Technische Hochschule Ingolstadt

Vergleich und Analyse des privaten Modus verschiedener Browser

Computer-Forensik und Vorfallsbehandlung

Carl Schünemann

Christoph Sell

29.08.2025

Inhaltsverzeichnis

1 Einleitung

Steigende Beliebtheit private Browsing: [11] ▪ Die Verwendung von PB wurde als die beliebteste Form der Online-Privatsphäre weltweit identifiziert. ▪ Aufgrund der gestiegenen Sensibilität und Öffentlichkeit für den Schutz der Privatsphäre und die Regulierung des eigenen digitalen Fußabdrucks im Internet werden PB-Technologien wahrscheinlich häufiger auf den Geräten der Nutzer eingesetzt. ▪ Auch wenn es schwierig ist, endgültige Nutzungsstatistiken für solche Aktionen zu erstellen, bietet der Konsens über den Online-Datenschutz einen Einblick. Im Jahr 2016 wurde die Verwendung eines PB-Fensters als die weltweit beliebteste Form der Online-Datenschutzmaßnahme identifiziert [1]. Allein in den USA nutzen Berichten zufolge rund 33 % der Nutzer ein PB-Fenster, wobei über 70 % zugeben, ihren Internetverlauf zu löschen [2]. - Eine umfassende Studie von Montasari und Peltola (2015) ergab, dass der Erfolg des privaten Modus bei verschiedenen Browsern sehr unterschiedlich ist

Vermeintliche Privatheim beim Browsen: [19] > Verschlüsselung ▪ Datenschutz und Datenverwendung sind Hauptbedenken der Internetnutzer geworden [5]. ▪ Fragen wie welche Daten von Unternehmen genutzt werden, mit wem sie geteilt werden und wie wertvoll sie sind, sind heute wichtige Themen. ▪ Daher versuchen Benutzer, sich so weit wie möglich zu schützen, insbesondere durch Begrenzung der Datenweitergabe. ▪ Lösungen wie Verschlüsselung auf HTTP-Ebene [6] und auf DNS-Ebene [7,8] sind Standard geworden und werden den Großteil des Datenverkehrs in den nächsten Jahren abdecken. ▪ Sie können jedoch nur End-to-End-Konversationen verschlüsseln, d.h. IP- und TCP- oder UDP-Informationen sind immer noch verfügbar. > VPNs ▪ Eine weitere beliebte Methode zum Schutz der Privatsphäre und zur Vermeidung von Datenverwendung ist die Verwendung von Virtual Private Networks (VPNs). ▪ Obwohl VPNs immer beliebter geworden sind und die meisten von ihnen den IP-Verkehr verschlüsseln und tunneln können, kann der Datenverkehr tatsächlich am Endpunkt des VPNs überwacht werden. ▪ Dies bedeutet, dass Akteure zwischen dem VPN-Servernetzwerk und dem Website-Server die Daten sehen und nutzen können. ▪ Der VPN-Anbieter kann sogar noch weiter gehen, da er auch die Identität des Clients kennt. > Tor und Brave: 1. Die Endpunkte der verschlüsselten Verbindungen, die von Tor und Brave hergestellt werden, nicht vollständig verschlüsselt sind. Daher können einige Informationen, wie z.B. die IP-Adresse des Benutzers, an den letzten Servern in der Kette sichtbar sein. 2. Einige Tor-Ausgangsknoten haben in der Vergangenheit die Aktivität ihrer Benutzer ausspioniert, um Daten zu sammeln und möglicherweise zu verkaufen. 3. Obwohl die Verwendung von Brave und Tor dazu beitragen kann, dass Benutzer online nicht nachverfolgt werden, werden sie nicht vor Verfolgung durch andere Methoden wie Standortverfolgung oder Geräte-Fingerprinting geschützt. 4. Schließlich können auch andere Schwachstellen in der Implementierung oder Konfiguration von Tor oder Brave dazu führen, dass Daten durchsickern und somit die Privatsphäre der Benutzer kompromittiert wird.

Immer mehr Kriminelle im Internet [13]: > Das Internet und seine Nutzer wachsen ständig, aber auch die Anzahl organisierter Verbrechen und illegale Aktivitäten nehmen zu.

"Webbrowser immer beliebter bla bla . . . " [12] > Webbrowser sind heutzutage ein wichtiger Werkzeug für Online-Aktivitäten wie Online-Banking, Online-Shopping und soziale Netzwerke.

Immer mehr Internet-Nuter:[12] ▪ Im Jahr 2019 gab es laut [13] fast 4,5 Milliarden Internetnutzer.

Zunehmende Bestrebungen nach Privatheit erschwert forensische Ermittlungen [16] > Zunehmende Verwendung von verschlüsselten Daten in der Dateispeicherung und Netzwerkkommunikation erschwert Ermittlungen. > Besonders schwierig ist das Tor-Protokoll, das sich auf den Schutz der Privatsphäre des Nutzers konzentriert. > Tor-Browser hinterlässt digitale Artefakte, die von Ermittlern genutzt werden können.

Motivation Portable Browser [8] ▪ Die Beliebtheit von tragbaren Webbrowsern nimmt aufgrund ihrer bequemen und kompakten Natur sowie des Vorteils, dass Daten einfach über einen USB-Stick gespeichert und übertragen werden können, zu. ▪ Entwickler arbeiten an Webbrowsern, die tragbar sind und zusätzliche Sicherheitsfunktionen wie den privaten Modus-Browsing, eingebaute Werbeblocker usw. bieten. ▪ Die erhöhte Wahrscheinlichkeit, tragbare Webbrowser für schädliche Aktivitäten zu nutzen, ist das Ergebnis von Cyberkriminellen, die der Ansicht sind, dass bei der Verwendung von tragbaren Webbrowsern im privaten Modus keine digitalen Fußabdrücke hinterlassen werden. ▪ Das Forschungspapier zielt darauf ab, eine vergleichende Studie von vier tragbaren Webbrowsern, nämlich Brave, TOR, Vivaldi und Maxthon, zusammen mit verschiedenen Speichererfassungstools durchzuführen, um die Menge und Qualität der aus dem Speicherauszug wiederhergestellten Daten in zwei verschiedenen Bedingungen zu verstehen, nämlich wenn die Browser-Tabs geöffnet und geschlossen waren, um forensische Ermittler zu unterstützen.

Private Browsing Motivation und Ausnutzen von Kriminellen: [15] ▪ Webbrowser werden täglich genutzt, um verschiedene Online-Aktivitäten durchzuführen. ▪ Webbrowser speichern eine große Menge an Daten über Benutzeraktivitäten, einschließlich besuchter URLs, Suchbegriffen und Cookies. ▪ Private Browsing-Modi wurden entwickelt, um Benutzern das Surfen im Internet zu ermöglichen, ohne Spuren zu hinterlassen. ▪ Dies kann von Kriminellen ausgenutzt werden, um ihre Aktivitäten zu verschleiern. ▪ Experimente werden auf jeder Browser-Modus durchgeführt, um zu untersuchen, ob sie Spuren auf der Festplatte oder im Arbeitsspeicher hinterlassen.

Motivation Private Browsing mit Portablen Browsern: [18] ▪ Das Internet ist ein unverzichtbares Werkzeug für alltägliche Aufgaben. ▪ Neben der üblichen Nutzung wünschen sich Benutzer die Möglichkeit, das Internet auf private Weise zu durchsuchen. ▪ Dies kann zu einem Problem führen, wenn private Internetsitzungen vor Computerermittlern verborgen bleiben müssen, die Beweise benötigen. ▪ Der Schwerpunkt dieser Forschung liegt darauf, verbleibende Artefakte aus privaten und portablen Browsing-Sitzungen zu entdecken. ▪ Diese Artefakte müssen mehr als nur Dateifragmente enthalten und ausreichend sein, um eine positive Verbindung zwischen Benutzer und Sitzung herzustellen. ▪ In den letzten 20 Jahren ist das Internet für alltägliche Aufgaben, die mit stationären und mobilen Computergeräten verbunden sind, drastisch unverzichtbar geworden. ▪ Benutzer wünschen sich neben der üblichen Internetnutzung auch Privatsphäre und die Möglichkeit, das Internet auf private Weise zu durchsuchen. ▪ Aus diesem Grund wurden neue Funktionen für das private Browsen entwickelt, die von allen gängigen Webbrowsern unterstützt werden. ▪ Unsere Forschung konzentriert sich auf die Entdeckung von Informationen von lokalen Maschinen, da die meisten Computeruntersuchungen auf der Suche und Beschlagnahme von lokalen Speichergeräten beruhen. ▪ Artefakte aus privaten und portablen Browsing-Sitzungen wie Benutzernamen, elektronische Kommunikation, Browsing-Verlauf, Bilder und Videos können für einen Computerermittler signifikante Beweise enthalten. ▪ Wir

werden auch flüchtige Daten analysieren, die in einer gängigen Incident-Response-Umgebung verfügbar wären.

Schwachstellen in Browsern, durch die Daten "lecken" [24] ▪ Private browsing ist seit 2005 eine beliebte Datenschutzfunktion in allen gängigen Browsern. ▪ Laut einer Studie (-> TODO: welche?) leiden alle Browser unter einer Vielzahl von Schwachstellen, von denen viele zuvor nicht bekannt waren. ▪ Die Probleme werden hauptsächlich durch eine laxere Kontrolle von Berechtigungen, inkonsistente Implementierungen der zugrunde liegenden SQLite-Datenbank, die Vernachlässigung von Cross-Mode-Interferenzen und eine fehlende Beachtung von Timing-Angriffen verursacht. ▪ Alle Angriffe wurden experimentell verifiziert und Gegenmaßnahmen vorgeschlagen.

Private Browsing Motivation und Ausnutzen von Kriminellen [21] ▪ Fast alle Aspekte des Lebens nutzen bereits das Internet, um auf das Internet zugreifen zu können, wird ein Webbrowser verwendet. ▪ Die Einführung des Internets hat das Leben der Menschen in vielen Bereichen verändert, darunter auch im Bereich der Kriminalität, insbesondere in der Verwendung von Webbrowser-Software für Transaktionen und Prozesse im Internet. ▪ Webbrowser speichern normalerweise Informationen wie URL-Verlauf, Suchbegriffe, Passwörter und andere Nutzeraktivitäten. ▪ Aus Sicherheitsgründen wurden einige Funktionen von Webbrowsern entwickelt, um den privaten Modus zu ermöglichen. ▪ Leider wird diese Funktion von einigen skrupellosen Menschen für kriminelle Aktivitäten durch die Anti-Forensik genutzt, um digitale Beweise in kriminellen Fällen zu minimieren oder zu verhindern.

Auswirkung von Darknet und Tor auf Forensiker [20] ▪ Personen, die Inhalte aus dem Darknet abrufen möchten, müssen nicht nur in einem regulären Browser Schlüsselwörter eingeben, sondern müssen es anonym über den TOR-Browser zugreifen, um ihre Identität wie IP-Adresse oder physische Lage zu verbergen. ▪ Aufgrund dieser Tatsachen ist es für Strafverfolgungsbehörden oder digitale forensische Experten schwierig, den Ursprung des Datenverkehrs, den Standort oder die Eigentümerschaft eines Computers oder einer Person im Darknet zu lokalisieren. ▪ Die Auswirkungen des Darknets traten auf, als das Federal Bureau of Investigation (FBI) im Oktober 2013 die Website Silk Road abschaltete, die ein Online-Schwarzmarkt und der erste moderne Darknet-Markt für den Verkauf illegaler Drogen war. ▪ Silk Road war nur über das TOR-Netzwerk zugänglich und vom Mainstream-Web verborgen. ▪ Da die meisten Darknet-Sites Transaktionen über anonyme digitale Währungen wie Bitcoin durchführen, die auf kryptografischen Prinzipien basieren, ist es für digitale forensische Experten sehr schwierig, solche Transaktionen zu verfolgen, da Benutzer und Dienste anonym sind. ▪ Das Ziel dieser Arbeit besteht darin, digitale forensische Techniken zu diskutieren, um solche Darknet-Verbrechen zu behandeln.

2 Theoretischer Hintergrund

Einleitend werden Struktur, Motivation und die abgeleiteten Forschungsfragen diskutiert.

2.1 Private Browsing

Definition Web Browser: > [21] ▪ Der Webbrowser ist eine Softwareanwendung zum Abrufen, Präsentieren und Durchsuchen von Informationsressourcen im Internet oder World Wide Web (WWW). ▪ Eine Informationsquelle wird durch einen Uniform Resource Identifier (URI) identifiziert und kann Webseiten, Bilder, Videos oder andere Inhalte enthalten.

> [12] ▪ Ein Webbrowser ist eine Software, die es Benutzern ermöglicht, das Internet über den von ihrem Dienstanbieter bereitgestellten Zugang zu nutzen. ▪ Die bekanntesten Webbrowser sind Google Chrome, Mozilla Firefox, Microsoft Edge und Brave. ▪ Webbrowser werden für alltägliche Aktivitäten wie das Anschauen von Videos, das Durchsuchen von Webseiten, das Posten von Bildern oder Videos in sozialen Medien und das Herunterladen und Hochladen von Dateien genutzt. ▪ Browser-Modi: Es gibt zwei verschiedene Browser-Modi: den normalen Browser-Modus und den privaten Browser-Modus.

Definition „Normal Browsing“: > [12] ▪ Der normale Browser-Modus speichert alle Browser-Aktivitäten wie Caches, Cookies, Suchbegriffe, Login-Daten und URL-Verlauf auf dem Computer. ▪ Die Cookies speichern Details des Benutzers wie z.B. Browsing-Muster, die anzeigen können, welche Websites der Benutzer häufig besucht oder welche Videos er/sie regelmäßig ansieht.

Definition “Private Browsing”: > [22] - Deshalb wurde eine neue Funktion in die Webbrowser aufgenommen, die den Internetnutzern eine größere Kontrolle über ihre Privatsphäre ermöglicht. Diese Funktion ist als "Private Browsing"bekannt und soll es den Nutzern ermöglichen, im Internet zu surfen, ohne Datenspuren auf ihrem Computer zu hinterlassen.

> [11] - Private Browsing"(PB) ist ein allgemeiner Begriff, der sich auf Mechanismen, die verhindern sollen, dass ein Nutzer Beweise für sein Web-Browsing-Verhaltens auf seinem lokalen Gerät gespeichert werden. - Von Anfang an muss betont werden, dass sich privates Surfen in diesem Zusammenhang nur auf Plattformen bezieht, die lokale Privatsphäre bieten, und dass diese von Anwendungen wie Tor (siehe https://www.torproject.org/) zu unterscheiden sind, die sich ebenfalls auf die Online-Privatsphäre konzentrieren, sowie von Einrichtungen, die die Verfolgung und Überwachung aus der Ferne verhindern, wie z. B. der Tracking Preference Expression des W3C (auch bekannt als "Do Not Track"). - Je nach Browser des Nutzers wird eine zugehörige PB-Funktion mit unterschiedlichen Begriffen bezeichnet: Ïnkognito-Modusïn Chrome, ÏnPrivateïn Edge und dem inzwischen nicht mehr unterstützten Internet Explorer sowie ein "privates Fensterïn Firefox.

Geschichte Private Browsing: > [22] - Die ADbC-Funktion "Privater Browsing-Modus"wurde erstmals 2005 mit Apple Safari 2.0 eingeführt. Drei Jahre später folgte Google Chrome 1.0 (Inkognito). Später,

im Jahr 2009, führten Microsoft Internet Explorer 8 und Mozilla Firefox 3.5 ihre Versionen von privaten Browsing-Modi ein, die als InPrivate bzw. Private Browsing bekannt sind (Dan, 2010).

> [15] ▪ Private Browsing-Modi haben je nach Browser unterschiedliche Namen, z.B. Ïncognito-Modusïn Chrome, ÏnPrivate Browsingïn Internet Explorer, "Private Browsingïn Firefox und Safari. ▪ erstmals 2005 von Apple Safari eingeführt, gefolgt von Google Chrome und Microsoft in 2008 und Mozilla in 2009.

Grund des privaten Modus: > [12] ▪ Private Browsing Mode wurde entwickelt, um die Privatsphäre und Anonymität beim Surfen im Internet zu verbessern, indem keine Spuren und Informationen von Browsing-Aktivitäten hinterlassen werden. ▪ Alle neuen Caches, die während des Surfens gespeichert wurden, werden entfernt, sobald der Browser geschlossen wird. ▪ Jeder Webbrowser bietet einen privaten Browser-Modus mit unterschiedlichen Bezeichnungen an, wie ÏnPrivate Browsing"für Internet Explorer und Microsoft Edge, Ïncognito-Modus"für Google Chrome und "Private Browsing"für Mozilla Firefox. > [1] zwei wesentliche Ziele des privaten Browsing: 1. (local) Besuchte Websites sollten im privaten Modus keine Spuren auf dem Computer des Benutzers hinterlassen. Wenn ein Familienmitglied den Browserverlauf überprüft, sollte keine Evidenz von im privaten Modus besuchten Websites gefunden werden können. 2. (website) Benutzer möchten möglicherweise ihre Identität vor den Websites, die sie besuchen, verbergen, indem sie es beispielsweise für Websites schwierig machen, die Aktivitäten des Benutzers im privaten Modus mit seinen Aktivitäten im öffentlichen Modus zu verknüpfen. Dies wird als Datenschutz vor einem Webangreifer bezeichnet. > [15] ▪ Private Modus Browser sollten in der Lage sein, die von besuchten Websites hinterlassenen Artefakte auf dem Computer des Benutzers zu verhindern. ▪ Browser sollten es Websites unmöglich machen, herauszufinden, ob ein bestimmter Benutzer sie zuvor besucht hat, indem sie verhindern, dass Websites die Aktivitäten von Benutzern im privaten und öffentlichen Modus verknüpfen.

Stakeholder Private Browsing: > Forensiker - [13] ▪ Die Entwicklung von Datenschutzfunktionen in Browsern stellt eine Herausforderung für digitale Forensiker dar, die Beweismittel sammeln möchten, um Kriminelle zu überführen. - [11] ▪ Durch die Möglichkeit des privaten Browsens besteht eine erhöhte Gefahr für illegale und schädliche Online-Aktivitäten. ▪ Die meisten privaten Browsing-Modi sind so konzipiert, dass sie lokal privat sind und Daten, die auf das Surfverhalten des Benutzers hinweisen, nicht auf dem Gerät gespeichert werden. ▪ Diese Handlungen können die Verfügbarkeit von Beweismaterial beeinträchtigen und stellen eine Herausforderung für Untersuchungen dar. - [11] ▪ Private browsing (PB) ist eine Funktion, die seit langem auf dem Radar von forensischen Praktikern steht. ▪ Risiko: PB kann dazu führen, dass potenziell beweiskräftiger Inhalt nicht auf einem lokalen Gerät gespeichert wird, was zu Untersuchungsbedenken führt. ▪ PB selbst hat viele legitime Anwendungen und ist nicht per se anti-forensisch, kann aber mit anti-forensischer Absicht verwendet werden. ▪ Fehlende Internetinhalte stellen ein Problem für Beweissammlung ▪ Private Browsing-Modi sollten die Aktivität des Nutzers vor forensischen Tools verbergen

> Kriminelle: - [13] ▪ Kriminelle nutzen vermehrt private Browser, um ihre Spuren zu verwischen und ihre illegalen Handlungen zu verbergen. ▪ Cyberkriminelle nutzen Private Browsing-Modi, um digitale Spuren auf dem Gerät zu verwischen und forensische Ermittler mit leeren Händen dastehen zu lassen. > Nutzerperspektive: - [11] ▪ Die Verwendung von PB wurde als die beliebteste Form der Online-Privatsphäre weltweit identifiziert. ▪ Aufgrund der gestiegenen Sensibilität und Öffentlichkeit für den Schutz der Privatsphäre und die Regulierung des eigenen digitalen Fußabdrucks im Internet werden PB-Technologien wahrscheinlich häufiger auf den Geräten der Nutzer eingesetzt. ▪ Auch wenn es schwierig ist, endgültige Nutzungsstatistiken für solche Aktionen zu erstellen, bietet der Konsens

über den Online-Datenschutz einen Einblick. Im Jahr 2016 wurde die Verwendung eines PB-Fensters als die weltweit beliebteste Form der Online-Datenschutzmaßnahme identifiziert [1]. Allein in den USA nutzen Berichten zufolge rund 33 % der Nutzer ein PB-Fenster, wobei über 70 % zugeben, ihren Internetverlauf zu löschen [2].

- [11] ▪ Die PB-Technologie wird aufgrund der gesteigerten Sensibilität und öffentlichen Aufmerksamkeit für den Schutz der Privatsphäre voraussichtlich häufiger auf Geräten verwendet. - [22] In den letzten Jahren (2010) haben jedoch viele der bekannten Webbrowser-Hersteller ihre Besorgnis über die Privatsphäre der Nutzer beim Surfen im Internet verstärkt. - Tatsächliche Gründe: [15] in [1] Experiment von Aggarwal et al.: Werbung auf Ad-Netzwerken geschaltet wurde, um verschiedene Kategorien von Websites einschließlich Erwachsenen- und Geschenk-Websites zu bewerben, um die Nutzung des privaten Modus mit der Art der besuchten Website zu korrelieren. –> Browsing-Modus auf Erwachsenen-Websites beliebter war als auf Geschenk-Websites. > Herstellerperspektive: - [15] Angeblich lt. Hersteller: o Einkaufen von Überraschungsgeschenken auf einem Familien-PC o Planung von Überraschungspartys

Stakeholder Private Browsing: > "Forensicher Ermittler [15] ▪ forensischer Ermittler kann forensische Browsing-Artefakte mit forensischen Tools und Techniken wiederherstellen > "Nutzer": - [22] > Tatsächlich ergab eine Studie, dass Private Browsing auf Websites für Erwachsene beliebter ist als auf Websites für den Geschenkekauf oder für Nachrichten. Dies deutet darauf hin, dass die Anbieter von Webbrowsern den Hauptnutzen dieses Tools möglicherweise falsch einschätzen, wenn sie es als ein Tool zum Kauf von Überraschungsgeschenken beschreiben (Aggarwal, Boneh, Bursztein, und Jackson, 2010). > "Browser Entwickler [13] Die Entwickler von Browsern haben den Mangel an Benutzerdatenschutz erkannt und einen privaten Browsermodus eingeführt, der das Schreiben von Browserdaten auf die Festplatte einschränkt oder idealerweise verhindert. - [22] Einem Artikel zufolge (Belani, Jones, 2005) behaupten die Hersteller all dieser Webbrowser, dass keine der besuchten Websites, Formularfelddaten, in die Adressleiste eingegebenen Adressen, besuchten Links und Suchanfragen auf dem lokalen Computer des Nutzers gespeichert werden (Brookman, 2010).

2.2 Angreifermodell

Definition Local Attacker nach [1]: - Z.B. Forensischer Prüfer - hat physischen Zugriff auf den Computer des Benutzers - versucht, auf dessen privaten Browserverlauf zuzugreifen. - beispielsweise ein Familienmitglied oder ein Freund sein, der den Computer des Benutzers nutzt, um auf dessen Browserverlauf zuzugreifen. - kann darauf installierte Programme verwenden, um Informationen zu sammeln. - hat keinen Zugriff auf die Maschine des Benutzers, bevor der Benutzer das private Surfen beendet hat. Ohne diese Einschränkung ist Sicherheit gegen einen lokalen Angreifer unmöglich. (z.B: Keylogger installieren, Benutzeraktionen aufzeichnen) - Durch die Beschränkung des lokalen Angreifers auf "forensische Untersuchungen nach dem Ereignis"kann man hoffen, Sicherheit zu gewährleisten, indem der Browser persistenten Zustandsänderungen während einer privaten Surfsitzung ausreichend löscht. - Der Angreifer wartet, bis der Benutzer den privaten Browsing-Modus verlässt, und erhält dann die vollständige Kontrolle über die Maschine. Dies bedeutet, dass der Angreifer auf forensische Daten angewiesen ist. - Während der aktiven Phase kann der Angreifer nicht mit Netzwerkelementen kommunizieren, die Informationen über die Aktivitäten des Benutzers im privaten Modus enthalten. Dies bedeutet, dass die Implementierung von Browser-seitigen Datenschutzmodi untersucht wird, nicht die serverseitigen Datenschutzmodi.

- Das Ziel des Angreifers besteht darin, für eine bestimmte Menge von HTTP-Anfragen, die er wählt, festzustellen, ob der Browser eine dieser Anfragen im privaten Browsing-Modus ausgeführt hat oder nicht. Wenn der lokale Angreifer dieses Ziel nicht erreichen kann, gilt die Implementierung des privaten Browsings als sicher. -> Local Attacker weiß, wonach er sucht!

- Es wird darauf hingewiesen, dass die Definition impliziert, dass der Angreifer nicht feststellen kann, welche Websites der Benutzer besucht hat oder was der Benutzer auf einer bestimmten Website getan hat. Darüber hinaus wird auf die Eigenschaften des privaten Browsings nicht formal eingegangen, wenn der Benutzer den privaten Browsing-Modus nie verlässt.

Problem: Local Attacker muss überarbeitet werden: [15] ▪ Es wurde festgestellt, dass das Konzept des lokalen Angreifers nicht ausreichend untersucht wurde und dass neue Experimente durchgeführt werden müssen, um ein besseres Verständnis für das Phänomen zu erlangen und herauszufinden, wie sich diese Funktion auf digitale forensische Untersuchungen auswirken könnte.

Definition Web Attacker nach [1] - Z.B. ISP - versucht Online-Aktivitäten des Benutzers im privaten Modus zu verfolgen und zu identifizieren, um diese mit seinen Aktivitäten im öffentlichen Modus in Verbindung zu bringen. - durch den Einsatz von Tracking-Tools oder das Sammeln von Informationen über die IP-Adresse des Benutzers oder andere Identifikationsmerkmale erfolgen. ▪ Kontrolliert die von Benutzer besuchten Websites und kann Informationen über Benutzeraktivitäten sammeln (-> z.B. ISP), aber nicht über den Computer des Benutzers. ▪ Webseiten können auch verschiedene Browser-Funktionen nutzen, um Browser zu identifizieren und sie über Privatsphäre-Grenzen hinweg zu verfolgen. ▪ Die Electronic Frontier Foundation hat eine Website namens Panopticlick (-> TODO: In Demo zeigen?) erstellt, die zeigt, dass die meisten Browser eindeutig identifiziert werden können, was die Ziele (1) und (2) des privaten Surfens in allen Browsern unterbricht.

Anti-Forensiche Grundsätze bei Browserentwicklung, um sich gegen Web-Attacker zu schützen nach [1] ▪ Browser haben drei Ziele, um die Privatsphäre der Benutzer zu schützen. o Ziel 1: Ein Benutzer, der im privaten Modus surft, soll nicht mit demselben Benutzer verknüpft werden können, der im öffentlichen Modus surft. o Ziel 2: Ein Benutzer in einer privaten Sitzung soll nicht mit demselben Benutzer in einer anderen privaten Sitzung verknüpft werden können. o Ziel 3: Eine Website soll nicht erkennen können, ob der Browser im privaten Modus ist. ▪ Ziele (1) und (2) sind schwierig zu erreichen, da die IP-Adresse des Browsers von Webseiten genutzt werden kann, um Benutzer über private Browsing-Grenzen hinweg zu verfolgen. ▪ Das Torbutton Firefox-Erweiterung (ein Tor-Client) macht große Anstrengungen, um Ziele (1) und (2) zu erreichen, indem es die IP-Adresse des Clients über das Tor-Netzwerk versteckt und Schritte unternimmt, um das Browser-Fingerprinting zu verhindern.

Beispiel: Web Attacker Angriffe: > IP-Angriffe [19] ▪ Obwohl Nutzer Verschlüsselung oder VPNs nutzen, ist ihre Privatsphäre oft ungeschützt, da mehrere Domains gleichzeitig besucht werden oder IP-Adressen von Cloud-Providern geteilt werden. ▪ Eine neue Methode zur Identifizierung von Web-Browsing wird vorgestellt, die nur auf den IP-Adressen basiert, mit denen der Nutzer verbunden war, ohne DNS Reverse-Resolution durchzuführen. ▪ Diese IP-Adresse-Sequenz wird in verschiedene Deep Learning Modelle eingespeist, um die tatsächlich besuchte Website zu identifizieren. ▪ Untersucht wurden auch andere Faktoren wie Abhängigkeit vom DNS-Server, Genauigkeit bei Top-Domains, Datenverstärkung durch Paket-Sampling-Simulation, Auswirkungen auf Paket-Sampling und Skalierbarkeit der Methode. ▪ Mit nur 10% der Pakete konnte die besuchte Website mit einer Genauigkeit und F1-Score von 94% bis 95% identifiziert werden. > ISP als „Web attacker“, um Kundenaktivität zu verfolgen [1] ▪ ISP können unsere Ergebnisse nutzen, um den Datenverkehr ihrer Kunden zu identifizieren. ▪ Dies ermöglicht

ISP, Daten für Marketingzwecke zu monetarisieren, sofern sie anonymisiert und mit Zustimmung der Kunden erfolgt. ▪ ISP müssen jedoch darauf achten, wer Zugriff auf Netzwerkverkehrsdaten hat. ▪ Das Weitergeben dieser Daten an Dritte kann zu potenziellen Datenschutzverletzungen bei Kunden führen. ▪ Hauptaufgabe ist eigentlich einfach, aber es können viele Komplikationen auftreten ▪ Hauptproblem ist das sogenannte "verwickelte Netz" ▪ Beim Verbinden mit einer Website muss der Webbrowser eine Kaskade von Verbindungen zu anderen Websites öffnen ▪ Grund dafür sind Bilder, Anzeigen, Banner, JavaScript-Bibliotheken, Social-Media-Links und vieles mehr

2.3 Private Browsing Artefakte

TODO: Common vs Uncommon Locations hier ansprechen

Residuale Daten > [12] ▪ Überraschenderweise besteht der private Browser in Chrome und Firefox aus wenigen residuellen Daten, die jedoch Beweise für Interessen wie Suchbegriffe, E-Mail-IDs und Passwörter liefern können ▪ Residuale Daten sind Daten, die von einem Gerät entfernt wurden, aber immer noch aufgespürt werden können. ▪ Diese Daten können mithilfe spezieller Tools, meist in Dateiüberresten oder lokalen Ordnern, identifiziert werden. ▪ Beispiele für residuale Daten sind Link-Dateien, Log-Dateien, Registry-Dateien, Prefetch-Dateien und Browser-Verlaufsdaten. ▪ Digitale Forensik kann solide elektronische Beweise aus solchen Überresten und Artefakten sammeln, um sie in Gerichtsverfahren zu verwenden.

Browser Artefakte: > [12] ▪ Jeder Browser hat unterschiedliche Artefakte im RAM des Geräts gespeichert ▪ Im normalen Browsing-Modus werden die Browsing-History-Details des Benutzers vor und nach dem Löschen des Verlaufs im Speicher gespeichert ▪ Benutzeraktivitäten und Daten beim Surfen können in normalen Browser-Modi wie Cookies, Caches, Downloads, Verlauf, anderen sensiblen Daten und temporären Dateien verfolgt und gespeichert werden, was digitalen Forensikern bei der Suche nach Beweisen hilft.

> [13] ▪ Browser speichern eine Vielzahl von Nutzerdaten, die von besuchten URLs bis zu Benutzernamen und Passwörtern reichen ▪ Das Wissen, dass Browser private Surfdaten preisgeben, ist schon etwas, aber der Standort dieser Artefakte ist von größter Bedeutung

> [22] - Webbrowser sind so konzipiert, dass sie eine Vielzahl von Informationen über die Aktivitäten ihrer Benutzer aufzeichnen und speichern können. Dazu gehören Caching-Dateien, besuchte URLs, Suchbegriffe, Cookies und andere. - Diese Dateien werden auf dem lokalen Computer gespeichert und können von jeder Person, die denselben Computer verwendet, leicht aufgerufen und abgerufen werden. Dies macht es auch für forensische Prüfer relativ einfach, die Internet-Aktivitäten eines Verdächtigen in Fällen zu untersuchen, in denen fragwürdige Websites besucht oder kriminelle Handlungen über das Internet durchgeführt wurden.

> [3] ▪ Bestimmte Datentypen aus HTTP-Protokoll-Transaktionen oder skriptgesteuerten Aktionen in HTML-Seiten werden separat im Dateisystem gespeichert und führen zu unterschiedlichen Datenbankeinträgen: Cookies, Web Storage und Indexed Database Storage.

Private Browsing Artefakte: > [1] 1. Änderungen, die von einer Website ohne jegliche Benutzerinteraktion initiiert werden. Beispiele hierfür sind das Setzen eines Cookies, das Hinzufügen eines Eintrags zur Verlaufsdatei und das Hinzufügen von Daten zum Browser-Cache. 2. Änderungen, die von einer Website

initiiert werden, aber eine Benutzerinteraktion erfordern. Beispiele hierfür sind das Generieren eines Client-Zertifikats oder das Hinzufügen eines Passworts zur Passwortdatenbank. 3. Änderungen, die vom Benutzer initiiert werden. Zum Beispiel das Erstellen eines Bookmarks oder das Herunterladen einer Datei. 4. Nicht benutzerspezifische Zustandsänderungen, wie das Installieren eines Browser-Patches oder das Aktualisieren der Phishing-Blockierungsliste. ▪ "geschützte Aktionen- Browser Artefakt, dass beim Verlassen des privaten Surfens gelöscht werden muss

Wie entstehen "Leckagen"von privaten Browsing Artefakten? [11] 1. Ein Fehler im Design und in der Entwicklung des Browsers 2. Das Betriebssystem übernimmt mehr Kontrolle über den Browser als es sollte, was dazu führt, dass Daten von außen abgegriffen werden

Common Locations: > Ort der Browserartefakte (“common locations“) ausführlich beschrieben in: [5]

> [12] ▪ Die Artefakte von Webbrowsern werden in bestimmten Ordnern im Betriebssystem gespeichert. ▪ Die genaue Lage variiert je nach Browser, die Dateiformate bleiben jedoch gleich. ▪ Es ist wichtig zu wissen, wo die Dateien gespeichert sind, um sie während des normalen und privaten Browsing-Modus untersuchen zu können. ▪ Tabelle 6 zeigt die Standorte der Artefakte von Google Chrome wie Verlauf, Caches und Cookies. ▪ Tabelle 7 stellt die häufigsten Standorte von Firefox-Artefakten wie Cookies, Cache, Verlauf und Lesezeichen vor. ▪ Alle Änderungen in Firefox, wie Lesezeichen, installierte Erweiterungen und gespeicherte Passwörter, werden im Profilordner gespeichert. ▪ Wie in der Tabelle gezeigt, werden Cookies in cookies.sqlite gespeichert, während Cache-Dateien im cache2-Ordner zu finden sind. ▪ Alle heruntergeladenen Lesezeichen, Dateien und der Verlauf werden in places.sqlite gespeichert. ▪ Mögliche Informationen, die aus der Browser-Forensik extrahiert werden können, sind Browsing-Verlauf, Cache, Cookies, Lesezeichen und Download-Liste.

> [21] ▪ Digitale Beweise in einem Webbrowser umfassen mindestens Caches, Verlauf, Cookies, Download-Dateilisten und Sitzungen. ▪ Zumindest ein Minimum an digitalen Beweisen aus einem Webbrowser ist sehr wichtig und wird von Ermittlern genutzt, um einen Fall bei Internetnutzung zu analysieren.

Gründe für Browser-Artefakte bei Private Browsing: [11] > Fehler im Design und Entwicklung des Browsers -> führt dazu, dass Daten von innen nach außen durchsickern, d. h. Browser ist schuld > Betriebssystem übernimmt mehr Kontrolle über den Browser als es sollte, was dazu führt, dass Daten von außen abgegriffen werden, d. h. Betriebssystem ist schuld

Definition private Browsing Artefakt: ==========
Strings, die Aktionen des Browsing-Protokolls zugeordnet werden können: Keywords, URLs, HTML-Fragmente, E-Mail-Adressen, Betreffzeilen etc.

Warum Computer-Forensik: [13] ▪ Die Untersuchung von digitalen Beweisen ist von großer Bedeutung, um Straftäter zu identifizieren und zur Rechenschaft zu ziehen.

Definition digitale Forensik [12] ▪ Digitale Forensik konzentriert sich auf die Wiederherstellung von Speichermedien, um digitale Beweise für Cybercrime-Untersuchungen zu sammeln. ▪ Die gewonnenen Beweise müssen jedoch in ihrem Originalzustand erhalten bleiben, um vor Gericht zulässig zu sein. ▪ Der Prozess der Erwerbung, Untersuchung, Analyse und Berichterstattung von digitalen Beweisen muss forensisch einwandfrei durchgeführt werden. ▪ Daher müssen Ermittlungsteams sich an die Phasen der digitalen Forensik halten, die auf weit verbreiteten Standards basieren. ▪ Digitale Forensik-Investigatoren verlassen sich auf die Artefakte, die aus diesen Browser-Records auf dem Gerät zurückbleiben, und

verwenden forensische Techniken, um die Artefakte zu erfassen, um Beweismittel zu finden. ▪ Die Artefakte werden im Computer-Speicher gespeichert, nachdem alle Browser-Verläufe, Caches und Cookies gelöscht wurden, was es für digitale forensische Gutachter einfach macht, die Daten zu extrahieren.

Definition Browser Forensics > [13] ▪ Web-Browser-Forensik sammelt und identifiziert Beweise und Informationen im Zusammenhang mit einem Verbrechen aus wiederhergestellten Browser-Sitzungen - Forensische Analyse des Webbrowsers beinhaltet die Wiederherstellung von Browsing-Artefakten, die Informationen über die Online-Aktivitäten eines Verdächtigen offenbaren. - Browser-Forensik wird für Ermittler immer wichtiger, da Suchverlauf, Download-Aktivität und Seitenaufrufe das Verständnis für das kriminelle Motiv verbessern können.

Ziel digitale Forensik [12] ▪ Digitale Forensik hat das Ziel, verwendbare Beweise für Computerkriminalität zu sammeln. ▪ Cyberkriminalität wie Hacking, betrügerische Transaktionen und Diebstahl geistigen Eigentums erhöhen den Bedarf an digitaler Forensik, um auf Cyberkriminalität mit einem digitalen Gerät zu reagieren. (2022) A Comparative Analysis of Residual Data

Live-Forensik: unterschiedliche Definitionen in Literatur > Live-Forensik als "moderner Trend"der Computer-Forensik [7] Im Gegensatz zur traditionellen (toten) digitalen Forensik wird bei der Live-Forensik versucht, flüchtige Daten aufzubewahren und Gegenmaßnahmen für verschlüsselte Dateien auf einem Live-System zu ergreifen. > [9]: TODO! > [12] ▪ „Live Forensics“ wird auch als „Live System Acquisition“ bezeichnet. ▪ Diese Methode wird angewendet, wenn das System in Betrieb ist, um potenzielle Artefakte im flüchtigen Arbeitsspeicher (RAM) zu finden, die als Beweismittel genutzt werden können. ▪ Viele Spuren von Computer-Sitzungen und Artefakte sind nur im flüchtigen Speicher zu finden und können nicht von externem Speicher aus ausgelesen werden. ▪ Die Daten können jedoch nicht gesammelt werden, da sie verloren gehen, sobald das System gestoppt oder neu gestartet wird. ▪ Die RAM-Daten müssen daher mit besonderen Verfahren behandelt werden, um ihre Integrität und Zuverlässigkeit während der Analyse zu gewährleisten. ▪ „Live Forensics“ ist nützlich, um auch Ereignisse zu untersuchen, die nur während der Nutzung des Systems aufgetreten sind, und um Daten effizient im flüchtigen Arbeitsspeicher zu speichern. ▪ Digitale Forensik kann dazu genutzt werden, die Gültigkeit von Beweismitteln bei Gerichtsverfahren zu untersuchen. ▪ Nach der Identifikation und Sammlung von potenziellen Beweismitteln wird in den meisten Fällen eine exakte Kopie der Daten erstellt, um sie als Backup zu nutzen und Veränderungen zu vermeiden. ▪ Es gibt zwei Arten von forensischen Techniken, um Speicherabbilder zu erstellen: „Dead Forensics“ und „Live Forensics“. ▪ Bei „Live Forensics“ hingegen wird das System im laufenden Betrieb untersucht, was oft schwieriger ist, aber auch wertvolle Informationen liefern kann. > [21] ▪ Forensische Untersuchung eines Systems, während es in Betrieb ist ▪ Daten gehen verloren, wenn das System heruntergefahren oder neu gestartet wird ▪ Verwendung bei flüchtigem Speicher wie RAM ▪ RAM-Erfassung durch RAM-Forensik-Tool ▪ Ziel ist es, den normalen Betrieb des Systems nicht zu beeinträchtigen ▪ Live Forensics liefert wichtige Informationen für die Analyse ▪ Analyse von digitalen Beweisen aus dem RAM mit Memory Analysis Tool. ▪ Eine Lösung für dieses Problem ist die Live-Forensik, um Daten aus dem Arbeitsspeicher zu extrahieren, bevor sie gelöscht werden. ▪ Diese Forschungsmethode wird verwendet, um Webbrowser im Allgemeinen und insbesondere tragbare Webbrowser zu analysieren.

Beispiele Live-Forensik > [24] ▪ Volatiler Speicher (Memory Inspection) kann eine wichtige Informationsquelle für forensische Untersuchungen sein ▪ DNS-Caching ist eine Bedrohung für private Browsing: Diese Schwachstelle entsteht, weil das Betriebssystem DNS-Anfragen des Browsers im Cache speichert, unabhängig davon, ob der Browser im privaten Modus ist oder nicht > [13] ▪ Registry

Snapshots: Um Veränderungen im System-Registry aufgrund der Browserinstallation zu verfolgen, wurde Regshot verwendet, um vor der Installation einen Snapshot der Registry aufzunehmen. - Ein zweiter Snapshot wurde nach der Installation des Browsers aufgenommen und mit dem ersten verglichen. - Regshot generiert einen Bericht über die Ergebnisse, der die neuen Dateien und Ordner zeigt, die dem Registry-Schlüssel hinzugefügt wurden.

Vorteile Live-Forensik > In Literatur bekannt: Die meisten Informationen im RAM > [11] ▪ Die meisten Daten können in den RAM-Speichergeräten des Betriebssystems gefunden werden. > [16] ▪ Da es wahrscheinlich ist, dass RAM-Aufnahmen Inhalte der Browsing-Session (z.B. durch Caching) aufzeigen, wurde dies in das Projekt aufgenommen, insbesondere da Warren (2017) dies aufgrund von Zeitbeschränkungen nicht tun konnte. > [16] ▪ Live Analyse während der Laufzeit einer Anwendung ist besonders vorteilhaft, um zu verstehen, wie das Betriebssystem und die Anwendung interagieren. ▪ Live Analyse kann potenziell mehr Informationen zur Browsing-Session liefern, da die Designbemühungen des Tor-Projekts darauf abzielten, Schreibzugriffe auf die Festplatte zu vermeiden.

Herausforderungen von Live-Forensik = Kontaminieren von Beweismitteln [7] Die größten Herausforderungen während des Datenerfassungsprozesses sind: Datenveränderung und Abhängigkeit vom Betriebssystem des verdächtigen Systems; wenn der Erfassungsprozess die Daten verändert, werden die Gerichte die Daten als forensisch untauglich abweisen

Definition Dead Forensik: > [12] ▪ Bei „Dead Forensics“ wird der Computer oder das Gerät, das untersucht werden soll, zuerst heruntergefahren, bevor das Speicherabbild erstellt wird. > [11] - Physische Speichererfassung ist nicht übliche Praxis und in den meisten Fällen nicht verfügbar > [9]: TODO! > [13] - Oft einzige Option: Analysen von Festplatten-Images von ausgeschalteten Geräten - Gründe für „einzige Option“: o Verzögerungen bei der Bearbeitung o Personalmangel bei den forensischen Untersuchern - also unrealistisch und unpraktisch, beschlagnahmte Geräte eingeschaltet zu lassen. - Ausschalten eines Geräts reduziert Risiko einer Datenänderung (versehentlich oder absichtlich) - isoliert es vom Netzwerk, um etwaige Versuche, es ferngesteuert zu löschen, zu verhindern, unter anderem. > [12] -> wiedersprüchlich? ▪ System wird heruntergefahren, bevor das Speicherabbild erstellt wird. ▪ Volatile Dateien gehen verloren: versteckte Dateien, ausgetauschte Dateien, Web-Aktivitäten, Artefakte und Log-Dateien ▪ Das Ziel ist es, eine genaue Kopie des nichtflüchtigen Speichers zu erstellen, bevor das System heruntergefahren wird, um die Originalität der Beweismittel zu erhalten.

Beispiele Dead Forensik: > Stichwortsuche in Festplatten-Images nach herunterfahren [24] > Timestamps von Dateien [24] > SQLite Datenbanken [24] > Unallocated Space [24] > Registry-Hives auf Festplatte, z.B. NTUSER.DAT [24]

Probleme bei Dead Forensik > [7]: TODO!

Wann Live-, wann Dead Forensik? [12] ▪ Die Wahl der Methode hängt von der Art der Untersuchung und der verfügbaren Zeit ab. ▪ Das Ziel ist es, eine genaue Kopie des Speichers zu erstellen, um die Integrität der Beweise zu bewahren und das Risiko von Veränderungen zu minimieren.

Definition: Darknet Forensik: [20] ▪ Motivation Darknet Forensik: o Terroristen, Kriminelle, extremistische Gruppen und Hassorganisationen nutzen das Darknet, um Cybercrime zu begehen. o Die Verwendung von TOR und Bitcoin auf dem Darknet erschwert die Verfolgung von Straftaten durch digitale Forensik-Experten. o Die vorgeschlagenen forensischen Techniken können digitale Forensik-Experten helfen, mit Cybercrime-Fällen im Zusammenhang mit dem Darknet umzugehen. ▪

Darknet-Forensik sind in zwei Kategorien unterteilt: 1. TOR-Browser-Forensik: ▪ vier Möglichkeiten zur Extraktion von TOR-Browser Artefakten: RAM-Forensik, Registry-Änderungen, Netzwerk-Forensik und Datenbank 2. Bitcoin-Transaktions-Forensik: Extrahieren von forensischen Artefakten aus Bitcoin-Wallet-Anwendung

3 Ziel der Arbeit

Wichtig: White-Box Ansatz gemäß local Attacker in [1] - Das Ziel des Angreifers besteht darin, für eine bestimmte Menge von HTTP-Anfragen, die er wählt, festzustellen, ob der Browser eine dieser Anfragen im privaten Browsing-Modus ausgeführt hat oder nicht. Wenn der lokale Angreifer dieses Ziel nicht erreichen kann, gilt die Implementierung des privaten Browsings als sicher. - Local Attacker weiß, wonach er sucht!

Forensiker müssen Funktionsweise von Private Browsing kennen [11] ▪ Die Kenntnis der Erfolgsrate der PB-Technologie unterstützt die Strafverfolgungsbehörden bei digitalen Untersuchungen von Internetinhalten ▪ Internetbeweise sind oft entscheidend für Untersuchungen ▪ Bestimmung des Umfangs und des Erfolgs von PB-Technologie unterstützt die Strafverfolgungsbehörden bei digitalen Untersuchungen von Internetinhalten ▪ Durch die Bestimmung des Umfangs und des Erfolgs von PB-Technologie können sie unnötige Datenverarbeitung und Zeitverschwendung vermeiden, die Untersuchungseffizienz verbessern und sicherstellen, dass keine wichtigen Inhalte übersehen werden. Daher können diese Punkte dazu beitragen, die Effektivität und Effizienz von Untersuchungen zu verbessern, insbesondere in Fällen, in denen Vor-Ort-Triage stattfindet oder in denen eine SHPO angeordnet wurde. Drei Punkte wichtig: ▪ Wenn der Verdacht besteht, dass PB stattgefunden hat, hilft es zu wissen, wie erfolgreich die PB-Funktion eines bestimmten Browsers ist, um unnötige Datenverarbeitung (und Zeitverschwendung) zu vermeiden, wenn tatsächlich keine Browserdaten auf einem Gerät vorhanden sind. ▪ Die Kenntnis darüber, wo PB möglicherweise Informationen zu Browsing-Sitzungen preisgibt, verbessert die Effizienz von Untersuchungen und verhindert, dass wichtige Inhalte übersehen werden. Dies ist besonders wichtig bei Vor-Ort-Triage, wie sie in einigen Fällen mit einer SHPO angeordnet wird.

Ziel der Arbeit: ========== - Welche Browsing Artefakte werden beim private Browsing auf einem Rechner hinterlassen, welche zeigen, dass eine Browsing Aktion vom Browser durchgeführt wurde? - Das heißt: o Es wird nach Browsing Artefakten gesucht, welche die Zuordnung „Durchgeführte Browsing Aktion“ <-> Browser ermöglichen o Vor, während und nach private Browsing Session nach Browsing Artefakten suchen, welche dem Browser zugeordnet werden können - Negativbeispiel: Suche in Hexdump nach im Browser gesuchtem String nicht als Beweis ausreichend, dass private Browsing Artefakte gefunden wurde. - Kategorisierung nach [18]: > Browsing History > Usernames/Email Accounts > Images

=> Thematisiert in [18]: o It appeared that the overall best way to recover residual data was to obtain the evidence from RAM or working memory, o Kritik: Oft nur String Match in RAM-Hex als Nachweis für PB genannt -> ausreichend? (Evtl. Gegenexperiment mit Editor)

Warum muss String-Artefakt Browser zugeordnet werden können? [12] ▪ Die Artefakte, die von den Browsing-Aktivitäten eines Kriminellen zurückgelassen wurden, können mit forensischen Tools extrahiert werden, um die Untersuchung des Ermittlers zu unterstützen. ▪ Die erlangten Beweise müssen vor Gericht zugelassen werden, insbesondere digitale Beweise, da sie ohne ordnungsgemäße Verfahren leicht manipuliert werden können. ▪ Es gibt bestimmte Merkmale von digitalen Beweisen, die Gerichte

nach folgenden Kriterien akzeptieren: 1. Durchsuchungsbefehle - Beweise, die ohne Genehmigung erlangt wurden, können vor Gericht nicht anerkannt werden. 2. Berichte - Alle Prozesse, Werkzeuge, Methoden, Techniken, spezifischen Zeit- und Datumsangaben sowie die Beweiskette müssen formell dokumentiert werden, um die Authentizität der digitalen Beweise vor Gericht zu demonstrieren und zu unterstützen. 3. Beweisauthentifizierung - Der ursprünglich erhaltene Beweis sollte durch Vergleich der Hash-Werte mit dem Kopiebeweis übereinstimmen. Der erworbene Beweis muss unverändert bleiben, um die Gerichte mit genauen Informationen zu überzeugen. Gerichte akzeptieren Kopien von Beweisen, wenn der ursprüngliche Beweis verloren gegangen oder zerstört wurde, die Kopie jedoch noch intakt ist.

Ziele anderer Arbeiten: ========== > [12] - Die Art der extrahierbaren Daten zu untersuchen - den Unterschied zwischen privatem und normalem Surfen zu vergleichen - zu analysieren, welcher Browser die vollständigeren residualen Daten liefert. > [15] ▪ ob bestimmte Arten von Browser-Daten gefunden werden konnten (Webseiten, Verlauf, Download-Verlauf, besuchte URLs und Suchbegriffe) > [21] ▪ Das Ziel dieser Studie ist es, eine Rahmenbedingung für die Analysephasen des Webbrowsers im privaten Modus und Anti-Forensik vorzuschlagen, um eine effektive und effiziente forensische Untersuchung zu ermöglichen. ▪ Die Studie nutzt Live-Forensik, um detailliertere Informationen über den Computer zu erhalten, während er noch in Betrieb ist, und eignet sich daher besser für die schnelle Datenerfassung in Echtzeit. > [24] ▪ umfassende Analyse der privaten Browsing-Funktion in den vier beliebtesten Webbrowsern (IE, Firefox, Chrome und Safari) vorgestellt. > [12] - digitalen Forensikern helfen, Artefakte von Geräten zu verfolgen, die Live-Memory-Erfassung verwenden > [16] - Methodik entwerfen, um folgende Fragen zu beantworten: 1. Kann Tor den Benutzer schützen, indem es Beweise für dessen Nutzung aus dem RAM löscht, wenn die Browsing-Sitzung geschlossen wird? 2. Kann die Tor-Nutzung zu vier Schlüsselmomenten erkannt werden: während das Browser-Fenster geöffnet ist, nach Schließen des Browser-Fensters, nach dem Löschen des Installationsverzeichnisses/ zugehöriger Dateien und nach dem Ausloggen des Benutzers? 3. Können Dateien aus dem Browsing-Protokoll in der Live-Forensik mit Tor 7.5.2 wiederhergestellt werden, der zum Zeitpunkt der Schreibens aktuellsten Version? - Die Experimente wurden im mobilen Modus mit Tor wiederholt, d.h. von einem USB-Stick ausgeführt. (!!!) zu bestätigen, dass die Existenz und Nutzung des Tor-Browsers in Windows 10 nachweisbar ist. (!!!) nachweisen, dass Artefakte des Tor-Browsing-Protokolls auf dem Zielcomputer wiederhergestellt werden können. > [12] > In dieser Studie werden die residualen Daten zwischen Google Chrome und Mozilla Firefox Webbrowsern im normalen und privaten Browsermodus mithilfe eines forensischen Tools analysiert und verglichen. > [15] ▪ In dem Projekt wurden die Effektivität der "privaten"Modus von vier weit verbreiteten Webbrowsern analysiert. > [24] ▪ Ziel: Bewertung der Sicherheit des privaten Surfens in den Browsern Chrome, Safari, Firefox und IE ▪ Die Autoren haben eine umfassende forensische Analyse durchgeführt, die sowohl Live-Memory-Analyse als auch Post-Mortem-Analyse umfasste. > [15] ▪ Vier getestet: Firefox, IE, Safari und Chrome

Keine Ziele der Arbeit: ========== - Private Browsing Ïndicators": Entering/Leaving Private Browsing [18] - Zeigen, dass ein Browser gestartet/geschlossen wurde - Zeigen, dass ein Browser im privaten Modus gestartet wurde - Zeigen, wann ein Browser gestartet/geschlossen wurde - Browser-Erweiterungen: [24] > Browser-Erweiterungen und ihre Auswirkungen auf das private Surfen wurden in einer Studie von Aggarwal et al. Im Jahr 2010 untersucht. –> Siehe Punkt „Add-Ons als Leck" > Die Chrome-Erweiterung „Incognito Inspector" kann im privaten Modus genutzt werden, um detaillierte Informationen über die Nutzeraktivitäten zu sammeln und in Echtzeit an einen Remote-Server zu senden. > Firefox-Erweiterungen sind standardmäßig im privaten Modus aktiviert und können genutzt werden, um Nutzeraktivitäten aufzuzeichnen. > Internet Explorer-Erweiterungen sind in der

Regel deaktiviert und erfordern die manuelle Aktivierung im privaten Modus. Die von den Autoren entwickelte Erweiterung funktionierte jedoch nicht, da sie aufgrund eingeschränkter Privilegien nicht auf die BHO-Klasse zugreifen konnte - [1] > Unterschiedliche Handhabung durch Browser: Gefährliche Leckage für private Browsing Artefakte > Entwickler von Add-Ons haben möglicherweise den privaten Browsing-Modus bei der Entwicklung ihrer Software nicht berücksichtigt, und ihr Quellcode wird nicht derselben rigorosen Überprüfung unterzogen wie die Browser selbst. > Gegenmaßnahme: [1] ▪ Es wurde eine Firefox-Erweiterung namens ExtensionBlocker entwickelt, um unsichere Erweiterungen im privaten Modus zu blockieren

4 Methodik

> Validation Stage (= Kapitel „Vergleich der Browser“)

Warum Methodik? > [1] Aufgrund der Komplexität moderner Browser ist eine systematische Methode erforderlich, um zu testen, ob der private Browsing-Modus ausreichend gegen die Bedrohungsmodelle aus Abschnitt 2 verteidigt. > [12] ▪ Die Verfahren für die digitale Forensik für Browser-Forensik müssen angemessen befolgt werden, um dem Ermittler bei der Durchführung der Untersuchung zu helfen. Die Verfahren unterscheiden sich je nachdem, wie die Untersuchung durchgeführt werden soll. > [11] ▪ Das Fehlen von Klarheit hat einen signifikanten Einfluss auf forensische Untersuchungen von Strafverfolgungsbehörden und deren Ansätze ▪ Eine Kette von Beweisführung muss dokumentiert werden, um die Integrität und Zuverlässigkeit der Daten sicherzustellen. ▪ Ein formaler forensischer Bericht wird dann vor Gericht präsentiert.

Bekanntes Computer Forensik Vorgehensmodell: [25]: Generic Model Computer Forensics Investigations (GCFIM) -> Daran orientieren sich alle in der Literatur

Phasen nach [15] ▪ Die forensische Analyse erfolgt in zwei Phasen. 1. Zunächst wird die Analyse an sowohl "üblichenäls auch üngewöhnlichenSSpeicherorten auf der Festplatte durchgeführt. 2. In der zweiten Phase wird der physische Arbeitsspeicher (RAM) untersucht.

Phasen nach [12]: ▪ Es gibt verschiedene Modelle für digitale Forensik, die sich in ihren Phasen unterscheiden können. ▪ Fünf Phasen sind besonders wichtig: Identifikation und Sammlung, Bewahrung, Erwerb, Analyse und Prüfung sowie Dokumentation. ▪ In der Identifikations- und Sammelphase werden alle potenziellen Beweismittel identifiziert, gekennzeichnet und gesammelt, um sie in der nächsten Phase zu verwenden. ▪ Beweismittel können z.B. Log-Dateien, temporäre Dateien, Netzwerkverbindungen, Browserverlauf und Cache sein. > Phasen: Preparation Phase o Versuchsplanung + Konfiguration der HW/SW + Durchführen des Experiments Acquisition Stage o Abbildung von der Festplatte (Static Forensics) und des RAMs (Live Forensics) Analysephase o Bilder der Speicherabbilder mit einem forensischen Tool untersuchen Validierungsphase o gefundenen Artefakte verglichen und dokumentiert

4.1 Preparation Stage

4.1.1 Konfiguration der Versuchsumgebung

VM Konfiguration

RAM: - Kaum Angaben in der Literatur: > [21]: 2 GB > [18]: 4 GB - Hier: 6 GB -> Ausblick: Kritik an Literatur, dass RAM-Größe kaum thematisiert wird, obwohl sie Auswirkungen auf Ergebnisse hat -> Siehe Kapitel X (TODO!)

Netzwerkeinstellungen: - TODO

Windows 10 Installation: - TODO

Tools auf VM: - Process Monitor - Regshot

Konfiguration des Analyse-Rechners

Volatility: Plugins-Liste: (Ähnlich zu [**Hariharan**] und [4]) - TODO

Autopsy: Evtl. hier Sleuthkit vs Autopsy thematisieren - TODO

Sonstige Tools: WinHex SQLite Viewer etc. - TODO

-> Evtl. am Schluss Tabelle mit allen Tool, Versionen und Plug-Ins

4.1.2 Browserauswahl

> Browserstudie [12] - Die Herstellerangaben unterschiedlicher Browser bzgl. Privatheit untersucht - Firefox 58.02: No Browsing History stored, No Cookies stored, No login Info stored, Tracking Protection Enabled: Disconnect, Download Files not Hidden - Chrome 63.0.3239: No Browsing History stored, No Cookies stored, No login Info stored, Tracking Protection Enabled: No, Download Files not Hidden

> design aim of Tor: [16] - preventing from writing to disk (Perry et al., 2018) - enabling secure deletion of the browser (Sandvik, 2013) (hier nicht relevant)

4.1.3 Browsing Szenario

- Wichtig für White-Box-Ansatz: Browsing Szenario ist bekannt - URL X ... (TODO!)

4.2 Acquisition Stage

> Browsing Szenario durchfüren > Zeitpunkte von -> Orientieren an: [16] - RAM-Dumps - VM-Snapshots (nur letzter Snapshot ist Post-Mortem Forensik) - Process Monitor Logfiles - Registry Snapshots

- Warum Process Monitor während Browsing? o Während Browsing Szenario Filechanges untersuchen: DaemonFS set to monitor all activity within local hard drive[18]

- Registry: [21] ▪ Das Windows-Registrierungsverzeichnis enthält viele Informationen zur Nutzung des Computers, Benutzerkonfigurationen, Anwendungen und Hardwaregeräte ▪ Informationen im Registrierungsverzeichnis werden nach Ausführungsreihenfolge, Suchschlüsselwörtern, zuletzt aufgerufenen Ordnern, Anwendungsprotokollen und anderen Kategorien sortiert.

4.3 Analysis Stage

> Analysis Stage (= Kapitel „Results“) - Analyse der akquirierten Artefakte der vorherigen Phase: VM-Snapshots, RAM-Dumps, Process Monitor Logfiles und Registry Snapshots mit ggf. zusätzlichen Tools - Oberster Leitsatz dabei: gefundenes Artefakt muss eindeutig Browser zugeordnet werden können: Deshalb einfache Stringsuche in RAM mit WinHex ungenügend -> Hier evtl. negatives Beispiel zu Stringsuche einflechten

4.3.1 Common Locations

Whitebox-Analyse: (gezieltes Suchen nach Dateien) [2] Definition: "White-Box"Computer Forensik bezieht sich auf eine forensische Untersuchungsmethode, bei der der forensische Analyst über umfassende Kenntnisse und Zugriff auf das untersuchte System verfügt. Im Kontext der Computerforensik bezieht sich "White-Box"darauf, dass der Analyst über volle Transparenz und Zugriff auf alle Informationen, Ressourcen und Artefakte des Systems verfügt.

Die "White-Box"Forensik kann verschiedene Techniken und Tools umfassen (z.B. Process Monitor, Regshot, Registry Explorer, Dekomprimierungstools), um Daten wiederherzustellen, gelöschte Informationen wiederherzustellen, Metadaten zu analysieren, Netzwerkaktivitäten zu überwachen und weitere forensische Analysen durchzuführen. Der Fokus liegt darauf, das System vollständig zu verstehen und alle relevanten Beweise zu sammeln.

Hier: In Orten gesucht, die 1. Process Monitor ermittelt hat und 2. in der Literatur vorgeschlagen wurden.

Definition: Common Location (= i.d.R. Installationsverzeichnisse der Browser) = „Bekannte Speicherorte“, z.B. bei Firefox - TODO: Quelle > Welche Dateien in Common Locations mit Process Monitor identifiziert > Wie haben sich Dateien verändert in verschiedenen Snapshots? > Was in Dateien gefunden? - Ziel: Befinden sich unter den Dateien, die ein Browser direkt auf die Festplatte schreibt private Browsing Artefakte? - Dateien sind Browserspezifisch, befinden sich in bekannten Pfaden. Beispiele: Datenbank-Dateien, Caches, temporäre Dateien. - String-Suche wäre nicht ausreichend, da Artefakte teilweise komprimiert (siehe .jsonlz4) - Beispiele: > Cache folder, Web history [15]

Schreiboperationen mit Process Monitor verfolgen

- Process Monitor: WriteFile Operationen von Browser - Vorgehen: (Siehe Aktivitätsdiagramm) o Basis = Process Monitor Logfile 1 und 2 o Processmonitor Filter: Browser-Prozess, Dateioperationen, nur WriteFile o Export als CSV o Datenaufbereitung in Excel o Irrelevante Spalten löschen: Time of Day (zeitl. Kontext nicht wichtig), Process Name (Da in Process Monitor bereits nach Namen gefiltert wurde -> Alle Prozesse haben gleichen Namen), Operation (Da in Process Monitor bereits nach Operation gefiltert wurde -> Alle Prozesse haben gleiche Operation „WriteFile“), Result, Detail o Gleiche Operationen (Duplikate) löschen o Neue Spalte mit Dateiendung -> Weiteres gruppieren, sortieren und analysieren ist browserspezifisch o Wenn Daten aufbereitet wurden: 1. Autopsy: Prüfen, ob Dateien noch in Snapshot Image vorhanden 2. Wenn ja, Dateien mit Autopsy extrahieren 3. Wenn nein, prüfen, ob Datei in RAM gecacht -> Hier beschreiben, wie mit Volatility filelist etc. Dateien

aus RAM wiederhergestellt werden können 4. Prüfen ob Browsing Artefakte in Dateien enhalten sind: Stringsuche nach Aktionen des Browsing-Protokolls

TODO: Allgemein: Nur Dateien untersucht, die gemäß Methodik (Kapitel X) entweder im Snapshot vorhanden sind oder sich über Autopsy Carving PlugIn bzw. RAM wiederherstellen lassen. > Wenn Temp-Dateien nicht mehr vorhanden, wird die nicht-Temp Datei aufgeführt

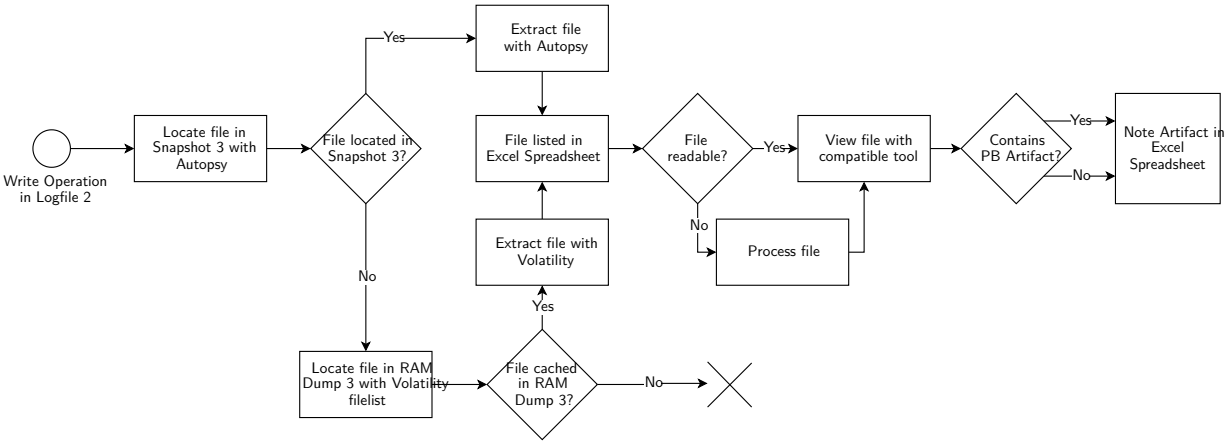


Abbildung 4.1: TODO: Process Monitor Write Operation to Excel Spreadsheet

SQLite-Datenbänke

- Gesondert betrachtet: Zeitlicher Vergleich von SQLite Datenbänken > Begründung: In Literatur ermittelt, dass SQLite DB von zentraler Bedeutung bei Browser History -> Hier wird i.d.R. Suchverlauf gespeichert > Zählt zu den wichtigsten "Common Locations> Vorgehen: Siehe Aktivitätsdiagramm TODO: WAL Checkpoint

4.3.2 Registry

- Registry: > Process Monitor: SetValue Operationen von Browser -> Values der Keys untersucht (je nach Datentyp) -> Sonderfall: REG_BIN - Kategorien der Keys auflisten Diagramm: z.B. Kreisdiagramm mit Anteil der Kategorien an gesamten Schreiboperationen > Stringsuche in Registry Hives mit Registry Explorer (Siehe Liste) - Suchbegriffe auflisten - Hives (Speicherorte) auflisten > ßhellactivities-ähnliche"Keys untersucht - Arbeit von ßhellactivities-ähnliche"Keys erklären

4.3.3 Uncommon Locations

Blackbox-Analyse: [2] (Stringsuchen im gesamten Image mithilfe von Tool) Definition: Auch "triage-style keyword search"[11] genannt, = Durchsuchung des Beweismaterials ohne Vorwissen über Browser-verhalten (d.h. welche Dateien geschrieben wurden) sowie ohne Vorverarbeitung der Dateien (z.B. Entpacken von Dateien). Stattdessen: Untersuchen der Images nur mittels vordefinierter Funktionen

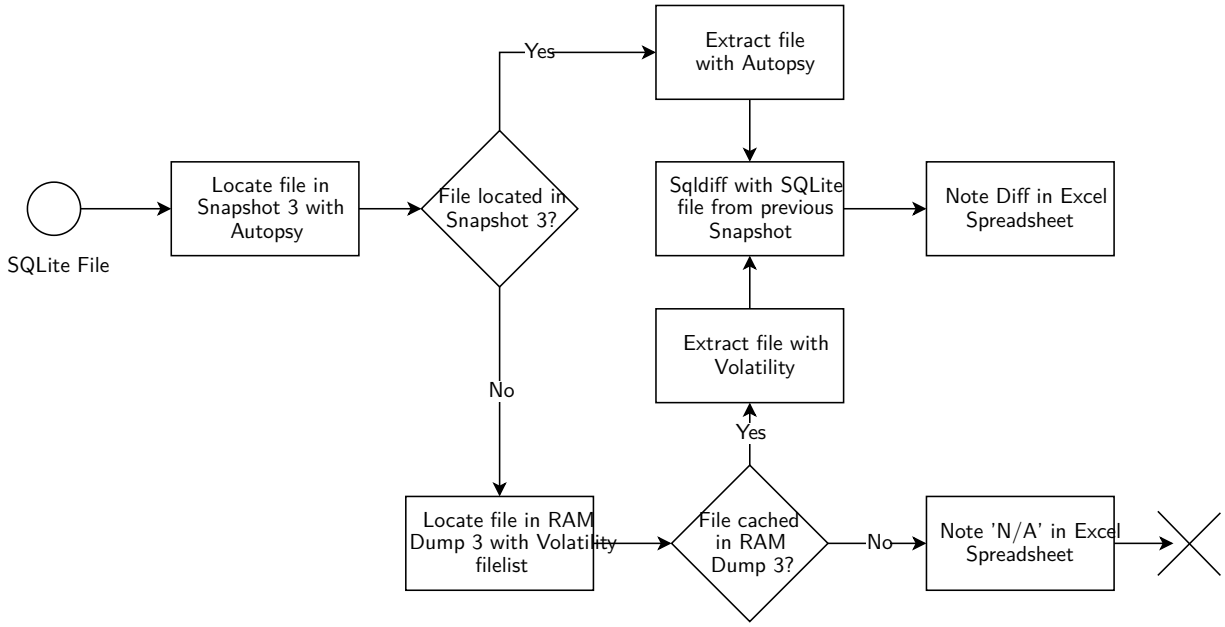


Abbildung 4.2: TODO: Process Monitor Write Operation to Excel Spreadsheet

von Forensik-Tools "Triage", da dies schnelles erstes Mittel von Forensikern, um nach Acquisition Phase Ergebnisse zu erhalten

Hier entscheidend Üncommon Locations": = „Unbekannte Speicherorte“, nur durch tiefgehende forensische Analyse entdeckt

- TODO: Quelle o Registry o Pagefile.sys o Unallocated Disk Space -> Suche nach „obfs4“ deckt Bridging IP-Adressen auf o Windows-Prefetching o Timestamps o $MFT o $Unalloc o $LogFile o Favicons o etilqs o Manifest.json o slack space

- Beispiele in der Literatur: > “$MFT”, “$LogFile”, “Favicons”, “etilqs”, “Manifest.json”, “pagefile.sys.”, “unallocated space” and “slack space” [15]

- Ziel: Untypische Orte, wo private Browsing Artefakte gefunden werden können. Im - Unterschied zu Common Locations: Weitergreifendes Konzept, umfasst Dateien, die nicht von Browsern in bekannten Browser-Ordnern gespeichert werden, sondern auch Speicherorte wie RAM, Registry oder Caches des Rechners, wie - In Literatur ermittelt: für private Browsing drei uncommon Locations relevant: o Stichwortsuchen in kompletten Speicherabbildern: Festplatte (Common Location Browser-Pfade ausgenommen) + RAM -> Wichtig: String-Treffer muss Browser zugeordnet werden können -> Negativbeispiele: o [21]: in WinHex: URLs, Passwörter gefunden -> Wie wird URL Browser zugeordnet? Reicht gefundener String in RAM-Hex als Beweis aus? o [14] WinHex: email account can be retrieved, retrieves all URL histories including the directories visited by a user o [15] Firefox: URLs und Keywords als Strings in WinHex gesucht und gefunden o [15] Chrome: URLs und Keywords als Strings in WinHex gesucht und gefunden

o In Literatur oft verwendet: Stichwortsuchen: > Autopsy Keyword-Suche außerhalb der Common Locations, in allen Partitionen ▪ Definition der gesuchten Strings ▪ Weiterführend: In Literatur nichts

über verwendete Plugins gefunden. Hier: o Automatische Kategorisierung von Dateien o Timeliner-Plugin (Wenn verwendbar?) > RAM: Yarascan Treffer -> String Kontext ▪ Definierte Yararules - TODO! ▪ HTML-Fragmente: [22] We were also able to find blocks of HTML code that constructs Web sites we visited. ▪ Image Carving: > Carved from Memdump [18] > Bildsuche mit: Griffeye's DI Analyze Pro with LACE plug-in [11]

- Windows: Prozess-Struktur im RAM: (–> TODO: Wo gefunden?) The EPROCESS data structure contains information about process instances, such as image name and ProcessID, the resources allocated in terms of memory allocations (how much and where), types (private, mapped, shareable, etc.), memory protections (combinations of read, write, execute, and reserved), modules loaded, and pointers to ETHREADs and the process environment block.

Both EPROCESS and ETHREAD are considered opaque objects by Microsoft [28], inhibiting analysis; fortunately, third-party work has been done to understand these struc tures [29], [30]. Microsoft does provide symbol files1, which help communicate the layout of data structures [31]. Indeed, Volatility uses these symbols for its own processing.

Included in EPROCESS, the ETHREAD object is an opaque structure which contains useful information about the stack. We calculated the size of a stack from the difference between its limit and base, both of which are attached to the ETHREAD.

Another member of the EPROCESS structure, the VAD tree, maps out the virtually allocated memory for a process [32]. VAD nodes refer to loaded modules (in the allocations in which they were referenced) and also have unique permission flags per node.

The PEB (process environment block) contains data about the number of heaps, which modules have been loaded into memory, and the command-line string that invoked the process [33]. The module list may not match the VAD tree's list exactly, the difference of these two sets indicating images of interest

In Literatur der Web Browser Forensik vorwiegend verwendet: - Autopsy Stichwort Suche nach PB Artefakten + Indizieren der Dateien durch Autopsy-PlugIns - Stichwortsuche in RAM mit Volatility Yarascan PlugIn. Vertiefende Untersuchung für jeden Yara-Rule-Treffer *** Hier werden Artefakte gefunden *** -> Flussdiagramm

Analyse mit Autopsy

Bei White-Box Analyse: Autopsy nur zur Dateiextraktion genutzt, hier: als konkretes forensisches Werkzeug Wichtig dabei: - Stichwortsuche -> Screenshot von Suchfunktion -> Suchbegriffe auflisten - Nutzen der Plug-Ins - evtl. "pagefile.sysProblematik ansprechen

Analyse mit Volatility

Bei White-Box Analyse: RAM nur zur Dateiextraktion genutzt, hier: als konkretes forensisches Werkzeug

Wichtig: Auf Ziel der Arbeit verweisen: gefundenes PB Artefakt muss zwingend Browser zugeordnet werden können -> d.h. gefundener String des Browsing Protocols in Hex des RAM-Dumps reicht

nicht als Beweis für gefundenes PB Artefakt aus. Stattdessen: gefundenes PB Artefakt im RAM muss Browser zugeordnet werden können -> Passendes Werkzeug = Volatility PlugIn "Yarascan> TODO: Definition Yarascan -> TODO: Auflistung der Yara-Rules -> Vorgehen: Siehe Baumdiagramm

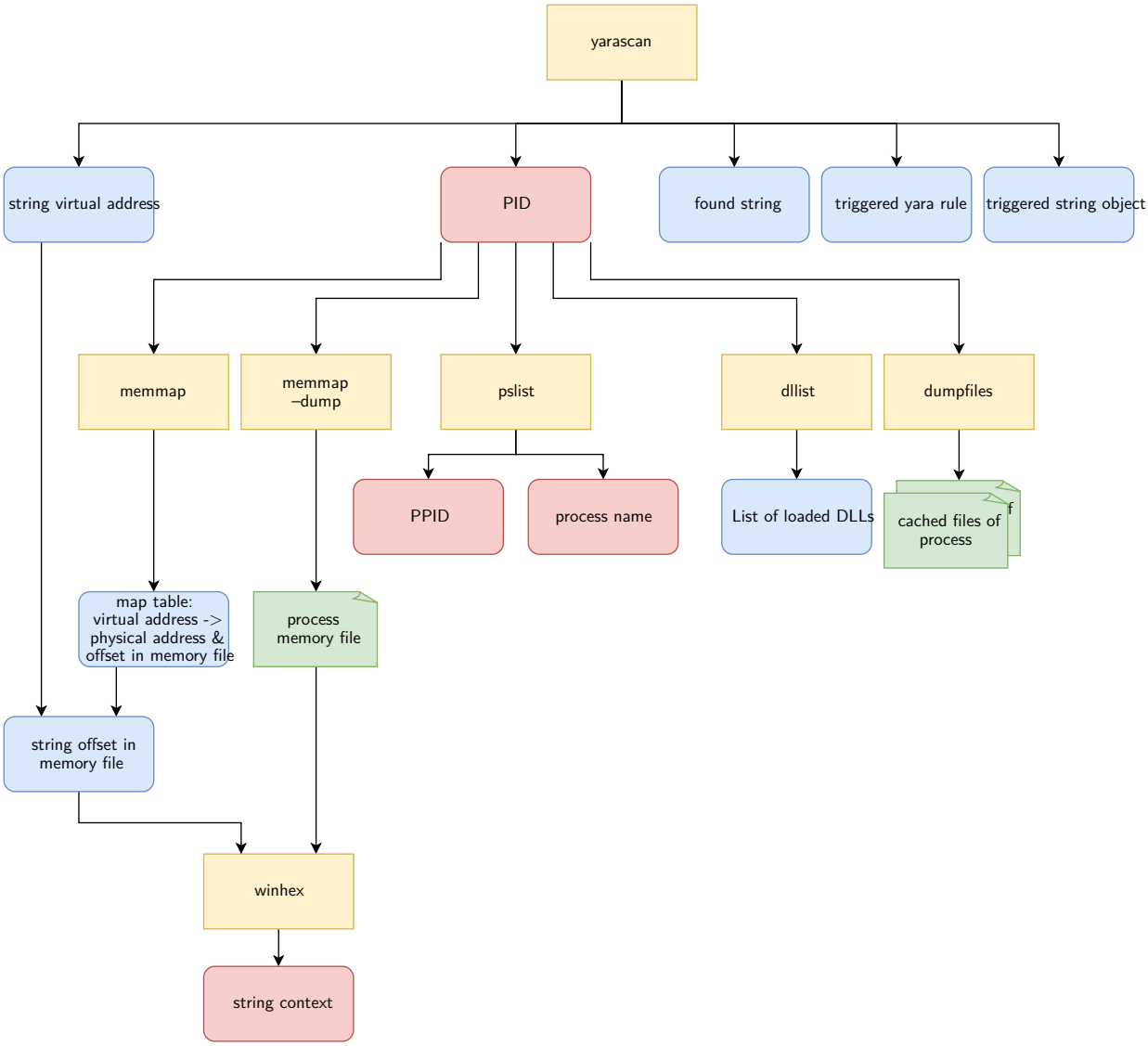


Abbildung 4.3: TODO: Process Monitor Write Operation to Excel Spreadsheet

5 Ergebnisse

= LÄNGSTES/AUSFÜHRLICHSTES KAPITEL!!!

Für jedes Unterkapitel gilt: > Erst allgemeines Vorgehen/Methodik definieren > Danach spezifisch für jeden Browser: Unterschied zwischen Snapshot-Zeitpunkten, insb. zwischen Live- und Dead-Forensik

5.1 Firefox

White-Box Analyse/Common Locations

Schreiboperationen mit Process Monitor verfolgen:

Im Anhang: Tabelle mit allen geschriebenen Dateien (markiert, wenn nicht mehr wiederherstellbar + markiert, wenn Datei "verändert"(siehe oben: temp, WAL))

Aux-Dateien, welche nicht mehr vorhanden waren, aber dafür "richtige"Dateien: - - - - - -

Ergebnis: Tabelle mit wiederherstellbaren Dateien: Logfile 1 vs. Logfile 2 + Tool mit dem Datei untersucht wurde - Dateien, die in beiden Logfiles nicht wiederherstellbar

		Logfile 1	Logfile 2
Cache	cache2\entries\037778A55E1B7E9BED3390289866D09402D6C913	Keine PB Artefakte	Keine Schreiboperation
	cache2\entries\1223A0378B8971FA4CD25EA1731C80B2B1676B42	Keine PB Artefakte	Keine Schreiboperation
	cache2\entries\250EE2BC03AFF526F1A1C3DB212A79DE3EB60D5E	Keine PB Artefakte	Keine Schreiboperation
	jumpListCache\ZKJGVJPzPe7w4w0KwEY0jw==.ico	Keine PB Artefakte	Keine Schreiboperation
	cache2\index.log	Keine Schreiboperation	Keine PB Artefakte
	cache2\index	Keine Schreiboperation	Keine PB Artefakte
Datareporting	datareporting\glean\events\pageload	Keine PB Artefakte	Keine Schreiboperation
	*datareporting\glean\db\data.safe	Keine PB Artefakte	Keine PB Artefakte
	*datareporting\archived\2023-05\1683405837882.9102466b-e465-4ecb-810f-74ae90c64c63.new-profile.jsonlz4	Keine Schreiboperation	Keine PB Artefakte
	*datareporting\archived\2023-05\1683405837905.86f4c992-6329-415b-8c29-911a2d4b7f9d.event.jsonlz4	Keine Schreiboperation	Keine PB Artefakte
	*datareporting\archived\2023-05\1683405837939.abf8b065-41a4-4e94-a044-1cead61e396a.main.jsonlz4	Keine Schreiboperation	Keine PB Artefakte
SQLite	storage\permanent\chrome\idb\3870112724rsegmnoittet-es.sqlite	Keine PB Artefakte	Keine Schreiboperation
	storage\permanent\chrome\idb\1657114595AmcateirvtiSty.sqlite	Keine PB Artefakte	Keine PB Artefakte
	places.sqlite	Keine PB Artefakte	Keine PB Artefakte
	cookies.sqlite	Keine PB Artefakte	Keine Schreiboperation
	formhistory.sqlite	Keine PB Artefakte	Keine Schreiboperation
	webappsstore.sqlite	Keine Schreiboperation	Keine PB Artefakte
	favicons.sqlite	Keine PB Artefakte	Keine Schreiboperation
	storage.sqlite	Keine PB Artefakte	Keine Schreiboperation
Sessionstore	*sessionstore-backups\recovery.jsonlz4	Keine PB Artefakte	Keine PB Artefakte
Sonstige Dateien	prefs-1.js	Keine PB Artefakte	Keine PB Artefakte
	*xulstore.json	Keine PB Artefakte	Keine PB Artefakte

Abbildung 5.1: Tabelle mit wiederherstellbaren Dateien: Logfile 1 vs. Logfile 2

Allgemein: Artefakte in zwei "Common Pfaden (Local) - (Roaming)

Kategorien der Logs: - Cache: > > > Zweck: "Firefox verwendet den Cache, um Webseiten und Ressourcen wie Bilder, Stylesheets, Skripte und andere Dateien temporär auf dem lokalen Computer

zu speichern. Dadurch können wiederholte Anfragen an den Server vermieden und die Ladezeiten verringert werden, da der Browser die Inhalte aus dem Cache abrufen kann, anstatt sie erneut herunterzuladen.Die tatsächlichen Inhalte dieser Datei sind binär und können je nach Art der Ressource variieren, beispielsweise HTML, Bild- oder Audiodateien.Änalyse: - Tool: MozillaCacheView - TODO: Screenshot > Zweck: "Die Indexdatei im Cache dient als Datenbank, die Informationen über die gespeicherten Dateien enthält. Sie ermöglicht dem Firefox-Browser, schnell auf die zwischengespeicherten Ressourcen zuzugreifen und diese effizient zu verwaltenÄnalyse: - Tool: siehe Github und HxD > - Tool: Windows Foto App - Enthält kleines "mÏcon

- datareporting: Allgemein: "Dateien im Ordner /datareporting/glean/dbßind Teil des Glean-Systems, das von Mozilla (dem Entwickler von Firefox) für die Sammlung von Telemetriedaten verwendet wird. Telemetrie-Daten sind anonyme Informationen über die Nutzung des Browsers, die zur Verbesserung der Software und zur Behebung von Problemen verwendet werden können. > Zweck: "Die "data.safe.binDatei enthält verschlüsselte Telemetrie-Daten, um ihre Integrität und Sicherheit zu gewährleisten.Änalyse: - Tool: HxD - keine PB Artefakte > Zweck: "Diese Dateien speichern Informationen über das Firefox-Profil, das von Glean verwendet wird, um Telemetriedaten zu sammeln.Änalyse: - Mit firefox propreritärem jsonlz4 Algorithmus verschlüsselt - können mit speziellem Tool "dejsonlz4"dekomprimiert werden (Quelle Github) - Dateien enthalten Systeminformationen im Json-Format (Screenshot?)

- Sessionstore-Backup: > Zweck: "Die Datei "recovery.jsonlz4ënthält eine Sicherungskopie der vorherigen Sitzung. Sie wird erstellt, wenn der Firefox-Browser nach einem Absturz oder einem unerwarteten Beenden neu gestartet wird.Änalyse: - jsolz4 Datei in sessionstore-backup lassen sich mit Online-Tool parsen (https://www.jeffersonscher.com/ffu/scrounger.html) - Ergebnise: Tab 1: Willkommen bei Firefox [6.5.2023, 22:25:06, about:welcome; Tab 2: Firefox Datenschutzhinweis — Mozilla [6.5.2023, 22:24:59], - Sind Seiten, die sich automatisch geöffnet haben, nachdem Firefox zum ersten Mal geöffnet wurde - keine PB Artefakte > Zweck: "Die Datei ßessionstore.jsonlz4ßpeichert den aktuellen Zustand der Firefox-Sitzung. Diese Datei wird während der Browsersitzung regelmäßig aktualisiert, um sicherzustellen, dass Änderungen im Zustand der Sitzung erfasst werden.Änalyse: - Lässt sich nicht mit Online-Tool aus Logfile 1 parsen - Stattdessen: dejsonlz4, danach Notepad++ mit JSON Plugin - kaum Einträge zu Sitzung, hauptsächlich CSS Daten zu Fenstergröße- und position, insb. keine PB Artefakte - Interessant: image-Eintrag als base64 entdeckt, in PNG umgewandelt (https://base64.guru/converter/decode/image), mit Windows Foto-App: "mÏcon (Mozilla-Logo)

- Sonstige Dateien: > Zweck: "Die Datei "prefs-1.jsënthält benutzerspezifische Einstellungen und Konfigurationen für den Firefox-Browser. Sie speichert die Präferenzen des Benutzers in Form von JavaScript-Objekten.

In dieser Datei werden verschiedene Arten von Einstellungen gespeichert, darunter:

Allgemeine Einstellungen: Dies umfasst Optionen wie die Standardsuchmaschine, die Startseite, den Zoomfaktor, die Spracheinstellungen und andere globale Einstellungen, die das Verhalten des Browsers beeinflussen.

Datenschutzeinstellungen: Hier werden Präferenzen bezüglich Cookies, Verlauf, Passwortverwaltung, Standortfreigabe und Tracking-Schutz gespeichert. Diese Einstellungen kontrollieren, wie der Browser mit persönlichen Daten und der Privatsphäre umgeht.

Add-On-Einstellungen: Wenn der Benutzer Erweiterungen oder Add-Ons installiert hat, können in dieser Datei die spezifischen Einstellungen und Konfigurationen für jedes Add-On gespeichert werden."

> Zweck: "Die Datei "xulstore.jsonßpeichert benutzerspezifische Anpassungen und Konfigurationen für den Firefox-Browser.Änalyse: - Weder JSON-Dateien (Notepad++) noch .tmp Dateien (HxD) enthalten PB Artefakte

- SQLite: (TODO: Abgleich mit Diffs-Exceltabelle, ob wirklich nur in places.sqlite geschrieben wurde) Dateien haben Sonderstellung: - Diese DBs dienen zur Verwaltung und Speicherung sämtlicher Browser Artefakte, insb. der Browser Historie - Aus diesem Grund: Dateien intensiver betrachtet Siehe Kapitel Methodik: > Entwicklung von Dateiinhalt in allen Snapshots (1, 2, 3 und 4) betrachtet > Für jeden Snapshot: - SQLite-Datei extrahiert und mit SQLite-Datei aus vorherigem Snapshot verglichen - Untersuchung der SQLite-Dateien mit SQLite-Viewer (GUI-Tools) - Wenn zu SQLite-Datei WAL-Datei existiert: mit sqlite3 Kommandozeilentool PRAGMA wal_checkpoint durchgeführt, danach neue SQLite-Datei mit ursprünglicher SQLite-Datei verglichen > Dabei gibt es drei Zustände: - leere Datei - neuer (nicht-leerer) Inhalt - gleichbleibender Inhalt Mit Process Monitor Logfiles festgestellt, dass in folgende SQLite-DBs geschrieben: - places.sqlite "Diese Datenbank enthält Informationen über die Lesezeichen, den Verlauf und die Tags im Firefox-Browser. Sie speichert die URLs der besuchten Websites, die Zeitstempel der Besuche, die Titel der Seiten und andere relevante Daten. cookies.sqlite Speichert Webseiten-Cookies Ïn dieser Datenbank werden die Cookies gespeichert, die von Websites im Firefox-Browser verwendet werden. Cookies sind kleine Textdateien, die von Websites auf dem Computer des Benutzers abgelegt werden und verschiedene Informationen speichern können, z. B. Anmeldeinformationen, Sitzungsdaten oder Präferenzen. storage.sqlite Speicher für Webseiten "Diese Datenbank wird von Firefox verwendet, um verschiedene Arten von Webdaten zu speichern, wie z. B. die IndexedDB-Datenbanken von Websites, Offline-Cache-Daten, Webseiten-Skriptdaten und andere lokale Speicherinformationen. favicons.sqlite Speichert Icons für Lesezeichen "Diese Datenbank speichert die Favicons, also die kleinen Symbole, die in der Adressleiste und bei den Lesezeichen angezeigt werden, um Websites visuell zu identifizieren. Sie enthält die gespeicherten Favicons für die besuchten Websites. webappsstore.sqlite Speicher für Webseiten "Diese Datenbank speichert Informationen über installierte Webanwendungen im Firefox-Browser. Sie enthält Daten wie Berechtigungen, Einstellungen und andere spezifische Informationen für Webanwendungen. formhistory.sqlite Ïn dieser Datenbank werden Informationen aus Webformularen gespeichert, die der Benutzer in Firefox ausfüllt. Sie enthält die eingegebenen Daten wie Name, E-Mail-Adresse, Adresse und andere Formulardaten, um das automatische Ausfüllen von Formularen zu ermöglichen. 1657114595AmcateirvtiSty.sqlite Äctivity Stream for Firefox is a collection of all the things you do in the browser that you care about displayed in a rich and meaningful way 3870112724rsegmnoittet-es.sqlite "Remote Settings in Firefox sind eine Funktion, die es ermöglicht, Browser-Einstellungen zentral zu verwalten und an die Benutzer zu übertragen, ohne ein vollständiges Browser-Update durchführen zu müssen.SSQLite DB ist Datenspeicher dazu

Ergebnisse: > Nach Browser-Installation noch keine SQLite-Datei angelegt (Snapshot 1) > Während Browsing Szenario alle DBs Initialisiert, außer "webappsstore.sqlite"(Snapshot 2) - Dabei wurden in places.sqlite die Seiten geschrieben, die sich automatisch nach Browserstart im public Modus geöffnet haben (Datenschutzhinweise zu Firefox) - Restliche Dateien ohne Inhalt, nur Spaltennamen - Nach WAL Checkpoints bleiben Dateien unverändert > Nach Schließen des Browsers (Snapshot 3) - in places.sqlite: Indizes bei eingetragenen Seiten aktualisiert - 1657114595AmcateirvtiSty.sqlite erhielt BLOB Eintrag, in HxD keine Muster erkennbar - webappsstore.sqlite: leer initialisiert, nur

	File	Snapshot 1: Browser Installation	Snapshot 2: After Browsing Scenario, Browser open		Snapshot 3: After Browsing Scenario, Browser closed		Snapshot 3: Browser closed	
			Vor WAL	Nach WAL	Vor WAL	Nach WAL	Vor WAL	Nach WAL
SQLite	places.sqlite	N/A	Initialisiert, Zeilen: Einträge für autom. geöffnete Seiten	no diff	Indizes bei vorhandenen Seiten aktualisiert	no diff	no diff	no diff
	cookies.sqlite	N/A	Leer initialisiert (Nur Spaltennamen)	leer	leer	leer	leer	leer
	storage.sqlite	N/A	Leer initialisiert (Nur Spaltennamen)	leer	leer	leer	leer	leer
	favicons.sqlite	N/A	Leer initialisiert (Nur Spaltennamen)	leer	leer	leer	leer	leer
	webappsstore.sqlite	N/A	N/A	N/A	Leer initialisiert (Nur Spaltennamen)	leer	leer	leer
	formhistory.sqlite	N/A	Leer initialisiert (Nur Spaltennamen)	leer	leer	leer	leer	leer
	1657114595AmcateirvtiSty.sqlite	N/A	Initialisiert, 1 Zeile: "origin: chrome"	no diff	Einträge (Binärdaten) eingefügt, keine PB Artefakte (HxD)	no diff	no diff	no diff
	3870112724rsegmnoittet-es.sqlite	N/A	Initialisiert, 1 Zeile: "origin: chrome"	no diff	no diff	no diff	no diff	no diff

Leer
Unverändert (nicht-leer)
Neuer (nicht-leerer) Inhalt

Abbildung 5.2: Comparison of found PB artifacts between RAM Dumps

Spaltennamen - restliche Dateien unverändert - nach WAL Checkpoints bleiben Dateien unverändert > Nach herunterfahren der VM (Snapshot 4) - Alle Dateien unverändert, auch nach WAL Checkpoint

- Zusammenfassung: in keiner Datei PB Artefakte

Quantitativ: (Diagramme) > Balkendiagramm: Für jede Logfilekategorie: Anzahl Schreiboperationen Logfile 1 vs Logfile 2

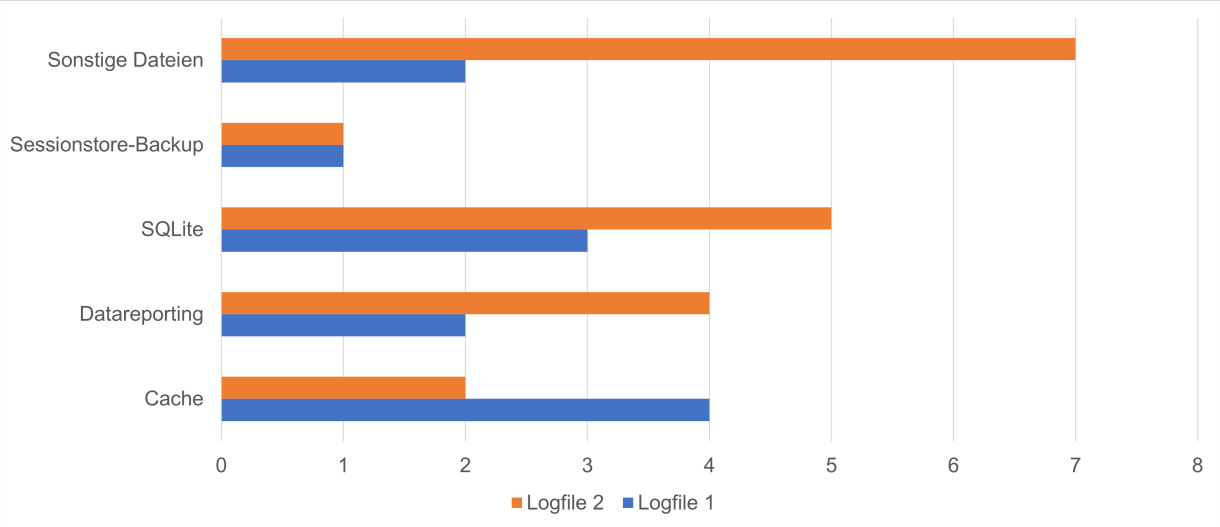


Abbildung 5.3: Comparison of found PB artifacts between RAM Dumps

Registry

> Process Monitor: SetValue Operationen von Browser TODO: Logfile 1 vs 2? Kategorien Registry Keys: 1) PreXULSkeletonUISettings: > Prefix: Absoluter Installationspfad von Firefox > Skeleton UI Einstellungen von Firefox Definition: > Der "PreXULSkeletonUISettings"Registry Key enthielt Einstellungen für die Benutzeroberfläche (UI) des Firefox-Browsers, insbesondere für das sogenannte SSkeleton UI". Das Skeleton UI ist eine vereinfachte Benutzeroberfläche, die während des Ladens des Browsers angezeigt wird, bevor die vollständige Benutzeroberfläche geladen ist. Es besteht aus grundlegenden Steuerelementen und Elementen, die dem Benutzer die Interaktion ermöglichen, während der Rest der Benutzeroberfläche noch geladen wird. > Der "PreXULSkeletonUISettingsSchlüssel enthielt

Konfigurationsoptionen wie Farben, Positionen und andere Einstellungen für das Skeleton UI. Durch das Bearbeiten dieses Schlüssels konnten Benutzer die Darstellung des Skeleton UI anpassen. Es ist jedoch wichtig zu beachten, dass das Ändern der Registrierungseinträge ein fortgeschrittenes Verfahren ist und Fehler zu Problemen mit dem Browser führen kann.

> Struktur der Keys: > Unterschiedliche UI Einstellungen - - - - - - - - - - > keine PB Artefakte unter UI Einstellungen 2) Business Activity Monitoring > Quelle: > BAM is a mostly undocumented feature that controls the programs executed in the background. DAM is a feature for devices supporting the "Connected Standby"mode (i.e when a device is turned on, but its display will be turned off). As a result, the BAM registry keys will contain data on any devices, while DAM registry keys will only contain data on mobile devices. > The BAM registry key contains multiple subkeys under bam
State
UserSettings, with one subkey per user, identified with the user SID. While the key is in the SYSTEM registry hive, program executions can thus still be tied to a specific user using this SID. > Each user-specific key contains a list of executed programs, with their full path and timestamp of last execution. > If a file is deleted, the eventual associated entry in the BAM is deleted as well after the system reboot. Additionally, BAM entries older than 7 days are deleted upon system boot. The BAM thus provides limited information on historic execution of programs > No entries are created in the BAM keys for executables on removable media and/or on network shares. > Key:

Quantitativ: (Diagramme) - Stacked Balkendiagramm jeweils für Logfile 1 und Logfile2: Anteil Kategorie 1 bzw.2 an allen Registry-Schreiboperationen

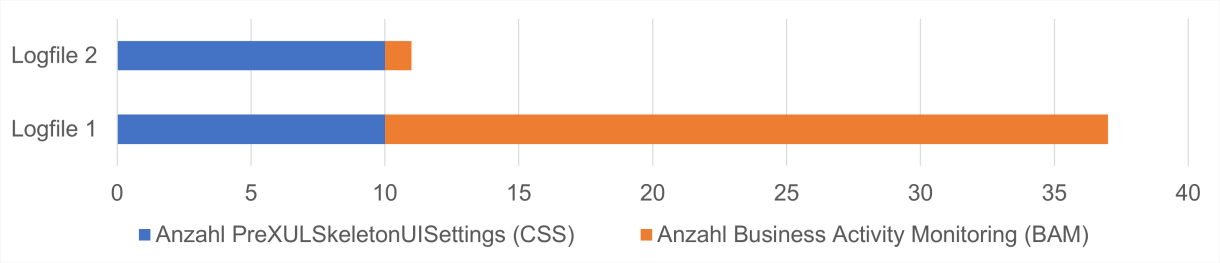


Abbildung 5.4: Comparison of found PB artifacts between RAM Dumps

> Stringsuche in Registry Hives mit Registry Explorer (Siehe Liste) In allen Hives kein Treffer für alle Suchbegriffe

Literatur: > angeblich in Shellactivities Ergebnisse. –> Nicht mehr vorhanden in aktueller Version (Verweis auf E-Mail)

Black-Box Analyse/Uncommon Locations

Analyse mit Autopsy

Bei White-Box Analyse/Common Locations: Autopsy nur zur Dateiextraktion genutzt, hier: als konkretes forensisches Werkzeug

Stichwortsuche: - In allen Snapshots keine Treffer (auch innerhalb $Carved) - TODO: Pagefile gefunden?

Von Autopsy automatisch indexierte Dateien: In allen Fällen: keine Dateien gelöscht, nur über Zeitraum der Snapshots neue dazugekommen - Web Bookmarks: Snapshot 1: > Bing.url (Unter

Source Name	S	C	O	URL	Title	Date Created	Program Name	Domain	Data Source
Bing.url			19	http://go.microsoft.com/fwlink/p/?LinkId=255142	Bing.url	2023-04-25 16:09:28 MESZ	Internet Explorer Analyzer	microsoft.com	CFV_Firefox_Klon_Snapshot_3.img
places.sqlite			6	https://support.mozilla.org/products/firefox	Hilfe erhalten	2023-05-06 22:25:00 MESZ	FireFox Analyzer	mozilla.org	CFV_Firefox_Klon_Snapshot_3.img
places.sqlite			6	https://support.mozilla.org/kb/customize-firefox-controls-b...	Firefox anpassen	2023-05-06 22:25:00 MESZ	FireFox Analyzer	mozilla.org	CFV_Firefox_Klon_Snapshot_3.img
places.sqlite			6	https://www.mozilla.org/contribute/	Mitmachen	2023-05-06 22:25:00 MESZ	FireFox Analyzer	mozilla.org	CFV_Firefox_Klon_Snapshot_3.img
places.sqlite			6	https://www.mozilla.org/about/	Über uns	2023-05-06 22:25:00 MESZ	FireFox Analyzer	mozilla.org	CFV_Firefox_Klon_Snapshot_3.img
places.sqlite			6	https://www.mozilla.org/firefox/?utm_medium=firefox-des...	Erste Schritte	2023-05-06 22:25:01 MESZ	FireFox Analyzer	mozilla.org	CFV_Firefox_Klon_Snapshot_3.img

Abbildung 5.5: Autopsy Web Bookmarks

C:/User/Forensik/Favorites/Links) enthält Bing Startseite Snapshot 2: > 5 Einträge in places.sqlite: (Firefox Standardseiten -> Deckt sich mit Beobachtungen aus Process Monitor Analyse, siehe Kapitel X) Snapshot 3: > unverändert zu 2 Snapshot 4: > unverändert zu 3 - Web Cookies: Snapshot 1: > 10

Source Name	S	C	O	URL	Date Accessed	Name	Value	Program Name	Domain	Data Source
WebCacheV01.dat			15	bing.com	2023-05-06 19:50:17 MESZ	SUID	M	Microsoft Edge Analyzer	bing.com	CFV_Firefox_Klon_Snapshot_3.img
WebCacheV01.dat			15	www.bing.com	2023-05-06 19:51:24 MESZ	MUIDB	31708C5FC3CF47068AFADC1CB47D0111	Microsoft Edge Analyzer	bing.com	CFV_Firefox_Klon_Snapshot_3.img
WebCacheV01.dat			15	bing.com	2023-05-06 19:50:17 MESZ	SRCHD	AF=NOFORM	Microsoft Edge Analyzer	bing.com	CFV_Firefox_Klon_Snapshot_3.img
WebCacheV01.dat			15	bing.com	2023-05-06 19:50:17 MESZ	SRCHUID	V=2&GUID=B2C50ADBCB9B4234A9FE14DBB1DCB91D&dm...	Microsoft Edge Analyzer	bing.com	CFV_Firefox_Klon_Snapshot_3.img
WebCacheV01.dat			15	bing.com	2023-05-06 19:50:17 MESZ	SRCHUSR	DOB=20230506	Microsoft Edge Analyzer	bing.com	CFV_Firefox_Klon_Snapshot_3.img
WebCacheV01.dat			15	bing.com	2023-05-06 19:50:20 MESZ	SRCHHPGUSR	SRCHLANG=de&LUT=1683402619223&IPMH=dee204058&I...	Microsoft Edge Analyzer	bing.com	CFV_Firefox_Klon_Snapshot_3.img
WebCacheV01.dat			15	bing.com	2023-05-06 19:50:19 MESZ	CortanaAppUID	C164AA3A4D47E127DDC66AD915CFD04C	Microsoft Edge Analyzer	bing.com	CFV_Firefox_Klon_Snapshot_3.img
WebCacheV01.dat			15	bing.com	2023-05-06 19:55:22 MESZ	ANON	A=A3B5B679A14D59B0AA027635FFFFFFFF	Microsoft Edge Analyzer	bing.com	CFV_Firefox_Klon_Snapshot_3.img
WebCacheV01.dat			15	live.com	2023-05-06 19:50:30 MESZ	MUID	118A534093A9626528C5404997A966B8	Microsoft Edge Analyzer	live.com	CFV_Firefox_Klon_Snapshot_3.img
WebCacheV01.dat			15	login.live.com	2023-05-06 19:51:06 MESZ	__Host-MSAAUTHP		Microsoft Edge Analyzer	live.com	CFV_Firefox_Klon_Snapshot_3.img

Abbildung 5.6: Autopsy Web Cookies

Einträge in WebCacheV01.dat (= DB des Internet Explorers zum speichern von Browserdaten): Cookies für bing.com und live.com (= outlook) Snapshot 2: > unverändert zu 1 Snapshot 3: > unverändert zu 2 Snapshot 4: > unverändert zu 3 - Web History: Snapshot 1: > 2 Einträge in WebCacheV01.dat:

Source Name	S	C	O	URL	Date Accessed	Referrer URL	Title	Program Name	Domain	Data Source	Username
places.sqlite			6	https://www.mozilla.org/de/privacy/firefox/	2023-05-06 22:25:00 MESZ	https://www.mozilla.org/privacy/firefox/	Firefox Datenschutzhinweis — Mozilla	FireFox Analyzer	mozilla.org	CFV_Firefox_Klon_Snapshot_3.img	
WebCacheV01.dat			15	https://login.live.com/oauth20_desktop.srf?lc=1031	2023-05-06 19:51:06 MESZ			Microsoft Edge Analyzer	live.com	CFV_Firefox_Klon_Snapshot_3.img	Forensik
WebCacheV01.dat			15	https://login.live.com/oauth20_authorize.srf?client_i...	2023-05-06 19:51:08 MESZ			Microsoft Edge Analyzer	live.com	CFV_Firefox_Klon_Snapshot_3.img	Forensik
WebCacheV01.dat				file:///Z:/Logfile_1	2023-05-06 20:29:36 MESZ			Microsoft Edge Analyzer		CFV_Firefox_Klon_Snapshot_3.img	Forensik
WebCacheV01.dat				file:///Z:/Logfile_2	2023-05-06 20:44:19 MESZ			Microsoft Edge Analyzer		CFV_Firefox_Klon_Snapshot_3.img	Forensik

Abbildung 5.7: Autopsy Web History

- 2x live.com (= outlook) Snapshot 2: > 1 Eintrag in places.sqlite: -> Zurückzuführen auf Seite, die sich automatisch geöffnet hat, als Firefox gestartet (bevor privates Fenster geöffnet wurde) > 1 neuer Einträge in WebCacheV01.dat: - file:///Z:/Logfile_1 (= Process Monitor Logfile, die in shared-Folder geladen wurde) -> Erklärung? Snapshot 3: > 1 neuer Eintrag in WebCacheV01.dat: - file:///Z:/Logfile_2 (= Process Monitor Logfile, die in shared-Folder geladen wurde) -> Erklärung? Snapshot 4: > unverändert zu 3 - Web Categories: Snapshot 1: > 2x WebCacheV01.dat aufgelistet =>

Source Name	S	C	O	Source Type	Score	...	...	...	Domain	Host	Name	File Path
WebCacheV01.dat			0	File	Unknown				bing.com	bing.com	Search Engine	/img_CFV_Firefox_Klon_Snapshot_3.img/vol_vol3/Users/Forensik/AppData/Local/Microsoft/Windows/WebCache/WebCacheV01.dat
WebCacheV01.dat			0	File	Unknown				live.com	login.live.com	Web Email	/img_CFV_Firefox_Klon_Snapshot_3.img/vol_vol3/Users/Forensik/AppData/Local/Microsoft/Windows/WebCache/WebCacheV01.dat

Abbildung 5.8: Autopsy Web Categories

Mit HxD untersucht, keine PB Artefakte Snapshot 2: > unverändert zu 2 Snapshot 3: > unverändert zu 3 Snapshot 4: > unverändert zu 4

Zusammenfassung: - keine PB Artefakte - Keine neuen Erkenntnisse vgl. mit intensiver Analyse mittels Process Monitor in Kapitel X - Eintrag von Datenschutzseite in places.sqlite wurde erkannt.

Analyse mit Volatility

Vorgehen: Siehe "Methodik"Kapitel - Ausgangslage: Volatility Yarascan Treffer - Für jeden Treffer: virtueller Offset des Strings, PID, getriggerte Yararule, getriggerte Yara Component z(= Variablenname des gesuchten Strings), gefundener String - Neue Spalte: "Prozessname> zu jeder PID Prozessnamen - Ergebnisse Aufbereitet nach folgendem Schema: > Für jeden RAM Dump > Für jede Yararule > Für jede Component > Filter: Prozessname = Firefox -> Anzahl zählen > Filter: Prozessname = Alle Prozesse außer Firefox -> Anzahl zählen

HTML Artefakte wurden in keinem RAM Dump gefunden => Nicht aufgeführt

Yararule "Keyword": Analyse: > Ausschließlich in RAM Dump 2 Keyword Artefakte gefunden >

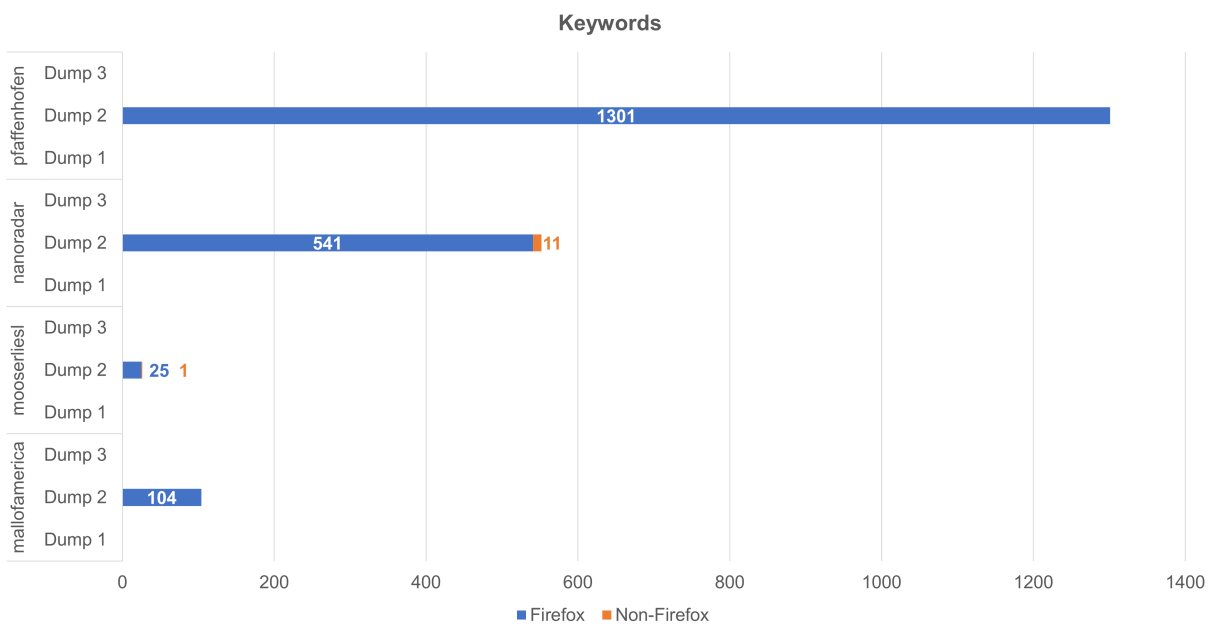


Abbildung 5.9: Keywords

Hauptsächlich in Firefox Prozess > Mit 1301 Artefakten, am häufigsten pfaffenhofen vertreten. Vermutung: Evtl. weil Google Maps viele zusätzliche Artefakte lädt.

Yararule ÜRL": Analyse: > Wie bei anderen Kategorien: Die meisten Artefakte in RAM Dump 2, in Firefox Prozessen > mooserliesl tritt am wenigsten auf, donaukurier am meisten (vmtl. auf Öffnen von Bild zurückzuführen) > Hier bemerkenswert, dass in RAM Dump 3 Artefakte von allen vier URLs zu finden sind > Bei genauerer Analyse des Process Monitor Logfiles herausgefunden: Artefakte alle in svchost.exe Prozess gefunden > Deshalb RAM Dump erneut mit Volatility windows.svcscan Plugin

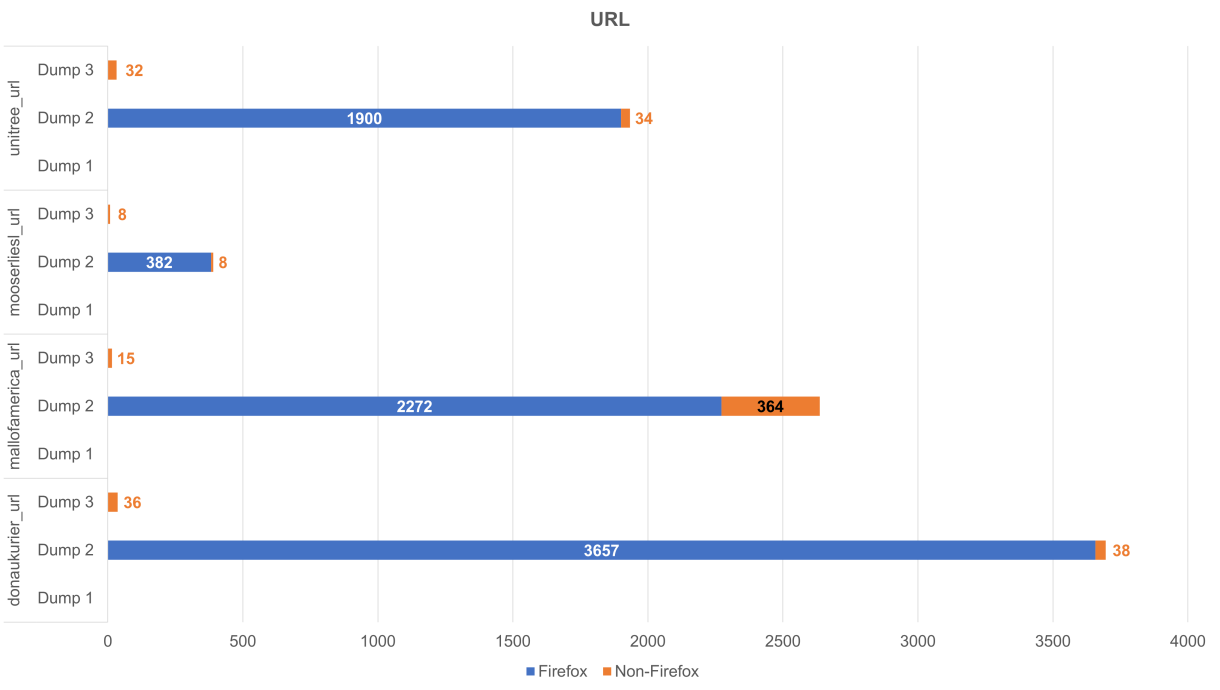


Abbildung 5.10: URL

untersucht: "The svcscan plugin allows the analyst to list out the services running. This plugin gives more detail to the running processes in the event that the analyst requires additional details such as the display name, binary path, or service type.» Ausgabe aller im RAM gefundener Services > Problem: Volatility svcscan liefert keine PID zu laufenden Services > Deshalb: "White-BoxÄnalyse: Snapshot 3 erneut aufgetaut, danach mit Process Explorer PID X (TODO!) von SVChost Prozess gesucht, in dem PB Artefakte gefunden wurden Def. Process Explorer: "Prozess Explorer zeigt Ihnen Informationen darüber an, welche Handles und DLLs-Prozesse geöffnet oder geladen wurden.Process Explorer, from Sysinternals, is a process management program that allows you to see the running processes on your computer and a great deal of information about each process. One of the nice features of Process Explorer is that it also gives you the ability to see what services a particular SVCHOST.EXE process is controlling.» Ergebnis: DNSCache Service mit PID X (TODO!) = DNScache Service TODO: Screenshot > Ausführung von ipconfig /displaydns liefert gecachte URLs TODO: Screenshot > Nach Löschen des DNSCaches mit ipconfig /flushdns + Schließen aller Process Monitor Instanzen + Beenden des DNSCaches Services + Erneuter RAM-Dump -> Keine PB Artefakte mehr gefunden! Yararule "Mail": Analyse: > Alle Mail Artefakte gefunden > Ausschließlich in RAM Dump 2 Mail Artefakte gefunden > Am häufigsten Absenderadresse "computerforensikvl@gmail.com"gefunden, als einziges Artefakt auch in anderen Prozessen gefunden. > Bemerkenswert: Passwort wurde 4x als Klartext im RAM gefunden! String Kontext: Offsets: PIDs: 0xb9ce29180c8 7420 0x2859f4ffd4e0 7420 0x24083b41858 8424 0x240840e5b08 8424

Memmap: Pid 7420 virtual physical size offset in file 0xb9ce2918000 0xcb20a000 0x1000 0x11dd4000 -> 0xb9ce29180c8 = 0x11dd40c8 0x2859f4ffd000 0x96812000 0x1000 0x12e23000 Disabled -> 0x2859f4ffd4e0 = 0x12e234e0 Memmap: Pid 8424 virtual physical size offset in file 0x24083b41000 0xc1d52000 0x1000 0x583000 Disabled -> 0x24083b41858 = 0x583858 0x240840e5000 0x2d3fb000

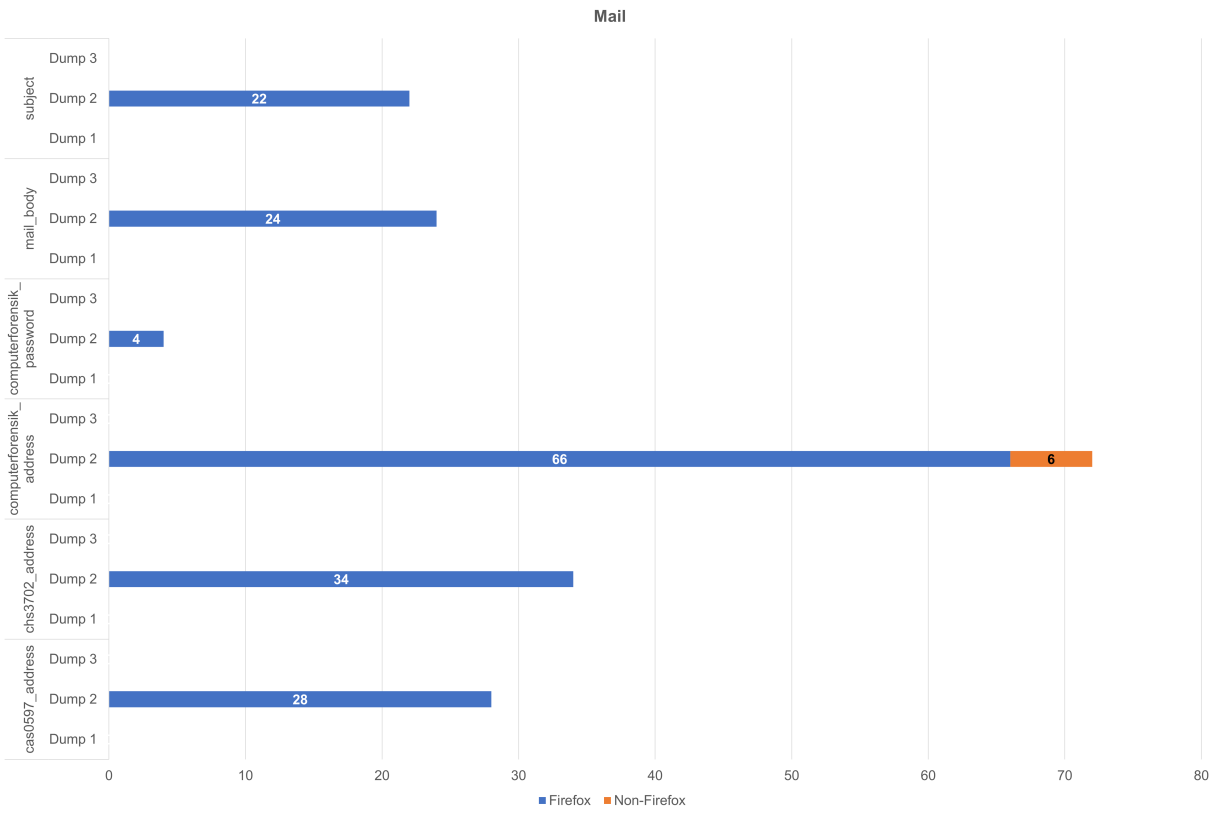


Abbildung 5.11: Mail

```
11DD4040  58 02 00 00 08 00 00 00 67 6D 70 41 64 64 6F 6E  X.......gmpAddon
11DD4050  4B 4B 4B 4B 4B 4B 4B DC F9 0E 50 4B 4B 4B 4B  KKKKKKKKÜù.PKKKK
11DD4060  58 02 00 00 0D 00 00 00 67 6D 70 44 6F 77 6E 6C  X.......gmpDownl
11DD4070  6F 61 64 65 72 4B 4B 4B 50 C3 FB EA 4B 4B 4B 4B  oaderKKKPÃûêKKKK
11DD4080  58 02 00 00 0D 00 00 00 47 4D 50 44 6F 77 6E 6C  X.......GMPDownl
11DD4090  6F 61 64 65 72 4B 4B 4B D0 6F AE 8A 4B 4B 4B 4B  oaderKKKÐo®ŠKKKK
11DD40A0  58 02 00 00 0D 00 00 00 5F 69 73 45 4D 45 45 6E  X......._isEMEEn
11DD40B0  61 62 6C 65 64 4B 4B 4B 8F 4F 0E 73 4B 4B 4B 4B  abledKKK.O.sKKKK
11DD40C0  58 02 00 00 0C 00 00 00 56 6F 72 6C 65 73 75 6E  X.......Vorlesun
11DD40D0  67 32 33 21 4B 4B 4B 4B F8 35 7D 48 4B 4B 4B 4B  g23!KKKKø5}HKKKK
11DD40E0  58 02 00 00 0F 00 00 00 5F 69 73 41 64 64 6F 6E  X......._isAddon
11DD40F0  45 6E 61 62 6C 65 64 4B 42 1D 99 C2 4B 4B 4B 4B  EnabledKB.™ÂKKKK
11DD4100  58 02 00 00 0F 00 00 00 6D 61 69 6C 2E 67 6F 6F  X.......mail.goo
11DD4110  67 6C 65 2E 63 6F 6D 4B 44 47 D9 2D 4B 4B 4B 4B  gle.comKDGÙ-KKKK
11DD4120  58 02 00 00 10 00 00 00 5F 75 70 64 61 74 65 4C  X......._updateL
11DD4130  61 73 74 43 68 65 63 6B 43 1F 7D 4B 4B 4B 4B 4B  astCheckC.}KKKKK
11DD4140  58 02 00 00 10 00 00 00 73 65 63 6F 6E 64 73 53  X.......secondsS
```

Abbildung 5.12: Password in memory page of PID 7420 at offset 0xb9ce29180c8

0x1000 0x96b000 Disabled -> 0x240840e5b08 = 0x96bb08 > In PID 8424: 2 Bytes pro Character, bspw. Unicode

Yararule Ïmage": Analyse: > Hex-Wert von Donaukurier Bild wurde im 2. RAM Dump in 3 Firefox Prozessen gefunden

Zusammenfassung = Stacked Bar Chart:

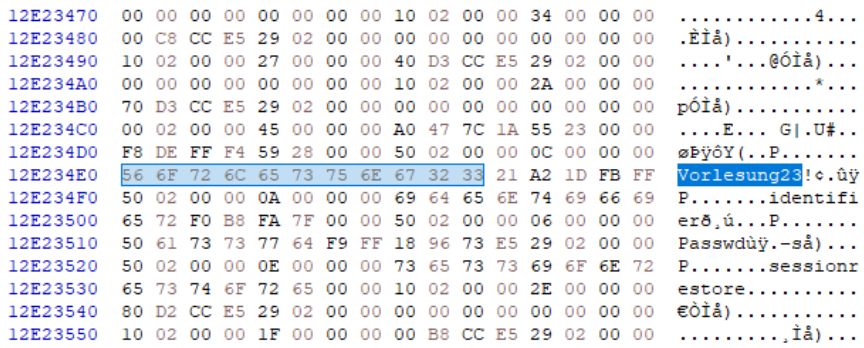


Abbildung 5.13: Password in memory page of PID 7420 at offset 0x2859f4ffd4e0

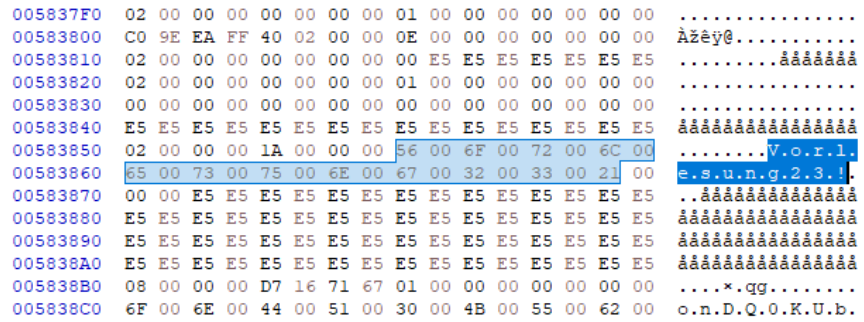


Abbildung 5.14: Password in memory page of PID 8424 at offset 0x24083b41858

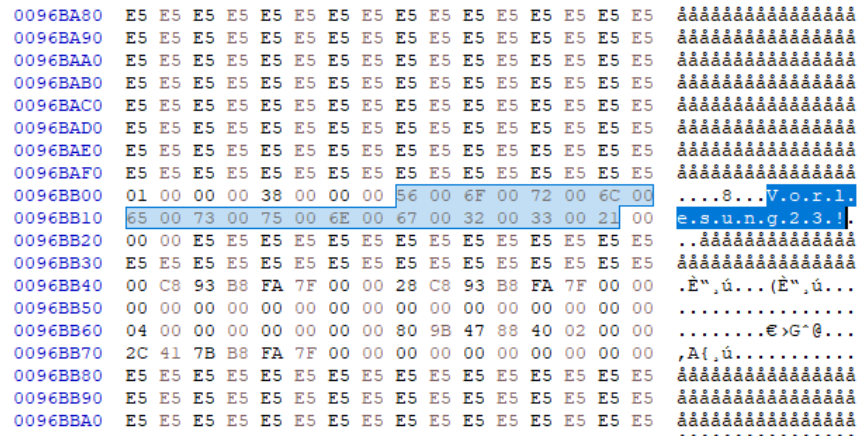


Abbildung 5.15: Password in memory page of PID 8424 at offset 0x240840e5b08

TODO: Kreisdiagramme/Balkendiagramme mit Gesamtzahl an (Non-)Firefox Yarascan-Treffer erst im Vergleich mit Tor

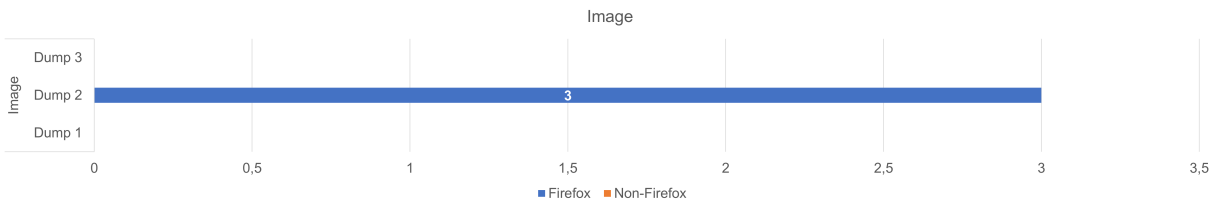


Abbildung 5.16: Image

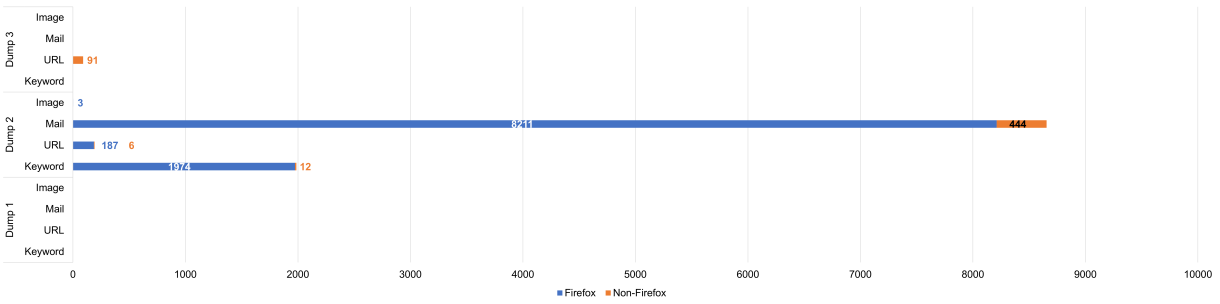


Abbildung 5.17: Summary

5.2 Tor

White-Box Analyse/Common Locations

Schreiboperationen mit Process Monitor verfolgen:

Im Anhang: Tabelle mit allen geschriebenen Dateien (markiert, wenn nicht mehr wiederherstellbar + markiert, wenn Datei "verändert"(siehe oben: temp, WAL))

Aux-Dateien, welche nicht mehr vorhanden waren, aber dafür "richtige"Dateien:

Ergebnis: Tabelle mit wiederherstellbaren Dateien: Logfile 1 vs. Logfile 2 + Tool mit dem Datei untersucht wurde - Dateien, die in beiden Logfiles nicht wiederherstellbar

Allgemein: Tor hat nur einen "Common Pfad - Dateien tauchen in zwei unterschiedlichen Ordnern auf:
- -

- Alle Schreibopertationen von Prozess "firefox.exe"durchgeführt, nicht "tor.exe"

=> Keine der Dateien enthält PB Artefakte, trotzdem nachfolgende genauere Betrachtung der wichtigsten Dateien im Zusammenhang des Tor Browsers

Kategorien der Logs: - Cache: > Zweck: "Die Datei ßtartupCache.8.littleïst eine interne Datei, die von Firefox und dem Tor Browser erstellt wird, um den Startvorgang des Browsers zu beschleunigen. Sie enthält im Wesentlichen eine Zwischenspeicherung von Daten, die beim Starten des Browsers benötigt werden.

Diese Datei enthält Informationen über bereits geladene Browser-Komponenten wie JavaScript-Code, CSS-Dateien, Bilder und andere Ressourcen. Indem der Browser diese Informationen zwischenspeichert,

Kategorie	Datei	Logfile 1	Logfile 2	Logfile 3
Cache	\Caches\profile.default\startupCache\startupCache.8.little	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
Datareporting	\datareporting\glean\db\data.safe.bin	Keine PB Artefakte	Keine Schreiboperationen	Keine PB Artefakte
	*\datareporting\state.json	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
SQLite	\storage\permanent\chrome\idb\3870112724rsegmnoittet-es.sqlite	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
	\storage\permanent\chrome\idb\1657114595AmcateirvtiSty.sqlite	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
	\places.sqlite	Keine PB Artefakte	Keine PB Artefakte	Keine PB Artefakte
	\cookies.sqlite	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
	\storage.sqlite	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
	\favicons.sqlite	Keine PB Artefakte	Keine Schreiboperationen	Keine PB Artefakte
	\webappsstore.sqlite	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
	\formhistory.sqlite	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
Sonstige Dateien	*\addonStartup.json.lz4	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
	\AlternateServices.txt	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
	*\broadcast-listeners.json	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
	*\extensions.json	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
	\extensions\staged\{73a6fe31-595d-460b-a920-fcc0f8843232}.xpi	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
	*\onion-aliases.json	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
	\prefs-1.js	Keine PB Artefakte	Keine PB Artefakte	Keine PB Artefakte
	\security_state\data.safe.bin	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
	\settings\data.safe.bin	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
	\SiteSecurityServiceState.txt	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
	\SiteSecurityServiceState-1.txt	Keine PB Artefakte	Keine Schreiboperationen	Keine Schreiboperationen
	*\profile.default\xulstore.json	Keine PB Artefakte	Keine Schreiboperationen	Keine PB Artefakte
	\profile.default\cert_override.txt	Keine Schreiboperationen	Keine PB Artefakte	Keine Schreiboperationen
	\profile.default\enumerate_devices.txt	Keine Schreiboperationen	Keine PB Artefakte	Keine Schreiboperationen
	\profile.default\sessionCheckpoints.json.tmp	Keine Schreiboperationen	Keine Schreiboperationen	Keine PB Artefakte
	\storage\default\moz-extension+++3041a34e-916a-4fca-8ea0-53f966d7a11f\userContextId=4294967295\.metadata-v2	Keine Schreiboperationen	Keine Schreiboperationen	Keine PB Artefakte
	Caches			
	Profile.Default			

Abbildung 5.18: Tabelle mit wiederherstellbaren Dateien: Logfile 1 vs. Logfile 2

kann er sie beim erneuten Starten des Browsers wiederverwenden, anstatt sie erneut herunterladen und verarbeiten zu müssen. Dadurch wird die Startzeit des Browsers verkürzt und die allgemeine Leistung verbessert.Änalyse: - Tool: HxD - kein PB Artefakte

- datareporting: > Zweck: "Die Datei ßtate.jsonïm Ordner /datareportingënthält Informationen über den Zustand und die Konfiguration des Firefox- oder Tor Browsers. Diese Datei kann Daten über die Verwendung des Browsers, wie z.B. installierte Add-Ons, zuletzt besuchte Websites, Browser-Einstellungen und andere Informationen enthalten. Sie wird verwendet, um dem Browser bei Bedarf den Zustand und die Einstellungen wiederherzustellen.Änalyse: - Tool Notepad++ mit JSON Plugin - keine PB Artefakte

- Sonstige Dateien: > Zweck: ënthält onion URLs, HTTP Alternative Services is a mechanism that allows servers to tell clients that the service they are accessing is available at another network location or over another protocol. This mapping can be stored in a file in the profile folder. This allows websites that do not support HTTPS to communicate in a secure way via port 443 (Opportunistic Encryption).» Zweck: Ïst "NoScriptËxtension. Wenn in Firefox geöffnet, kann installiert werden-> TODO: Screenshot, wenn in Firefox per "drag-and-drop"gezogen > Zweck: Enthält SecureDrop Adressen: z.B. sueddeutsche.securedrop.tor.onion (z.B. > Zweck: The file containing the updated security data > Entielt früher private Browsing Artefakte (https://gitlab.torproject.org/tpo/applications/tor-browser/-/issues/18589), jetzt aber keine private Browsing Artefakte => Keine der Dateien enthält PB Artefakte

- SQLite: Aus Process Monitor Logfiles erkennbar: Tor verwaltet und beschreibt die exakt gleichen SQLite Datenbanken wie Firefox.

Hier ebenfalls gesondert betrachtet: Fokus auf die Entwicklung von Dateiinhalt in allen Snapshots (1, 2, 3-1, 3 und 4) betrachtet

Ergebnisse: > Nach Browser-Installation noch keine SQLite-Datei angelegt (Snapshot 1) > Während Browsing Szenario alle DBs Initialisiert, außer "webappsstore.sqlite"(Snapshot 2) - Dabei wurden in places.sqlite automatisch .onion URLs geschreiben, die zu Tor Standardseiten führen, wie "The Tor

	File	Snapshot 1: Browser Installation	Snapshot 2: After Browsing Scenario, Browser open		(Tor only) Snapshot 3-1: After Identity reset		Snapshot 3: After Browsing Scenario, Browser closed		Snapshot 4: VM Shutdown	
			Vor WAL	Nach WAL	Tor (Diff)	Nach WAL	Tor	Nach WAL	Vor WAL	Nach WAL
SQLite	places.sqlite	N/A	Initialisiert, Zeilen: Onion URLs für Tor Standardseiten, wie "The Tor Blog" oder "Tor Browser Manual" und Spenden-Seite (http://rzuwtpc4wb3xdzrj3yeajsvm3fkq4vbeubm2tdxsqruzzegs5dwcmlsd.onion/).	no diff	Indizes bei vorhandenen Seiten aktualisiert	no diff	Indizes bei vorhandenen Seiten aktualisiert	no diff	no diff	no diff
	cookies.sqlite	N/A	Leer initialisiert (Nur Spaltennamen)	leer	leer	leer	leer	leer	leer	leer
	storage.sqlite	N/A	Leer initialisiert (Nur Spaltennamen)	leer	leer	leer	leer	leer	leer	leer
	favicons.sqlite	N/A	Initialisiert, Einträge mit Präfix fake-favicon-uri", enthalten onion URLS für Tor Standardseiten, wie "The Tor Blog" oder "Tor Browser Manual" und Spenden-Seite (http://rzuwtpc4wb3xdzrj3yeajsvm3fkq4vbeubm2tdxsqruzzegs5dwcmlsd.onion/)	no diff	no diff	no diff	in allen drei tabellen Indizes aktualisiert	no diff	no diff	no diff
	webappsstore.sqlite	N/A	Leer initialisiert (Nur Spaltennamen)	leer	leer	leer	leer	leer	leer	leer
	formhistory.sqlite	N/A	Leer initialisiert (Nur Spaltennamen)	leer	leer	leer	leer	leer	leer	leer
	1657114595Amcatsirvtisty.sqlite	N/A	Leer initialisiert (Nur Spaltennamen)	leer	leer	leer	leer	leer	leer	leer
	3870112724rcogmaoittet-es.sqlite	N/A	Initialisiert, 1 Zeile: "origin: chrome"	no diff	no diff	no diff	gleich bleibend	no diff	no diff	no diff

Leer
Unverändert
Neuer (nicht-leerer) Inhalt

Abbildung 5.19: Comparison of found PB artifacts between RAM Dumps

Blogöder "Tor Browser Manual"bzw. die Tor Spenden-Seite, obwohl keine dieser Seiten aufgerufen wurde TODO: Screenshot von URLs? - in Favicons.sqlite wurden die exakt gleichen Enträge geschrieben, mit dem Präfix "Fake-favicon-uri". Ein tatsächliches Icon wurde nicht in die DB geschrieben - remote settings Datenbank enthielt den gleichen Eintrag wie es bereits bei Firefox der Fall war. Keine PB Artefakte - Restliche Dateien ohne Inhalt, nur Spaltennamen - Nach WAL Checkpoints bleiben Dateien unverändert > Nach Zurücksetzen der Browser-Identität (Snapshot 3-1) - in places.sqlite: Indizes bei eingetragenen Seiten aktualisiert - restliche Dateien unverändert > Nach Schließen des Browsers (Snapshot 3) - in places.sqlite sowie favicons.sqlite: Indizes bei eingetragenen Seiten aktualisiert - restliche Dateien unverändert - nach WAL Checkpoints bleiben Dateien unverändert > Nach herunterfahren der VM (Snapshot 4) - Alle Dateien unverändert, auch nach WAL Checkpoint

- Zusammenfassung: in keiner Datei PB Artefakte

Quantitativ: (Diagramme) > Balkendiagramm: Für jede Logfilekategorie: Anzahl Schreiboperationen Logfile 1 vs Logfile 2

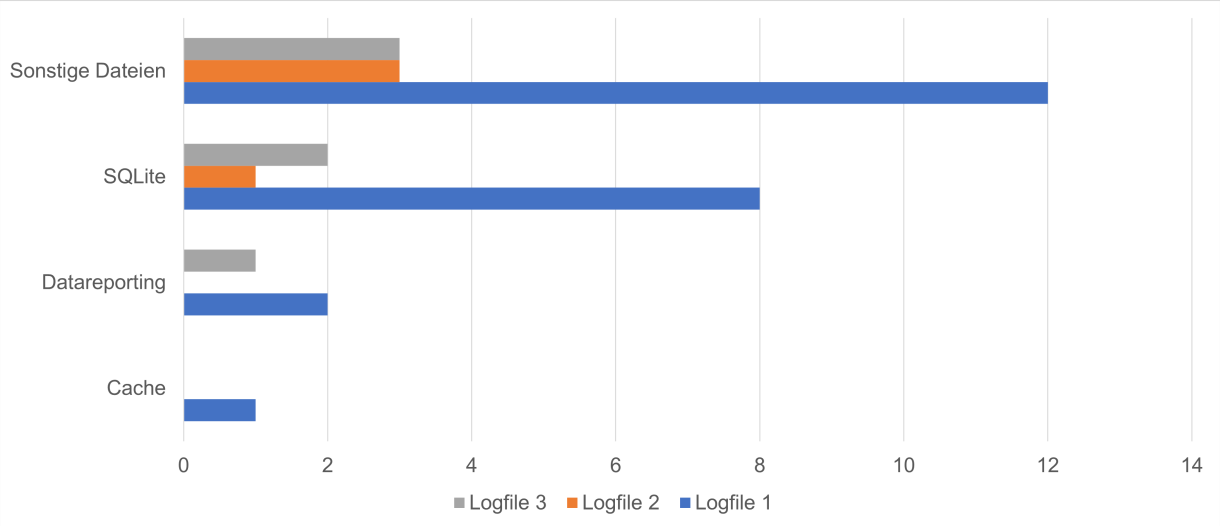


Abbildung 5.20: Comparison of found PB artifacts between RAM Dumps

Registry

> Process Monitor: SetValue Operationen von Browser Kategorien Registry Keys: Analog zu Firefox 1) PreXULSkeletonUISettings: > Prefix: Absoluter Installationspfad von Firefox > Skeleton UI Einstellungen von Firefox Definition: > Der "PreXULSkeletonUISettings"Registry Key enthielt Einstellungen für die Benutzeroberfläche (UI) des Firefox-Browsers, insbesondere für das sogenannte SSkeleton UI". Das Skeleton UI ist eine vereinfachte Benutzeroberfläche, die während des Ladens des Browsers angezeigt wird, bevor die vollständige Benutzeroberfläche geladen ist. Es besteht aus grundlegenden Steuerelementen und Elementen, die dem Benutzer die Interaktion ermöglichen, während der Rest der Benutzeroberfläche noch geladen wird. > Der "PreXULSkeletonUISettingsSchlüssel enthielt Konfigurationsoptionen wie Farben, Positionen und andere Einstellungen für das Skeleton UI. Durch das Bearbeiten dieses Schlüssels konnten Benutzer die Darstellung des Skeleton UI anpassen. Es ist jedoch wichtig zu beachten, dass das Ändern der Registrierungseinträge ein fortgeschrittenes Verfahren ist und Fehler zu Problemen mit dem Browser führen kann.

> Struktur der Keys: > Unterschiedliche UI Einstellungen - - - - - - - - - - > keine PB Artefakte unter UI Einstellungen 2) Business Activity Monitoring > Quelle: > BAM is a mostly undocumented feature that controls the programs executed in the background. DAM is a feature for devices supporting the "Connected Standby"mode (i.e when a device is turned on, but its display will be turned off). As a result, the BAM registry keys will contain data on any devices, while DAM registry keys will only contain data on mobile devices. > The BAM registry key contains multiple subkeys under bam
State
UserSettings, with one subkey per user, identified with the user SID. While the key is in the SYSTEM registry hive, program executions can thus still be tied to a specific user using this SID. > Each user-specific key contains a list of executed programs, with their full path and timestamp of last execution. > If a file is deleted, the eventual associated entry in the BAM is deleted as well after the system reboot. Additionally, BAM entries older than 7 days are deleted upon system boot. The BAM thus provides limited information on historic execution of programs > No entries are created in the BAM keys for executables on removable media and/or on network shares. > Key:

Quantitativ: (Diagramme) - Stacked Balkendiagramm jeweils für Logfile 1 und Logfile2: Anteil Kategorie 1 bzw.2 an allen Registry-Schreiboperationen

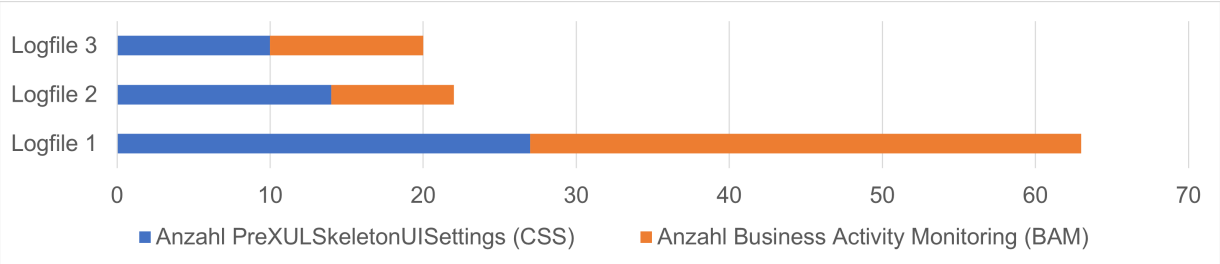


Abbildung 5.21: Comparison of found PB artifacts between RAM Dumps

> Stringsuche in Registry Hives mit Registry Explorer (Siehe Liste) In allen Hives kein Treffer für alle Suchbegriffe

Literatur: > Wie bei Firefox: Shellactivities Key existiert nicht mehr –> Nicht mehr vorhanden in aktueller Version (Verweis auf E-Mail)

Black-Box Analyse/Uncommon Locations

Analyse mit Autopsy

Bei White-Box Analyse/Common Locations: Autopsy nur zur Dateiextraktion genutzt, hier: als konkretes forensisches Werkzeug

Stichwortsuche: - In allen Snapshots keine Treffer (auch innerhalb $Carved) - TODO: Pagefile gefunden?

Von Autopsy automatisch indexierte Dateien: In allen Fällen: keine Dateien gelöscht, nur über Zeitraum der Snapshots neue dazugekommen - Web Bookmarks: Snapshot 1: > Bing.url (Unter

Source Name	S	C	O	URL	Title	Date Created	Program Name	Domain
Bing.url			19	http://go.microsoft.com/fwlink/p/?LinkId=255142	Bing.url	2023-04-25 16:09:28 MESZ	Internet Explorer Analyzer	microsoft.com

Abbildung 5.22: Autopsy Web Bookmarks

C:/User/Forensik/Favorites/Links) enthält Bing Startseite Snapshot 2: > unverändert zu 1 Snapshot 3-1: > unverändert zu 2 Snapshot 3-2: > unverändert zu 3-1 Snapshot 4: > unverändert zu 3-2 - Web Cookies: Snapshot 1: > 9 Einträge in WebCacheV01.dat (= DB des Internet Explorers zum speichern

Source Name	S	C	O	URL	Date Accessed	Name	Value	Program Name	Domain
WebCacheV01.dat			15	bing.com	2023-05-07 13:56:11 MESZ	SUID	A	Microsoft Edge Analyzer	bing.com
WebCacheV01.dat			15	bing.com	2023-05-07 13:47:01 MESZ	ANON	A=D8530A977A1F5DAD20B78D8CFFFFFFFF	Microsoft Edge Analyzer	bing.com
WebCacheV01.dat			15	login.live.com	2023-05-07 12:26:40 MESZ	__Host-MSAAUTHP		Microsoft Edge Analyzer	live.com
WebCacheV01.dat			15	live.com	2023-05-07 12:25:56 MESZ	MUID	0EDC58E500A76FCE1B0F4BEF04A76BD6	Microsoft Edge Analyzer	live.com
WebCacheV01.dat			15	www.bing.com	2023-05-07 12:25:44 MESZ	MUIDB	31708C5FC3CF47068AFADC1CB47D0111	Microsoft Edge Analyzer	bing.com
WebCacheV01.dat			15	bing.com	2023-05-07 12:25:44 MESZ	SRCHD	AF=NOFORM	Microsoft Edge Analyzer	bing.com
WebCacheV01.dat			15	bing.com	2023-05-07 12:25:44 MESZ	SRCHUID	V=2&GUID=62F5FD78E9D94446BAFDF9DEC8188103&dm...	Microsoft Edge Analyzer	bing.com
WebCacheV01.dat			15	bing.com	2023-05-07 12:25:44 MESZ	SRCHUSR	DOB=20230507	Microsoft Edge Analyzer	bing.com
WebCacheV01.dat			15	bing.com	2023-05-07 12:25:44 MESZ	SRCHHPGUSR	SRCHLANG=de	Microsoft Edge Analyzer	bing.com

Abbildung 5.23: Autopsy Web Cookies

von Browserdaten): Cookies für bing.com und live.com (= outlook) Snapshot 2: > unverändert zu 1 Snapshot 3-1: > unverändert zu 2 Snapshot 3-2: > unverändert zu 3-1 Snapshot 4: > unverändert zu 3-2 - Web History: Snapshot 1: > 2 Einträge in WebCacheV01.dat: - 2x live.com (= outlook) Snapshot

Source Name	S	C	O	URL	Date Accessed	Program Name	Domain	Username
WebCacheV01.dat				file:///Z:/Logfile2-2	2023-05-07 14:28:06 MESZ	Microsoft Edge Analyzer		Forensik
WebCacheV01.dat				file:///Z:/Logfile2-1	2023-05-07 14:17:05 MESZ	Microsoft Edge Analyzer		Forensik
WebCacheV01.dat				file:///Z:/Logfile1	2023-05-07 13:55:54 MESZ	Microsoft Edge Analyzer		Forensik
WebCacheV01.dat			15	https://login.live.com/oauth20_authorize.srf?client_id=000...	2023-05-07 12:26:43 MESZ	Microsoft Edge Analyzer	live.com	Forensik
WebCacheV01.dat			15	https://login.live.com/oauth20_desktop.srf?lc=1031	2023-05-07 12:26:40 MESZ	Microsoft Edge Analyzer	live.com	Forensik

Abbildung 5.24: Autopsy Web History

2: > 1 neuer Einträge in WebCacheV01.dat: - file:///Z:/Logfile_1 (= Process Monitor Logfile, die in

shared-Folder geladen wurde) -> Erklärung? Snapshot 3-1: > 1 neuer Eintrag in WebCacheV01.dat: - file:///Z:/Logfile_2-1 (= Process Monitor Logfile, die in shared-Folder geladen wurde) -> Erklärung? Snapshot 3-2: > 1 neuer Eintrag in WebCacheV01.dat: - file:///Z:/Logfile_2-2 (= Process Monitor Logfile, die in shared-Folder geladen wurde) -> Erklärung? Snapshot 4: > unverändert zu 3-2 - Web Categories: Snapshot 1: > 2x WebCacheV01.dat aufgelistet => Mit HxD untersucht, keine

Source Name	S	C	O	Source Type	Score	Conclusion	Configuration	Justification	Domain	Host	Name
WebCacheV01.dat			0	File	Unknown				bing.com	bing.com	Search Engine
WebCacheV01.dat			0	File	Unknown				live.com	login.live.com	Web Email

Abbildung 5.25: Autopsy Web Categories

PB Artefakte Snapshot 2: > unverändert zu 2 Snapshot 3-1: > unverändert zu 3 Snapshot 3-2: > unverändert zu 3-1 Snapshot 4: > unverändert zu 3-2

Zusammenfassung: - keine PB Artefakte - Keine neuen Erkenntnisse vgl. mit intensiver Analyse mittels Process Monitor in Kapitel X - .onion URL Einträge in places.sql nicht erkannt

Analyse mit Volatility

Vorgehen: Siehe "Methodik"Kapitel - Ausgangslage: Volatility Yarascan Treffer - Für jeden Treffer: virtueller Offset des Strings, PID, getriggerte Yararule, getriggerte Yara Component z(= Variablenname des gesuchten Strings), gefundener String - Neue Spalte: "Prozessname> zu jeder PID Prozessnamen - Ergebnisse Aufbereitet nach folgendem Schema: > Für jeden RAM Dump > Für jede Yararule > Für jede Component > Filter: Prozessname = Firefox -> Anzahl zählen > Filter: Prozessname = Alle Prozesse außer Firefox -> Anzahl zählen

Wie bei Firefox: HTML Artefakte wurden in keinem RAM Dump gefunden => Nicht aufgeführt

Yararule "Keyword": Analyse: > Ausschließlich in RAM Dump 2 und RAM Dump 3-1 Keyword Artefakte gefunden > In RAM Dump 3-1 bei jedem Keyword deutlich weniger Artefakte als in RAM Dump 2 => Identitäts-Reset reduziert Keyword Artefakte deutlich > Hauptsächlich in Firefox Prozess, kein Artefakt in Tor.exe Prozess > Mit 4833 Artefakten in RAM Dump 2 am häufigsten "pfaffenhofen"vertreten. Vermutung: Evtl. weil Google Maps viele zusätzliche Artefakte lädt. > Nach Schließen von Tor Browser: keine Keyword Artefakte mehr in RAM

Yararule ÜRL": Analyse: > Wie bei Yararule "Keyword": Ausschließlich in RAM Dump 2 und RAM Dump 3-1 Keyword Artefakte gefunden > In RAM Dump 3-1 bei jedem Keyword deutlich weniger Artefakte als in RAM Dump 2 => Identitäts-Reset reduziert URL Artefakte deutlich > Hauptsächlich in Firefox Prozess, danach am häufigsten Tor.exe Prozess und am wenigsten Artefakte in anderen Prozessen > Bemerkenswert: "mallofamerica.comïst mit 26.505 mal in RAM Dump 2 am häufigsten als Artefakt gefunden worden. Vergleich: "mooserliesl.de"wurde nur 508 mal in RAM Dump 2 gefunden > Nach Schließen von Tor Browser: keine URL Artefakte mehr in RAM

> TODO: DNSCache?

Yararule "Mail": Analyse: > Alle Mail Artefakte gefunden > Artefakte ausschließlich in Firefox Prozess gefunden > Artefakte fast ausschließlich in RAM Dump 2 Mail gefunden > Nur die Absenderadresse

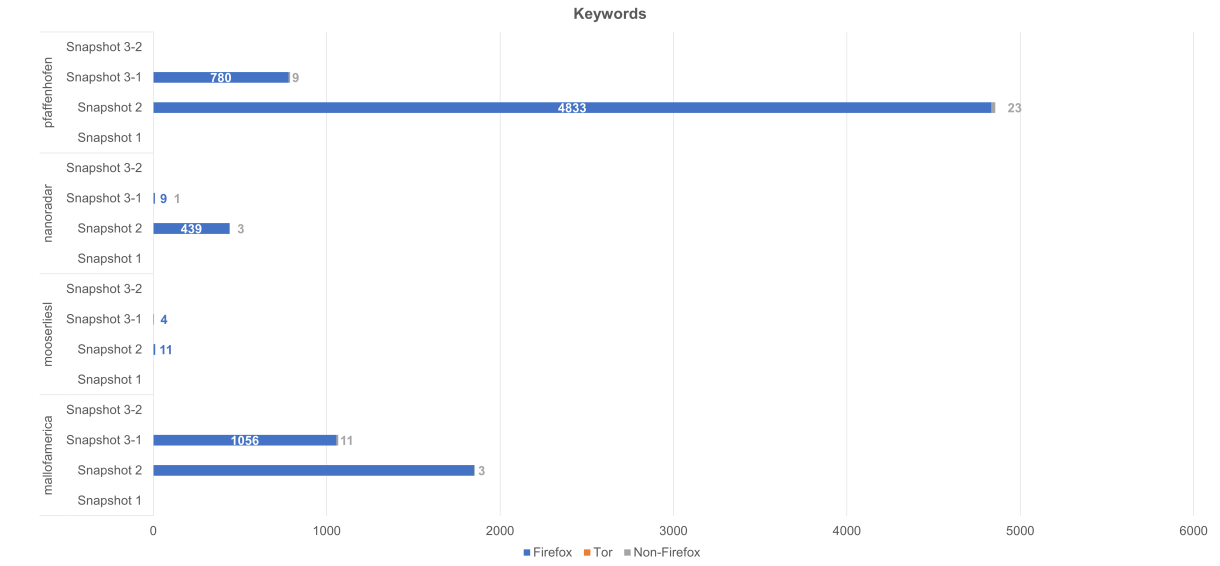


Abbildung 5.26: Keywords

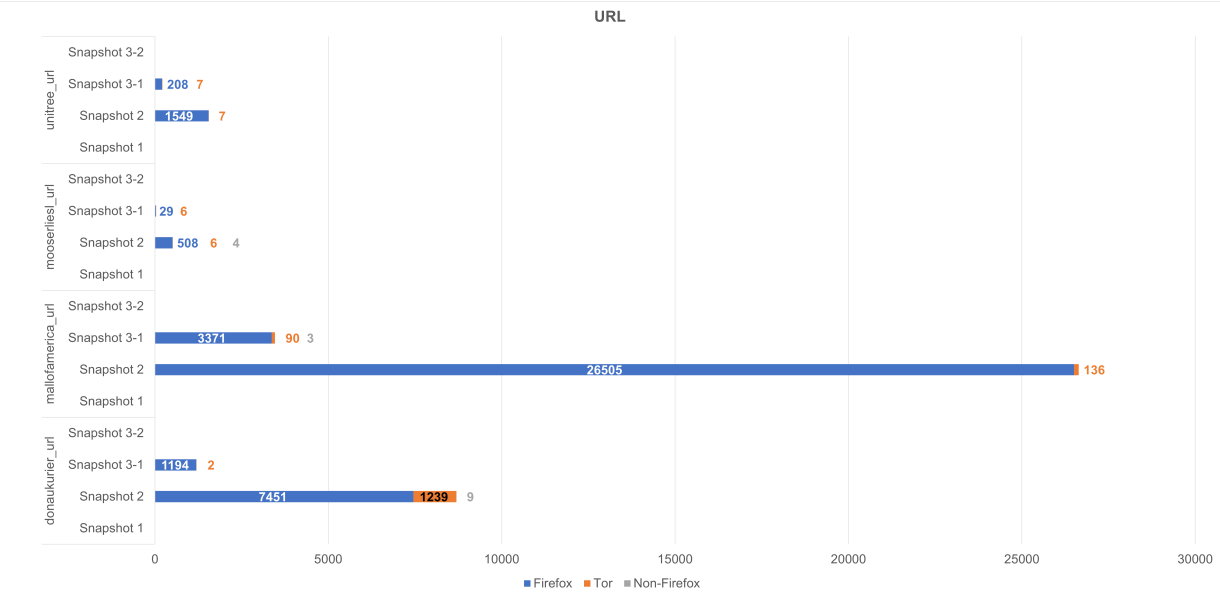


Abbildung 5.27: URL

"computerforensikvl@gmail.com"wurde nach Identitäts-Reset in RAM Dump 3-1 gefunden > Absenderadresse ist häufigstes Mail Artefakt > Bemerkenswert: Passwort wurde 2x als Klartext im RAM gefunden! String Kontext: Offsets: PIDs: 0xb9ce29180c8 7420 0x2859f4ffd4e0 7420 0x24083b41858 8424 0x240840e5b08 8424

Yararule Ïmage": Analyse: > Hex-Wert von Donaukurier Bild wurde ein einzigees mal im 2. RAM Dump in einem Firefox Prozess gefunden

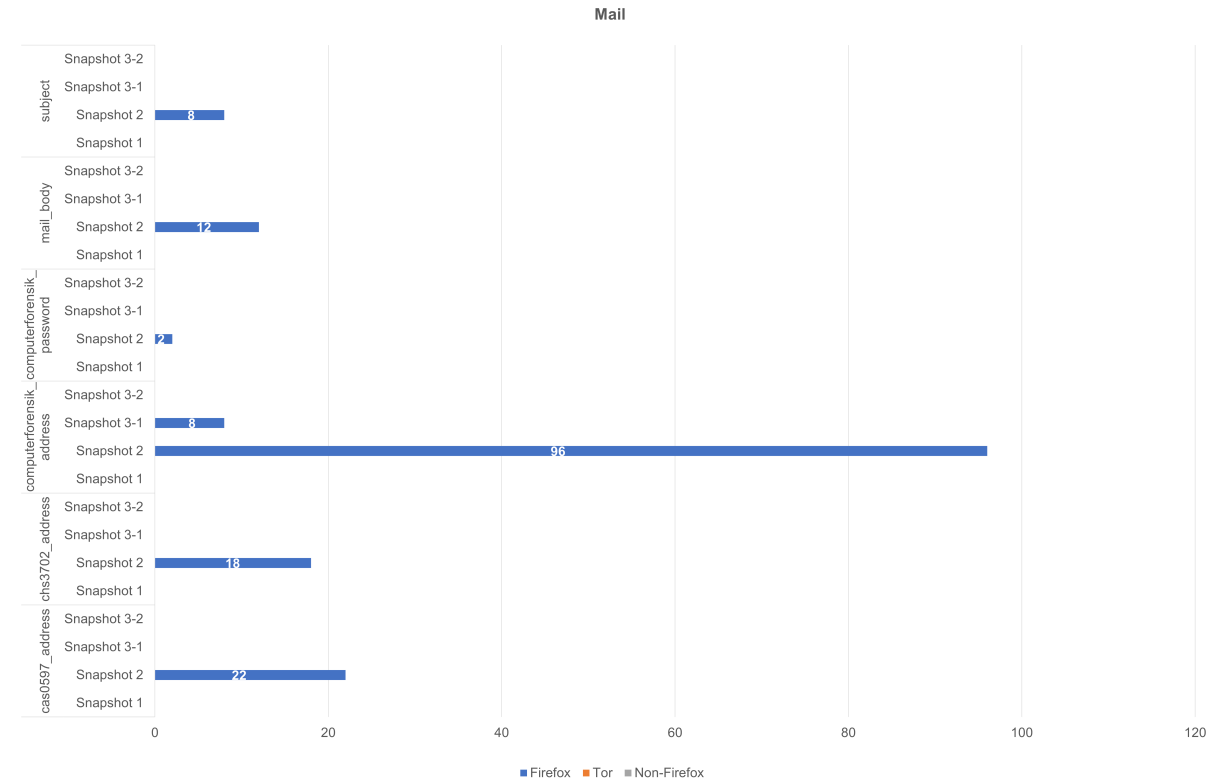


Abbildung 5.28: Mail

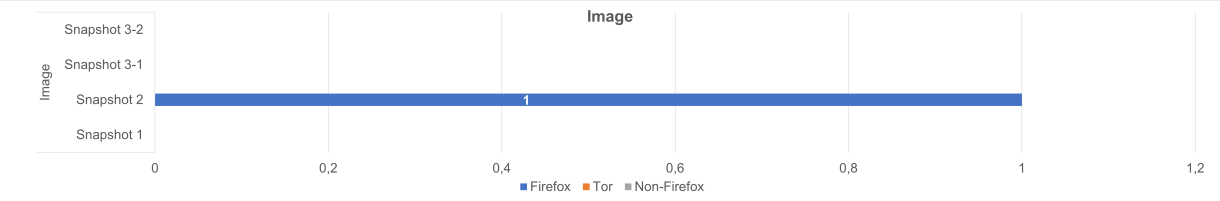


Abbildung 5.29: Image

Zusammenfassung = Stacked Bar Chart: - PB Artefakte ausschließlich in RAM Dump 2 und 3-1 gefunden - Nach Identitäts-Reset deutlich weniger Artefakte in vorhanden - Am meisten URL-Artefakte gefunden, wobei mallofamerica.com dominant - HTML Artefakte wurden in keinem RAM Dump gefunden

TODO: Kreisdiagramme/Balkendiagramme mit Gesamtzahl an (Non-)Firefox Yarascan-Treffer erst im Vergleich mit Tor

Uncommon Locations

Literatur:

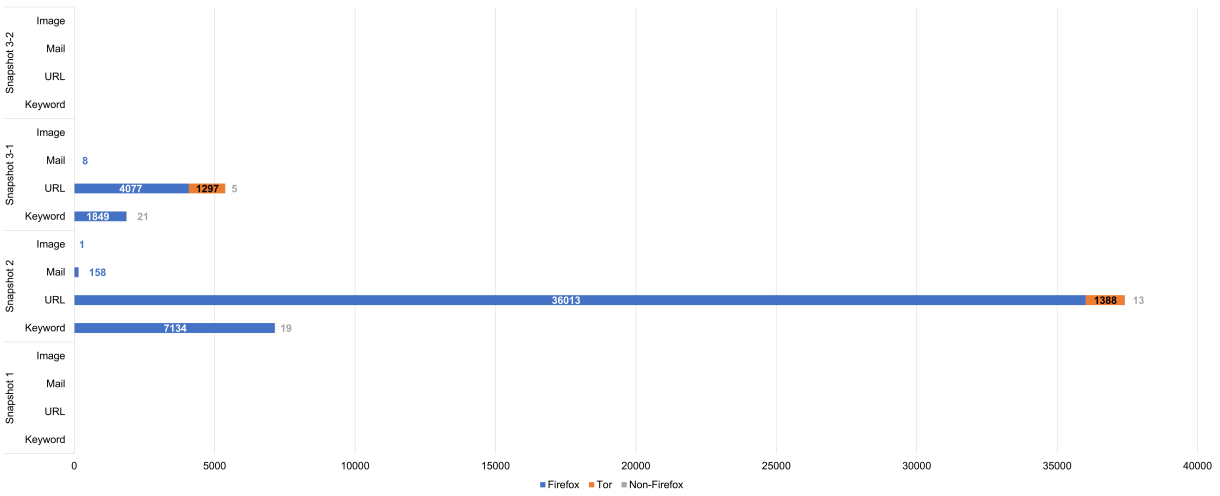


Abbildung 5.30: Summary

5.3 Chrome

Uncommon Locations

o Autopsy Keyword-Suche: > Chrome and Edge produced five artefacts as reported by both tools. (FTK, Autopsy) [6] –> Artefakte werden nicht genannt! > only two temporary files (Figure 7) were recovered with Minitool Power Data Recovery but it was a dead end; Location: appdata/. . . /Chrome/. . . / Preferences/RF1533fa.TMP [5] > pagefile.sys file showed no traces at all [22]

5.4 Brave

6 Vergleich der Browser

- Zusammenfassung: Vergleich Tor v. Firefox und Brave v. Chrome

Firefox vs Tor: > Gestacktes Balkendiagramm zu veränderten SQLite DBs => Erst bei Vergleich mit Tor!

- Firefox v. Chrome (SStandardbrowser") - Tor v. Brave (SSichere Browser") - Zum Schluss: Ëine große Tabelle"mit den wichtigsten Kategorien?

7 Diskussion

> Artefakte im DNS Cache: [24] ▪ DNS-Caching ist eine Bedrohung für private Browsing ▪ Diese Schwachstelle entsteht, weil das Betriebssystem DNS-Anfragen des Browsers im Cache speichert, unabhängig davon, ob der Browser im privaten Modus ist oder nicht ▪ Mehrere Jahre nach der Meldung dieser Schwachstelle besteht sie immer noch in allen Browsern fort ▪ Es wurden einige Erweiterungen von Drittanbietern entwickelt, um dieses Problem zu beheben, aber keine davon wurde von den Browserherstellern übernommen.

> Viele RAM-Artefakte - Firefox [16] ▪ Darcie et al. (2014) fanden Beweise für das Web-Browsing in Form von JPEG- und HTML-Dateien in Live-Forensik, aber eine statische Forensik war erfolglos. ▪ Eine vorherige Live-Forensik-Analyse des Firefox-Browsers zeigte, dass Artefakte aus einer privaten Browsing-Sitzung aus dem Speicher wiederhergestellt werden konnten. (Findlay and Leimich, 2014).

> IE hinterlässt viele Spuren im Gegensatz zu Ergebnissen: [14] o hidden folders are usually stored at C/Users/User/AppData o evidence searches are conducted extensively in the C: partition o bookmarks remain and can be viewed o downloads remain in the downloads folder until the user manually deletes them o CacheView trace entire URL and browsing histories including the temporary files CacheView enables to find the image's URL and from specific website

> Urteil über die Privatheit von Tor nach [16] The design aim of preventing Tor from writing to disk (Perry et al., 2018) is not achieved in this version. ▪ Configuration files, downloaded files, and browserrelated data are recoverable from the file system. ▪ Significant data-leakage from the browsing session occurred: HTTP header information, titles of web pages and an instance of a URL were found in registry files, system files, and unallocated space. ▪ The data-leakage contained the German word for 'search' in reference to a Google search. This hints at the locale of the Tor server used to exit the network (exit relay). The Tor Project's design aim of enabling secure deletion of the browser (Sandvik, 2013) is not achieved in this version. ▪ References to: the installation directory, Firefox SQLite files, bridging IPs/ports, default bookmarks, Tor-related DLLs and Tor product information were all recovered after the browser was deleted. ▪ In a scenario where the operating system paged memory, an instance

Weiterführende Arbeiten: > Cross-mode interference [10]: o the Chrome://memory page displays all the opened tabs in the browser regardless if they are in the usual or private mode -> Nicht mehr aktuell -> Stattdessen: Chrome Task-manager (Ctrl + Esc), Funktioniert auch bei Firefox > Unser Scope: Process Monitor nach Prozessnamen gefiltert - Weiterführend: Nach Pathnamen filtern: "Common Locations"

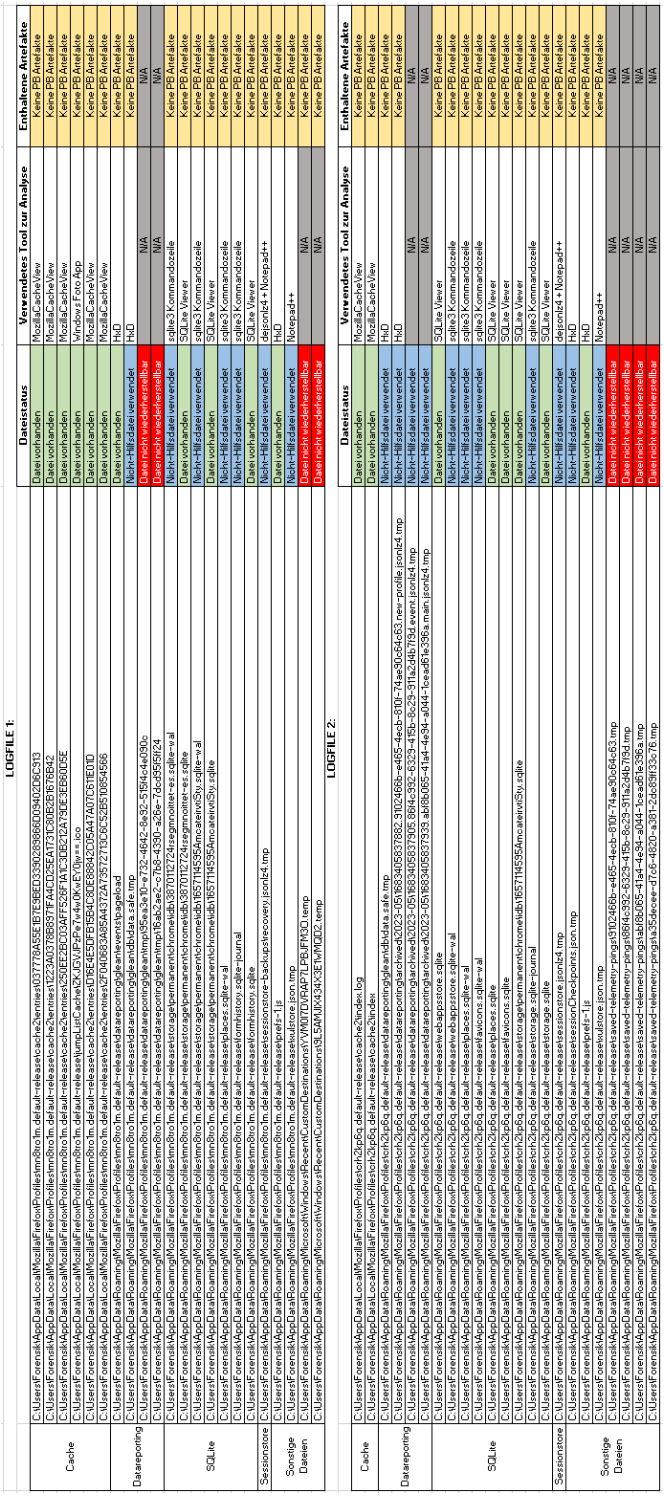
> Für wen wird Browser entwickelt > Warum und für wen wird Private Browsing analysiert? > Ist das Auffinden privater Browsing Artefakte Schuld von Browser Entwicklern? (Oder Schuld des Betriebssystem, wie in (TODO!) erwähnt)

8 Fazit

Einleitend werden Struktur, Motivation und die abgeleiteten Forschungsfragen diskutiert.

Appendices

All File Operations Firefox

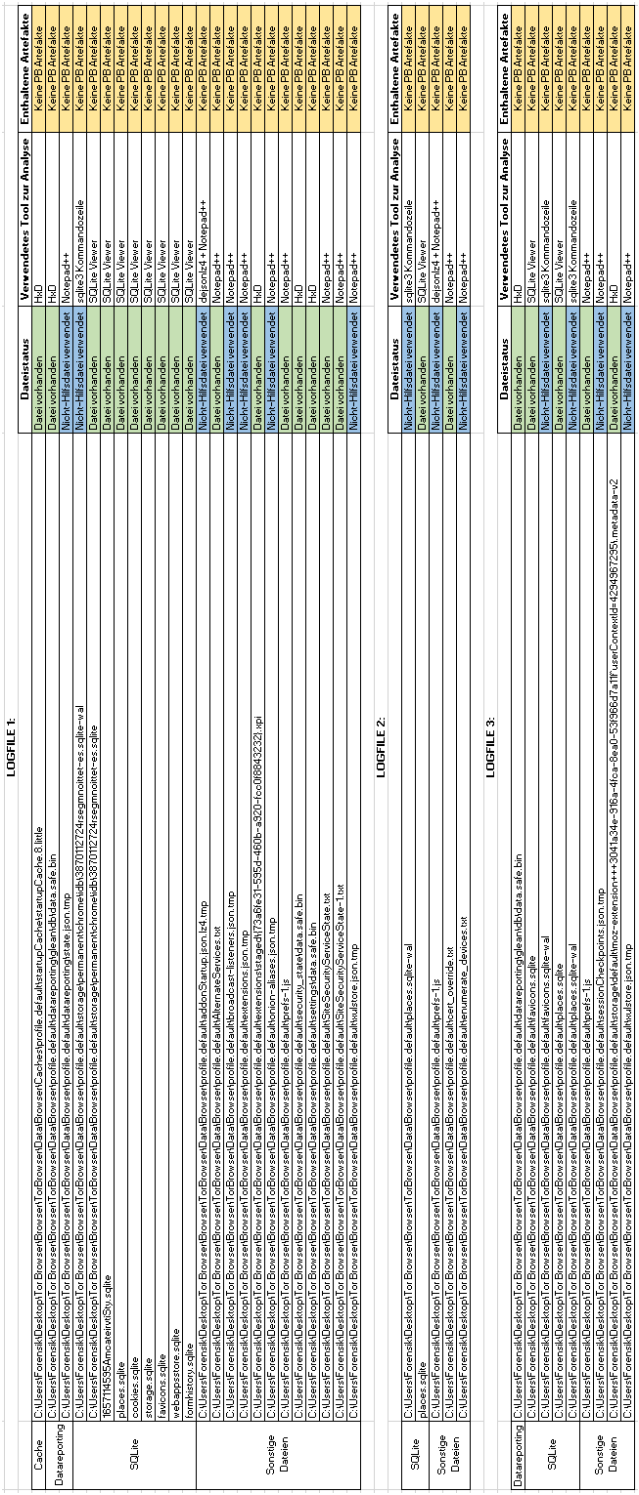


LOGFILE 1:				
		Dateistatus	Verwendetes Tool zur Analyse	Enthaltene Artefakte
Cache	C:\Users\Forensik\AppData\Local\Mozilla\Firefox\Profiles\mn8no1n.default-release\cache2\entries\037778A55E1B7E98ED33902898660D9402D6C313	Datei vorhanden	MozillaCacheView	Keine PB Artefakte
	C:\Users\Forensik\AppData\Local\Mozilla\Firefox\Profiles\mn8no1n.default-release\cache2\entries\1223A037B6B97FFA4CDC5EA1731C80B2BF1678642	Datei vorhanden	MozillaCacheView	Keine PB Artefakte
	C:\Users\Forensik\AppData\Local\Mozilla\Firefox\Profiles\mn8no1n.default-release\cache2\entries\250EE2BC03AFF5226F1A7E3DB215A79DE3EB60D5E	Datei vorhanden	MozillaCacheView	Keine PB Artefakte
	C:\Users\Forensik\AppData\Local\Mozilla\Firefox\Profiles\mn8no1n.default-release\jumpListCache\ZkJGVJP+Pw7v4w0KvREYDw==.ico	Datei vorhanden	Windows Foto App	Keine PB Artefakte
	C:\Users\Forensik\AppData\Local\Mozilla\Firefox\Profiles\mn8no1n.default-release\cache2\entries\7D6E4E5DFB15B4C90E88842C05A47A07C61E01D	Datei vorhanden	MozillaCacheView	Keine PB Artefakte
	C:\Users\Forensik\AppData\Local\Mozilla\Firefox\Profiles\mn8no1n.default-release\cache2\entries\7F04D6B3A85A4372A75727139C6C52B5100B54566	Datei vorhanden	MozillaCacheView	Keine PB Artefakte
Datareporting	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\mn8no1n.default-release\datareporting\glean\pageload	Datei vorhanden	HxD	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\mn8no1n.default-release\datareporting\glean\pending_pings\e732-4642-8c92-51514c4e090c	Nicht-Hilfsdatei verwendet	HxD	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\mn8no1n.default-release\datareporting\glean\tmp\5abc2ae2-c7b8-4330-a26e-7dcd95f5bf24	Datei nicht wiederherstellbar	N/A	N/A
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\mn8no1n.default-release\datareporting\glean\tmp\b5ea3e10-e732-4642-8c92-51514c4e090c	Datei nicht wiederherstellbar	N/A	N/A
SQLite	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\mn8no1n.default-release\storage\permanent\chrome\idb\3870112724rsgmnoitet-es.sqlite	Nicht-Hilfsdatei verwendet	sqlite3 Kommandozeile	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\mn8no1n.default-release\storage\permanent\chrome\idb\3870112724rsgmnoitet-es.sqlite-wal	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\mn8no1n.default-release\storage\permanent\chrome\idb\1657114595Amcateirvtiy.sqlite	Nicht-Hilfsdatei verwendet	sqlite3 Kommandozeile	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\mn8no1n.default-release\storage\permanent\chrome\idb\1657114595Amcateirvtiy.sqlite-wal	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\mn8no1n.default-release\storage\permanent\chrome\idb\1657114595Amcateirvtiy.sqlite-journal	Nicht-Hilfsdatei verwendet	sqlite3 Kommandozeile	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\mn8no1n.default-release\places.sqlite-wal	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\mn8no1n.default-release\storage\default\moz-extension+++ca2ea5ec-5c0b-4e3b-8f8c-d5ac1ed8fa0c\idb\3647222921wleabcEoxlt-eengsairo.sqlite	Nicht-Hilfsdatei verwendet	sqlite3 Kommandozeile	Keine PB Artefakte
Sessionstore	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\mn8no1n.default-release\sessionstore-backups\recovery.jsonlz4.tmp	Nicht-Hilfsdatei verwendet	dejsonlz4 + Notepad++	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\mn8no1n.default-release\prefs-1.js	Datei vorhanden	HxD	Keine PB Artefakte
Sonstige Dateien	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\mn8no1n.default-release\sulnstore.json.tmp	Nicht-Hilfsdatei verwendet	Notepad++	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Microsoft\Windows\Recent\CustomDestinations\VM07DVRAP7LPEJFH3O.temp	Datei nicht wiederherstellbar	N/A	N/A
	C:\Users\Forensik\AppData\Roaming\Microsoft\Windows\Recent\CustomDestinations\9L5AMJX434X3ETWMQD2.temp	Datei nicht wiederherstellbar	N/A	N/A

LOGFILE 2:				
		Dateistatus	Verwendetes Tool zur Analyse	Enthaltene Artefakte
Cache	C:\Users\Forensik\AppData\Local\Mozilla\Firefox\Profiles\cln2kp6q.default-release\cache2\index.log	Datei vorhanden	MozillaCacheView	Keine PB Artefakte
	C:\Users\Forensik\AppData\Local\Mozilla\Firefox\Profiles\cln2kp6q.default-release\cache2\index	Datei vorhanden	MozillaCacheView	Keine PB Artefakte
Datareporting	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\datareporting\glean\pending_pings\1902466b-e465-4ecb-810f-74ae90c64c63.tmp	Nicht-Hilfsdatei verwendet	HxD	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\datareporting\archived\2023-05\1683405837882.1902466b-e465-4ecb-810f-74ae90c64c63.new-profile.jsonlz4.tmp	Nicht-Hilfsdatei verwendet	HxD	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\datareporting\archived\2023-05\1683405837905.8684c392-6323-415b-8c29-311a2d4b7f9d.event.jsonlz4.tmp	Nicht-Hilfsdatei verwendet	HxD	N/A
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\datareporting\archived\2023-05\1683405837933.abf80665-41a4-4a94-a044-1cea861e336a.main.jsonlz4.tmp	Nicht-Hilfsdatei verwendet	HxD	N/A
SQLite	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\webappsstore.sqlite	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\webappsstore.sqlite-wal	Nicht-Hilfsdatei verwendet	sqlite3 Kommandozeile	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\places.sqlite-wal	Nicht-Hilfsdatei verwendet	sqlite3 Kommandozeile	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\favicons.sqlite-wal	Nicht-Hilfsdatei verwendet	sqlite3 Kommandozeile	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\places.sqlite	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\favicons.sqlite	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\storage\permanent\chrome\idb\1657114595Amcateirvtiy.sqlite	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\storage\permanent\chrome\idb\1657114595Amcateirvtiy.sqlite-journal	Nicht-Hilfsdatei verwendet	sqlite3 Kommandozeile	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\storage.sqlite	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
Sessionstore	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\sessionstore-backups\recovery.jsonlz4.tmp	Nicht-Hilfsdatei verwendet	dejsonlz4 + Notepad++	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\sessionCheckpoints.json.tmp	Nicht-Hilfsdatei verwendet	HxD	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\prefs-1.js	Datei vorhanden	HxD	Keine PB Artefakte
Sonstige Dateien	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\sulnstore.json.tmp	Nicht-Hilfsdatei verwendet	Notepad++	Keine PB Artefakte
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\saved-telemetry-pings\3f1902466b-e465-4ecb-810f-74ae90c64c63.tmp	Datei nicht wiederherstellbar	N/A	N/A
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\saved-telemetry-pings\89f04c392-6323-415b-8c29-311a2d4b7f9d.tmp	Datei nicht wiederherstellbar	N/A	N/A
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\saved-telemetry-pings\ab86065-41a4-4a94-a044-1cea861e336a.tmp	Datei nicht wiederherstellbar	N/A	N/A
	C:\Users\Forensik\AppData\Roaming\Mozilla\Firefox\Profiles\cln2kp6q.default-release\saved-telemetry-pings\a35edeee-d7c6-4820-a381-2dc89f133c76.tmp	Datei nicht wiederherstellbar	N/A	N/A

Abbildung .1: All File Operations Firefox: Logfile 1 vs. Logfile 2

All File Operations Tor



LOGFILE 1:

		Dateistatus	Verwendetes Tool zur Analyse	Enthaltene Artefakte
Cache	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\Cache\startupCache.bin	Datei vorhanden	HxD	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\startupCache\startupCache.8.little	Datei vorhanden	HxD	Keine PB Artefakte
Datareporting	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\datareporting\lastversion.bin	Datei vorhanden	HxD	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\datareporting\state.json.tmp	Nicht-Hilfsdatei verwendet	Notepad++	Keine PB Artefakte
SQLite	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\storage\permanent\chrome\idb\3870112724rsegmnoittet-es.sqlite-wal	Nicht-Hilfsdatei verwendet	sqlite3 Kommandozeile	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\storage\permanent\chrome\idb\3870112724rsegmnoittet-es.sqlite	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
	1657114595AmcateirvtiSu.sqlite	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
	places.sqlite	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
	cookies.sqlite	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
	storage.sqlite	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
	favicons.sqlite	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
	webappsstore.sqlite	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
	formhistory.sqlite	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
Sonstige Dateien	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\addonStartup.json.lz4.tmp	Nicht-Hilfsdatei verwendet	dejsonlz4 + Notepad++	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\AlternateServices.txt	Datei vorhanden	Notepad++	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\broadcast-listeners.json.tmp	Nicht-Hilfsdatei verwendet	Notepad++	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\extensions.json.tmp	Nicht-Hilfsdatei verwendet	Notepad++	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\extensions\staged\173a9e31-595d-460b-a920-fcc0f8843232}.xpi	Datei vorhanden	HxD	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\extensions\staged\173a9e31-595d-460b-a920-fcc0f8843232}.xpi	Nicht-Hilfsdatei verwendet	Notepad++	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\prefs-1.js	Datei vorhanden	Notepad++	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\security_state\data.safe.bin	Datei vorhanden	HxD	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\settings\data.safe.bin	Datei vorhanden	HxD	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\SiteSecurityServiceState.txt	Datei vorhanden	Notepad++	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\SiteSecurityServiceState-1.txt	Datei vorhanden	Notepad++	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\xulstore.json.tmp	Nicht-Hilfsdatei verwendet	Notepad++	Keine PB Artefakte

LOGFILE 2:

		Dateistatus	Verwendetes Tool zur Analyse	Enthaltene Artefakte
SQLite	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\places.sqlite-wal	Nicht-Hilfsdatei verwendet	sqlite3 Kommandozeile	Keine PB Artefakte
	places.sqlite	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
Sonstige Dateien	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\prefs-1.js	Nicht-Hilfsdatei verwendet	dejsonlz4 + Notepad++	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\cert_override.txt	Datei vorhanden	Notepad++	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\enumerate_devices.txt	Nicht-Hilfsdatei verwendet	Notepad++	Keine PB Artefakte

LOGFILE 3:

		Dateistatus	Verwendetes Tool zur Analyse	Enthaltene Artefakte
Datareporting	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\datareporting\glean\db\data.safe.bin	Datei vorhanden	HxD	Keine PB Artefakte
SQLite	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\avicons.sqlite	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\avicons.sqlite-wal	Nicht-Hilfsdatei verwendet	sqlite3 Kommandozeile	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\places.sqlite	Datei vorhanden	SQLite Viewer	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\places.sqlite-wal	Nicht-Hilfsdatei verwendet	sqlite3 Kommandozeile	Keine PB Artefakte
Sonstige Dateien	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\prefs-1.js	Datei vorhanden	Notepad++	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\sessionCheckpoints.json.tmp	Nicht-Hilfsdatei verwendet	Notepad++	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\storage\default\moz-extension+++3041a34e-3f6a-4fca-8ea0-53966d7a1ff\userContextId=4294967295.metadata-v2	Datei vorhanden	HxD	Keine PB Artefakte
	C:\Users\Forensik\Desktop\Tor Browser\Browser\TorBrowser\Data\Browser\profile.default\xulstore.json.tmp	Nicht-Hilfsdatei verwendet	Notepad++	Keine PB Artefakte

Abbildung .2: All File Operations Firefox: Logfile 1 vs. Logfile 2 vs. Logfile 3

Literatur

[1] Gaurav Aggarwal u. a. "An Analysis of Private Browsing Modes in Modern Browsers."
In: *USENIX security symposium*. 2010, S. 79–94.

[2] Gabriele Bonetti u. a. "Black-box forensic and antiforensic characteristics of solid-state drives".
In: *Journal of Computer Virology and Hacking Techniques* 10 (2014), S. 255–271.

[3] Howard Chivers. "Private browsing: A window of forensic opportunity".
In: *Digital Investigation* 11.1 (2014), S. 20–29.

[4] Divya Dayalamurthy. "Forensic memory dump analysis and recovery of the artefacts of using tor bundle browser–the need". In: (2013).

[5] Hasan Fayyad-Kazan u. a.
"Forensic analysis of private browsing mechanisms: Tracing internet activities". In: (2021).

[6] Ryan M Gabet, Kathryn C Seigfried-Spellar und Marcus K Rogers.
"A comparative forensic analysis of privacy enhanced web browsers and private browsing modes of common web browsers".
In: *International Journal of Electronic Security and Digital Forensics* 10.4 (2018), S. 356–371.

[7] Ms Pooja Gupta. "Capturing Ephemeral Evidence Using Live Forensics".
In: *IOSR J. Electron. Commun. Eng* (2013), S. 109–113.

[8] Meenu Hariharan, Akash Thakar und Parvesh Sharma. "Forensic Analysis of Private Mode Browsing Artifacts in Portable Web Browsers Using Memory Forensics". In: *2022 International Conference on Computing, Communication, Security and Intelligent Systems (IC3SIS)*.
IEEE. 2022, S. 1–5.

[9] Nihad A Hassan. *Digital forensics basics: A practical guide using Windows OS*. Apress, 2019.

[10] Ashley Hedberg. *The privacy of private browsing*. Techn. Ber.
Technical Report, Tufts University, MA, USA, 2013.

[11] Graeme Horsman u. a. "A forensic examination of web browser privacy-modes".
In: *Forensic Science International: Reports* 1 (2019), S. 100036.

[12] Aina Izzati und Nurul Hidayah Ab Rahman. "A Comparative Analysis of Residual Data Between Private Browsing and Normal Browsing Using Live Memory Acquisition".
In: *Applied Information Technology And Computer Science* 3.2 (2022), S. 68–83.

[13] Ahmed Redha Mahlous und Houssam Mahlous. "Private Browsing Forensic Analysis: A Case Study of Privacy Preservation in the Brave Browser".
In: *International Journal of Intelligent Engineering Systems* 13.06 (2020), S. 294–306.

[14] Raihana Md Saidi u. a. "Analysis of Private Browsing Activities". In: *Regional Conference on Science, Technology and Social Sciences (RCSTSS 2016) Theoretical and Applied Sciences*.
Springer. 2018, S. 217–228.

[15] Reza Montasari und Pekka Peltola. "Computer forensic analysis of private browsing modes". In: *Global Security, Safety and Sustainability: Tomorrow's Challenges of Cyber Security: 10th International Conference, ICGS3 2015, London, UK, September 15-17, 2015. Proceedings 10*. Springer. 2015, S. 96–109.

[16] Matt Muir, Petra Leimich und William J Buchanan. "A forensic audit of the tor browser bundle". In: *Digital Investigation* 29 (2019), S. 118–128.

[17] Rebecca Nelson, Atul Shukla und Cory Smith. "Web browser forensics in google chrome, mozilla firefox, and the tor browser bundle". In: *Digital Forensic Education: An Experiential Learning Approach* (2020), S. 219–241.

[18] Donny Jacob Ohana und Narasimha Shashidhar. "Do private and portable web browsers leave incriminating evidence? a forensic analysis of residual artifacts from private and portable web browsing sessions". In: *2013 IEEE Security and Privacy Workshops*. IEEE. 2013, S. 135–142.

[19] Daniel Perdices u. a. "Web browsing privacy in the deep learning era: Beyond VPNs and encryption". In: *Computer Networks* 220 (2023), S. 109471.

[20] Digvijaysinh Rathod. "Darknet forensics". In: *future* 11 (2017), S. 12.

[21] Tri Rochmadi, Imam Riadi und Yudi Prayudi. "Live forensics for anti-forensics analysis on private portable web browser". In: *Int. J. Comput. Appl* 164.8 (2017), S. 31–37.

[22] Huwida Said u. a. "Forensic analysis of private browsing artifacts". In: *2011 International Conference on Innovations in Information Technology*. IEEE. 2011, S. 197–202.

[23] Priya P Sajan u. a. "Tor Browser Forensics". In: *Turkish Journal of Computer and Mathematics Education (TURCOMAT)* 12.11 (2021), S. 5599–5608.

[24] Kiavash Satvat u. a. "On the privacy of private browsing–a forensic approach". In: *Data Privacy Management and Autonomous Spontaneous Security: 8th International Workshop, DPM 2013, and 6th International Workshop, SETOP 2013, Egham, UK, September 12-13, 2013, Revised Selected Papers*. Springer. 2014, S. 380–389.

[25] Yunus Yusoff, Roslan Ismail und Zainuddin Hassan. "Common phases of computer forensics investigation models". In: *International Journal of Computer Science & Information Technology* 3.3 (2011), S. 17–31.