

## Technischer Bericht in der Digitalen Forensik

Herausgegeben vom Lehrstuhl für Informatik 1 der Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU) in Kooperation mit dem Masterstudiengang Digitale Forensik (Hochschule Albstadt-Sigmaringen, FAU, Wolfgang-Goethe-Universität Frankfurt am Main)

### Anwendungsanalyse des Passwortmanagers KeePass Version 2.34 (Portable) Microsoft Windows 7 32-Bit

Christian Hainzinger

15.02.2017

Technischer Bericht Nr. 12

#### **Zusammenfassung**

Bei „**KeePass**“ handelt sich um einen beliebten Passwortmanager. Die freie Softwarelösung, welche einen großen Funktionsumfang besitzt und einer ständigen Weiterentwicklung unterliegt, ist aufgrund verschiedener Portierungen auf nahezu allen Betriebssystemen verwendbar. Die mittels „KeePass“ gespeicherten „Zugangsdaten“ für Onlinedienste werden standardmäßig in einer verschlüsselten und mittels Passwort gesicherten Datei gespeichert und vorgehalten. Für eine forensische Untersuchung ist es relevant zu wissen, ob ein Passwort-Manager verwendet und Passwörter gespeichert wurden. Für diesen Bericht wurde die „portable“ Version des „KeePass“-Programms analysiert und im Bericht werden nun die „digitalen“ Spuren aufgezeigt, welche auf eine Verwendung dieses Programms schließen lassen. Die Arbeit entstand im Rahmen des Moduls Browser- und Anwendungsforensik des Studiengangs Digitale Forensik im Wintersemester 2016/2017 unter der Anleitung von Felix Freiling, Holger Morgenstern, Michael Gruhn und Gaston Pugliese.

#### **Hinweis:**

Technische Berichte in Digitaler Forensik werden herausgegeben vom Lehrstuhl für Informatik 1 der Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU) in Kooperation mit dem Masterstudiengang Digitale Forensik (Hochschule Albstadt-Sigmaringen, FAU, Goethe-Universität Frankfurt am Main). Die Reihe bietet ein Forum für die schnelle Publikation von Forschungsergebnissen in Digitaler Forensik in deutscher Sprache. Die in den Dokumenten enthaltenen Erkenntnisse sind nach bestem Wissen entwickelt und dargestellt. Eine Haftung für die Korrektheit und Verwendbarkeit der Resultate kann jedoch weder von den Autoren noch von den Herausgebern übernommen werden. Alle Rechte verbleiben beim Autor. Einen Überblick über die bisher erschienenen Berichte sowie Informationen zur Publikation neuer Berichte finden sich unter <https://www1.cs.fau.de/df-whitepapers>.

# Inhalt

<b>1. Einführung</b> .....	3
<b>1.1 Untersuchungsobjekt</b> .....	3
<b>1.2 Arbeitsumgebung</b> .....	3
<b>2. Technische Anwendungsanalyse</b> .....	4
<b>2.1 KeePass – Ein Überblick</b> .....	4
<b>2.2 Generelle Vorgehensweisen</b> .....	5
<b>2.2.1. Ereignismethode</b> .....	5
<b>2.2.2. Zustandsmethode</b> .....	6
<b>2.2.3. Regshot und Wireshark</b> .....	6
<b>2.2.4. Analyse der Dateien</b> .....	6
<b>2.3 Persistente Spurenmenge</b> .....	7
<b>2.3.1. Dateisystem</b> .....	7
<b>2.3.2. Windows Registry</b> .....	13
<b>2.3.3. Prefetch</b> .....	15
<b>2.3.4. Dateianalyse</b> .....	16
<b>2.3.5. Netzwerkspuren</b> .....	21
<b>3. Zusammenfassung</b> .....	21

# 1. Einführung

Dieser Bericht beschreibt die Analyse der digitalen Spuren des frei verfügbaren Passwortmanagers „KeePass“ in der Version 2.34 (Portable) für das Betriebssystem Windows 7 32-Bit. Es werden vor allem die beiden nachfolgenden Fragen beantwortet:

- Welche Sachverhalte der Anwendung kann man wo im Dateisystem finden?
- Wie kann man die Spuren auslesen?

## 1.1 Untersuchungsobjekt

Als Untersuchungsobjekt der Anwendungsanalyse wurde der Passwortmanager „KeePass“ ausgewählt. Aufgrund der Vielzahl von Onlinediensten (wie z.B. Web-Mail und Onlineshops) bei denen heutzutage ein durchschnittlicher Internetnutzer registriert ist und der grundlegenden Sicherheitsempfehlung verschiedene Passwörter für diese Onlinedienste zu verwenden, macht eine Verwaltung und Speicherung der Passwörter notwendig. Als Softwarelösung für dieses Problem haben sich in den letzten Jahren die sogenannten „Passwort Manager“ etabliert, bei welchen es sich um Computerprogramme handelt, welche die Passwörter in einer verschlüsselten und mittels Passwort geschützten Datenbank abspeichern und weitere Verwaltungsfunktionen (z.B. Generieren eines Passworts) anbieten.

Bei „KeePass“ handelt es sich um freie Softwarelösung, welche einen großen Funktionsumfang besitzt und einer ständigen Weiterentwicklung unterliegt. Aufgrund verschiedener Portierungen (z.B. als AndroidApp) ist KeePass auf nahezu allen Betriebssystemen verwendbar. Die mittels „KeePass“ gespeicherten „Zugangsdaten“ für Onlinedienste werden standardmäßig in einer verschlüsselten und mittels Passwort gesicherten Datei gespeichert und vorgehalten. Für eine forensische Untersuchungen beziehungsweise „polizeilichen“ Ermittlungen ist es relevant zu wissen, ob ein Passwort-Manager verwendet und Passwörter gespeichert wurden. Neben einer Installationsvariante gibt es KeePass auch als „portable“ Version, bei welcher keine Installation notwendig ist und welche z.B. auch ganz einfach von einem USB-Stick gestartet werden kann.

Da bei der „portablen“ Windows-Version von „KeePass“ bereits die „Spuren“ der Installation wegfallen war es aus Sicht des Autors interessant zu wissen, welche anderen digitalen Spuren aufgefunden werden können, welche auf eine Verwendung des Programms „KeePass“ schließen lassen.

## 1.2 Arbeitsumgebung

Für die Analyse des Programms „KeePass“ wurde mit Hilfe der Virtualisierungssoftware Oracle VirtualBox (Version 5.0.32) eine virtuelle Maschine mit dem Betriebssystem Windows 7 (32-bit Version) als Arbeitsumgebung aufgesetzt. Ein „Gemeinsamer Ordner“ wurde zum Datenaustausch von Host- zu Gastsystem verwendet. Auf der virtuellen Maschine wurden außerdem Softwareprogramme „Process Monitor“ (Version 3.31) und „Process Explorer“ (Version 16.20) aus der Sysinternals-Suite der Fa. Microsoft verwendet<sup>1</sup>. Außerdem wurde die Software Regshot (Version 1.9.0) verwendet um die Änderungen an der Registry zu protokollieren. Außerdem wurde für die Verwendung des **idifference2.py** Programms im Zuge

---

<sup>1</sup> Heruntergeladen von <https://technet.microsoft.com/de-de/sysinternals/bb842062>

der Verwendung der Zustandsmethode nach Dewald auch noch die virtuelle Maschine FIWALK (Linux OS) aus Modul 105 verwendet.

Auf dem Gastsystem wurden die Programme *Notepad++* (Version 7.2.2, Texteditor für die Bearbeitung der Skripte), *7-zip File Manager* (Version 16.04, Archivierungsprogramm), *Cygwin* (Version 2.6.1, Ausführung der Skripte) und *Microsoft Word 2013* (Version 15.0.4893.1000, Erstellung des Berichts) als auch das *MS Snipping Tool* (Version 1607, Erstellung von Screenshots) verwendet. Außerdem wurden die Programme *WinPrefetchView* (v.1.35), *RegViewer* (Version 1.3.0.0), *Wireshark* (Version 2.2.3) und *Autopsy* (Version 4.3.0) für die weitere Analyse der festgestellten Dateien verwendet.

## 2. Technische Anwendungsanalyse

In diesem Kapitel erfolgt eine Beschreibung der technischen Vorgehensweise und es werden die Ergebnisse aus der Anwendungsanalyse des Passwortmanagers „KeePass“ aufgezeigt. Im ersten Abschnitt wird die Anwendung „KeePass“ und die untersuchten Funktionen in Kurzfassung dargestellt. Im darauffolgenden Kapitel wird auf die generelle Vorgehensweise zur Gewinnung der Spurenmenge durch z.B. Zustands- und Ereignismethode eingegangen. Als letzter Unterabschnitt folgt die Auflistung der gefundenen persistenten Spuren.

### 2.1 KeePass – Ein Überblick

Die Software KeePass ist ursprünglich von **Dominik Reichl** in der Programmiersprache C++ entwickelt worden und seit Version 2.x basiert diese auf C#. KeePass ist als freies Programm unter den Bedingungen der GNU General Public License (GPL) verfügbar. Laut den Angaben auf der offiziellen Webseite <http://keepass.info/features.html> wird die Passwortdatenbank mit Hilfe des Advanced Encryption Standard (AES, Rijndael) und des Twofish Algorithmus verschlüsselt. Das Passwort für die Datenbank wird außerdem noch mittels SHA-256 gehasht. Als zusätzlicher Schutz kann neben einem Passwort auch noch eine Key-Datei für eine Passwortdatenbank erstellt werden.

Das Programm KeePass bietet außerdem unter anderem die folgenden weiteren Funktionalitäten an:

- Portabilität, keine Installation notwendig.
- Der Inhalt der Datenbank kann in verschiedene Formate (wie z.B. CSV, XML und TXT) exportiert werden.
- Einträge für die Datenbank können aus verschiedenen Formaten importiert werden.
- Such- und Sortierungsfunktionen für die Passwortdatenbank.
- Generierung eines zufälligen Passwortes mittels eigenem Generator.
- Funktionen die ausgewählten Passwörter in die Zwischenablage beziehungsweise in ausgewählte Webforms zu kopieren.
- Erweiterbarkeit durch Plugins.

Für die Anwendungsanalyse wurde die Portable Version von KeePass verwendet, da durch den Autor auf der Webseite <http://keepass.info/download.html> diese unter anderem wie folgt angepriesen wird: „KeePass runs without any additional installation and won't store any settings outside the application directory.“

Im Zuge der Anwendungsanalyse wurde sich auf die nachfolgenden Funktionen beziehungsweise Aktionen konzentriert:

- Erstmaliges Starten (einmal mit Aktivierung und einmal mit Deaktivierung der „automatischen Updatefunktion“)
- Erstellung einer Passwortdatenbank
- Starten des Programms über die Kommandozeile
- Erstellen, Ändern, Suchen und Löschen eines Eintrags
- Export der Einträge in einer XML-Datei
- Import von Einträgen aus einer XML-Datei
- Schließen des Programms.

Die Passwortdatenbank wurde nur mit Hilfe eines Passwortes gesichert, die Anwendungsanalyse enthält also keine Aussagen über die Verwendung einer Key-Datei. Um eine möglichst hohe Automatisierung zu erreichen wurde, außerdem das Plugins in Form der KPScript.exe<sup>2</sup> verwendet. Dieses wurde ebenfalls von Dominik Reichl in C# entwickelt um „Scripting“ mit KeePass zu ermöglichen.

## **2.2 Generelle Vorgehensweisen**

Zur Erhebung der Spurenmenge, welche durch die Verwendung von „KeePass“ bei den obengenannten Aktionen entstehen wurde die im Kapitel 1.3 genannte Arbeitsumgebung verwendet. Es wurde sowohl die Ereignismethode unter Verwendung des Programms „Process Monitors“ als auch die Zustandsmethode unter Verwendung von überarbeiteten Skripten, basierend auf den Skripten aus der Hausarbeit zu Modul 105, verwendet. Dateien, welche aufgrund der Ergebnisse dieser Methoden als relevant angesehen werde, wurde anschließend genauer mittels der Programme Autopsy, WinPrefetchView, RegViewer und Notepad++ analysiert.

Die obengenannten Aktivitäten wurden chronologisch durchlaufen und sowohl vor als auch nach einer Aktion wurde der aktuelle Systemzustand mittels VirtualBox gesichert. Dadurch konnte ein Zurückkehren in die jeweiligen Phasen ermöglicht werden, falls etwaige Fehler auftreten sollten.

### **2.2.1. Ereignismethode**

Die Programme „Process Monitor“ und „Process Explorer“ wurden verwendet um die Ereignismethode durchzuführen. Mit Hilfe des „Process Explorer“ wurde die Prozessstruktur der KeePass.exe näher untersucht und mittels Screenshots gesichert. Der „Process Monitor“ wurde verwendet um Ereignisse bei der Ausführung der KeePass.exe beziehungsweise Aktionen innerhalb des Programms zu protokollieren. Dazu wurde der „Process Monitor“ zuerst gestartet und ein Filter auf die „KeePass.exe“ eingestellt, damit nur Ereignisse dieser protokolliert werden. Nach der Durchführung einer Aktion wurden jeweils die aufgezeichneten Ereignisse in einer nativen PML-Datei abgespeichert und anschließend verworfen. Die

---

<sup>2</sup> Mehr Informationen diesbezüglich sind unter [http://keepass.info/help/v2\\_dev/scr\\_index.html](http://keepass.info/help/v2_dev/scr_index.html) zu finden.

PML-Dateien<sup>3</sup> wurden später dann auf dem Hostsystem mit Hilfe des „Process Monitors“ (32-bit Modus) ausgewertet.

### 2.2.2. Zustandsmethode

Die Zustandsmethode nach Dewald wurde basierend auf dem Verfahren, welches vom Autor für die Hausarbeit im Modul 105 erstellt wurde, durchgeführt. Dabei wurde in die jeweiligen Skripte angepasst. Die Skripte können im Anhang A eingesehen werden. Das Skript *vboxmanage.sh* (siehe Punkt A.1.1) wurde verwendet um mittels des Programms *VBoxManage.exe* (= Bestandteil von *VirtualBox*) automatisiert die zu untersuchenden Aktionen auf der virtuellen Maschine auszuführen, die jeweiligen Systemzustände zu sichern und Festplattenabbilder zu erzeugen. Das Skript *idiff2.sh* wurde in der virtuellen Maschine **fiwalk** gestartet um mit Hilfe des Programms *idifference2.py* die Spurenmenge und die charakteristischen Spuren der jeweiligen Aktionen zu berechnen.

### 2.2.3. Regshot und Wireshark

Die verwendete virtuelle Maschine wurde über die Kommandozeile dahingehend konfiguriert, dass während der durchgeführten und zu analysierenden Aktionen der Netzwerkverkehr durch *VirtualBox* automatisch mitgeschnitten wird und einer PCAP Datei abgespeichert wird. Die so erstellte PCAP Datei wurde anschließend mit Hilfe des Software *Wireshark* ausgewertet.

```
VBoxManage modifyvm m117_win7-Klon --nictrace1 on --nictracefile1 D:\Forensik\M117\pcap\file.pcap
```

Die Änderungen an der Windows-Registry wurden neben den Aufzeichnungen mittels des „ProcessMonitors“ auch noch mit Hilfe der Software *Regshot* protokolliert. Daher wurde mittels *Regshot* vor der Ausführung einer Aktion ein Abbild der Registry genommen. Anschließend wurde die zu untersuchende Aktion

ausgeführt und ein zweites Abbild der Registry mit Hilfe von *Regshot* erstellt. Anschließend wurde *Regshot* verwendet um die beiden Abbilder zu vergleichen und die Änderungen in einer Textdatei zu speichern

### 2.2.4. Analyse der Dateien

Falls ein einfaches Kopieren von interessanten Dateien nicht möglich war. Wurden die im Zuge der Zustandsmethode erstellten Festplattenabbilder mit Hilfe der Forensikprogramms **Autopsy** untersucht und eine Dateiwiederherstellung mittels *Carving* angestoßen. Die nähere Betrachtung der Hexwerte beziehungsweise lesbarer Strings innerhalb einer Datei wurde mit Hilfe von *Autopsy* beziehungsweise *Notepad++* vorgenommen.

---

<sup>3</sup> Dieser Bericht enthält aufgrund des vorgegebenen Umfangs nur Auszüge aus diesen Dateien. Auf Nachfrage werden die Dateien gerne übermittelt.

## 2.3 Persistente Spurenmenge

Mit Hilfe der im vorherigen Kapitel genannten Methoden wurden die persistente Spurenmenge des Passwortmanagers KeePass ermittelt und analysiert. In den nachfolgenden Unterabschnitten werden die gewonnenen Erkenntnisse in Bezug auf die persistenten als auch charakteristischen Spuren aufgelistet. Die Spuren werden in die Bereiche Dateisystem, Registry und Prefetch unterteilt. Die Unterscheidung zwischen Dateisystem und Registry erfolgt da eine inhaltliche Änderung auf der Ebene der Registry nicht auf der Dateisystemebene feststellbar ist.

### 2.3.1. Dateisystem

Der für diese Analyse verwendete Windowsbenutzer hatte die Kennung „m117“. Die KeePass.exe befand sich im Verzeichnis C:\keepass\keepass-2.34\ und die verwendete Datenbank **database.kdbx** wurde im Verzeichnis C:\keepass\ abgespeichert.

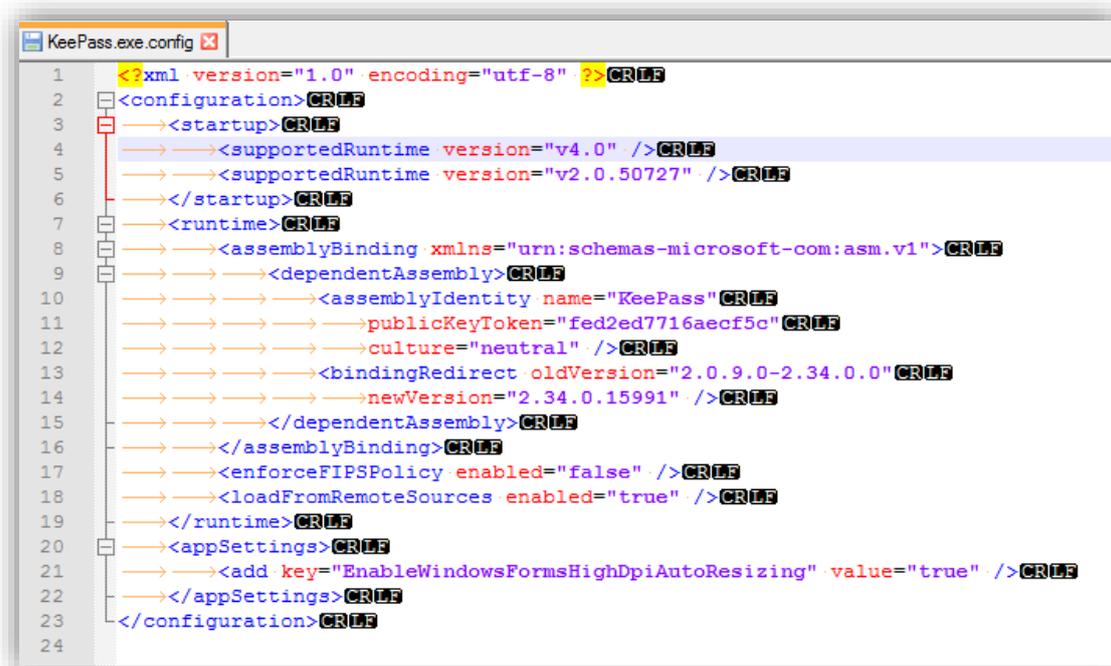
#### Anwendungsverzeichnis:

Durch das Entpacken des KeePass-2.34.zip Archivs, welches von der offiziellen Webseite „keepass.info“ heruntergeladen war, wurden die folgenden Dateien beziehungsweise Ordner im Zielordner erstellt:

Plugins	Ordner in welchem Plugins hinterlegt werden müssen. Ordner ist leer.
XSL	Ordner in welchem sich Dateien für die XSL Transformation befinden.
XSL\KDBX_DetailsFull.xsl	XSL Stylesheet für den XML-Export aller Informationen.
XSL\KDBX_DetailsLite.xsl	XSL Stylesheet für den XML-Export aller Informationen ohne Gültigkeitszeitraum der Passwörter.
XSL\KDBX_PasswordsOnly.xsl	XSL Stylesheet für den XML-Export nur von Passwörtern.
XSL\KDBX_Styles.css	CSS-Datei mit Informationen für Schriftart, -größe und -farbe.
XSL\KDBX_Tabular.xsl	XSL Stylesheet für den XML-Export aller Informationen in tabellarischer Form.
XSL\TableHeader.gif	GIF-Datei, für den TableHeader.
KeePass.chm	Kompilierte HTML-Hilfedatei mit der Dokumentation zu KeePass 2.x
KeePass.exe	Ausführbare Datei des KeePass – Passwortmanagers
KeePass.exe.config	Konfigurationsdatei für die KeePass.exe im XML-Format. Enthält z.B. die Versionsnummer des verwendeten KeePass.
KeePass.XmlSerializers.dll	Native Windowsbibliothek für die XML-Serialisierung.
KeePassLibC32.dll	Native Windowsbibliothek mit dem „Kerncode“ von KeePass für 32-bit <sup>4</sup>
KeePassLibC64.dll	Native Windowsbibliothek mit dem „Kerncode“ von KeePass für 64-bit.
License.txt	Textdatei mit Lizenzinformationen

---

<sup>4</sup> Informationen von <https://sourceforge.net/p/keepass/discussion/329221/thread/23e38a99/>



```

1  <?xml version="1.0" encoding="utf-8" ?>CRLE
2  <configuration>CRLE
3  <startup>CRLE
4      <supportedRuntime version="v4.0" />CRLE
5      <supportedRuntime version="v2.0.50727" />CRLE
6  </startup>CRLE
7  <runtime>CRLE
8      <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">CRLE
9          <dependentAssembly>CRLE
10             <assemblyIdentity name="KeePass" CRLE
11                 publicKeyToken="fed2ed7716aecf5c" CRLE
12                 culture="neutral" />CRLE
13             <bindingRedirect oldVersion="2.0.9.0-2.34.0.0" CRLE
14                 newVersion="2.34.0.15991" />CRLE
15             </dependentAssembly>CRLE
16          </assemblyBinding>CRLE
17          <enforceFIPSPolicy enabled="false" />CRLE
18          <loadFromRemoteSources enabled="true" />CRLE
19        </runtime>CRLE
20  <appSettings>CRLE
21      <add key="EnableWindowsFormsHighDpiAutoResizing" value="true" />CRLE
22  </appSettings>CRLE
23 </configuration>CRLE
24

```

Abbildung 1: Inhalt von KeePass.exe.config

Deinstallation überprüft, ob das .NET Framework installiert ist.

#### Identifikation relevanter Dateien mittels der Zustandsmethode:

Die im Zuge der Zustandsmethode erstellten Dateien mit den Spuren für die einzelnen Aktionen sind im Anhang unter den Punkten 3.1. bis 3.7. einsehbar.

Beim erstmaligen Starten von KeePass.exe wird die Datei **C:\Windows\Prefetch\KEEPASS.exe-CC926147.pf** erstellt. Beim späteren erneuten Starten von KeePass wird diese Datei ebenfalls gelesen. Mehr zu dieser Datei unter dem nachfolgenden Punkt 2.3.3.

Beim erstmaligen Starten von KeePass.exe wurde ebenfalls eine Datei **GDIPFONTCACHEV1.DAT** unter **C:\Users\m117\AppData\Local\** erstellt.

Nach der Erstellung einer Datenbank-Datei „database.kdbx“ und Speicherung unter C:\keepass\ mittels KeePass konnten nach dieser Aktion natürlich Lese- und Schreibzugriffe mittels der Zustandsmethode bzgl. dieser Datei und diesem Verzeichnis festgestellt werden. Außerdem konnten nach dieser Aktion auch noch die Erstellung von .Windows-Verknüpfungen namens **keepass.lnk** und **database.kdbx.lnk** unter **C:\Users\m117\AppData\Roaming\Microsoft\Windows\Recent\** festgestellt werden.

Beim Beenden der KeePass.exe nach der Erstellung beziehungsweise bei Veränderungen bzgl. Datenbank-Datei wird die Konfigurationsdatei **KeePass.config.xml** im Anwendungsverzeichnis (hier **C:\keepass\KeePass-2.34\**) erstellt beziehungsweise modifiziert. In dieser Datei ist unter anderem, der Pfad der zuletzt verwendeten Datenbank-Datei enthalten.



## II. Erstellung einer Datenbank-Datei

**Threads:** Laden und Verwenden von msftedit.dll, mssvp.dll, mapi32.dll, urlmon.dll, wininet.dll, actxprxy.dll, imageres.dll, ieproxy.dll, xmlite.dll, comctl32.dll und propsys.dll.

**Registry:** Lesen der Einträge aus HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\FolderDescriptions für die Auswahl des Speicherorts der Datenbank-Datei.

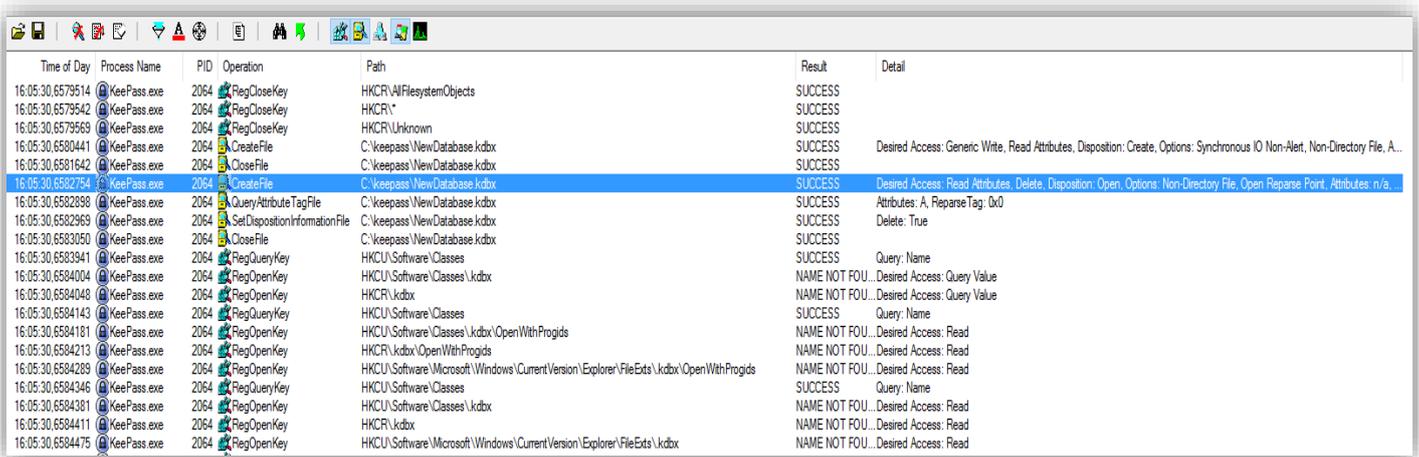
Lesen von Einträgen bzgl. des Öffnens von **.kdbx** Dateien zum Beispiel in HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\FileExts\.

Hinzufügen eines Eintrags in HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\LastVisitedPidIMRU und -\OpenSavePidIMRU\kdbx.

**Dateien:** Zugriff auf C:\Users\m117\Documents und C:\Users\m117\AppData\Roaming.

Erstellung von KeePass.exe.Local, Zugriffe auf die dll's wie z.B. cmdlg32.dll, shellstyle.dll, propsys.dll, ntshui.dll, xmlite.dll, msftedit.dll.

Erstellung der Datei **C:\keepass\NewDatabase.kdbx**.



The screenshot shows the Windows Process Monitor tool recording the actions of KeePass.exe. The table below summarizes the key operations shown in the image:

Time of Day	Process Name	PID	Operation	Path	Result	Detail
16:05:30,65739514	KeePass.exe	2064	RegCloseKey	HKCR\FileSystemObjects	SUCCESS	
16:05:30,65739542	KeePass.exe	2064	RegCloseKey	HKCR*	SUCCESS	
16:05:30,65739568	KeePass.exe	2064	RegCloseKey	HKCR\Unknown	SUCCESS	
16:05:30,6580441	KeePass.exe	2064	CreateFile	C:\keepass\NewDatabase.kdbx	SUCCESS	Desired Access: Generic Write, Read Attributes, Disposition: Create, Options: Synchronous IO Non-Alert, Non-Directory File, A...
16:05:30,6581842	KeePass.exe	2064	CloseFile	C:\keepass\NewDatabase.kdbx	SUCCESS	
16:05:30,6582754	KeePass.exe	2064	CreateFile	C:\keepass\NewDatabase.kdbx	SUCCESS	Desired Access: Read Attributes, Delete, Disposition: Open, Options: Non-Directory File, Open Reparse Point, Attributes: n/a
16:05:30,6582898	KeePass.exe	2064	QueryAttribute TagFile	C:\keepass\NewDatabase.kdbx	SUCCESS	Attributes: A, Reparse Tag: 0x0
16:05:30,6582968	KeePass.exe	2064	SetDispositionInformationFile	C:\keepass\NewDatabase.kdbx	SUCCESS	Delete: True
16:05:30,6583050	KeePass.exe	2064	CloseFile	C:\keepass\NewDatabase.kdbx	SUCCESS	
16:05:30,6583941	KeePass.exe	2064	RegQueryKey	HKCU\Software\Classes	SUCCESS	Query: Name
16:05:30,6584004	KeePass.exe	2064	RegOpenKey	HKCU\Software\Classes\kdbx	NAME NOT FOUND	Desired Access: Query Value
16:05:30,6584048	KeePass.exe	2064	RegOpenKey	HKCR\kdbx	NAME NOT FOUND	Desired Access: Query Value
16:05:30,6584143	KeePass.exe	2064	RegQueryKey	HKCU\Software\Classes	SUCCESS	Query: Name
16:05:30,6584181	KeePass.exe	2064	RegOpenKey	HKCU\Software\Classes\kdbx\OpenWithProgids	NAME NOT FOUND	Desired Access: Read
16:05:30,6584213	KeePass.exe	2064	RegOpenKey	HKCR\kdbx\OpenWithProgids	NAME NOT FOUND	Desired Access: Read
16:05:30,6584283	KeePass.exe	2064	RegOpenKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\FileExts\kdbx\OpenWithProgids	NAME NOT FOUND	Desired Access: Read
16:05:30,6584346	KeePass.exe	2064	RegQueryKey	HKCU\Software\Classes	SUCCESS	Query: Name
16:05:30,6584381	KeePass.exe	2064	RegOpenKey	HKCU\Software\Classes\kdbx	NAME NOT FOUND	Desired Access: Read
16:05:30,6584411	KeePass.exe	2064	RegOpenKey	HKCR\kdbx	NAME NOT FOUND	Desired Access: Read
16:05:30,6584475	KeePass.exe	2064	RegOpenKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\FileExts\kdbx	NAME NOT FOUND	Desired Access: Read

Abbildung 3: Process Monitor – Erstellung einer Datenbank-Datei

## III. Beenden von KeePass:

**Threads:** Die Threads als auch der Prozess von KeePass.exe werden beendet.

**Dateien:** Die Dateizugriffe auf die Dateien KeePass.exe und KeePass.XmlSerializers.dll werden geschlossen.

Vorgenommene Änderungen an den Einstellungen von KeePass beziehungsweise an den Einträgen der Datenbank-Datei werden in temporären Dateien gespeichert. Mit diesen Dateien werden die ursprünglichen Dateien dann später überschrieben. So wird z.B. aus KeePass.config.xml.tmp die neue KeePass.config.xml und aus database.kdbx.tmp wird die neue database.kdbx.

## IV. Erstellung eines Datenbankeintrags:

**Threads:** Laden von KeePassLibC32.dll.

**Registry:** Auslesen des Dateipfads für die rsaenh.dll HKLM\SOFTWARE\Microsoft\Cryptography\Defaults\Provider\Microsoft Enhanced RSA and AES Cryptographic Provider\Type.

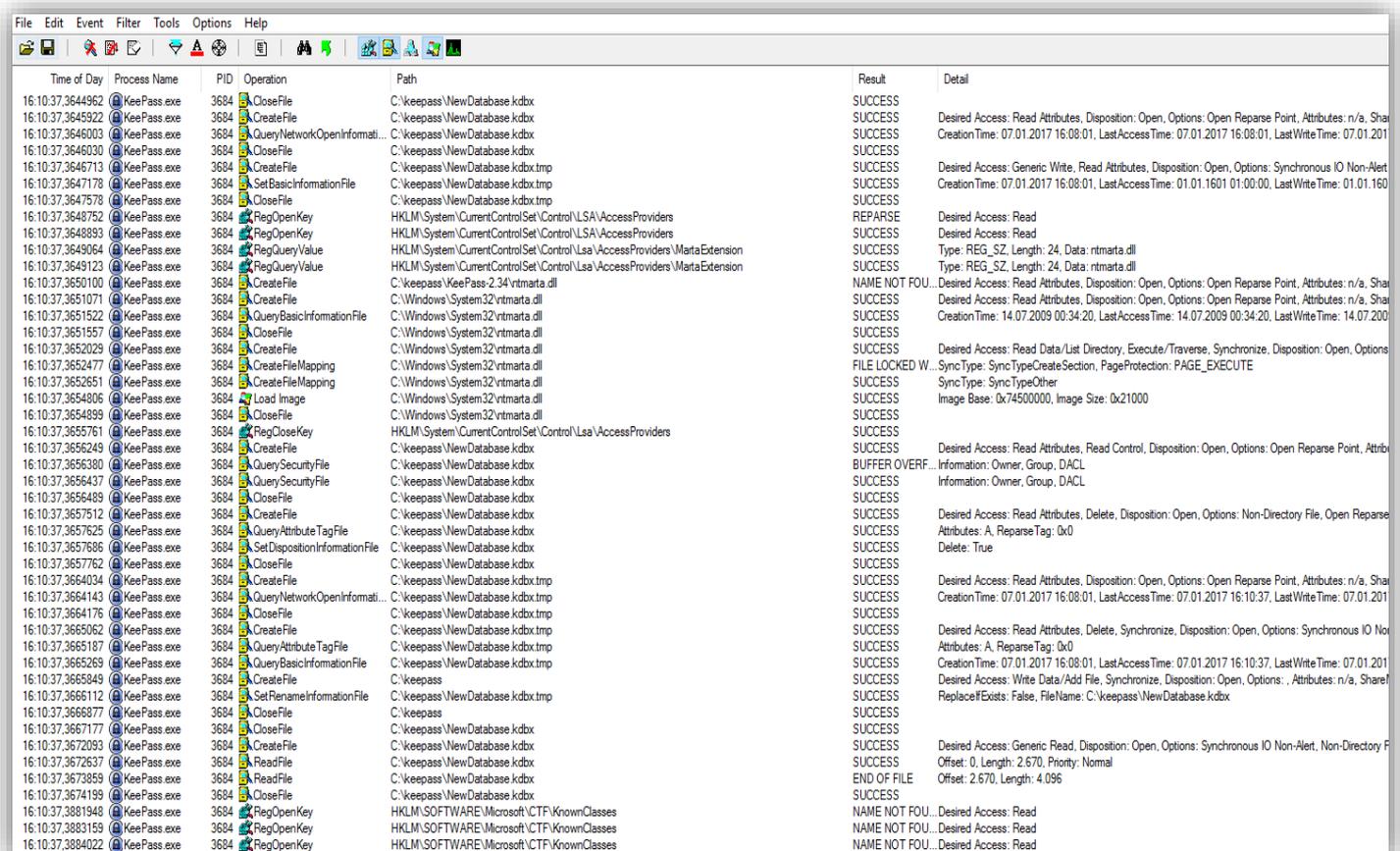
**Dateien:** Erstellen einer database.kdbx.tmp und anschließender Umbenennung in database.kdbx.

## V. Ändern eines Eintrags

**Threads:** Laden eines Images von ntmarta.dll.

**Registry:** Lesen aus HKLM\System\CurrentControlSet\Control\Lsa\AccessProviders\MartaExtension.

**Dateien:** Erstellen einer database.kdbx.tmp und anschließender Umbenennung in database.kdbx.  
Lesen von ntmarta.dll aus C:\Windows\System32.



Time of Day	Process Name	PID	Operation	Path	Result	Detail
16:10:37.3644962	Keepass.exe	3684	CloseFile	C:\keepass\NewDatabase.kdbx	SUCCESS	
16:10:37.3645922	Keepass.exe	3684	CreateFile	C:\keepass\NewDatabase.kdbx	SUCCESS	Desired Access: Read Attributes, Disposition: Open, Options: Open Reparse Point, Attributes: n/a, Share
16:10:37.3646003	Keepass.exe	3684	QueryNetworkOpenInformati...	C:\keepass\NewDatabase.kdbx	SUCCESS	CreationTime: 07.01.2017 16:08:01, LastAccessTime: 07.01.2017 16:08:01, LastWriteTime: 07.01.2017
16:10:37.3646030	Keepass.exe	3684	CloseFile	C:\keepass\NewDatabase.kdbx	SUCCESS	
16:10:37.3646713	Keepass.exe	3684	CreateFile	C:\keepass\NewDatabase.kdbx.tmp	SUCCESS	Desired Access: Generic Write, Read Attributes, Disposition: Open, Options: Synchronous IO Non-Alert
16:10:37.3647178	Keepass.exe	3684	SetBasicInformationFile	C:\keepass\NewDatabase.kdbx.tmp	SUCCESS	CreationTime: 07.01.2017 16:08:01, LastAccessTime: 01.01.1601 01:00:00, LastWriteTime: 01.01.1601
16:10:37.3647578	Keepass.exe	3684	CloseFile	C:\keepass\NewDatabase.kdbx.tmp	SUCCESS	
16:10:37.3648752	Keepass.exe	3684	RegOpenKey	HKLM\System\CurrentControlSet\Control\Lsa\AccessProviders	REPARSE	Desired Access: Read
16:10:37.3648993	Keepass.exe	3684	RegOpenKey	HKLM\System\CurrentControlSet\Control\Lsa\AccessProviders	SUCCESS	Desired Access: Read
16:10:37.3649064	Keepass.exe	3684	RegQueryValue	HKLM\System\CurrentControlSet\Control\Lsa\AccessProviders\MartaExtension	SUCCESS	Type: REG_SZ, Length: 24, Data: ntmarta.dll
16:10:37.3649123	Keepass.exe	3684	RegQueryValue	HKLM\System\CurrentControlSet\Control\Lsa\AccessProviders\MartaExtension	SUCCESS	Type: REG_SZ, Length: 24, Data: ntmarta.dll
16:10:37.3650100	Keepass.exe	3684	CreateFile	C:\keepass\KeePass-2.34\ntmarta.dll	NAME NOT FOUND	Desired Access: Read Attributes, Disposition: Open, Options: Open Reparse Point, Attributes: n/a, Share
16:10:37.3651071	Keepass.exe	3684	CreateFile	C:\Windows\System32\ntmarta.dll	SUCCESS	Desired Access: Read Attributes, Disposition: Open, Options: Open Reparse Point, Attributes: n/a, Share
16:10:37.3651522	Keepass.exe	3684	QueryBasicInformationFile	C:\Windows\System32\ntmarta.dll	SUCCESS	CreationTime: 14.07.2009 00:34:20, LastAccessTime: 14.07.2009 00:34:20, LastWriteTime: 14.07.2009
16:10:37.3651557	Keepass.exe	3684	CloseFile	C:\Windows\System32\ntmarta.dll	SUCCESS	
16:10:37.3652029	Keepass.exe	3684	CreateFile	C:\Windows\System32\ntmarta.dll	SUCCESS	Desired Access: Read Data/List Directory, Execute/Traverse, Synchronize, Disposition: Open, Options
16:10:37.3652477	Keepass.exe	3684	CreateFileMapping	C:\Windows\System32\ntmarta.dll	FILE LOCKED W...	SyncType: SyncTypeCreateSection, PageProtection: PAGE_EXECUTE
16:10:37.3652651	Keepass.exe	3684	CreateFileMapping	C:\Windows\System32\ntmarta.dll	SUCCESS	SyncType: SyncTypeOther
16:10:37.3654806	Keepass.exe	3684	Load Image	C:\Windows\System32\ntmarta.dll	SUCCESS	Image Base: 0x74500000, Image Size: 0x21000
16:10:37.3654899	Keepass.exe	3684	CloseFile	C:\Windows\System32\ntmarta.dll	SUCCESS	
16:10:37.3655761	Keepass.exe	3684	RegCloseKey	HKLM\System\CurrentControlSet\Control\Lsa\AccessProviders	SUCCESS	
16:10:37.3656249	Keepass.exe	3684	CreateFile	C:\keepass\NewDatabase.kdbx	SUCCESS	Desired Access: Read Attributes, Read Control, Disposition: Open, Options: Open Reparse Point, Attrib
16:10:37.3656380	Keepass.exe	3684	QuerySecurityFile	C:\keepass\NewDatabase.kdbx	BUFFER OVERFLO...	Information: Owner, Group, DACL
16:10:37.3656437	Keepass.exe	3684	QuerySecurityFile	C:\keepass\NewDatabase.kdbx	SUCCESS	Information: Owner, Group, DACL
16:10:37.3656489	Keepass.exe	3684	CloseFile	C:\keepass\NewDatabase.kdbx	SUCCESS	
16:10:37.3657512	Keepass.exe	3684	CreateFile	C:\keepass\NewDatabase.kdbx	SUCCESS	Desired Access: Read Attributes, Delete, Disposition: Open, Options: Non-Directory File, Open Reparse
16:10:37.3657625	Keepass.exe	3684	QueryAttribute Tag File	C:\keepass\NewDatabase.kdbx	SUCCESS	Attributes: A, Reparse Tag: 0x0
16:10:37.3657686	Keepass.exe	3684	SetDispositionInformationFile	C:\keepass\NewDatabase.kdbx	SUCCESS	Delete: True
16:10:37.3657762	Keepass.exe	3684	CloseFile	C:\keepass\NewDatabase.kdbx	SUCCESS	
16:10:37.3660304	Keepass.exe	3684	CreateFile	C:\keepass\NewDatabase.kdbx.tmp	SUCCESS	Desired Access: Read Attributes, Disposition: Open, Options: Open Reparse Point, Attributes: n/a, Share
16:10:37.3664143	Keepass.exe	3684	QueryNetworkOpenInformati...	C:\keepass\NewDatabase.kdbx.tmp	SUCCESS	CreationTime: 07.01.2017 16:08:01, LastAccessTime: 07.01.2017 16:10:37, LastWriteTime: 07.01.2017
16:10:37.3664176	Keepass.exe	3684	CloseFile	C:\keepass\NewDatabase.kdbx.tmp	SUCCESS	
16:10:37.3665062	Keepass.exe	3684	CreateFile	C:\keepass\NewDatabase.kdbx.tmp	SUCCESS	Desired Access: Read Attributes, Delete, Synchronize, Disposition: Open, Options: Synchronous IO No
16:10:37.3665187	Keepass.exe	3684	QueryAttribute Tag File	C:\keepass\NewDatabase.kdbx.tmp	SUCCESS	Attributes: A, Reparse Tag: 0x0
16:10:37.3665269	Keepass.exe	3684	QueryBasicInformationFile	C:\keepass\NewDatabase.kdbx.tmp	SUCCESS	CreationTime: 07.01.2017 16:08:01, LastAccessTime: 07.01.2017 16:10:37, LastWriteTime: 07.01.2017
16:10:37.3665949	Keepass.exe	3684	CreateFile	C:\keepass\NewDatabase.kdbx	SUCCESS	Desired Access: Write Data/Add File, Synchronize, Disposition: Open, Options: Attributes: n/a, Share
16:10:37.3666112	Keepass.exe	3684	SetRenameInformationFile	C:\keepass\NewDatabase.kdbx.tmp	SUCCESS	ReplaceIfExists: False, FileName: C:\keepass\NewDatabase.kdbx
16:10:37.3666877	Keepass.exe	3684	CloseFile	C:\keepass\NewDatabase.kdbx	SUCCESS	
16:10:37.3667177	Keepass.exe	3684	CloseFile	C:\keepass\NewDatabase.kdbx	SUCCESS	
16:10:37.3672093	Keepass.exe	3684	CreateFile	C:\keepass\NewDatabase.kdbx	SUCCESS	Desired Access: Generic Read, Disposition: Open, Options: Synchronous IO Non-Alert, Non-Directory F
16:10:37.3672637	Keepass.exe	3684	ReadFile	C:\keepass\NewDatabase.kdbx	SUCCESS	Offset: 0, Length: 2.670, Priority: Normal
16:10:37.3673859	Keepass.exe	3684	ReadFile	C:\keepass\NewDatabase.kdbx	END OF FILE	Offset: 2.670, Length: 4.096
16:10:37.3674199	Keepass.exe	3684	CloseFile	C:\keepass\NewDatabase.kdbx	SUCCESS	
16:10:37.3881948	Keepass.exe	3684	RegOpenKey	HKLM\SOFTWARE\Microsoft\CTF\KnownClasses	NAME NOT FOUND	Desired Access: Read
16:10:37.3883159	Keepass.exe	3684	RegOpenKey	HKLM\SOFTWARE\Microsoft\CTF\KnownClasses	NAME NOT FOUND	Desired Access: Read
16:10:37.3884022	Keepass.exe	3684	RegOpenKey	HKLM\SOFTWARE\Microsoft\CTF\KnownClasses	NAME NOT FOUND	Desired Access: Read

Abbildung 4: Process Monitor –Änderung an einem Datensatz

## VI. Löschen eines Eintrags:

**Threads:** Laden eines Images von xmllite.dll.

**Dateien:** Erstellen einer database.kdbx.tmp und anschließender Umbenennung in database.kdbx.  
Lesen von xmllite.dll aus C:\Windows\System32.

## VII. Export der Einträge in eine XML-Datei

**Threads:** Laden und Verwenden von msftedit.dll, mssvp.dll, mapi32.dll, urlmon.dll, wininet.dll, actxprxy.dll, imageres.dll, ieproxy.dll, xmllite.dll, comctl32.dll und propsys.dll.

**Registry:** Auslesen des Dateipfads für die zuzuladenden DLL-Dateien aus der Registry. Für die prpsys.dll z.B. aus HKCR\CLSID\{1f486a52-3cb1-48fd-8f50-b8dc300d9f9d}\InProcServer32\{Default}.



Auslesen der Pfade zu den dll-Dateien z.B. für die shell32.dll aus HKCR\CLSID\{0E5AAE11-A475-4c5b-AB00-C66DE400274E}\InProcServer32\{Default}.

### Dateien:

Lesen des Verzeichnis und der XML-Datei, welche importiert werden sollen.

Lesen der shell32.dll, msxml3.dll, shellstyle.dll, mssvp.dll, msTracer.dll, msfte.dll, mapi32.dll und msftedit.dll.

Time of Day	Process Name	PID	Operation	Path	Result	Detail
16:14:34.3840969	KeePass.exe	3684	RegSetValue	HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\Modules\NavPane\ExpandedState	SUCCESS	Type: REG_BINARY, Length: 638, Data: 06 00 00 00 16 00 14 00 1F 80 80 A6 3C 32 4D C2
16:14:34.3841019	KeePass.exe	3684	RegOpenKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\Modules\NavPane	SUCCESS	
16:14:34.3848469	KeePass.exe	3684	RegOpenKey	HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\ShellCompatibility\Objects\{031E4825-7B94-	NAME NOT FOU...	Desired Access: Query Value
16:14:34.3850324	KeePass.exe	3684	RegCreateKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\Modules\NavPane\	SUCCESS	Desired Access: Read/Write, Disposition: REG_OPENED_EXISTING_KEY
16:14:34.3850586	KeePass.exe	3684	RegSetValue	HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\Modules\NavPane\ExpandedState	SUCCESS	Type: REG_BINARY, Length: 638, Data: 06 00 00 00 16 00 14 00 1F 80 80 A6 3C 32 4D C2
16:14:34.3856420	KeePass.exe	3684	RegCloseKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\Modules\NavPane	SUCCESS	
16:14:34.3878157	KeePass.exe	3684	RegCloseKey	HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\CommandStore\shell\Windows.orga...	SUCCESS	
16:14:34.3878212	KeePass.exe	3684	RegCloseKey	HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\CommandStore	SUCCESS	
16:14:34.3879251	KeePass.exe	3684	RegCloseKey	HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\CommandStore\shell\Windows.orga...	SUCCESS	
16:14:34.3879285	KeePass.exe	3684	RegCloseKey	HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\CommandStore	SUCCESS	
16:14:34.4062476	KeePass.exe	3684	Thread Exit		SUCCESS	Thread ID: 2892, User Time: 0.0000000, Kernel Time: 0.0000000
16:14:34.4281070	KeePass.exe	3684	RegOpenKey	HKLM\Software\Microsoft\Windows\Tablet PC\	SUCCESS	Desired Access: Read
16:14:34.4281225	KeePass.exe	3684	RegQuery/Value	HKLM\SOFTWARE\Microsoft\Windows\Tablet PC\IsTabletPC	SUCCESS	Type: REG_DWORD, Length: 4, Data: 0
16:14:34.4281278	KeePass.exe	3684	RegCloseKey	HKLM\SOFTWARE\Microsoft\Windows\Tablet PC	SUCCESS	
16:14:34.4286626	KeePass.exe	3684	RegOpenKey	HKLM\SOFTWARE\Microsoft\CTF\KnownClasses	SUCCESS	
16:14:34.4289713	KeePass.exe	3684	CreateFile	C:\keepass\KeePass-2.34	SUCCESS	Desired Access: Read
16:14:34.4289839	KeePass.exe	3684	QueryNetworkOpenFormat...	C:\keepass\KeePass-2.34	SUCCESS	Desired Access: Read Attributes, Disposition: Open, Options: Open Reparse Point, Attributes: n/a, Sh...
16:14:34.4289877	KeePass.exe	3684	CloseFile	C:\keepass\KeePass-2.34	SUCCESS	
16:14:34.4290501	KeePass.exe	3684	CreateFile	C:\keepass\KeePass-2.34	SUCCESS	Desired Access: Execute/Traverse, Synchronization: Disposition: Open, Options: Directory, Synchronou...
16:14:34.4290628	KeePass.exe	3684	CloseFile	C:\Users\m117\Documents	SUCCESS	
16:14:34.4421209	KeePass.exe	3684	CloseFile	C:\Windows\System32\de-DE\mssvp.dll.mui	SUCCESS	
16:14:35.8074874	KeePass.exe	3684	RegOpenKey	HKLM\SOFTWARE\Microsoft\CTF\KnownClasses	NAME NOT FOU...	Desired Access: Read
16:14:35.8887882	KeePass.exe	3684	RegOpenKey	HKLM\SOFTWARE\Microsoft\CTF\KnownClasses	NAME NOT FOU...	Desired Access: Read
16:14:35.9040058	KeePass.exe	3684	RegOpenKey	HKLM\SOFTWARE\Microsoft\CTF\KnownClasses	NAME NOT FOU...	Desired Access: Read
16:14:35.9373852	KeePass.exe	3684	CreateFile	C:\Users\m117\Documents\NewDatabase.xml	SUCCESS	Desired Access: Generic Read, Disposition: Open, Options: Synchronous IO Non-Alert, Non-Directory
16:14:35.9384085	KeePass.exe	3684	ReadFile	C:\Users\m117\Documents\NewDatabase.xml	SUCCESS	Offset: 0, Length: 4,096, Priority: Normal
16:14:35.9386743	KeePass.exe	3684	QueryStandardInformationFile	C:\Users\m117\Documents\NewDatabase.xml	SUCCESS	AllocationSize: 16,384, EndOfFile: 12,743, NumberOfLinks: 1, DeletePending: False, Directory: False
16:14:35.9388954	KeePass.exe	3684	ReadFile	C:\Users\m117\Documents\NewDatabase.xml	SUCCESS	Offset: 4,096, Length: 4,096, Priority: Normal
16:14:35.9393319	KeePass.exe	3684	ReadFile	C:\Users\m117\Documents\NewDatabase.xml	SUCCESS	Offset: 8,192, Length: 4,096, Priority: Normal
16:14:35.9406434	KeePass.exe	3684	ReadFile	C:\Users\m117\Documents\NewDatabase.xml	SUCCESS	Offset: 12,288, Length: 455, Priority: Normal
16:14:35.9406674	KeePass.exe	3684	ReadFile	C:\Users\m117\Documents\NewDatabase.xml	END OF FILE	Offset: 12,743, Length: 4,096, Priority: Normal
16:14:35.9408787	KeePass.exe	3684	ReadFile	C:\Users\m117\Documents\NewDatabase.xml	END OF FILE	Offset: 12,743, Length: 4,096, Priority: Normal
16:14:35.9408962	KeePass.exe	3684	CloseFile	C:\Users\m117\Documents\NewDatabase.xml	SUCCESS	
16:14:35.9444127	KeePass.exe	3684	RegOpenKey	HKLM\SOFTWARE\Microsoft\Fusion\PublisherPolicy\Default\policy.2.34.KeePass.resources_de-DE...	NAME NOT FOU...	Desired Access: Read
16:14:35.9444705	KeePass.exe	3684	RegOpenKey	HKLM\SOFTWARE\Microsoft\Fusion\PublisherPolicy\Default\policy.2.34.KeePass.resources_de_fed...	NAME NOT FOU...	Desired Access: Read
16:14:35.9498218	KeePass.exe	3684	RegOpenKey	HKLM\SOFTWARE\Microsoft\CTF\KnownClasses	NAME NOT FOU...	Desired Access: Read
16:14:35.9659502	KeePass.exe	3684	RegCloseKey	HKCU\Software\Microsoft\Internet Explorer\Main	SUCCESS	
16:14:35.9659600	KeePass.exe	3684	RegCloseKey	HKCU\Software\Microsoft\Internet Explorer\Settings	SUCCESS	
16:14:42.0548597	KeePass.exe	3684	RegOpenKey	HKLM\SOFTWARE\Microsoft\CTF\KnownClasses	NAME NOT FOU...	Desired Access: Read
16:14:42.1286614	KeePass.exe	3684	RegOpenKey	HKLM\SOFTWARE\Microsoft\CTF\KnownClasses	NAME NOT FOU...	Desired Access: Read
16:14:42.1540284	KeePass.exe	3684	ReadFile	C:\Windows\assembly\NativeImages_v2.0.50727_32\mscorlib\62a1b3e4b40ec0e0c5cfad0c8848e64...	SUCCESS	Offset: 8,958,976, Length: 4,096, I/O Flags: Non-cached, Paging I/O, Synchronous Paging I/O, Prior...
16:14:42.1732899	KeePass.exe	3684	ReadFile	C:\Windows\assembly\NativeImages_v2.0.50727_32\mscorlib\62a1b3e4b40ec0e0c5cfad0c8848e64...	SUCCESS	Offset: 7,300,096, Length: 16,384, I/O Flags: Non-cached, Paging I/O, Synchronous Paging I/O, Prior...
16:14:42.2094389	KeePass.exe	3684	RegOpenKey	HKLM\SOFTWARE\Microsoft\CTF\KnownClasses	NAME NOT FOU...	Desired Access: Read
16:14:42.2105907	KeePass.exe	3684	RegOpenKey	HKLM\SOFTWARE\Microsoft\CTF\KnownClasses	NAME NOT FOU...	Desired Access: Read
16:14:42.2235828	KeePass.exe	3684	RegOpenKey	HKLM\SOFTWARE\Microsoft\CTF\KnownClasses	NAME NOT FOU...	Desired Access: Read
16:14:47.5620342	KeePass.exe	3684	Thread Exit		SUCCESS	Thread ID: 880, User Time: 0.0000000, Kernel Time: 0.0000000

Abbildung 6: Process Monitor - Import aus einer XML-Datei

## 2.3.2. Windows Registry

Mit Hilfe der Software Regshot konnte nachfolgendes festgestellt werden.

Beim Starten von KeePass werden unter anderem zwei neue Schlüssel unter **HKLM\SOFTWARE\Microsoft\Tracing\** names **KeePass\_RASAPI32** und **KeePass\_RASMANCS** angelegt. Beide Schlüssel erhalten die folgenden Werte:

Name :	Wert
EnableFileTracing	0x00000000
EnableConsoleTracing	0x00000000
FileTracingMask	0xFFFF0000
ConsoleTracingMask	0xFFFF0000
MaxFileSize	0x00100000
FileDirectory	"%windir%\tracing"

Beim Erstellen einer Datenbank-Datei werden unter anderem die folgenden Schlüssel neu in der Registry erstellt:

HKU\SID<sup>5</sup>\Software\Microsoft\Windows\CurrentVersion\Explorer\FileExts\kdbx  
HKU\SID\Software\Microsoft\Windows\CurrentVersion\Explorer\FileExts\kdbx\OpenWithList  
HKU\SID\Software\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\OpenSavePidIMRU\kdbx  
HKU\SID\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\kdbx  
HKU\SID\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\Folder

Beim **Schließen** von KeePass, dem **Erstellen**, **Suchen**, **Ändern** und **Löschen** eines Datenbankeintrages konnten keine weiteren signifikanten neuen Schlüssel festgestellt werden. Jedoch folgt bei jeder der untersuchten Aktionen eine Änderung von Werten des Schlüssels:

HKU\SID\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\{CEBFF5CD-ACE2-4F4F-9178-9926F41749EA}\Count\HRZR\_PGYFRFFVBA.

Hierbei handelt es sich um einen Eintrag im sogenannten UserAssist-Schlüssel. Eine windowseigene Funktionalität speichert für jedes ausführbare Programm in diesem Schlüssel beziehungsweise in Unterschlüssel, unter anderem wie oft es bereits ausgeführt beziehungsweise wann es zuletzt ausgeführt wurde. Die in diesem Schlüssel hinterlegten Werte sind ROT13 verschlüsselt<sup>6</sup>.

Beim **Export** beziehungsweise beim **Import** von Datenbankeinträge in eine **XML-Datei** beziehungsweise aus einer XML-Datei werden unter anderem die folgenden -3- Schlüssel neu erzeugt:

HKU\SID\Software\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\OpenSavePidIMRU\xml  
HKU\SID\Software\Microsoft\Windows\CurrentVersion\Explorer\FileExts\xml\OpenWithList  
HKU\SID\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\xml

Die Schlüssel erhalten dann die folgenden Werte: -- hinzufügen Screenshots

Besonders beim Importieren von Daten aus einer XML-Datei werden die beiden Werte von HKU\SID\Software\Microsoft\Windows\CurrentVersion\Explorer\FileExts\xml\OpenWithList wie folgt

gesetzt:	Name	Wert
	\a	"KeePass.exe"
	\MRUList	"a"

Eine weitere Überprüfung mit Hilfe von Regshot zeigt, dass bei der Verwendung einer Key-Datei zur zusätzlichen Absicherung der Datenbank, identische Einträge für die Datei-Endung **.key** in den drei nachfolgenden Schlüsseln zu finden sind:

HKU\SID\Software\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\OpenSavePidIMRU\key  
HKU\SID\Software\Microsoft\Windows\CurrentVersion\Explorer\FileExts\key\OpenWithList

<sup>5</sup> Die Security Identifier (S-1-5-21-2373602983-835143386-2166487477-1000) des Benutzer m117 wurde zu besserer Lesbarkeit und zur Verallgemeinerung durch Abkürzung SID ersetzt.

<sup>6</sup> Informationen aus <https://www.aldeid.com/wiki/Windows-userassist-keys> und <https://splotied.blogspot.de/2012/12/sans-forensic-artifact-6-userassist.html>

### 2.3.3. Prefetch

Wie bereits unter Punkt 2.3.1. genannt konnte mit Hilfe der Zustands- als auch der Ereignismethode festgestellt werden, dass eine Prefetch-Datei **KEEPASS.EXE-CC926147.pf** erstellt wurde. Die Prefetch-Dateien werden automatisch von Windows zur Optimierung und Beschleunigung des Starts von Anwendungen erstellt. In dieser Prefetch-Datei werden unter anderem Informationen über die für diese Anwendung notwendigen Dateien, als auch wann die Anwendung das letzte Mal ausgeführt wurde abgespeichert. Die Prefetch-Dateien können mit Hilfe von anderen bereits existierten Werkzeugen wie z.B. *WinPrefetchView v.1.35 von NirSoft*<sup>7</sup> ausgewertet werden. Aus der obengenannten Prefetch-Datei können mit Hilfe dieses Programms unter anderem die Dateipfade für die verwendeten DLL-Dateien als auch für die zuletzt verwendete Datenbankdatei entnommen werden.

Das Vorhandensein einer Prefetch-Datei kann als gute Spur für die Ausführung einer bestimmten .EXE-Datei angesehen werden, da ein Großteil der Windows-Nutzer von deren Existenz wissen dürfte und für die Beseitigung aller Spuren durch die Prefetch-Dateien eine genaue Analyse der eigentlichen Anwendung notwendig ist.

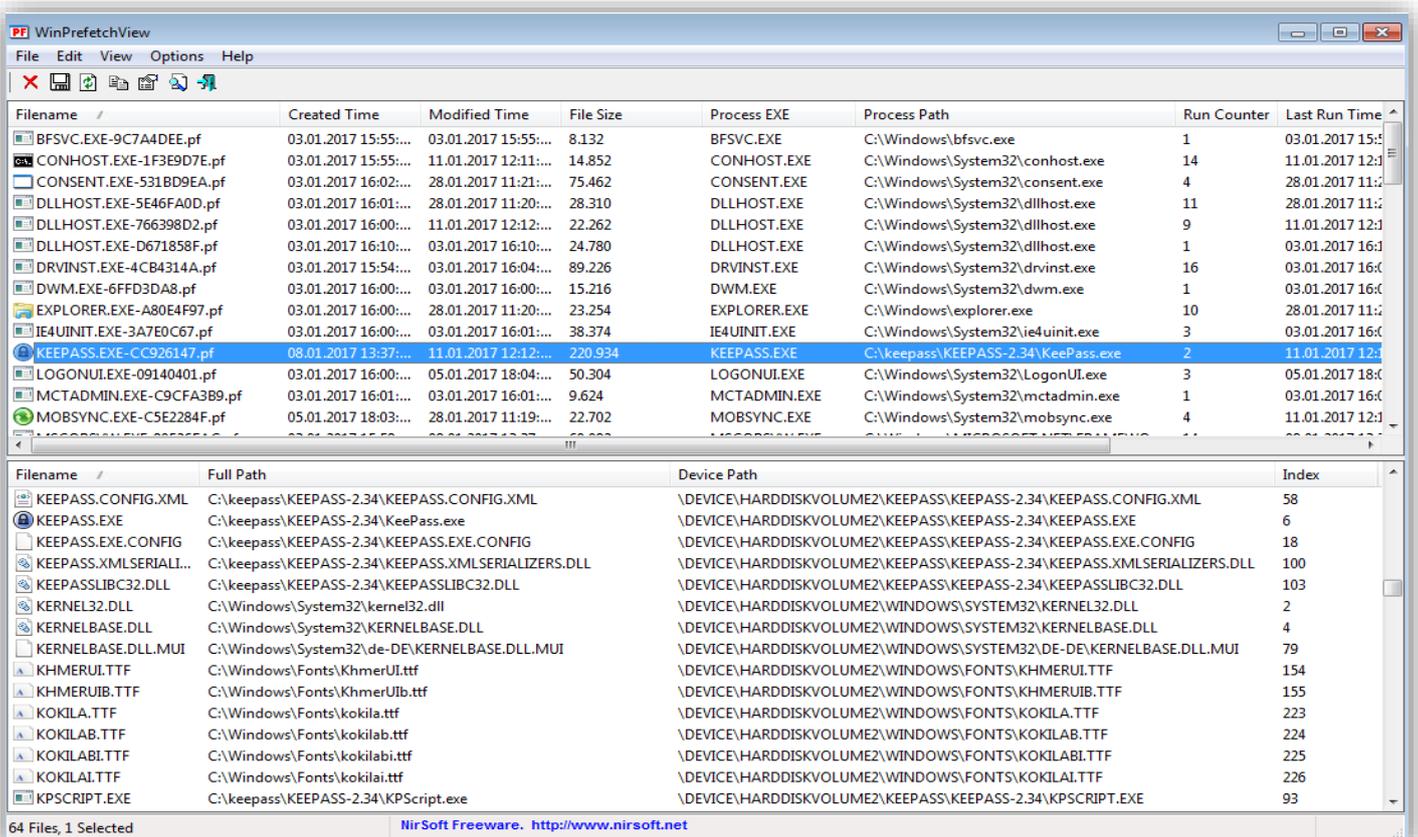


Abbildung 7: Inhalt der Prefetch-Datei von KeePass - Teil 1

<sup>7</sup> Mehr Informationen unter [http://www.nirsoft.net/utills/win\\_prefetch\\_view.html](http://www.nirsoft.net/utills/win_prefetch_view.html)

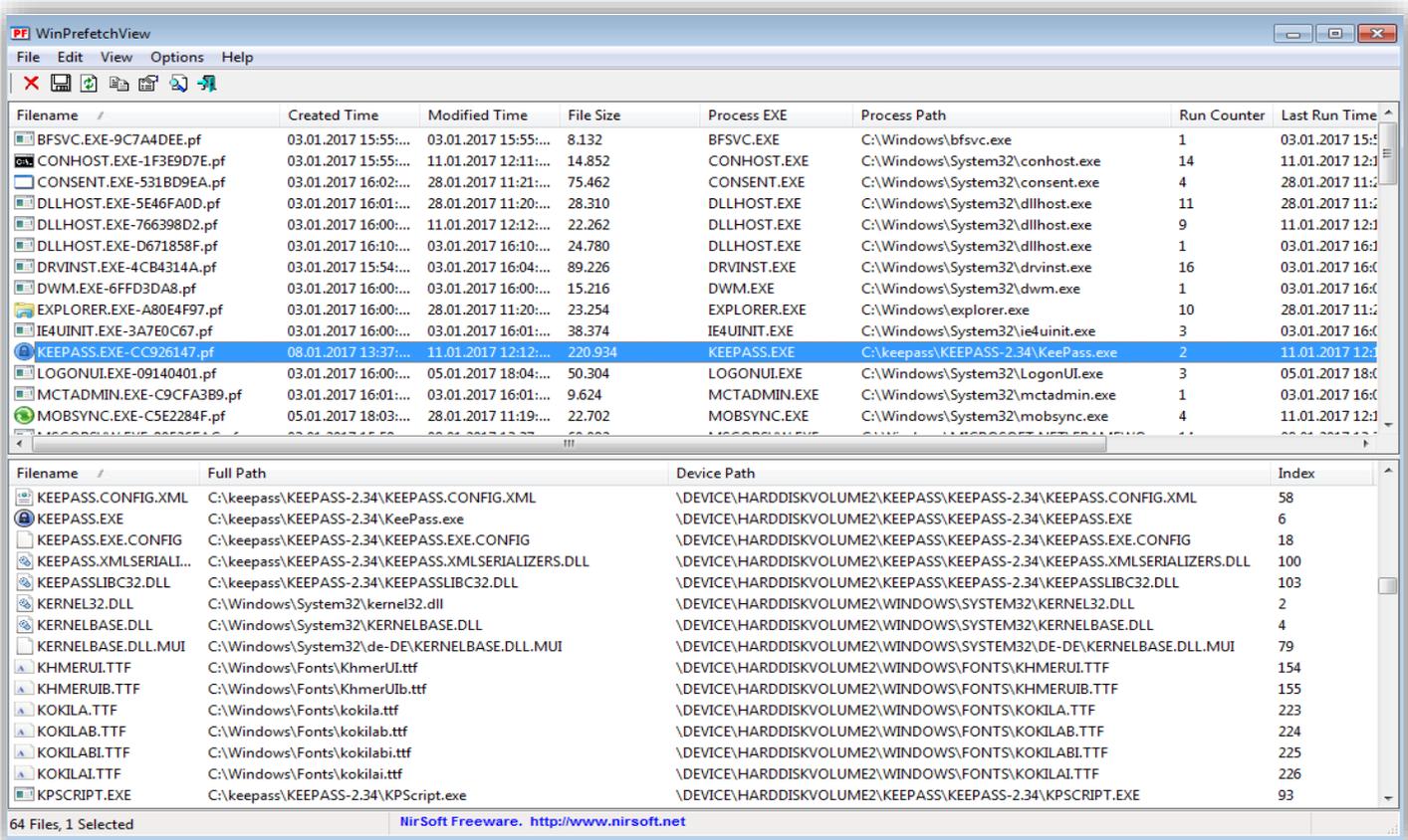


Abbildung 8: Inhalt der Prefetch-Datei von KeePass - Teil 2

### 2.3.4. Dateianalyse

#### database.kdbx.lnk

Eine Überprüfung der Verknüpfungsdatei database.kdbx.lnk erbrachte, dass in dieser lesbare Strings enthalten sind, welche neben dem Dateinamen auch nach die Pfadangabe zum Speicherort der Datei in relativer und absoluter Form enthalten sind.

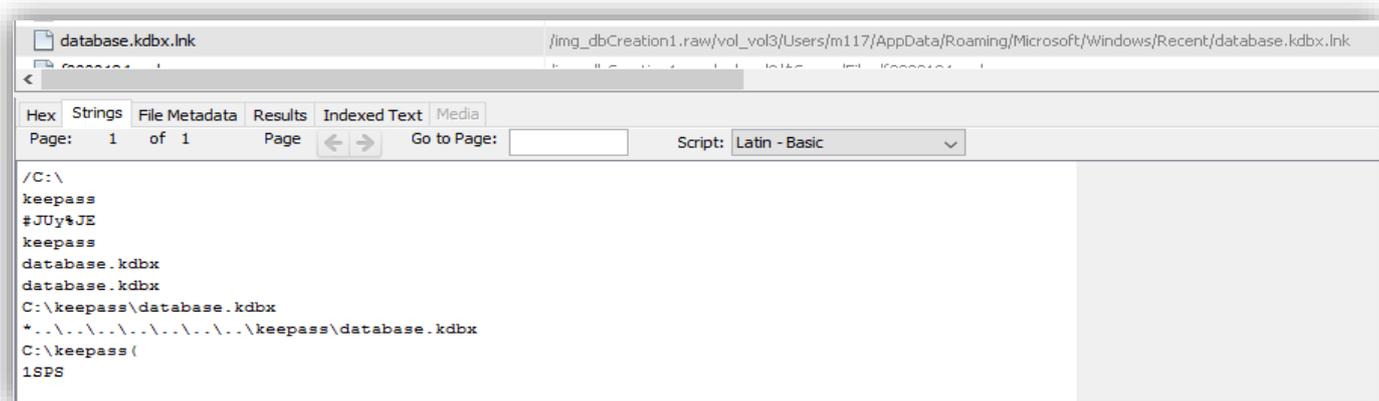


Abbildung 9: Inhalt der database.kdbx.lnk Datei

#### GDIPFONTCACHEV1.DAT

Eine Analyse dieser Datei mit Hilfe von Autopsy erbrachte, dass in dieser die jeweiligen Dateipfade zu den **.TTF-Dateien** der jeweiligen Schriftarten wie z.B. ARIAL enthalten sind. Eine Internetrecherche erbrachte, dass es sich bei dieser Datei tatsächlich um einen „Fontcache“ handelt, welcher automatisch von

Programmen erstellt wird, welche das .NET-Framework der Fa. Microsoft verwenden<sup>8</sup>. Laut der Webseite keepass.info benötigt die Version 2.x von KeePass unter Windows mindestens eine Version  $\geq 2.0$  vom Microsoft .NET Framework<sup>9</sup>.

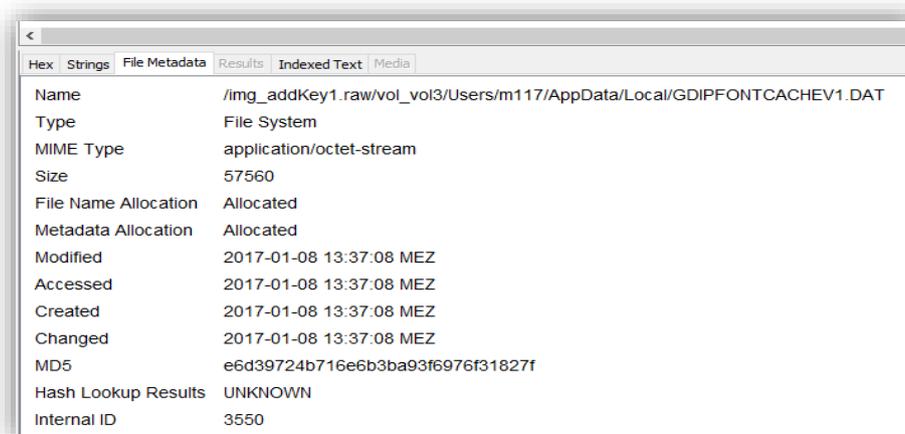


Abbildung 11: GDIPFONTCACHEV1.DAT - Metadaten

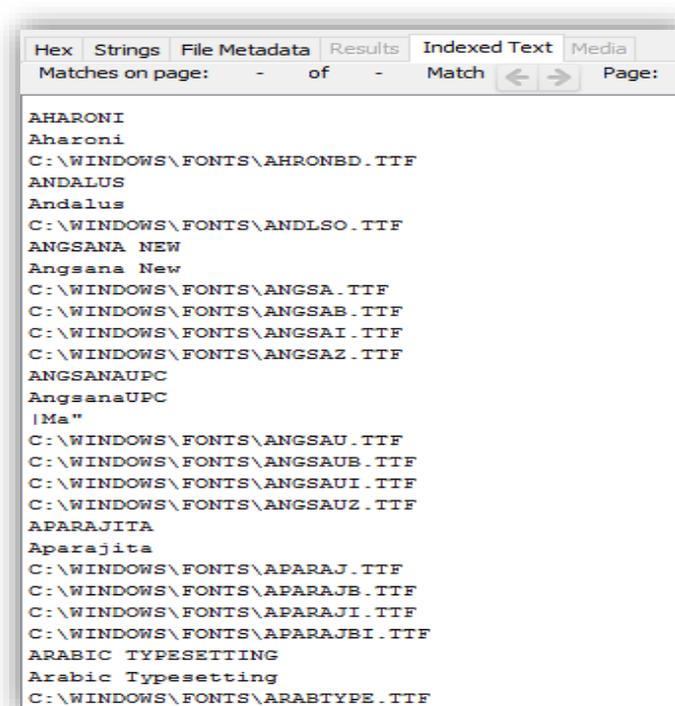


Abbildung 10: Inhalt von GDIPFONTCACHEV1.DAT

### **Filecarving der .kdbx-Dateien**

Durch die Verwendung Filecarving-Programms Photorec, welches Bestandteil der Forensik Software Autopsy ist, konnte bei der Untersuchung der Festplattenabbildern der einzelnen Aktionen mehrere angebliche .kdbx-Dateien wiederhergestellt werden. Der Autor vermutete, dass es sich bei den wiederhergestellten Dateien um ältere Versionen der Datenbank-Datei database.kdbx handelt, welche nach einer Änderung des Datenbankinhalts z.B. Hinzufügen eines Eintrags, nicht mehr verwendet wurde. Diese Vermutung basierte vor allem aus den Erkenntnissen der Anwendungsanalyse, welche gezeigt hatte, dass

<sup>8</sup> Gefunden auf <https://www.sevenforums.com/general-discussion/101631-what-gdipfontcachev1-dat.html>

<sup>9</sup> Siehe [http://keepass.info/help/base/faq\\_tech.html#sysreq2x](http://keepass.info/help/base/faq_tech.html#sysreq2x)

KeePass einer temporären Version der Datenbank-Datei verwaltet und dort die vom User vorgenommenen Änderungen speichert. Diese temporäre Datenbank-Datei wird nachdem Speichern durch den Benutzer dann zur neuen Datenbank-Datei.

Das automatische Carving der .kdbx-Dateien mittels Photorec ist jedoch fehlerhaft, so wurde zwar eine eindeutige Headersignatur immer richtig gefunden, jedoch wurde keine „eindeutige“ Footersignatur gefunden.

```
0x00000000: 03 D9 A2 9A 67 FB 4B B5 01 00 03 00 02 10 00 31 ....g.K.....1
0x00000010: C1 F2 E6 BF 71 43 50 BE 58 05 21 6A FC 5A FF 03 ....qCP.X.!j.Z..
0x00000020: 04 00 01 00 00 00 04 20 00 4B DC 2D 4A 81 BE 71 ..... .K.-J..q
```

Abbildung 12: Headersignatur der .kdbx Datei

Was dazu führte, dass die mittels Photorec wiederhergestellten Dateien viel zu groß waren.

Dateinamen	Dateigröße	Anmerkung:
database.kdbx	2142	letzte und aktuellste Version
f0003832.kdbx	45056	vorletzte Version
f0033216.kdbx	120864768	älteste Version

File	File Path	Comment	Modified Time	Changed Time	Accessed Time
GDIPFONTCACHE1.DAT	/img_addKey1.raw/vol_vol3/Users/m117/AppData/Local/G...		2017-01-08 13:37:08 MEZ	2017-01-08 13:37:08 MEZ	2017-01-08 13:37:08 MEZ
KEEPASS.EXE-CC926147.pf	/img_addKey1.raw/vol_vol3/Windows/Prefetch/KEEPASS.E...		2017-01-11 12:41:45 MEZ	2017-01-11 12:41:45 MEZ	2017-01-08 13:37:11 MEZ
f0033032.kdbx	/img_addKey1.raw/vol_vol3/\$CarvedFiles/f0033032.kdbx		0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00
database.kdbx	/img_addKey1.raw/vol_vol3/keepass/database.kdbx		2017-01-11 13:11:56 MEZ	2017-01-11 13:11:56 MEZ	2017-01-08 14:05:05 MEZ

Hex	Strings	File Metadata	Results	Indexed Text	Media
Name		/img_addKey1.raw/vol_vol3/\$CarvedFiles/f0033032.kdbx			
Type		Carved			
MIME Type		application/octet-stream			
Size		120889344			
File Name Allocation		Unallocated			
Metadata Allocation		Unallocated			
Modified		0000-00-00 00:00:00			
Accessed		0000-00-00 00:00:00			
Created		0000-00-00 00:00:00			
Changed		0000-00-00 00:00:00			
MD5		81906f6e63547929f603b9f33afc855			
Hash Lookup Results		UNKNOWN			
Internal ID		118878			

Abbildung 13: Filecarving - wiederhergestellte .kdbx Datei 1

File	File Path	Comment	Modified Time	Changed Time	Accessed Time
f0033216.kdbx	/img_deleteKey1.raw/vol_vol3/\$CarvedFiles/f0033216.kdbx		0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00
f0003832.kdbx	/img_deleteKey1.raw/vol_vol3/\$CarvedFiles/f0003832.kdbx		0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00
database.kdbx	/img_deleteKey1.raw/vol_vol3/keepass/database.kdbx		2017-01-11 13:52:00 MEZ	2017-01-11 13:52:00 MEZ	2017-01-08 14:05:05 MEZ

Hex	Strings	File Metadata	Results	Indexed Text	Media
Name		/img_deleteKey1.raw/vol_vol3/\$CarvedFiles/f0033216.kdbx			
Type		Carved			
MIME Type		application/octet-stream			
Size		120864768			
File Name Allocation		Unallocated			
Metadata Allocation		Unallocated			
Modified		0000-00-00 00:00:00			
Accessed		0000-00-00 00:00:00			
Created		0000-00-00 00:00:00			
Changed		0000-00-00 00:00:00			
MD5		f6b636d3e6e80d85ddda8ad2851cf35d			
Hash Lookup Results		UNKNOWN			
Internal ID		118879			

Abbildung 14: Filecarving - wiederhergestellte .kdbx Datei 2

Der Autor hat dann die entsprechenden Hexwerte der Dateien in Autopsy näher betrachtet und festgestellt, dass bei den wiederhergestellten Dateien ab dem Bereich 0x000008c0 beziehungsweise 0x00000820 größere Bereiche mit Nullen folgten. Daher vermutete der Autor, dass es sich hierbei um die Dateienden handelt könnte. Der Autor verwendete anschließend das Unix-Programm **dd** um die „wiederhergestellten“ Dateien zu „zuschneiden“. Es wurde die folgenden Konsolenbefehle verwendet:

```
$ dd if=118877-f0003832.kdbx of=carved_database.kdbx bs=1 count=2238
$ dd if=118879-f0033216.kdbx of=carved_database2.kdbx bs=1 count=2078
```

Die zugeschnittenen Dateien konnten dann mit KeePass unter Eingabe des Passwortes geöffnet werden. Dabei handelte es sich tatsächlich um ältere Versionen der Datenbank-Datei.

The screenshot displays the Autopsy interface. At the top, there's a 'Directory Listing' section with a table of files. Below that, the 'Hex' view is active, showing a hex dump of the selected file. The hex dump shows the file content starting with 'm..u..\$.q...b...' and ending with a large block of zeros from offset 0x000008c0 onwards.

File	File Path	Comment	Modified Time	Changed Time
f0033216.kdbx	/img_deleteKey1.raw/vol_vol3/\$CarvedFiles/f0033216.kdbx		0000-00-00 00:00:00	0000-00-00 00:00:00
f0003832.kdbx	/img_deleteKey1.raw/vol_vol3/\$CarvedFiles/f0003832.kdbx		0000-00-00 00:00:00	0000-00-00 00:00:00
database.kdbx	/img_deleteKey1.raw/vol_vol3/keepass/database.kdbx		2017-01-11 13:52:00 MEZ	2017-01-11 13:52:00 MEZ

Hex Editor View:

```

Page: 1 of 3 Page: < > Go to Page: 1 Jump to Offset 0

0x000007f0: 6D D0 A8 75 06 0D 24 E0 71 D5 BD B8 62 89 FE F5 m..u..$.q...b...
0x00000800: 31 CF 4C E0 6E 1E C6 AA 66 2E B1 F7 82 90 CF 05 1.L.n...f.....
0x00000810: 1B 11 C0 C1 AB 7C C1 2F 59 87 83 A6 32 C9 EE AE .....|/Y...2...
0x00000820: 95 83 FD C2 3B 64 21 D1 E8 D9 58 C4 E7 8E 8E 16 ...;d!...X....
0x00000830: 76 6F 3E A9 AC E9 29 7D 48 9B A1 96 E8 5C 26 5E vo>...)}H....\&^
0x00000840: 40 72 4E A6 7F 5F 12 F8 A1 CB DC 48 DB 02 F9 59 @rN.....H...Y
0x00000850: 69 07 3D BE 1B 45 E3 20 DD 0F 24 C4 2E 70 84 90 i...E...$.p...
0x00000860: 27 98 BA D6 C8 54 EF 8D FE 8F 7C 0F C2 3D 46 27 '...T...|...=F'
0x00000870: 00 9A 9C EC 3F 33 B6 48 E6 71 8A CE 01 8B 59 1F .....?3.H.q....Y.
0x00000880: FA 6E 93 FF AF 22 4D 87 1F 22 6D B7 DD B5 84 E5 .n... "M... "m.....
0x00000890: D2 F6 98 18 20 4B FF 36 91 31 3C EA 19 9A 81 92 ... K.6.1<.....
0x000008a0: BA ED 5E 2D BD 10 C7 52 83 EE 41 2F 34 E5 B1 17 ...^...R...A/4...
0x000008b0: 0E 92 D3 42 DD FC BE 90 79 30 C3 B5 AD 59 00 00 ...B...y0...Y..
0x000008c0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x000008d0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x000008e0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x000008f0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000900: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000910: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000920: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000930: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000940: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000950: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000960: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000970: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000980: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000990: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x000009a0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x000009b0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x000009c0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x000009d0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x000009e0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x000009f0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000a00: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000a10: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000a20: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000a30: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000a40: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000a50: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000a60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000a70: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000a80: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000a90: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000aa0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000ab0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000ac0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000ad0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....

```

Abbildung 15: Dateiende von wiederhergestellten .kdbx Datei 1

Directory Listing

Bookmark File Tags

Table Thumbnail

File	File Path	Comment	Modified Time	Changed Time
f0033216.kdbx	/img_deleteKey1.raw/vol_vol3/\$CarvedFiles/f0033216.kdbx		0000-00-00 00:00:00	0000-00-00 00:00:00
f0003832.kdbx	/img_deleteKey1.raw/vol_vol3/\$CarvedFiles/f0003832.kdbx		0000-00-00 00:00:00	0000-00-00 00:00:00
database.kdbx	/img_deleteKey1.raw/vol_vol3/keepass/database.kdbx		2017-01-11 13:52:00 MEZ	2017-01-11 13:52:00 MEZ

---

Hex Strings File Metadata Results Indexed Text Media

Page: 1 of 7377 Page Go to Page: 1 Jump to Offset 0

0x00000710:	96 FC EF 13	6D DA 30 4F	76 2B F7 86	1A 2E A6 39	....m.00v+....9
0x00000720:	80 B5 50 E3	6A 0E 56 69	6F 4B 62 4C	AF AE 9C 67	..P.j.VioKbL...g
0x00000730:	C9 DD CA FA	E6 A7 DA 39	3C 79 AB FD	0D AE 69 F8	.....9<y....i
0x00000740:	01 70 BF 5E	6C CA FA 2E	80 40 F9 62	93 9E 31 50	..p.^1....@.b...1P
0x00000750:	7F 8C 4C A6	97 34 54 A6	E2 72 43 10	A0 28 FD 14	..L..4T...rC...()
0x00000760:	EA 81 F2 64	94 7E B0 FA	F9 BA D2 DC	F5 80 C4 3E	...d.~...>
0x00000770:	2F 17 87 2F	F8 D1 06 D3	58 81 C3 74	17 1E FD 3D	/.../...X...t...=
0x00000780:	4E CC 91 74	D9 1C A7 6D	4C 62 27 E6	96 A4 D5 A9	N...t...m.b'.....
0x00000790:	D8 D6 56 77	5C A2 C3 21	57 BF EA A5	07 CD 09 06	..Vw\...!W.....
0x000007a0:	D2 FE F8 FC	38 89 3D 48	1A FE B1 FD	D9 C5 2A 7D	....8.=H.....*
0x000007b0:	58 A2 58 AF	96 5F F8 BE	E9 E0 76 52	75 16 85 62	X.X...vRu...b
0x000007c0:	0C 57 29 FD	B8 91 B0 8D	D7 3C C9 DB	24 DD 5B 59	.W).....<...\$.{Y
0x000007d0:	42 FC 37 A6	9C 25 4A 22	CF 66 7A 04	16 A4 D4 74	B.7...%J".fz...t
0x000007e0:	ED 3C 5F 3C	2C 64 01 07	12 8E 74 E9	AF 4A 7F EA	..<_<.d...t...J..
0x000007f0:	A9 25 DE A3	FE D0 09 CD	0E EA F1 BA	58 53 02 3D	..%.....XS.=
0x00000800:	84 74 18 03	9B BE C2 C3	D3 78 4E B9	A0 A7 BB 4D	..t.....xN....M
0x00000810:	3C 35 08 C6	DB 88 3D 65	72 E8 47 67	80 9B 00 00	<5....=er.Gg....
0x00000820:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x00000830:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x00000840:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x00000850:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x00000860:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x00000870:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x00000880:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x00000890:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x000008a0:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x000008b0:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x000008c0:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x000008d0:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x000008e0:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x000008f0:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x00000900:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x00000910:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x00000920:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x00000930:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x00000940:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x00000950:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x00000960:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x00000970:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x00000980:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x00000990:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x000009a0:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x000009b0:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x000009c0:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x000009d0:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x000009e0:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....
0x000009f0:	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	.....

Abbildung 16:Dateiende von wiederhergestellten .kdbx Datei 2

database.kdbx - KeePass

File Edit View Tools Help

Title	User Name	Password	URL	Notes
Sample En...	User Name	*****	http://keepa...	Notes
Sample En...	Michael321	*****	http://keepa...	

0 of 2 selected Ready.

Abbildung 17: geöffnete database.kdbx Datei

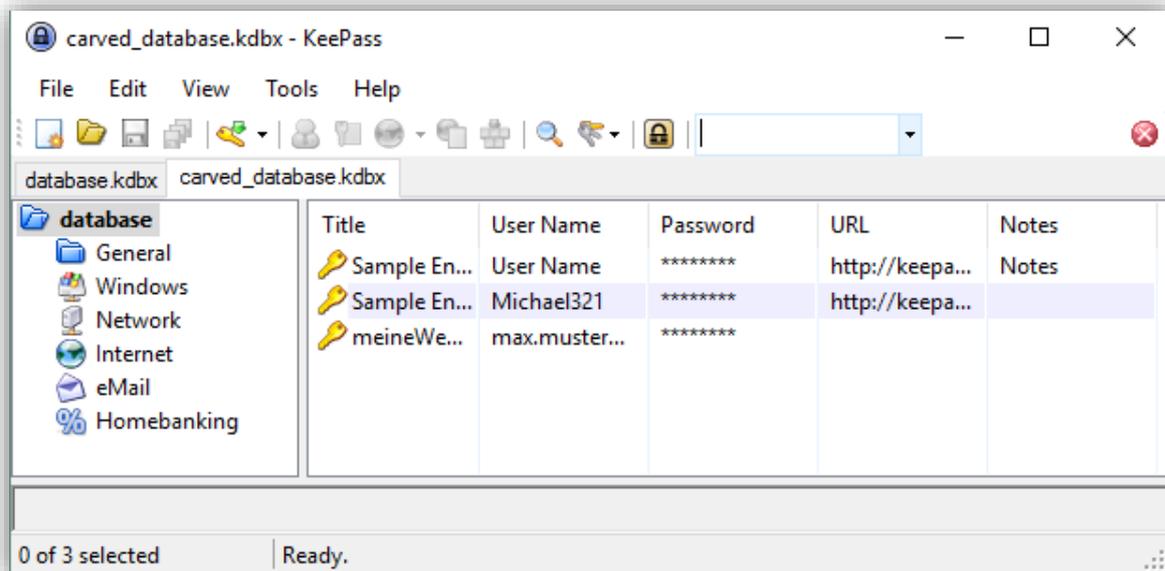


Abbildung 138: geöffnete wiederhergestellte .kdbx Datei 1

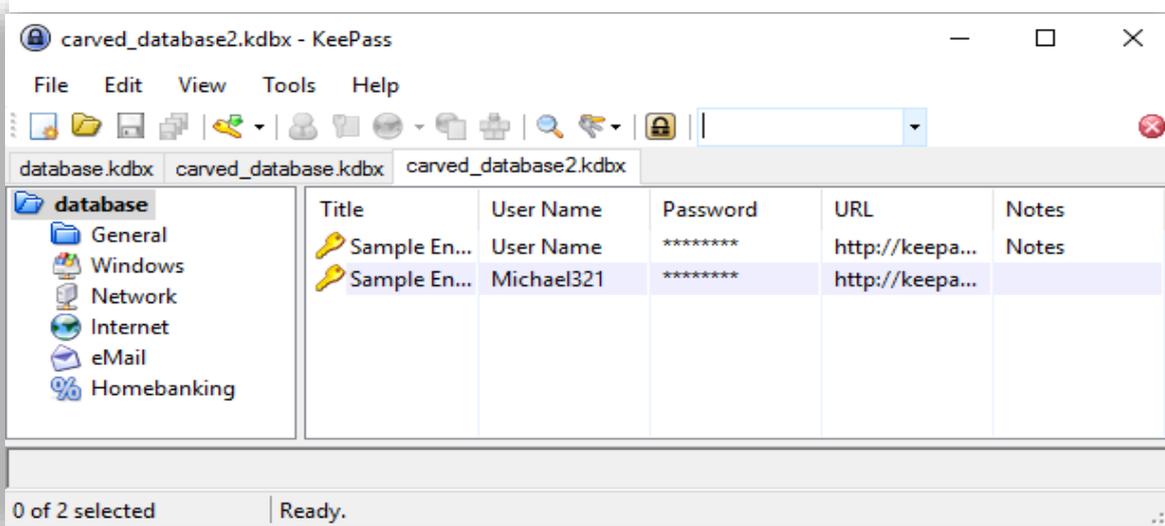


Abbildung 19: geöffnete wiederhergestellte .kdbx Datei 2

### 2.3.5. Netzwerkspuren

Die Überprüfung des ausgezeichneten Netzwerkverkehrs zeigte nur im Fall des Startens von KeePass mit Option „Automatic Update Check enabled“ einen Netzwerkverkehr zum Webserver **keepass.info** (IP-Adresse: 46.252.18.237). Bei allen anderen Aktionen könnte kein Netzwerkverkehr festgestellt werden. Die Auswertung des festgestellten Netzwerkverkehrs zeigte, dass aufgrund einer Anforderung mittels „GET / http/1.1.“ eine Kopie der HTML-Datei von keepass.info an den anfragten Client-Rechner übersendet wird. Daraus wird dann durch das Programm die aktuellste Version extrahiert u. mit der verwendeten Version von KeePass verglichen. Falls ein Unterschied und somit eine neuere Version festgesellt wird, erscheint im KeePass-Programm ein neues Fenster mit der Überschrift „Update Check“.

## 3. Zusammenfassung

Die in diesem Dokument genannten Ergebnisse der Anwendungsanalyse haben gezeigt, dass sich Spuren über die Verwendung der portablen Version von KeePass sowohl im Dateisystem, in der Registry als auch in den Prefetch-Dateien finden lassen. Die Anwendung inkl. aller notwendigen Dateien, als auch der Datenbank-Datei beziehungsweise Key-Datei mögen sich zwar auf einem externen Datenträger abspeichern

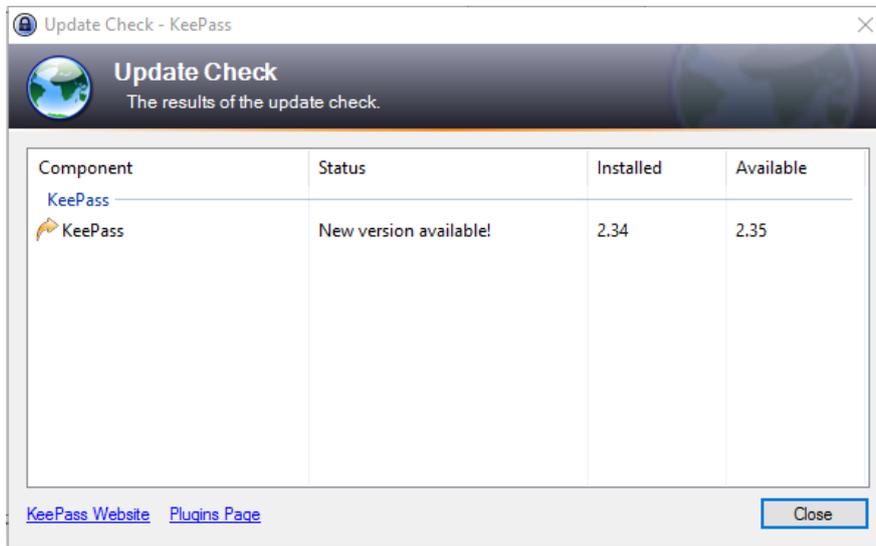


Abbildung 20: Update Check

lassen. Die Spuren (z.B. Prefetch, Registry-Einträge), welche jedoch automatisch durch das Windows-Betriebssystem für die Anwendung entstehen, lassen sich aber kaum vermeiden.

Bei der Präsentation der festgestellten Erkenntnisse handelt es sich um die persönliche Auswahl des Autors. Es wurde versucht sich auch im Hinblick auf den vorgegeben Umfang der Arbeit auf die signifikantesten Spuren zu konzentrieren.

Aufgrund des vorgegebenen Umfangs der Arbeit wurde seitens des Unterzeichners verzichtet weitere Analysen wie z.B. Untersuchung des Hauptspeichers, Auswertung des Journals des NTFS-Dateisystems, der Windows Event Logs und Reverse Engineering der KeePass.exe verzichtet. Diese Analysen könnten Teil einer weitergehenden Arbeit sein.