Forensic Analysis of the Tor Browser Bundle on OS X, Linux, and Windows

Runa A. Sandvik
runa@torproject.org

Tor Tech Report 2013-06-001
June 28, 2013

1 Introduction

With an estimated 100,000 downloads every month[1] the Tor Browser Bundle is the most popular software package offered on the Tor Project website. A lot of work has been put into making the Tor Browser safe for use with Tor[2], including the use of extra patches against this browser to enhance privacy and security. The Tor Browser Bundle also aims to ensure that the user is able to completely and safely remove the bundle without leaving other traces on her computer.

In an effort to further enhance the security of the Tor Browser Bundle, we performed a forensic analysis of the bundle (version 2.3.25-6, 64-bit) on three different operating systems: OS X 10.8, Debian 6.0 Squeeze Linux, and Windows 7. Our objective was to find traces left by the Tor Browser Bundle and then find ways to counter forensic analysis in three different scenarios:

(a) On a machine that the user does not own, such as a machine in a library or Internet café.

(b) On a machine that the user does own, but does not have administrative rights on.

(c) On a machine that the user does have administrative rights on, but where the user is non-technical and does not know where to find traces of the Tor Browser Bundle or how to remove them.

In the following, we discuss the objective, scope, and limitations for this analysis. We then look into the traces found on the different operating systems and suggest possible mitigations for some of them. We conclude with ideas for further analysis work.

[1] https://webstats.torproject.org/webalizer/www.torproject.org/usage_201305.html
[2] https://www.torproject.org/projects/torbrowser/design/

2 Scope

The primary scope of this forensic analysis was to set up, use, and analyze three operating systems for any changes that may have been made specifically by the use of the Tor Browser Bundle. We built three separate virtual machines, one for each operating system, with default installation settings. We did not download the Tor Browser Bundle using a browser, but instead connected an external drive which we then copied the bundle from. We made a decision to only consider traces left by the Tor Browser Bundle after the bundle had been deleted and the system had been completely shut down.

3 Limitations

The objective, scope, and tools used during this analysis introduced a few limitations that we feel is worth considering when reading this report. Additionally, we had to assume a number of things about the end user, her system, and how she is using the Tor Browser Bundle.

3.1 Objective

The objective assumes that the user either does not have administrative rights on the machine, or does not know how to find and remove traces of the Tor Browser Bundle. A technical user with administrative rights on her system will be able to mitigate a number of the traces found.

3.2 Scope

All three operating systems were installed with default settings and values. The Tor Browser Bundle was copied from an attached external drive to the user's Desktop or home directory. Once the user finished browsing, the Tor Browser Bundle directory and archive was moved to the trash can, and the trash can was then emptied. The system was completely shut down once the bundle had been deleted.

We did not consider traces which are not directly related to the Tor Browser Bundle, such as the presence of an external drive. Additionally, we did not consider traces left after using the Tor Browser Bundle while the bundle was still present on the system, or the system had not been completely shut down.

We believe it is likely that a different scenario would reveal additional traces of the Tor Browser Bundle on the user's system.

3.3 Tools

We used a range of different tools to perform the forensic analysis, all of which are free and available online. The following three tools were all used both before and after we ran the Tor Browser Bundle:

- **dd**[3] - create a backup image of the virtual drive.

[3] http://www.debianhelp.co.uk/ddcommand.htm

- **rsync**[4] - copy all the files on the system over to an external drive.
- **md5deep** and **hashdeep**[5] - compute hashes for every file on the drive, and later compare hashes of the clean image against hashes of the tainted image. A new or changed hash indicates a new or changed file.

We also performed a run-time analysis of the Tor Browser Bundle on Windows 7 using Noriben[6] and procmon[7]. This allowed us to create a report of everything the Tor Browser Bundle did while it was running. A similar analysis was not performed on OS X or Linux due to time constraints.

An analyst with access to a different set of tools, such as commercial tools, might find traces which we were unable to find.

4 Process

We followed roughly the same testing process for all three operating systems. We set up a separate virtual machine for each operating system, logged in with the account we created during the installation process, installed available updates and shut it down cleanly. We used a normal user account on Linux, a non-root administrative account on OS X, and an administrative account on Windows.

Once the operating system had been set up, we connected the virtual drive to another virtual machine, used dd to create an image of the drive, used hashdeep to compute hashes for every file on the drive, and then rsync to copy all the files over to an external drive. It is important to note that we used hashdeep and rsync on the original virtual drive, not on the copy we created with dd.

After having secured a copy of the clean virtual machine, we rebooted the system, connected an external drive, and copied the Tor Browser Bundle from the external drive to the Desktop or user's home directory.

We started the Tor Browser Bundle by clicking on the package archive to extract it, and then clicking on the Tor Browser Bundle executable to run it. On Debian Linux, we also used the command line to extract the archive with *tar -zxvf* and start the bundle with *./start-tor-browser*.

We waited for the Tor Browser to confirm we were connected to the network by loading https://check.torproject.org/. We then browsed a couple of different pages and clicked on a few links before shutting it down by closing the Tor Browser and clicking on the *Exit*-button in Vidalia. The Tor Browser did not crash and we did not see any error messages.

We deleted the Tor Browser Bundle folder and package archive by moving all components into the Trash/Recycle Bin, clicking on it and choosing Empty Trash/Empty Recycle Bin. On Linux, we also deleted the Tor Browser folder and package archive using *rm -rf* on the command line.

We repeated the steps with dd, rsync, and hashdeep to create a copy of the tainted virtual machine. On Windows, we also used Noriben and procmon as previously noted.

[4] http://packages.debian.org/squeeze/rsync
[5] http://packages.debian.org/squeeze/md5deep
[6] https://www.novainfosec.com/2013/04/17/noriben-your-personal-portable-malware-sandbox/
[7] http://technet.microsoft.com/en-us/sysinternals/bb896645.aspx

5 Results

The following sections list the traces found which directly relate to the Tor Browser Bundle. Each issue has its own ticket in the bug tracker[8]. The full list of traces can be found in #8166 for Linux, #6846 for OS X, and #6845 for Windows.

The majority of the issues found show traces of the Tor Browser Bundle package on the user's system. *Issue 6* describes the only known instance of browsing history leakage, other than perhaps swap files/partitions.

A number of the issues below are related to default operating system behavior, such as the use of Spotlight on OS X and Windows Search. The easiest way to avoid leaving traces on a computer system is to use *The Amnesic Incognito Live System (TAILS)*[9].

5.1 OS X

5.1.1 Issue 1: Apple System Log (ASL)

The Apple System Log is a background process that allows messages from different parts of the operating system to be recorded in several ways. We were able to find traces of the Tor Browser Bundle in the following files:

- /var/log/asl/2013.05.22.U0.G80.asl
- /var/log/asl/2013.05.22.U501.asl

We were not able to examine the following files, but they may contain traces of the bundle:

- /var/log/asl/StoreData
- /var/log/asl/SweepStore

This issue has been documented as #8982.

5.1.2 Issue 2: Crash Reporter and Diagnostic Messages

The Crash Reporter on OS X will collect information about any application that crashes or hangs. We did not encounter any problems when running the Tor Browser Bundle, but we still found traces of the bundle in the following files:

- /Library/Application Support/CrashReporter/
  Intervals_00000000-0000-1000-8000-000C2976590B.plist
- /var/log/DiagnosticMessages/2013.05.22.asl

We were not able to examine the following file, but it might contain traces of the bundle:

- /var/log/DiagnosticMessages/StoreData

This issue has been documented as #8983.

[8] https://bugs.torproject.org/
[9] https://tails.boum.org/

5.1.3 Issue 3: FSEvents API

The FSEvents API allows applications to register for notifications of changes to a given directory tree. Whenever the filesystem is changed, the kernel passes notifications to a process called fseventsd.

The following file contains the path to the attached external drive, the path to the Tor Browser Bundle on the Desktop, and the path to the Tor Browser Bundle in the Trash:

- /.fseventsd/0000000000172019

We were not able to examine the other files in the *.fseventsd* directory, which may also contain traces of the bundle. This issue has been documented as #8984.

5.1.4 Issue 4: HFS+

HFS+ is the default filesystem on OS X; it supports journaling, quotas, Finder information in metadata, hard and symbolic links, aliases, etc. HFS+ also supports hot file clustering, which tracks read-only files that are frequently requested and then moves them into a "hot zone". The hot file clustering scheme uses an on-disk B-Tree file for tracking.

We were not able to examine the following files, which may contain traces of the bundle:

- /.hotfiles.btree
- /.journal

This issue has been documented as #8985.

5.1.5 Issue 5: Preferences

OS X applications store preference settings in plist files, and the files below are related to system fonts, the file manager, recent items, and the Tor Browser Bundle. These files all contain traces of the Tor Browser Bundle:

- /Users/runa/Library/Preferences/com.apple.ATS.plist
- /Users/runa/Library/Preferences/com.apple.finder.plist
- /Users/runa/Library/Preferences/com.apple.recentitems.plist
- /Users/runa/Library/Preferences/org.mozilla.torbrowser.plist

This issue has been documented as #8986.

5.1.6 Issue 6: Saved Application State

Resume is one of the new features in OS X 10.7 and 10.8. The feature allows applications to save their last known state when they are closed, and then return to this state when they are later reopened.

While the Tor Browser does not use this feature, it does leak information in the files which are written to the */Users/runa/Library/Saved Application State/* directory:

- /Users/runa/Library/Saved Application State/org.mozilla.torbrowser.savedState/data.data
- /Users/runa/Library/Saved Application State/org.mozilla.torbrowser.savedState/window_3.data
- /Users/runa/Library/Saved Application State/org.mozilla.torbrowser.savedState/windows.plist

The *windows.plist* file contains the HTML title tag of the last active tab in the Tor Browser (or currently active tab, if the browser is still open). This has been documented as #8987.

Thanks to community review of our findings, we have a potential fix for this issue which we will include in version 3.0alpha2 of the Tor Browser Bundle.

5.1.7 Issue 7: Spotlight

Spotlight, and the Metadata Server (mds), indexes all items and files on a system and allows the user to perform system-wide searches for all sorts of items; documents, pictures, applications, system preferences, etc.

We were not able to examine the following files, but it is likely that Spotlight and mds picked up the Tor Browser Bundle at some point:

- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/.store.db
- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/0.indexGroups
- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/0.indexHead
- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/0.indexIds
- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/0.indexUpdates
- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/journalAttr.3
- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/journals.live/journal.20916
- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/journals.live/journal.21051
- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/live.0.indexGroups
- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/live.0.indexHead
- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/live.0.indexIds
- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/live.0.indexUpdates

- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/permStore
- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/reverseDirectoryStore
- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/reverseStore.updates
- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/shutdown_time
- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/store.updates
- /.Spotlight-V100/Store-V2/5D1FD6C7-8789-4860-9B72-6325801BFADD/tmp.spotlight.loc
- /var/db/mds/messages/se_SecurityMessages

This issue has been documented as #8988.

5.1.8 Issue 8: Swap

OS X relies on swap files and paging for memory and cache management. We were not able to examine the swap file, but it is likely that the following file contains traces of the bundle:

- /var/vm/swapfile0

This issue has been documented as #8989.

5.1.9 Issue 9: Temporary data

OS X stores per-user temporary files and caches in */var/folders/*. The following files contain the path to the Tor Browser Bundle on the Desktop and in the Trash:

- /var/folders/fb/v5wqpgls029d8tp_pcjy0yth0000gn/C/com.apple.LaunchServices-036501.csstore
- /var/folders/fb/v5wqpgls029d8tp_pcjy0yth0000gn/C/com.apple.QuickLook.thumbnailcache/index.sqlite
- /var/folders/zz/zyxvpxvq6csfxvn_n0000000000000/C/com.apple.LaunchServices-0360.csstore
- /var/folders/fb/v5wqpgls029d8tp_pcjy0yth0000gn/C/com.apple.QuickLook.thumbnailcache/thumbnails.data

These files also contain strings such as *org.torproject.torbrowserbundle*, *org.mozilla.torbrowser*, *torbrowser_en-us.app*, *torbrowser.app*, *net.vidalia-project.vidalia*, and *vidalia.app*.

We were not able to examine the last file, *thumbnails.data*, but it might contain traces of the bundle as well. This issue has been documented as #8990.

5.2 Debian GNU/Linux with GNOME

5.2.1 Issue 10: Bash History

Bash is the default shell/command processor on Linux and keeps a record of commands typed by the user. The file below contains lines showing we extracted and ran the Tor Browser Bundle. This trace is specific to the user shell being */bin/bash*. Other shells and window managers will give different results:

- /home/runa/.bash_history:

This issue has been documented as #8697.

5.2.2 Issue 11: GVFS

GVFS is the virtual filesystem for the GNOME desktop. This result will vary depending on the window manager used. The following file contains the filename of the Tor Browser Bundle tarball, *tor-browser-gnu-linux-x86_64-2.3.25-5-dev-en-US.tar.gz*:

- /home/runa/.local/share/gvfs-metadata/home

This issue has been documented as #8695.

After deleting the Tor Browser Bundle by moving the folder and package archive into the Trash/Recycle Bin, clicking on it and choosing Empty Trash/Empty Recycle Bin, we noticed that the following file contained lines indicating that the Tor Browser Bundle had been deleted:

- /home/runa/.local/share/gvfs-metadata/home-c0ca7993.log

Traces in this file include lines such as */.local/share/Trash/expunged/3864782161/start-tor-browser* and */.local/share/Trash/expunged/3864782161/App/tor*. This issue has been documented as #8707.

5.2.3 Issue 12: Recently Used

The following file contains information about recently used files, including the Tor Browser Bundle. The file contains the filename of the Tor Browser Bundle tarball, *tor-browser-gnu-linux-x86_64-2.3.25-5-dev-en-US.tar.gz*, as well as the time and date the bundle was added, modified, and visited:

- /home/runa/.recently-used.xbel

The file *.recently-used* could also exist. This issue has been documented as #8706.

5.2.4 Issue 13: X Session Manager

In the X Window System, an X session manager is a session management program, a program that can save and restore the current state of a set of running applications. The file listed below contains the following string, "*Window manager warning: Buggy client sent a _NET_ACTIVE_WINDOW message with a timestamp of 0 for 0x3800089 (Tor Browse)*":

- /home/runa/.xsession-errors

The file *.xsession-errors.old* could also exist. This issue has been documented as #8696.

5.3 Windows

5.3.1 Issue 14: Prefetch

Windows keeps track of the way the system starts and which programs the user commonly opens. This information is saved as a number of small files in the *Prefetch* folder. The files below may contain data and elements of executable code:

- C:\Windows\Prefetch\START TOR BROWSER.EXE-F5557FAC.pf
- C:\Windows\Prefetch\TBB-FIREFOX.EXE-350502C5.pf
- C:\Windows\Prefetch\TOR-BROWSER-2.3.25-6_EN-US.EX-1354A499.pf
- C:\Windows\Prefetch\TOR.EXE-D7159D93.pf
- C:\Windows\Prefetch\VIDALIA.EXE-5167E0BC.pf

The following cache files are most likely similar to prefetch files. We were not able to examine these files, but they may contain traces of the Tor Browser Bundle:

- C:\Users\runa\AppData\Local\Microsoft\Windows\Caches\cversions.1.db
- C:\Users\runa\AppData\Local\Microsoft\Windows\Caches{AFBF9F1A-8EE8-4C77-AF34-C647E37CA0D9}.1.ver0x0000000000000006.db
- C:\Windows\AppCompat\Programs\RecentFileCache.bcf

This issue has been documented as #8916.

5.3.2 Issue 15: Thumbnail Cache

Windows stores thumbnails of graphics files, and certain document and movie files, in Thumbnail Cache files. The following files contain the Onion Logo icon associated with the Tor Browser Bundle:

- C:\Users\Runa\AppData\Local\Microsoft\Windows\Explorer\thumbcache_32.db

- C:\Users\Runa\AppData\Local\Microsoft\Windows\Explorer\thumbcache_96.db
- C:\Users\Runa\AppData\Local\Microsoft\Windows\Explorer\thumbcache_256.db

Other Thumbnail Cache files, such as *thumbcache_1024.db*, *thumbcache_sr.db*, *thumbcache_idx.db*, and *IconCache.db*, may also contain the Onion Logo icon. This issue has been documented as #8921.

One possible solution would be to drop the Onion Logo icon and use a standard Windows icon instead, assuming this does not confuse our Windows users too much.

5.3.3 Issue 16: Windows Paging File

Microsoft Windows uses a paging file, called *pagefile.sys,* to store frames of memory that do not currently fit into physical memory. The file *C:\pagefile.sys* contains information about the attached external drive, as well as the filename for the Tor Browser Bundle executable. This issue has been documented as #8918.

5.3.4 Issue 17: Windows Registry

The Windows Registry is a database that stores various configuration settings and options for the operating system. *HKEY_CURRENT_USER*, abbreviated *HKCU*, stores settings that are specific to the currently logged-in user. Each user's settings are stored in files called *NTUSER.DAT* and *UsrClass.dat*.

The path to the Tor Browser Bundle executable is listed in the following two files:

- C:\Users\runa\AppData\Local\Microsoft\Windows\UsrClass.dat
- C:\Users\runa\AppData\Local\Microsoft\Windows\UsrClass.dat.LOG1

We did not find traces of the Tor Browser Bundle in any of the *NTUSER.DAT* files. It is likely that we would have seen different results had we used Windows XP, due to a change in registry handling between Windows XP/Vista and Windows 7. This issue has been documented as #8919.

5.3.5 Issue 18: Windows Search

Windows Search, which is enabled by default, builds a full-text index of files on the computer. One component of Windows Search is the Indexer, which crawls the file system on initial setup, and then listens for file system notifications to index changed files. Windows Search writes a number of files to *C:\ProgramData\Microsoft\Search\Data\Applications\Windows*:

- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\GatherLogs\SystemIndex\SystemIndex.1.Crwl
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\GatherLogs\SystemIndex\SystemIndex.1.gthr

- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\MSS.chk
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\MSS.log
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\MSS00007.log
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\MSS00008.log
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\Indexer\CiFiles\00010004.ci
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\Indexer\CiFiles\00010004.dir
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\Indexer\CiFiles\00010004.wid
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\Indexer\CiFiles\00010004.wsb
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\Indexer\CiFiles\CiAB0002.001
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\Indexer\CiFiles\CiAB0002.002
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\Indexer\CiFiles\CiAD0002.001
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\Indexer\CiFiles\CiAD0002.002
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\Indexer\CiFiles\INDEX.000
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\Indexer\CiFiles\INDEX.001
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\Indexer\CiFiles\INDEX.002
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\PropMap\CiPT0000.000
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\PropMap\CiPT0000.001
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\PropMap\CiPT0000.002

- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\SecStore\CiST0000.000
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\SecStore\CiST0000.001
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\SecStore\CiST0000.002
- C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Windows.edb

We were not able to examine the Windows Search database files, but it is likely that Windows Search picked up the Tor Browser Bundle at some point. This issue has been documented as #8920.

6 Further work

The Tor Browser Bundle aims to ensure that no traces are left on the user's system. However, a number of the traces listed in this report are related to default operating system settings, some of which the bundle might not be able to remove. We therefore propose the creation of a document which lists steps our users can take to mitigate these traces on the different operating systems.

The scope of this analysis covered traces left by the Tor Browser Bundle itself, not traces left by other applications while downloading the bundle. The results in this report would have been slightly different had we included traces of downloading the bundle from a browser. We propose to expand the scope of a future analysis to also include downloading the Tor Browser Bundle with a default browser.

The goal of this analysis was to identify traces left behind by the Tor Browser Bundle after extracting, using, and deleting the bundle. The Tor Browser Bundle uses Firefox Private Browsing mode by default, which should prevent browsing history from being written to disk. We propose to watch the Tor Browser Bundle directory itself for browsing history leaks, before the bundle is deleted, for example via automated tests to watch for regressions by either Mozilla or us.

The forensic analysis was performed with one specific version of the Tor Browser Bundle. Other packages, such as the Pluggable Transports Tor Browser Bundle[10] and the experimental Tor Browser Bundle without Vidalia[11], and newer versions of the bundle may leave a different set of traces on the user's system. We propose to include forensic analysis in our build infrastructure so that we can test a number of Tor Browser Bundle packages on a regular basis.

As noted in the tools section, we performed a run-time analysis of the Tor Browser Bundle on Windows 7. We were not able to perform a similar analysis on OS X and Linux due to time constraints. We propose to perform a run-time analysis of the Tor Browser Bundle on OS X and Linux to rule out any additional traces.

[10] https://www.torproject.org/docs/pluggable-transports.html.en#download
[11] https://blog.torproject.org/blog/announcing-tor-browser-bundle-30alpha1

Acknowledgments

Thanks to Mike Perry, Philipp Winter, and Steve Lord, for providing valuable feedback for this technical report.