 3.4-1 below specifies the statistics used for comparison across the test VMs after completing the test script.

Table 3.4-1

Bulk_extractor Feature File	Description
domain_histogram.txt	domains visited on the VM and the number of times each was visited
domain.txt	domains found on the VM, including dotted-quad addresses found in text
email_domain_histogram.txt	email domains used on the VM, and the number of times each was used
email-histogram.txt	email addresses used on the VM, and the number of times each was used
email.txt	email addresses used on the VM
json.txt	JSON file structures identified on the VM
url_searches.txt	histogram of terms used in Internet searches from services such as Google, Bing, Yahoo, and others.
url_services.txt	histogram of the domain name portion of all the URLs found on the VM.
sqlite_carved	directory of extracted sqlite.db structures
url_histogram.txt	URLs, typically found in browser caches, email messages, and pre-compiled into executables and the number of times each was used.
url.txt	URLs, typically found in browser caches, email messages, and pre-compiled into executables.

After generating the output directory with Bulk_extractor, two custom commands per search term were run at the Linux command line to recursively parse the output directory and generate the final statistics for comparison across the test VMs. The commands were as follows:

1. `grep -a -R search_string output_directory | cut -d : -f 1 | sort -d | uniq -c`
2. `grep -a -R search_string output_directory | cut -d : -f 1 | grep sqlite_carved | sort -d | wc -l`

where *search_string* is the specific search string to recursively search for and *output_directory* is the target output directory. The search strings used were identical to the "Search String" and "Subject" columns in Tables 3.3-3 and 3.3-4 respectively.

4. Analysis and Findings

The results of the custom Linux commands were collected and entered into a table for final review and comparison between VMs. The following sections depict the tabularized data for each VM. The tables show the findings for each search string by Bulk_extractor feature file. Autopsy analysis results are also included where appropriate.

4.1. Base VM Control

This base control case represents the location of artifacts found on the base VM without running the test script.

	Search term	domain_histogram.txt	Domain.txt	email_domain_histogram.txt	email_histogram.txt	email.txt	json.txt	url_searches.txt	url_services.txt	sqlite.db	url_histogram.txt	url.txt	Base VM Delta
BASE	checkboxolympics.com	-	-	-	-	-	-	-	-	-	-	-	-
	guardian.co.tt	-	-	-	-	-	-	-	-	-	-	-	-
	wikirecreation.com	-	-	-	-	-	-	-	-	-	-	-	-
	burymewithmymoney.com	-	-	-	-	-	-	-	-	-	-	-	-
	JoePandaRocks@hotmail.com	-	-	-	-	-	-	-	-	-	-	-	-
	JoePandaRocks@yahoo.com	-	-	-	-	-	-	-	-	-	-	-	-
	Hotmail.com	1	1	1	1	1	-	-	-	-	-	-	**5**
	Yahoo.com	89	345	1	2	6	2	1	89	-	-	341	**876**
	Twitter.com	2	17	-	-	-	3	-	2	-	5	19	**48**
	Connection search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Convict search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Symptom search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Deprive search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Flood search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Nightmare search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Craftsman search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Tolerate search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Flow search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Spill search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	unanimous_email_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	continental_email_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	TOTAL	**92**	**363**	**2**	**3**	**7**	**5**	**1**	**91**	**-**	**5**	**360**	**929**

Figure 4.1-1 Bulk_extractor results for Base VM before browsing activity

As expected, there are no artifacts associated with the specific search terms. Additionally, the existence of artifacts associated with the Hotmail.com, Yahoo.com, and Twitter.com domains should be noted. This is likely due to activity that took place on the original base image. The Autopsy results were similar.

4.2. Base VM Post Test Script using Chrome in Normal Mode

This control case was used to depict the artifacts that would be generated using normal Chrome browsing.

	Search term	domain_histogram.txt	Domain.txt	email_domain_histogram.txt	email_histogram.txt	email.txt	json.txt	url_searches.txt	url_services.txt	sqlite.db	url_histogram.txt	url.txt	Base VM Delta
BASE AFTER TEST SCRIPT	checkboxolympics.com	2	286	-	-	-	61	4	2	4,227	34	317	**4,933**
	guardian.co.tt	11	3,260	-	-	-	83	4	11	6,298	574	5,165	**15,406**
	wikirecreation.com	4	844	-	-	-	59	5	4	6,971	143	902	**8,932**
	burymewithmymoney.com	3	268	-	-	-	42	2	3	5,613	30	321	**6,282**
	JoePandaRocks@hotmail.com	-	8	-	3	8	2	-	-	81	4	4	**110**
	JoePandaRocks@yahoo.com	-	36	-	2	47	6	-	-	71	4	4	**170**
	Hotmail.com	3	148	1	5	12	14	8	3	2,267	46	174	**2,676**
	Yahoo.com	173	4,501	2	15	71	154	49	177	2,997	1,082	5,301	**13,646**
	Twitter.com	31	3,771	1	69	301	74	7	31	7,998	673	3,882	**16,790**
	Connection search_research_test	-	-	-	-	-	-	3	-	1,948	19	118	**2,088**
	Convict search_research_test	-	-	-	-	-	-	3	-	1,376	16	72	**1,467**
	Symptom search_research_test	-	-	-	-	-	4	9	-	549	20	69	**651**
	Deprive search_research_test	-	-	-	-	-	4	5	-	655	8	9	**681**
	Flood search_research_test	-	-	-	-	-	7	1	-	302	9	46	**365**
	Nightmare search_research_test	-	-	-	-	-	5	1	-	300	9	1	**316**
	Craftsman search_research_test	-	-	-	-	-	3	1	-	38	5	9	**56**
	Tolerate search_research_test	-	-	-	-	-	-	1	-	754	4	59	**818**
	Flow search_research_test	-	-	-	-	-	-	1	-	571	5	56	**633**
	Spill search_research_test	-	-	-	-	-	-	1	-	429	5	42	**477**
	unanimous_email_research_test	-	-	-	-	-	-	-	-	36	-	-	**36**
	continental_email_research_test	-	-	-	-	-	-	-	-	32	-	-	**32**
	Base VM Delta	**135**	**12,759**	**2**	**91**	**432**	**513**	**104**	**140**	**43,513**	**2,685**	**16,191**	**76,565**

Figure 4.2-1 Bulk Extractor results for Base VM in Normal Mode

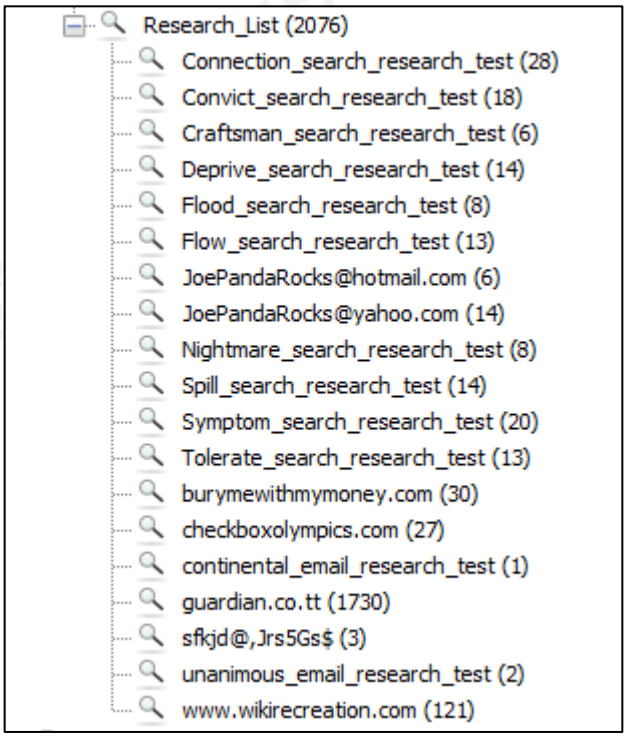


Figure 4.2-2 Autopsy results for Base VM in Normal Mode

To produce these results, the test script was followed on a clone of the Base VM using the Chrome Browser in normal mode. Overall, Bulk_extractor identified 8,242% more artifacts than the Base VM. Also, the Autopsy results to the left show a significant number of artifacts generated when executing the test script using Chrome in Normal mode.

4.3. Chrome Incognito Mode

On the surface, the Bulk Extractor Chrome Incognito browsing results were promising.

	Search term	domain_histogram.txt	Domain.txt	email_domain_histogram.txt	email_histogram.txt	email.txt	json.txt	url_searches.txt	url_services.txt	sqlite.db	url_historgram.txt	url.txt	Base VM Delta
CHROME INCOGNITO BROWSING	checkboxolympics.com	-	-	-	-	-	-	-	-	-	-	-	-
	guardian.co.tt	-	-	-	-	-	-	-	-	-	-	-	-
	wikirecreation.com	-	-	-	-	-	-	-	-	-	-	-	-
	burymewithmymoney.com	1	5	-	-	-	-	-	1	-	1	5	**13**
	JoePandaRocks@hotmail.com	-	-	-	-	-	-	-	-	-	-	-	-
	JoePandaRocks@yahoo.com	-	2	-	1	2	-	-	-	2	-	-	**7**
	Hotmail.com	1	1	1	1	1	-	-	-	-	-	-	-
	Yahoo.com	90	196	1	1	2	8	1	90	27	171	196	**(93)**
	Twitter.com	4	297	1	63	263	3	-	4	99	67	99	**852**
	Connection search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Convict search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Symptom search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Deprive search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Flood search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Nightmare search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Craftsman search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Tolerate search_research_test	-	-	-	-	-	-	1	-	1	1	1	**4**
	Flow search_research_test	-	-	-	-	-	-	1	-	1	1	1	**4**
	Spill search_research_test	-	-	-	-	-	-	1	-	4	1	5	**11**
	unanimous2_email_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	continental2_email_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Base VM Delta	**4**	**138**	**1**	**63**	**261**	**6**	**3**	**4**	**134**	**237**	**(53)**	**798**

Figure 4.3-1 Bulk Extractor results for Chrome Incognito

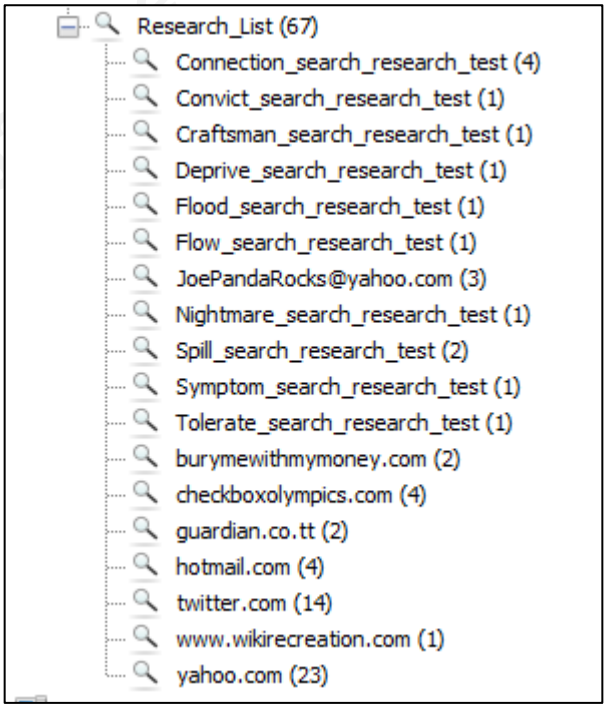


Figure 4.3-2 Autopsy results for Chrome Incognito

At a macro level, completing the test script using Chrome in Incognito mode generated 86% more artifacts than on the base VM, and only 2.2% of the artifacts generated when completing the script with Chrome in normal browsing mode. This is a significant improvement over traditional browsing. However, Autopsy was able to extract artifacts on almost all test script items. A deeper review of the Autopsy results reveals exactly how this was achieved.

Google rightly claims that all browsing history, searches, and cookies

are deleted once all Incognito windows are closed. Only bookmarks are saved when specifically requested by the user. The screenshots below confirm Google's statements.

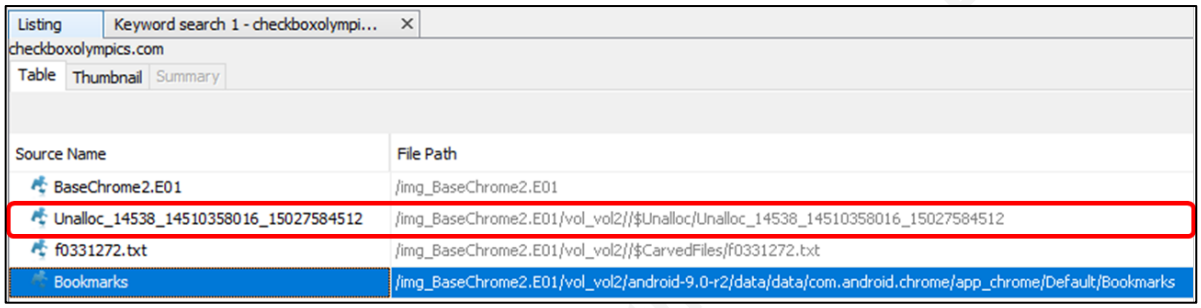


Figure 4.3-3 Screenshot of Autopsy search results for checkboxolympics.com for Chrome Incognito


```
},
"synced": {
  "children": [ {
    "date_added": "13285354407104607",
    "guid": "2535cf02-ae95-47d2-9252-4ea564b3c078",
    "id": "5",
    "name": "Checkbox Olympics",
    "type": "url",
    "url": "https://checkboxolympics.com/"
  }, {
    "date_added": "13285354473062801",
    "guid": "79ac2ab9-dc3c-45c5-9860-9224fb6ac1b7",
    "id": "6",
    "name": "Home - Trinidad Guardian",
    "type": "url",
    "url": "https://guardian.co.tt/"
  } ],
  "date_added": "13285354363115422",
  "date_modified": "13285354473062801",
  "guid": "4cf2e351-0e85-532b-bb37-df045d8f8d0f",
  "id": "3",
  "name": "Mobile bookmarks",
  "type": "folder"
}
```


Figure 4.3-5 Content of Bookmarks artifact

The line highlighted in blue in Figure 4.3-3 indicates the bookmarks found by Autopsy. Note the path indicated in the "File Path" column. This is one of the well-known paths where browsing artifacts are stored (See Figure 2.1-2). Figure 4.3-4 shows the actual content of the file. Note the existence of the two URLs specifically bookmarked in the test script.

The line circled in red in Figure 4.3-3 above represents unallocated space as the name suggests. According to the Autopsy website (https://sleuthkit.org/autopsy/docs/user-docs/3.1/uilayout_page.html) unallocated space are chunks of the file system that are currently not being used. This type of space often stores deleted files. It is this space from which Autopsy and other digital forensics tools parse and carve deleted files and artifacts. Figure 4.3-5 below shows the contents of the unallocated space block. Note the existence of all four test script URLs, regardless of whether they were bookmarked or not.

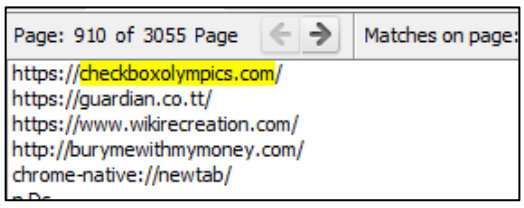


Figure 4.3-4 Unallocated space content

4.4. Firefox Private Browsing

Using Firefox in Private mode yielded similar, but improved, results to Chrome in Incognito mode.

	Search term	domain_histogram.txt	Domain.txt	email_domain_histogram.txt	email_histogram.txt	email.txt	json.txt	url_searches.txt	url_services.txt	sqlite.db	url_historgram.txt	url.txt	Base VM Delta
FIREFOX PRIVATE BROWSING	checkboxolympics.com	-	-	-	-	-	-	-	-	-	-	-	-
	guardian.co.tt	-	-	-	-	-	-	-	-	-	-	-	-
	wikirecreation.com	-	-	-	-	-	-	-	-	-	-	-	-
	burymewithmymoney.com	-	-	-	-	-	-	-	-	-	-	-	-
	JoePandaRocks@hotmail.com	-	-	-	-	-	-	-	-	-	-	-	-
	JoePandaRocks@yahoo.com	-	-	-	-	-	-	-	-	-	-	-	-
	Hotmail.com	1	1	1	1	1	-	-	-	-	-	-	-
	Yahoo.com	89	250	1	2	6	2	1	89	5	166	245	**(20)**
	Twitter.com	4	209	1	52	163	3	-	4	83	9	65	**545**
	Connection search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Convict search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Symptom search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Deprive search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Flood search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Nightmare search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Craftsman search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Tolerate search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Flow search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Spill search_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	unanimous2_email_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	continental2_email_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Base VM Delta	**2**	**97**	**1**	**52**	**163**	**-**	**-**	**2**	**88**	**170**	**(50)**	**525**

Figure 4.4-1 Bulk Extractor results for Firefox Private Browsing

Figure 4.4-2 Autopsy results for Firefox Private

Firefox in Private mode produced 57% more artifacts than the base VM and only 1.9% of the artifacts generated when completing the script with Chrome in normal browsing mode. It should be noted that Bulk Extractor was unable to find any artifacts related to the test scripts. Rerunning the test script confirmed the Bulk Extractor results. Autopsy confirmed the improvement in residual artifacts. However, Autopsy did find the persistent artifacts related to the URLs that were bookmarked in accordance with the test script (checkboxolympics.com and guardian.co.tt). See Figure 4.4-2.

Mozilla's statements regarding browsing artifacts in Private mode were confirmed by the test scripts. See the screenshots below.

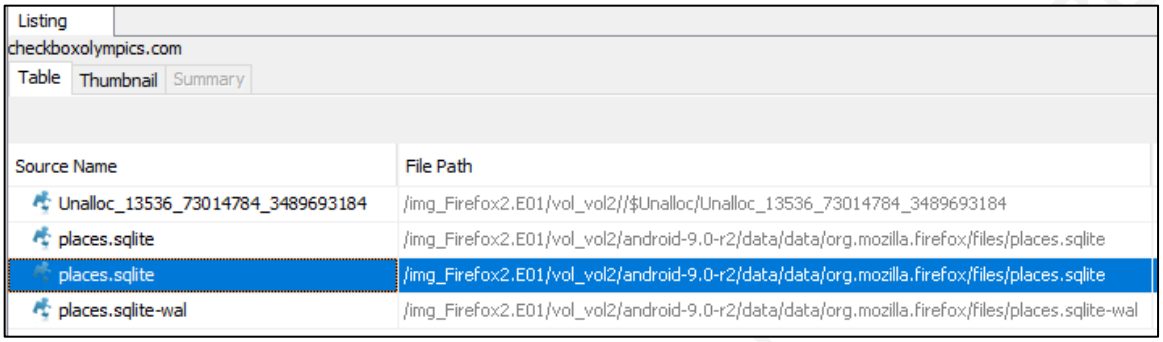


Figure 4.4-3 Screenshot of Autopsy search results for checkboxolympics.com for Firefox Private

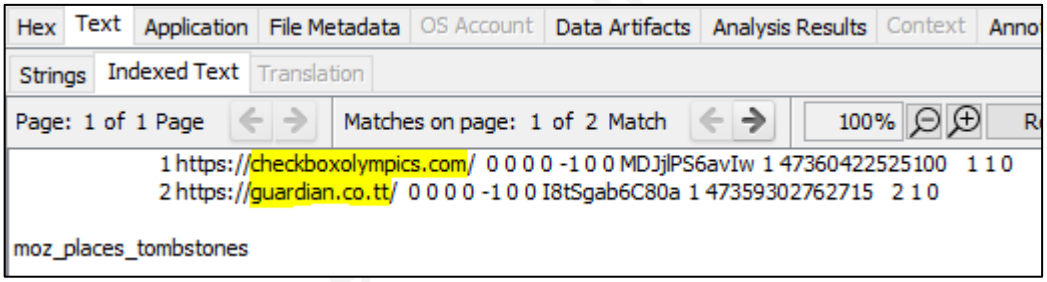


Figure 4.4-4 Content of the places.sqlite artifact in Firefox Private

4.5. Tor Browser

Like Chrome and Firefox, Tor Browser demonstrated a significant reduction in artifacts versus the base VM.

	Search term	domain_histogram.txt	Domain.txt	email_domain_histogram.txt	email_histogram.txt	email.txt	json.txt	url_searches.txt	url_services.txt	sqlite.db	url_histogram.txt	url.txt	Base VM Delta
Tor Browser	checkboxolympics.com	2	6	-	-	-	-	-	2	-	3	6	**19**
	guardian.co.tt	2	3	-	-	-	-	-	2	-	3	3	**13**
	wikirecreation.com	-	-	-	-	-	-	-	-	-	-	-	**-**
	burymewithmymoney.com	-	-	-	-	-	-	-	-	-	-	-	**-**
	JoePandaRocks@hotmail.com	-	-	-	-	-	-	-	-	-	-	-	**-**
	JoePandaRocks@yahoo.com	-	20	-	1	20	-	-	-	-	-	-	**41**
	Hotmail.com	1	1	1	1	1	1	-	-	-	-	-	**1**
	Yahoo.com	89	346	1	2	6	4	1	89	-	166	343	**171**
	Twitter.com	2	228	1	64	210	4	-	2	60	6	20	**549**
	Connection search_research_test	-	-	-	-	-	-	-	-	-	-	-	**-**
	Convict search_research_test	-	-	-	-	-	-	-	-	-	-	-	**-**
	Symptom search_research_test	-	-	-	-	-	-	-	-	-	-	-	**-**
	Deprive search_research_test	-	-	-	-	-	-	-	-	-	-	-	**-**
	Flood search_research_test	-	-	-	-	-	-	-	-	-	-	-	**-**
	Nightmare search_research_test	-	-	-	-	-	-	-	-	-	-	-	**-**
	Craftsman search_research_test	-	-	-	-	-	-	-	-	-	-	-	**-**
	Tolerate search_research_test	-	-	-	-	-	-	-	-	-	-	-	**-**
	Flow search_research_test	-	-	-	-	-	-	-	-	-	-	-	**-**
	Spill search_research_test	-	-	-	-	-	-	-	-	-	-	-	**-**
	unanimous2_email_research_test	-	-	-	-	-	-	-	-	-	-	-	**-**
	continental2_email_research_test	-	-	-	-	-	-	-	-	-	-	-	**-**
	Base VM Delta	**4**	**241**	**1**	**65**	**230**	**4**	**-**	**4**	**60**	**173**	**12**	**794**

Figure 4.5-1 Bulk Extractor results for the Tor Browser

Figure 4.5-2 Autopsy results for Tor Browser

Completing the test script with Tor Browser generated 85% more artifacts than the base VM and 2.2% of the artifacts generated when completing the script with Chrome in normal browsing mode. This was the expected result given that as previously mentioned, the latest Tor Browser was developed based on Mozilla's Firefox source code. Bulk Extractor found the expected bookmark artifacts. Autopsy also found the bookmark artifacts and confirmed the similarity to Firefox (note the path to the browser artifact sqlite database and the similarity in the data stored.)

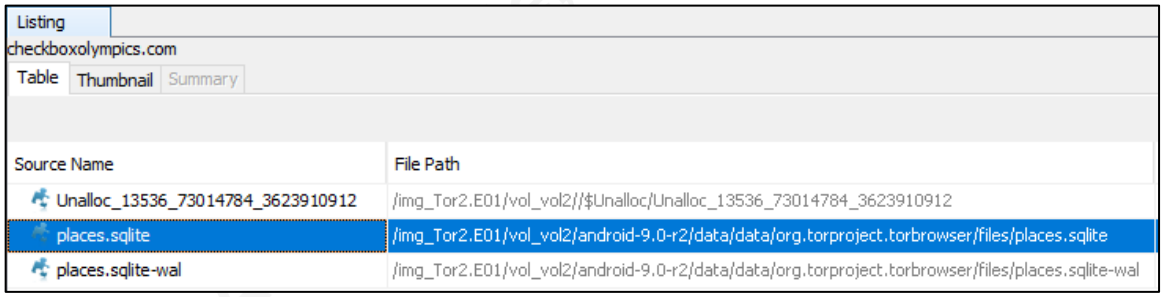


Figure 4.5-3 Screenshot of Autopsy search results for checkboxolympics.com for Tor Browser

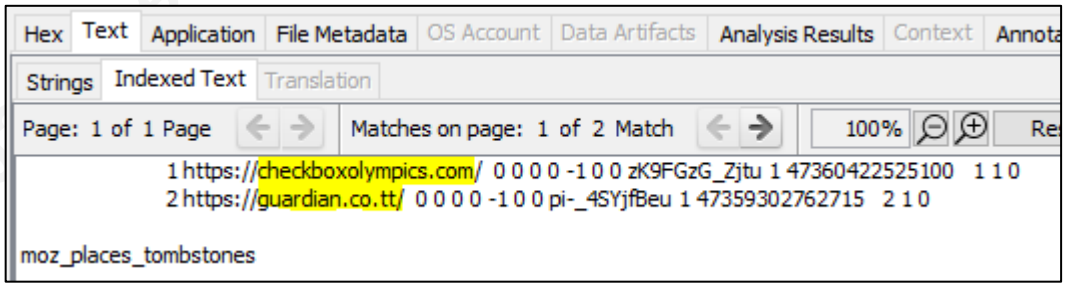


Figure 4.5-4 Content of the places.sqlite artifact in Tor Browser

4.6. DuckDuckGo

DuckDuckGo fared far worse than Chrome, Firefox, and Tor Browser.

	Search term	domain_histogram.txt	Domain.txt	email_domain_histogram.txt	email_histogram.txt	email.txt	json.txt	url_searches.txt	url_services.txt	sqlite.db	url_historgram.txt	url.txt	Base VM Delta
DuckDuckGo	checkboxolympics.com	1	20	-	-	-	-	-	1	42	5	20	**89**
	guardian.co.tt	3	69	-	-	-	5	-	3	229	41	80	**430**
	wikirecreation.com	-	-	-	-	-	-	-	-	-	-	-	-
	burymewithmymoney.com	1	9	-	-	-	-	-	1	9	7	9	**36**
	JoePandaRocks@hotmail.com	-	7	-	2	7	2	-	-	13	-	-	**31**
	JoePandaRocks@yahoo.com	-	2	-	1	2	-	-	-	4	1	2	**12**
	Hotmail.com	2	11	1	4	10	2	-	1	59	3	3	**91**
	Yahoo.com	123	873	1	1	2	44	20	123	745	453	913	**2,422**
	Twitter.com	14	902	1	65	201	19	-	25	522	297	820	**2,818**
	Connection search_research_test	-	-	-	-	-	2	4	-	8	19	24	**57**
	Convict search_research_test	-	-	-	-	-	-	3	-	4	10	16	**33**
	Symptom search_research_test	-	-	-	-	-	3	6	-	4	16	17	**46**
	Deprive search_research_test	-	-	-	-	-	4	5	-	2	8	9	**28**
	Flood search_research_test	-	-	-	-	-	1	1	-	3	2	2	**9**
	Nightmare search_research_test	-	-	-	-	-	1	1	-	7	5	5	**19**
	Craftsman search_research_test	-	-	-	-	-	1	1	-	3	2	2	**9**
	Tolerate search_research_test	-	-	-	-	-	1	1	-	1	1	1	**5**
	Flow search_research_test	-	-	-	-	-	1	1	-	1	1	1	**5**
	Spill search_research_test	-	-	-	-	-	1	1	-	21	2	19	**44**
	unanimous2_email_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	continental2_email_research_test	-	-	-	-	-	-	-	-	-	-	-	-
	Base VM Delta	**52**	**1,530**	**1**	**70**	**215**	**82**	**43**	**63**	**1,677**	**868**	**1,583**	**6,184**

Figure 4.6-1 Bulk Extractor results for DuckDuckGo

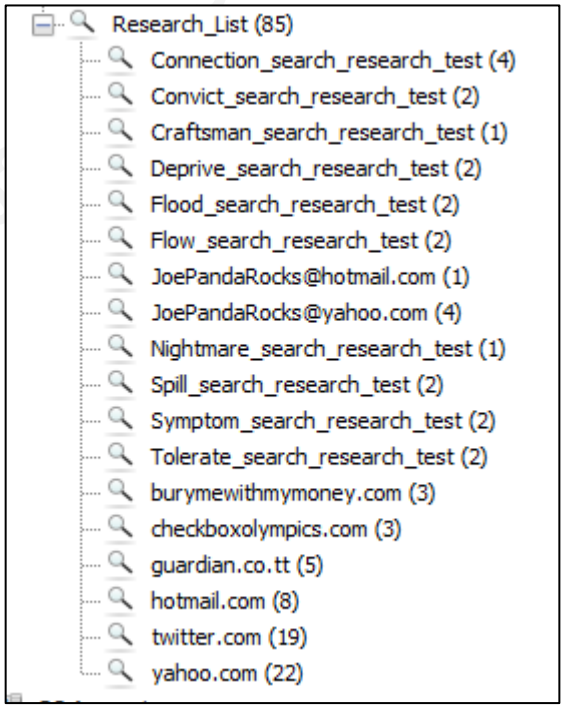


Figure 4.6-2 Autopsy results for DuckDuckGo

Completing the test script with DuckDuckGo and then clicking on the "Fire Button" generated 665% more artifacts than the base VM and 9.2% of the artifacts generated when completing the script with Chrome in normal browsing mode. While this performance is an improvement over normal browsing, the results were more than expected. Bulk Extractor found residual artifacts for most of the test script items, including regular searches using Google, Yahoo, YouTube, and Bing. Autopsy analysis yielded similar results (See Figure 4.6-2).

Figure 4.6-3 Screenshot of Autopsy search results for checkboxolympics.com for DuckDuckGo

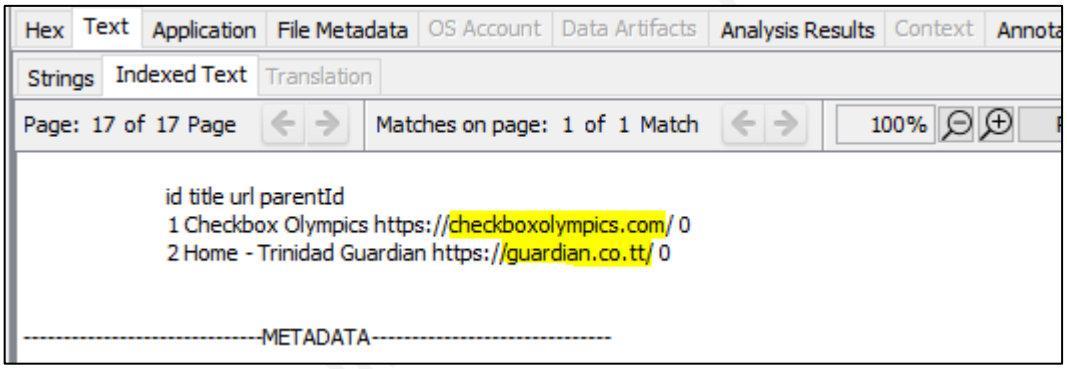


Figure 4.6-4 Content of the app.db artifact in DuckDuckGo

Autopsy identified the expected bookmark artifacts along with the associated persistent file path. However, it should also be noted, that much like Chrome Incognito mode, most of the artifacts were found in unallocated space (See examples in Figure 4.6-5).

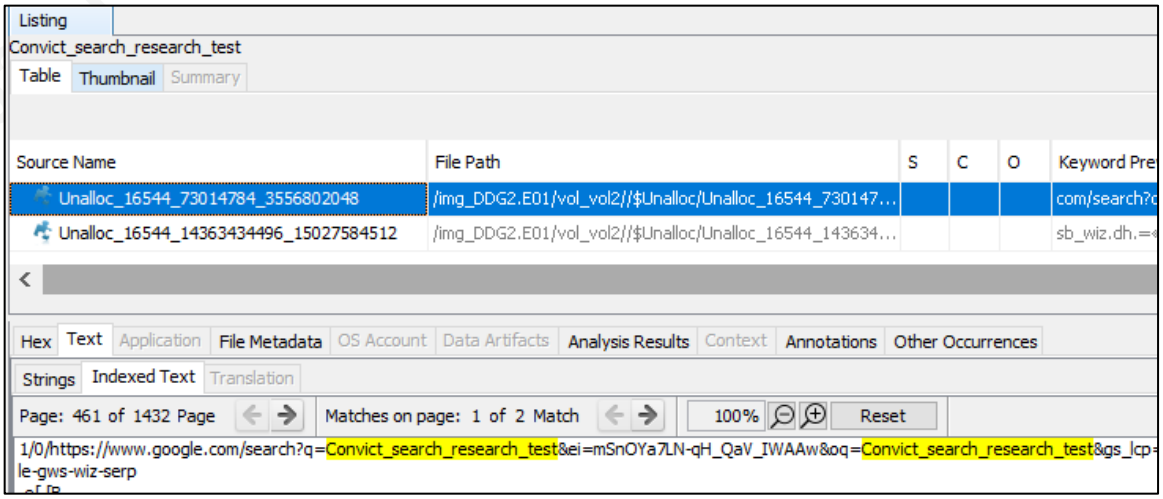
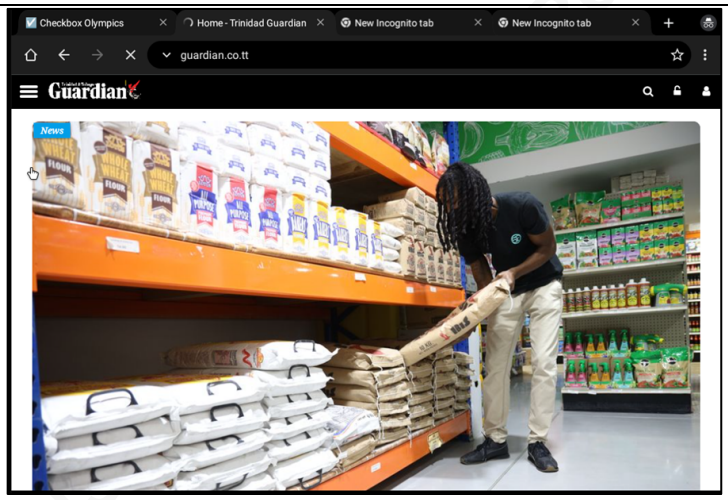
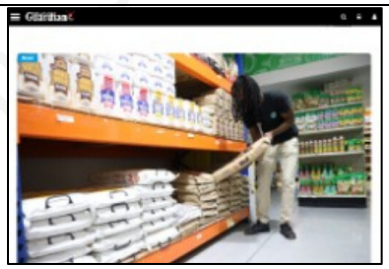
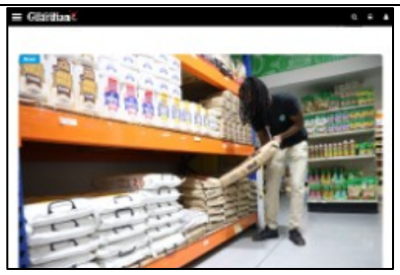


Figure 4.6-5 Screenshot of Autopsy search results for Convict_search_research_test for DuckDuckGo

Author Name, email@address

Additionally, both Bulk Extractor and Autopsy analysis revealed the existence of images that appear to be screenshots of browsing activity. The table below represents one of the examples found. The remaining examples can be found in Appendix A.

Table 4.6-1 Sample screenshots of test script URL and carved images for DuckDuckGo

<table>
<tr><td colspan="2">Manual screenshot taken during test script for URL: https://guardian.co.tt</td></tr>
<tr><td colspan="2"> 
</td></tr>
<tr><td>Bulk Extractor carved image</td><td>Autopsy carved image</td></tr>
<tr><td>  </td><td>  </td></tr>
</table>

Note that the images were not part of a persistent file and path. All forensically relevant images were carved from unallocated space, which indicates that DuckDuckGo did attempt to delete the pictures and remove them from the traditional filesystem. However, as demonstrated by this research, simple free tools are capable of carving and extracting the deleted files and, in the case of DuckDuckGo, revealing browsing habits as well as the content of emails.

Author Name, email@address

5. Conclusion

This research aimed to explore the effectiveness of private browsing at mitigating the generation and persistence of filesystem-based artifacts on Android-based mobile devices. Four popular browsers used by advocates of private browsing were studied. The four browsers were Chrome (Incognito mode), Firefox (Private mode), DuckDuckGo, and Tor Browser. The research methodology called for two controls, and a standardized test script which was run against each of the four browsers and one of the controls. An EnCase image of each filesystem was then captured for follow-on low-level analysis by Bulk_extractor and Autopsy. The results of the analyses were then collected and summarized.

5.1. Key Findings

1. None of the tested browsers completely avoided the generation and persistence of filesystem-based artifacts.
2. Different browsers performed well at different aspects of private browsing.
3. DuckDuckGo generated a significantly higher number of artifacts than expected, as well as images that have the potential to reveal detailed browsing habits and email contents.
4. Overall, Firefox performed the best of the four browsers tested and generated the least number of artifacts. However, Tor Browser, which as previously mentioned is based on the Firefox source code, also performed very well. Given Tor Browser's additional network traffic masking capabilities, it would be the best choice for users who wish to minimize their local device and network footprints.
5. All browsers generated additional unexpected artifacts.
6. While there are observable reductions in persistent filesystem-based artifacts, a low-level analysis of an Android device's disk image will reveal browsing activities when one of the four tested private browsers is used.

It should be noted that while this research does identify various instances of artifacts existing in unallocated space, the reality is that gaining access to full disk images on modern Android devices is a difficult task. This is because according to the Android

Open Source project (https://source.android.com/security/encryption/full-disk) Google introduced full-disk encryption beginning with Android 4.4, and file-based encryption (FBE) in Android 7.0 (https://source.android.com/security/encryption). Currently, any devices running Android 10 or later only support FBE. While the scope of this research does not cover the nuances of Android device encryption, it is important to understand that on modern devices, even with root access, it is very difficult to gain a full disk image that includes unallocated space. As such, achieving results similar to the research findings using physical Android devices may prove challenging.

5.2. Research Gaps and Recommendations

Four important gaps were identified as the research evolved:

1. Android version: Originally an attempt was made to utilize Android 10 or Android 11. However, it was eventually determined that of the tested Android emulators for PC, which included Bliss OS, Prime OS, Phoenix OS, and Android x86, only Android x86 offered a truly stable option, of which the most recent release was based on Android 9. As mentioned previously, Android 9, commonly known as Pie, only represents 14% of the worldwide Android user-base, and that percentage is trending downwards. This research runs the risk of quickly becoming irrelevant. Once a newer version of Android x86 is available, the research should be enhanced and repeated.
2. Other browsers: Although the four browsers researched are popular with online privacy advocates, several other private browsers enjoy a robust following. They include Brave Browser, Cake Browser, Dolphin Zero, Frost Browser, and InBrowser. A complete effort should seek to analyze these as well.
3. Available forensics software: As demonstrated, both Bulk_extractor and Autopsy are extremely competent, open-source tools. However, research that relies on a small subset of available tools runs the risk of becoming irrelevant if that subset can be demonstrated to produce questionable results. Other digital forensics applications should be incorporated into a broader analysis.

This was the original intent of the study. However, time, funding, and limited tool knowledge were limiting factors in the research effort.

4. Deeper analysis of DuckDuckGo: The results of the DuckDuckGo browser were particularly surprising given the reputation that the browser has garnered in the cyber security and privacy communities. Often during research, other associated paths that warrant deeper analysis become apparent. This was the case with DuckDuckGo. A deeper understanding is required of what exactly causes the snapshot images to be taken.

In conclusion, private browsing continues to grow in popularity and general users may be surprised by the results of this study which indicate that despite significant efforts of well-known, and often well-funded developers, private browsing may not be as private as they believe it to be. While they may provide the user with a degree of privacy from other users of the device, a forensic analysis of the device is very likely capable of revealing detailed personal browsing activity and habits.

Author Name, email@address

References

Aggarwal, G., Bursztein, E., Jackson, C., & Boneh, D. (2010). *An Analysis of Private Browsing Modes in Modern Browsers*. 79–94.

Arshad, M. R., Hussain, M., Tahir, H., Qadir, S., Ahmed Memon, F. I., & Javed, Y. (2021). Forensic Analysis of Tor Browser on Windows 10 and Android 10 Operating Systems. *IEEE Access*, *9*, 141273–141294. https://doi.org/10.1109/ACCESS.2021.3119724

Asim, M., Amjad, M. F., Iqbal, W., Afzal, H., Abbas, H., & Zhang, Y. (2019). AndroKit: A toolkit for forensics analysis of web browsers on android platform. *Future Generation Computer Systems*, *94*, 781–794. https://doi.org/10.1016/j.future.2018.08.020

Boerman, S. C., Kruikemeier, S., & Zuiderveen Borgesius, F. J. (2021). Exploring Motivations for Online Privacy Protection Behavior: Insights From Panel Data. *Communication Research*, *48*(7), 953–977. https://doi.org/10.1177/0093650218800915

Br, October 1, on V. in B. D. on, 2021, & Pst, 11:03 Am. (n.d.). *Consumer privacy study finds online privacy is of growing concern to increasingly more people*. TechRepublic. Retrieved December 1, 2021, from https://www.techrepublic.com/article/consumer-privacy-study-finds-online-privacy-is-of-growing-concern-to-increasingly-more-people/

Bursztein, G. A. E., Jackson, C., & Boneh, D. (n.d.). *AnAnalysisofPrivateBrowsingModesinModernBrowsers*. 15.

Chen, B. X. (2021, September 16). The Battle for Digital Privacy Is Reshaping the Internet. *The New York Times*. https://www.nytimes.com/2021/09/16/technology/digital-privacy.html

Chivers, H. (2014). Private browsing: A window of forensic opportunity. *Digital Investigation*, *11*(1), 20–29. https://doi.org/10.1016/j.diin.2013.11.002

Author Name, email@address

DuckDuckGo Privacy. (n.d.). DuckDuckGo. Retrieved December 29, 2021, from https://duckduckgo.com/privacy

Flowers, C., Mansour, A., & Al-Khateeb, H. (2016). Web browser artefacts in private and portable modes: A forensic investigation. *International Journal of Electronic Security and Digital Forensics*, *8*, 99–117. https://doi.org/10.1504/IJESDF.2016.075583

FTK Imager. (n.d.). Exterro. Retrieved December 6, 2021, from https://www.exterro.com/ftk-imager

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

Garfinkel, S. L. (2013). Digital media triage with bulk data analysis and bulk_extractor. *Computers & Security*, *32*, 56–72. https://doi.org/10.1016/j.cose.2012.09.011

Garfinkel, S. L. (2021). *Building bulk_extractor* [C++]. https://github.com/simsong/bulk_extractor (Original work published 2012)

Hikmatyar, F., & Sugiantoro, B. (2019). Digital Forensic Analysis on Android Smartphones for Handling Cybercrime Cases. *IJID (International Journal on Informatics for Development)*, *7*, 19. https://doi.org/10.14421/ijid.2018.07204

Hitchcock, B., Le-Khac, N.-A., & Scanlon, M. (2016). Tiered forensic methodology model for Digital Field Triage by non-digital evidence specialists. *Digital Investigation*, *16*, S75–S85. https://doi.org/10.1016/j.diin.2016.01.010

Hodge, R. (n.d.). *Tor browser FAQ: What is it and how does it protect your privacy?* CNET. Retrieved December 29, 2021, from https://www.cnet.com/tech/services-and-software/tor-browser-faq-what-is-it-and-how-does-it-protect-your-privacy/

Author Name, email@address

Horsman, G., Findlay, B., Edwick, J., Asquith, A., Swannell, K., Fisher, D., Grieves, A., Guthrie, J., Stobbs, D., & McKain, P. (2019). A forensic examination of web browser privacy-modes. *Forensic Science International: Reports*, *1*, 100036. https://doi.org/10.1016/j.fsir.2019.100036

Johansen, A. (2020, October 8). *What is private browsing? How to use it on any browser*. https://us.norton.com/internetsecurity-privacy-what-is-private-browsing.html

Mobile and Desktop Operating Systems Market Share in 2021. (2021, January 22). *HostingTribunal*. https://hostingtribunal.com/blog/operating-systems-market-share/

Mobile Android Version Market Share Worldwide. (n.d.). StatCounter Global Stats. Retrieved January 12, 2022, from https://gs.statcounter.com/android-version-market-share/mobile/worldwide/

Nimje, H., & Honwadkar, D. K. N. (2015). *Digital Forensic Investigation and Analysis of Android Mobile*. https://doi.org/10.21275/v4i12.nov152071

Noorulla, E. S. (2014). *Web Browser Private Mode Forensics Analysis*. 58.

Ohana, D. J., & Shashidhar, N. (2013). Do private and portable web browsers leave incriminating evidence?: A forensic analysis of residual artifacts from private and portable web browsing sessions. *EURASIP Journal on Information Security*, *2013*(1), 6. https://doi.org/10.1186/1687-417X-2013-6

OpenText Security (n.d.). *OpenText EnCase Forensic Software*. OpenText. Retrieved January 12, 2022, from http://security.opentext.com/encase-forensic

Patil, S. (2017). *Data Extraction Techniques for Android Based Devices*. *5*(2), 4.

Author Name, email@address

Pretorius, S., Ikuesan, A. R., & Venter, H. S. (2017). Attributing users based on web browser history. *2017 IEEE Conference on Application, Information and Network Security (AINS)*, 69–74. https://doi.org/10.1109/AINS.2017.8270427

Private_Browsing.pdf. (n.d.). Retrieved December 7, 2021, from https://duckduckgo.com/download/Private_Browsing.pdf

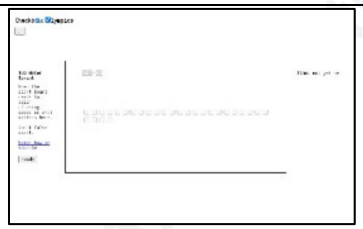
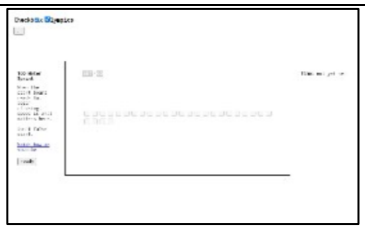
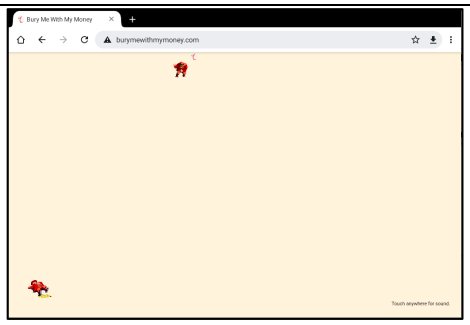
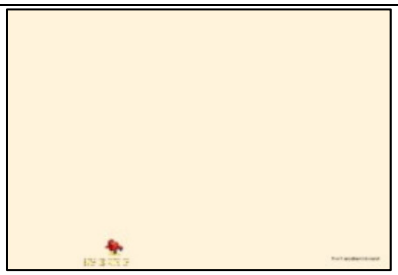
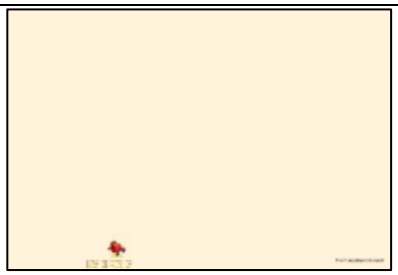
Rogers, M., Goldman, J., Mislan, R., Wedge, T., & Debrota, S. (2006). Computer Forensics Field Triage Process Model. *Journal of Digital Forensics, Security and Law*, *1*(2). https://doi.org/10.15394/jdfsl.2006.1004

The Web Browser: A Critical Source for Digital Forensic and Cybersecurity Investigations. (2018, June 22). Exabeam. https://www.exabeam.com/information-security/the-web-browser-a-critical-source-for-digital-forensic-and-cybersecurity-investigations/

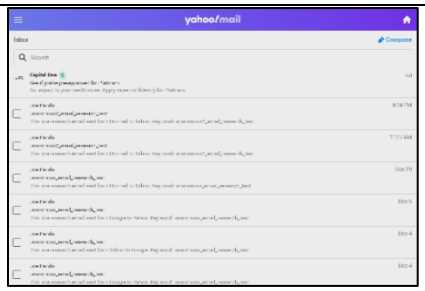
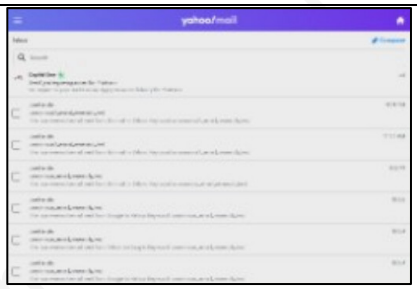
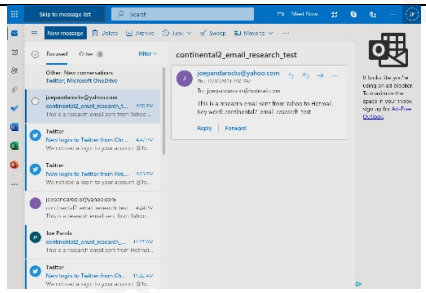
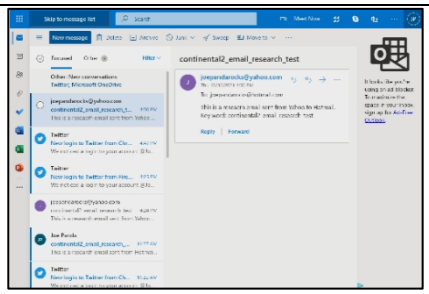
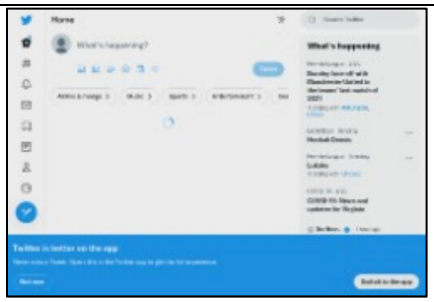
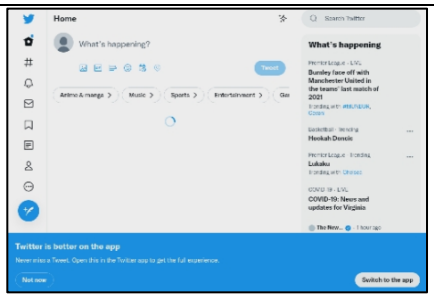
Younis, L. B., Sweda, S., & Alzu'bi, A. (2021). Forensics Analysis of Private Web Browsing Using Android Memory Acquisition. *2021 12th International Conference on Information and Communication Systems (ICICS)*, 273–278. https://doi.org/10.1109/ICICS52457.2021.9464591

Author Name, email@address

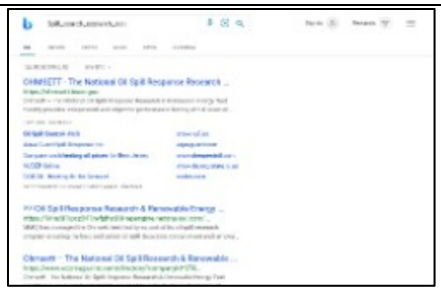
Appendix A
DuckDuckGo Carved Images

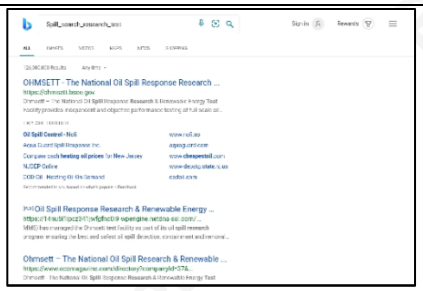
<table>
<tr><td colspan="2">Manual screenshot taken during test script for URL: https://checkboxolympics.com</td></tr>
<tr><td colspan="2">  </td></tr>
<tr><td>Bulk Extractor carved image</td><td>Autopsy carved image</td></tr>
<tr><td>  </td><td>  </td></tr>
<tr><td colspan="2">Manual screenshot taken during test script for URL: http://burymewithmymoney.com</td></tr>
<tr><td colspan="2">  </td></tr>
<tr><td>Bulk Extractor carved image</td><td>Autopsy carved image</td></tr>
<tr><td>  </td><td>  </td></tr>
</table>

Author Name, email@address

Carved Images of Yahoo Mail interface	
Bulk Extractor carved image	Autopsy carved image
	
Carved Images of Hotmail interface	
Bulk Extractor carved image	Autopsy carved image
	
Carved Images of Twitter.com interface	
Bulk Extractor carved image	Autopsy carved image
	
Carved Images of Bing search interface for “Spill_search_research_test”	
Bulk Extractor carved image	Autopsy carved image

Author Name, email@address





Author Name, email@address