Smartphone-Forensik

Marko Schuba, Hans Höfken, Thomas Schaefer

32 GB an Daten – Datenmengen, die Ermittler vor Jahren nur auf PCs vorfanden, passen heute mit dem Smartphone bequem in die Hosentasche. Und die Inhalte sind zum Teil lohnend. Neben Emails und Office-Dokumenten finden sich Fotos, Filme, Kontakte, Anruflisten, Kurznachrichten und Geodaten: alles potentielle Beweismittel im Rahmen einer Untersuchung. Von den Daten der Facebook-App ganz zu schweigen.

IN DIESEM ARTIKEL ERFAHREN SIE...

- welche speziellen Probleme die forensische Untersuchung von Smartphones mit sich bringt.
- anhand eines Fallbeispiels zu Windows Phone 7, wie man neue Smartphones analysierbar macht und welche Daten man findet.

WAS SIE VORHER WISSEN SOLLTEN...

- Sie sollten Grundkenntnisse der IT-Forensik mitbringen, wie das gerichtsverwertbare Sichern von Daten und die Funktionsweise forensischer Tools. Zudem sollten Sie wissen, wie Sicherheitsmechanismen auf IT-Systemen funktionieren, da Smartphone-Hersteller den Zugriff auf die Daten üblicherweise schützen.

Der Weg, forensisch korrekt an diese Daten heran zu kommen, unterscheidet sich jedoch grundlegend von der klassischen „Festplatten-Forensik“. In diesem Artikel, wird das Abenteuer Smartphone-Forensik genauer unter die Lupe genommen.

Der Smartphone–Markt boomt. Laut IDC [1] wuchs der weltweite Absatz im 3. Quartal 2011 im Vergleich zum Vorjahr um 42.6%. Insgesamt wurden 118,1 Millionen Smartphones ausgeliefert, im 3. Quartal 2010 nur 82,8 Millionen. Bei den Betriebssystemen wird der Markt derzeit von vier Herstellern dominiert: Google mit Android gefolgt von Apple mit iOS, dann Nokias Symbian und Research in Motion mit Blackberry OS. Nokia hat darüber hinaus kürzlich erste Telefone mit dem Microsoft Betriebssystem Windows Phone 7 herausgebracht, weshalb dieses System wohl in naher Zukunft seine Marktanteile (zu Lasten von Symbian) vergrößern dürfte.

Was Smartphones von herkömmlichen Handys unterscheidet ist ihre große Funktionalität. Smartphones sind nicht mehr für die Telefonie sondern für eine Breite Nutzung von Anwendungen optimiert. Deutlich wird dies durch die Benutzerschnittstelle: sie besteht aus einem hochauflösenden Multitouch-Display mit intuitiver Bedienung. Auf der leistungsfähigen Hardware läuft ein komplettes Betriebssystem mit offenen Schnittstellen, die es Benutzern erlaubt, Apps von Drittanbietern zu installieren und nutzen. Netzwerktechnisch bieten Smartphones eine Reihe von Optionen, z.B. WLAN und Bluetooth (für lokalen, kabellosen Zugriff), GSM, UMTS, HSDPA usw. (für den mobilen Zugriff unterwegs) und USB (z.B. für die Synchronisation mit dem PC). Smartphones verfügen über eine Reihe von Sensoren, welche Bewegungen, Lage, Magnetfelder, Licht oder Objekte in der Nähe erkennen. Ein wichtige Komponente ist dabei der in den meisten Smartphones vorhandenen GPS-Empfänger, der eine genaue Positionsbestimmung ermöglicht, die in Bereichen schlechter Satellitenabdeckung zusätzlich durch WLAN- bzw. Mobilfunk-Ortung ergänzt wird. Die Kombination dieses großen Funktionsumfangs mit preisgünstigen Daten-Flatrates, wie sie heutzutage von dem meisten Netzbetreibern angeboten werden, hat zu einer schier unendlichen Anzahl von Smartphone-Apps geführt. Abgesehen von Programmen, die man von PCs her kennt, wurden hier Anwendungen entwickelt, die speziellen Fähigkeiten der Telefone optimal ausnutzen. So nutzen Spiele die Bewegungssensoren und das Touchscreen von Smartphones aus und ermöglichen so ein ganz neues Spielgefühl. Navigationssoftware, die früher eine spezielle Hardware erforderten, können nun einfach entwickelt werden, weil das Tele-

fon seine Position und Bewegungsrichtung kennt und Kartenmaterial online verfügbar ist.

Herausforderungen der Smartphone-Forensik

Mit der Fülle der Einsatzmöglichkeiten steigt auch die Menge der Informationen, die für digitale Ermittler von Interesse sein könnten. Lieferten Handys üblicherweise Daten wie Anruflisten, SMS-Nachrichten, Kontakte oder Fotos, ergeben sich bei Smartphones eine Flut an weiteren potentiellen Beweisen. Dazu gehören insbesondere Geodaten und Daten verschiedenster Anwendungen, wie Chat, Emails oder Office-Dokumente. Die Extraktion und Analyse dieser Daten stellt die Ermittler aber vor eine Reihe von Herausforderungen.

Isolation des Geräts: Wird ein IT-System beschlagnahmt, so wird es üblicherweise zur Untersuchung in ein forensisches Labor transportiert. Bei Computern erfolgt dieser Transport meist in abgeschaltetem Zustand. Zur weiteren Analyse wird die Festplatte ausgebaut, kopiert und forensische Untersuchungen werden anschließend auf dem Duplikat durchgeführt. Bei Mobiltelefonen ergibt sich hier ein Problem: wird die Stromversorgung des Geräts unterbrochen, lässt es sich ohne Kenntnis des PINs der passenden SIM-Karte nicht mehr starten. Der PIN lässt sich zwar beim Netzwerkbetreiber erfragen, allerdings kostet dies Zeit und ist auch nur dann ohne weiteres möglich, wenn der Betreiber im gleichen Land sitzt wie die Ermittler. Aus diesen Gründen bevorzugen es Ermittler, die Geräte in eingeschaltetem Zustand ins Labor zu bringen. Aber auch der eingeschaltete Modus birgt Nachteile. Der schlimmste Fall wäre ein „Remote Wipe“ des Geräts, bei dem sämtliche Daten über ein Funksignal gelöscht würden. Viele Smartphones bieten diese Funktionalität z.B. für verlorene Telefone an. Aber auch ganz normale Funkaktivitäten wie *Location Updates* oder eingehende Anrufe oder Nachrichten können das Beweismittel verändern. Deshalb müssen beschlagnahmte, eingeschaltete Geräte funktechnisch isoliert werden, bis sie in einem abgeschirmten Raum weiter untersucht werden können. Verschiede Hersteller bieten dazu Lösungen an, z.B. Beutel aus Metallgewebe. Der resultierende Farradaysche Käfig soll die Funksignale des Telefons unterbinden. Mit einer zusätzlichen Batterie im Beutel kann sichergestellt werden, dass der Handy-Akku auch einen längeren Transport übersteht, auch wenn das Handy die Sendeleistung hochschraubt (es versucht ja weiterhin Basisstationen zu erreichen). Kürzlich vorgestellte Forschungsergebnisse der Purdue University [2] haben jedoch gezeigt, dass diese Lösungen keineswegs so abschirmsicher sind, wie angenommen. Es ist deshalb Vorsicht geboten!

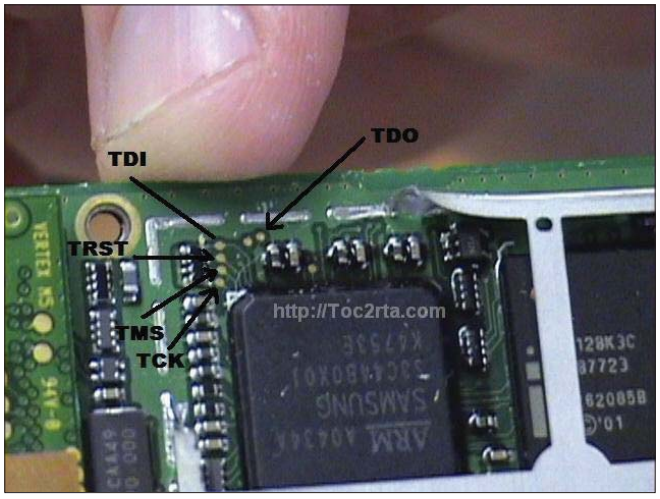


Abbildung 1. *JTAG-Schnittstelle (Quelle: http://Toc2rta.com)*

Gerätevielfalt: Es gibt sehr viele sehr unterschiedliche Geräte. Und ständig kommen neue hinzu. Ein Beispiel: Einer der führenden Tool-Hersteller unterstützt

Abbildung 2. *Chat-App (Quelle: http://www.whatsapp.com/)*

mit seiner letzten Release mehr als 7000 Geräteprofile [3]. Auch wenn dies nicht alles Smartphones sind, zeigt es, auf wie viele unterschiedliche Handy-Typen sich Ermittler einstellen müssen. Smartphones unterscheiden sich z.B. hinsichtlich Hersteller, Plattform, Betriebssystem oder Stecker-Typen. Für den Ermittler bedeutet dies, das je nach Smartphone für die Analyse unterschiedliche Kabel und Extraktionsmethoden anzuwenden sind. Dass eine Methode bei einem Gerät funktioniert bedeutet dabei nicht, dass sie nach einem Firmware-Upgrade des Telefons immer noch benutzt werden kann.

Datenextraktion: Wie bekommt man die Daten aus dem Speicher des Smartphones heraus? Der Ausbau des Flashspeichers analog zum Ausbau einer Festplatte ist möglich, bedeutet aber die Zerstörung des Telefons und manchmal auch der Daten. Der Grund liegt darin, dass der Chip aus dem Telefon heraus gelötet werden muss. Die dabei entstehenden hohen Temperaturen lösen oft nicht nur das Lötzinn, sondern auch die Daten des Chips auf. Hat der Flashspeicher die Prozedur überstanden, können die Daten recht einfach und komplett vom Chip gelesen werden. Man spricht dabei von einem physikalischen Abbild des Speichers. Möchte oder kann man den Flashspeicher nicht auslöten, bleibt nur die Möglichkeit über die Schnittstellen des Telefons auf die Daten zuzugreifen. Eine Hardware-nahe Möglichkeit ist die JTAG-Schnittstelle, welche bei manchen Platinen existiert und zum Testen und Debuggen der elektronischen Hardware eingesetzt wird. Leider ist diese Schnittstelle nicht immer vorhanden oder dokumentiert, weshalb diese Option manchmal ebenfalls entfällt. Bleibt nur noch der Zugriff über die normalen Schnittstellen des Telefons, z.B. mittels USB. Hier unterscheidet man zwischen der logischen Extraktion (möglichst viele Daten werden über bekannte Schnittstellen wie AT Kommandos, OBEX o.ä. ausgelesen), der Dateisystem-Extraktion (mehr Daten als bei der logischen Analyse, da gesamtes Dateisystem ausgelesen wird) bzw. der physikalischen Extraktion (komplettes Speicherabbild) unterschieden. Die physikalische Extraktion liefert die umfangreichsten Ergebnisse ist aber nicht bei jedem Telefon möglich, da entsprechende Schutzmaßnahmen des Herstellers umgangen werden müssen, was man z.B. durch modifizierte Bootloader oder Ausnutzen von Schwachstellen erreicht.

Datendekodierung: Liegen das Dateisystem oder das physikalische Speicherabbild vor, so ist der Ermittler noch nicht fertig: die Daten müssen dekodiert werden, bevor Sie überhaupt angesehen werden können. Schwierigkeiten ergeben sich hier durch die Vielzahl der meist proprietären und nicht dokumentierten Dateisysteme. Reverse Engineering ist dabei häufig die einzige Methode die zum Ziel führt.

Verschlüsselung: Wie bei der Festplatten-Forensik stellt auch bei Smartphones die Verschlüsselung der Daten die Ermittler immer häufiger vor Probleme. Beispielsweise wird bei iOS ab Version 4 der Speicher komplett verschlüsselt (AES 256 Bit), wobei die Verschlüsselung durch den Passcode des Geräts geschützt ist. Ohne Kenntnis des Passcodes ist eine Analyse der Daten nicht möglich. Zwar existieren für das iOS erste Ansätze, auch auf Passcode-geschützte Daten zuzugreifen [4], allerdings funktioniert der Ansatz nur für die aus 4 Zahlen bestehenden, einfachen Passcodes.

Apps: Eine weitere Herausforderung der Smartphone-Forensik ist die Vielzahl der Apps. Immer mehr wichtige Daten werden nicht an den Standardorten, son-

Abbildung 3. *Cellebrite UFED (Quelle: www.cellebrite.com)*

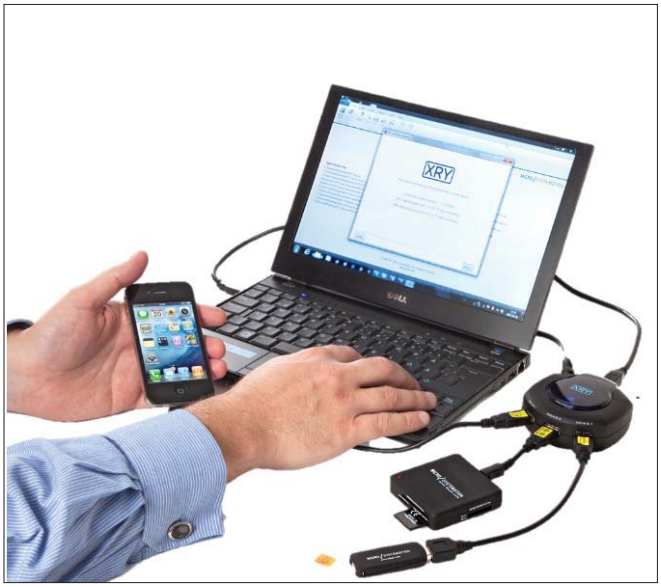


Abbildung 4. *Micro Systemation XRY (Quelle: www.msab.se)*

dern den Speicherbereichen der Apps gespeichert. So wird SMS-Kommunikation seit einiger Zeit zunehmend durch Chat-Apps oder Apps sozialer Netze ersetzt (wie z.B. WhatsApp im Bild unten). Um Spuren dieser Nachrichten zu finden, müssen die Daten der App gefunden, ausgelesen und dekodiert werden. Ein weiterer, wichtiger Informationstyp, der von Apps gespeichert wird sind Geodaten. Viele Apps (Wetter, Soziale Netze, Jogging, Navigation etc.) speichern Ortsinformationen ab. Wie können diese Geodaten einfach von Ermittlern gefunden und herausgefiltert werden?

Existierende Tools

Es gibt eine Reihe von Tools, die für die forensische Analyse von Smartphones eingesetzt werden können. Hier wird nur kurz und ohne weitere Bewertung ein Überblick über die bekanntesten kommerziellen Werkzeuge sowie zwei an deutschen Hochschulen entwickelten Tools gegeben werden. Weiterführende Informationen finden sich in den angegebenen Quellen.

ADEL: Der Android Data Extractor Lite (ADEL) wurde an der Universität Erlangen-Nürnberg entwickelt und spezialisiert sich auf die Auswertung von SQLite-Datenbanken von Android-Smartphones [5].

AFT: Das Android Forensic Toolkit (AFT) der FH Aachen ermöglicht das Rooten von Android-Geräten und die Auswertung von Anwendungsdaten [6]. Fokus ist dabei die Analyse und grafische Auswertung von Geodaten.

Cellebrite UFED: Ein sehr weit verbreitetes Tool, welches laut Herstellerangaben über 7000 Geräteprofile unterstützt (logische, Dateisystem- und physikalische Extraktion), ist UFED des Herstellers Cellebrite [7] (siehe auch Abbildung 3).

Compelson MOBILedit!: Ebenfalls häufig benutzt wird das Werkzeug MOBILedit!. Es unterstützt die Betriebssysteme Android, Apple iOS, Blackberry und Windows Mobile 6.5 [8].

Encase Forensic: Encase ist eines der Spitzenprogramme bei der forensischen Analyse von Computern.

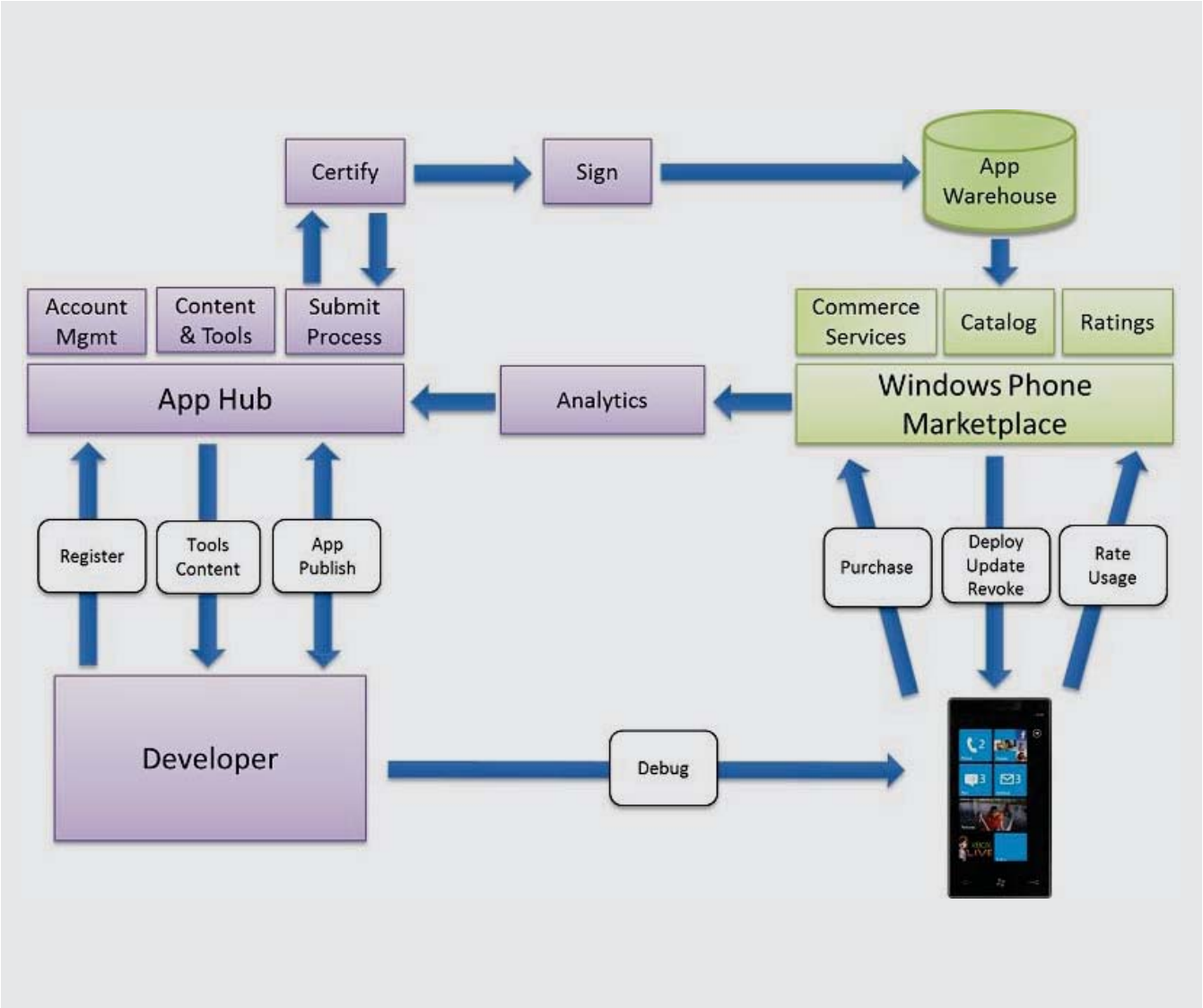


Abbildung 5. *Lebenszyklus von WP7-Apps [16]*

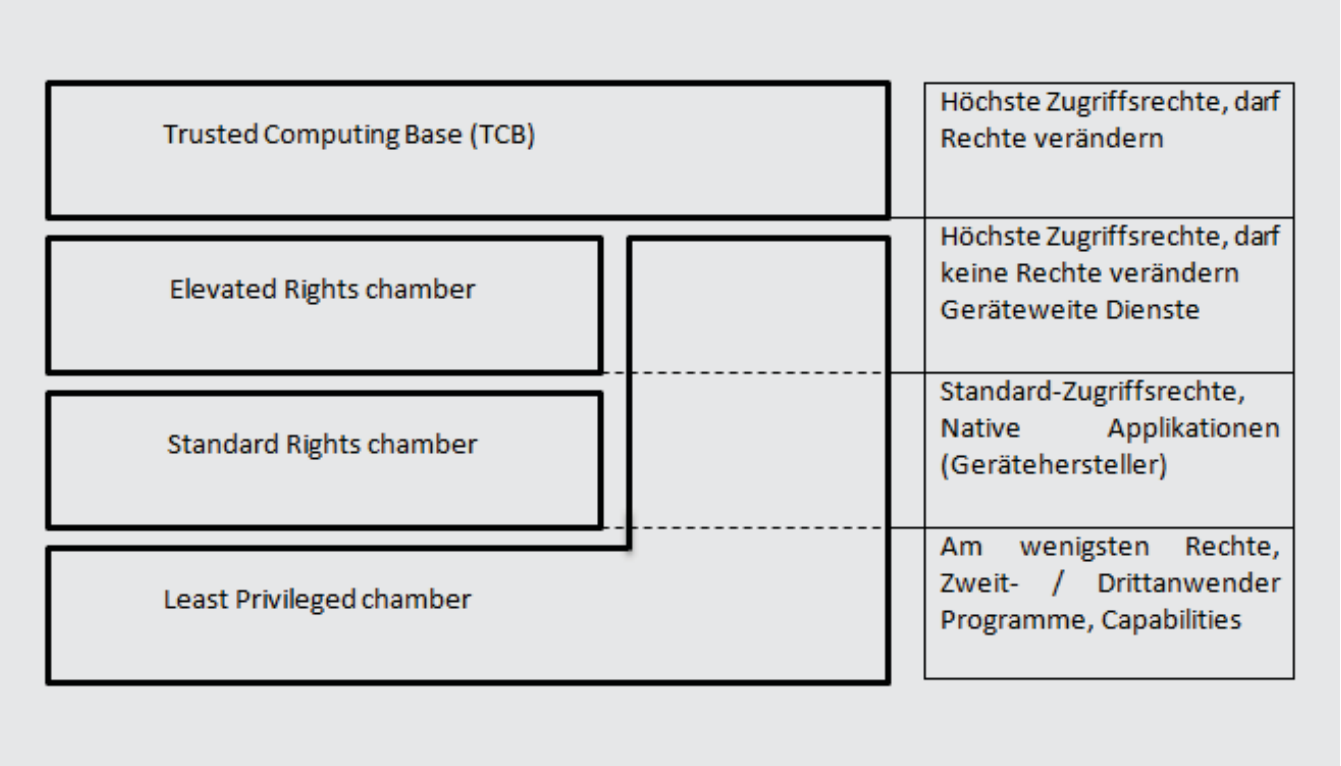


Abbildung 6. *Sicherheitsmodell WP7*

Inzwischen unterstützt es auch Smartphones verschiedener Herstelle (Android, Apple, Blackberry, Windows Mobile, Symbian) [9].

Katana Lantern: Lantern ist ausschließlich für die iOS-Forensik konzipiert und unterstützt explizit die neusten Apple-Produkte bzw. Firmware-Versionen. Außerdem kann das Tool den Passcode eines verschlüsselten Devices mit Hilfe des iTunes-Zertifikates zurücksetzen, iTunes-Backups einlesen und RAW-Images von iOS-Devices analysieren [10].

Micro Systemation XRY: Ähnlich umfangreich wie UFED, unterstützt XRY fast 6000 Geräteprofile von der logischen bis hin zur physikalischen Daten-Extraktion [11]. Abbildung 4 zeigt das Tool XRY.

Paraben Device Seizure: Das Tool liest die Daten von über 4000 Mobiltelefonen, PDAs oder GPS-Geräten aus. Die meisten Smartphone-Betriebssysteme werden unterstützt (meist logisch, manchmal auch physikalisch) [12].

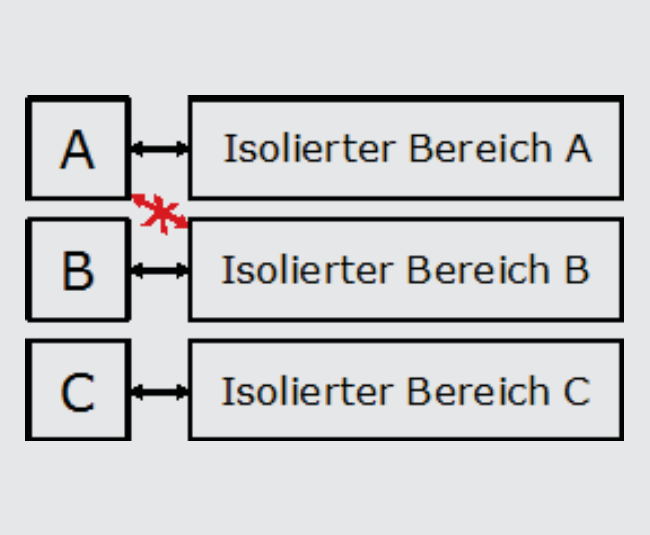


Abbildung 7. *Isolierte Bereich und Zugriffsrechte*

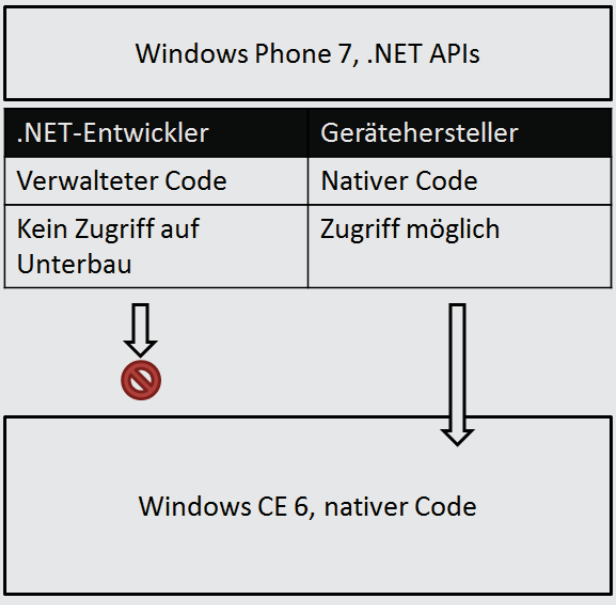


Abbildung 8. *Zugriff auf Windows CE 6 Kernel von Benutzer-Apps und Native Apps*

viaForensics viaExtract: Dieses Werkzeug hat sich auf die logische Analyse von Android-Smartphones spezialisiert [13].

Fallbeispiel 1: Windows Phone 7

Im Folgenden wird anhand eines Fallbeispiels erläutert, wie die Dateisystem-Extraktion bei einem Smartphone durchgeführt werden kann. Der Ansatz wurde Anfang des Jahres an der FH Aachen entwickelt [14] und bezieht sich auf das neue Microsoft Smartphone-Betriebssystem Windows Phone 7, welches Ende 2010 auf den Markt kam.

Windows Phone 7 (WP7) basiert auf dem Windows CE 6 Kernel. Es bietet ein neues Benutzer-Interface sowie ein Multi-Touchscreen mit virtuellem On-Screen-Keyboard. WP7-Smartphones werden mit vorinstallierten Apps ausgeliefert. Zu diesen gehören normalerweiser ein Webbrowser (Internet Explorer Mobile), Email-Clients (ein Outlook Client, der aber auch für Hotmail, Yahoo!Mail oder Googlemail benutzt werden kann), Multimedia-Player (für Musik, Videos oder Bil-

Abbildung 9. *Interface der Una App auf einem WP7-Smartphone*

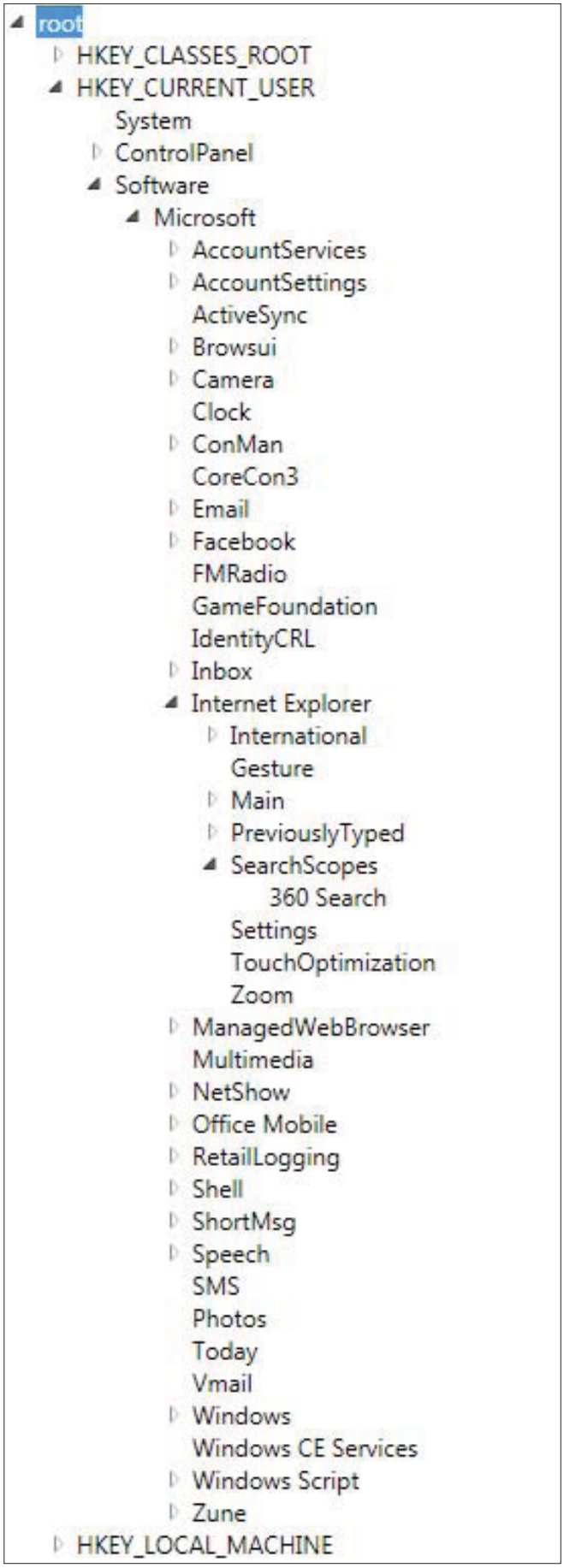


Abbildung 10. *Extrahierte Registry*

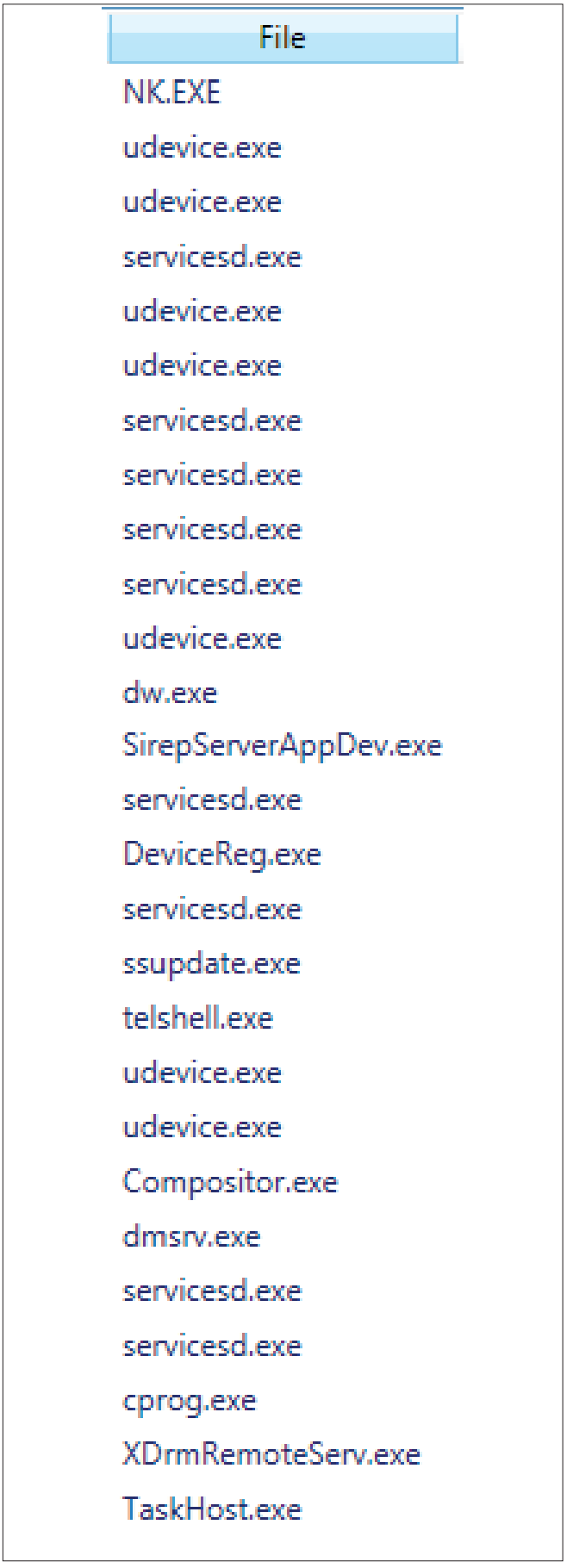


Abbildung 11. *Laufende Prozesse auf einem WP7-Handy*

der) sowie einer Office-Suite (welche Interoperabilität mit Microsofts Desktop-Produkten bietet).

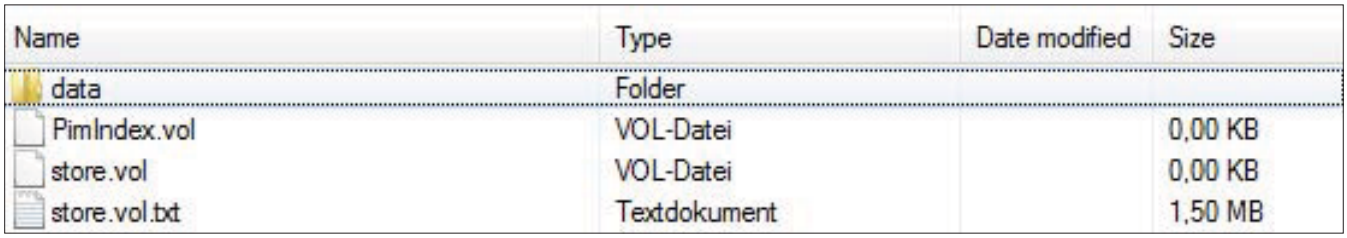
Auch WP7 erlaubt die Installation von Apps, die von Drittanbietern geliefert werden. Zusammen mit Musik und Videos werden solche Apps online im „Windows Phone Marketplace“ angeboten. Dieser Marktplatz wird von Microsoft betrieben. Möchte ein Anbieter seine App im Marktplatz platzieren, muss die App zunächst einen Zulassungsprozess durchlaufen, bei dem die App geprüft und von Microsoft signiert wird, bevor sie von Anwendern heruntergeladen werden kann. Um ihre eigene App vor der Einreichung zu testen, bieten sich Drittanbietern zwei Optionen [15]. Zunächst einmal gibt es die Möglichkeit, Benutzer-Apps in einer Emulator-Umgebung zu testen, die mit Microsofts Application Development Framework ausgeliefert wird. Alternativ kann man sich bei Microsoft registrieren, die App auf dem App Hub von Microsoft publizieren und dann eine nicht-signierte Version der App auf einem bestimmten, registrierten Smartphone testen. Abbildung 3 zeigt, wie diese zweite Testmöglichkeit in den allgmeinen Application Development Lifecycle Process der WP7 Apps integriert ist.

Das Sicherheitsmodell von WP7 folgt dem Prinzip der geringsten möglichen Rechte (Least Privilege), welche in der Form von sogenannten Kammern (Chambers) realisiert wird [17]. Eine Kammer definiert dabei eine Menge von Rechten, die einem bestimmten Prozess gewährt werden. Im WP7-Sicherheitsmodell gibt es vier unterschiedliche Kammern, welche in Abbildung 4 dargestellt sind.

Benutzer-Apps laufen in der Least Privileged Chamber, d.h. sie haben sehr eingeschränkte Rechte. Möchten Benutzer-Apps Daten persistent vorhalten, können sie diese in einem isolierten Bereich (Isolated Area) speichern [18]. Jede Benutzer-App generiert und verwaltet seinen eigenen isolierten Bereich. Zugriff auf den isolierten Bereich anderer Benutzer-Apps ist nicht möglich. Das Konzept wird für drei Benutzer-Apps A, B und C in Abbildung 5 veranschaulicht.

Die Ausführungsumgebung von Benutzer-Apps ist ebenfalls beschränkt: Jede Benutzer-App läuft in einer eigenen Sandbox und kann nicht direkt auf den Speicherbereich anderer Apps oder interne Teile des Betriebssystems zugreifen. Für bestimmte Systemfunktionen kann dies durch Benutzerfreigabe aufgehoben werden, z.B. zur Speicherung von Daten in den Kontakten oder um einen Anruf aus einer App heraus zu tätigen.

Apps der Gerätehersteller (sogenannte Native Apps) agieren in der Standard Rights Chamber und haben damit erweiterte Rechte. In Bezug auf Daten-Extraktion ist es wichtig festzuhalten, dass Native Apps Zugang zum Windows CE 6 Kernel haben, und dadurch



Name	Type	Date modified	Size
data	Folder		
PimIndex.vol	VOL-Datei		0,00 KB
store.vol	VOL-Datei		0,00 KB
store.vol.txt	Textdokument		1,50 MB

Abbildung 12. *Erzeugung einer Kopie von store.vol*

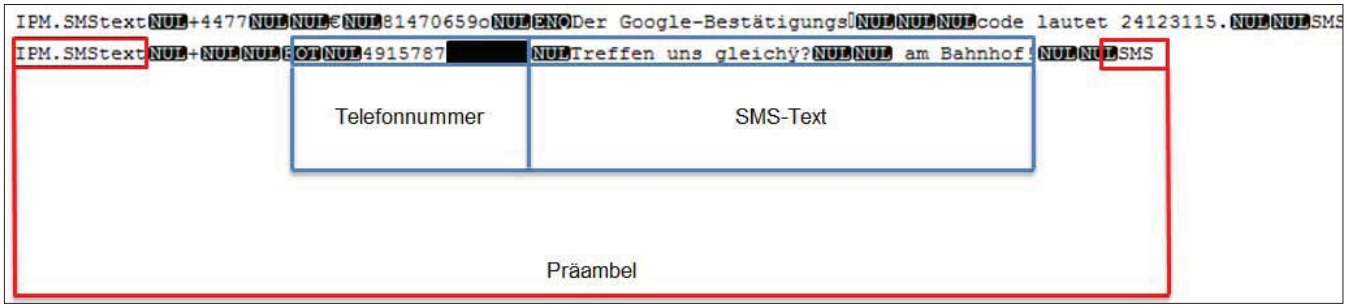


Abbildung 13. *Dekodieren von SMSs in der Datei store.vol.txt*

im Gegensatz zu Benutzer-Apps Daten aus den isolierten Bereichen anderer Apps lesen können (siehe Abbildung 6).

Diese Eigenschaft macht sich der hier vorgestellte forensische Ansatz zu Nutze. Auf dem Zielgerät wird eine kleine App installiert, welche Zugriff auf weite Teile des Handy-Speichers hat. Durch die Installation wird zwar ein kleiner Bereich des Flashspeichers überschrieben; im Vergleich zur Alternative „gar keine Daten zu erhalten“ ist dies, insbesondere wenn es gut dokumentiert wird, das kleinere Übel.

Zunächst muss eine App gebaut werden, die nativen Code ausführen kann. Das Fallbeispiel nutzt dazu die von Thomas Hounsell vorgestellte Methode, welche auf der DLL “Windows.Phone.interopService” basiert [19]. Diese DLL liefert die Methode "RegisterComDLL", welche native Hersteller-DLLs importieren kann. Verwendet man die “Windows.Phone.interopService” DLL in einer .NET Benutzer-App, wird es möglich, native Code auszuführen und Zugriff auf das gesamte Dateisystem inklusive isolierter Bereiche zu erhalten.

Als nächstes muss die fertige App auf dem Smartphone installiert werden. Wie bereits erläutert, erfordert dies im Normalfall die Registrierung des Entwicklers und des Telefons bei Microsoft, eine Prozedur die ein Ermittler wohl kaum für jedes zu untersuchende Telefon durchlaufen möchte. Deshalb kommt hier ein kleines Tool zum Einsatz, welches dabei hilft, diesen Prozess zu umgehen. Das Tool nennt sich ChevronWP7, und wurde eigentlich als Jailbreak-Methode für WP7 entwickelt [20], liefert also die Möglichkeit, Apps, die nicht im Microsoft Marketplace registriert sind, auf das Gerät zu laden.

Zur Daten-Extraktion auf dem WP7-Gerät wurden zwei unterschiedliche Apps benutzt. Die App “TouchXperience” stammt aus dem Projekt einer Entwickler-Community [21]. Die andere App, genannt „Una“, ist eine Entwicklung der FH Aachen. Damit die Apps mit ihrem Server auf dem PC kommunizieren können, erfordern beide Apps die Microsoft Zune Software [22] auf dem Rechner.

TouchXperience: Die App TouchXperience wird in Kombination mit dem Windows Mobile Device Manager (WPDM), einer Management Software für WP7 geliefert. Die zwei Programme bilden eine Client-Server-Architektur, wobei der Client (TouchXperience App) Daten wie das Dateisystem vom Smartphone extrahiert, und an den Server WPDM liefert, der sie dann in ein lesbares, graphisches Format umwandelt. Aus forensischer Sicht besonders interessant ist dabei der File Explorer, welcher Lese-, Schreib- und Ausführungsrechte für fast alle Dateien des WP7-Geräts ermöglicht. Je nach Hersteller des WP7-Geräts, können die von TouchXperience gelieferten Datenmengen unterschiedlich aussehen.

Una: Una wurde an der FH Aachen entwickelt. Es basiert ebenfalls auf einer Client-Server-Architektur und kann Daten aus WP7-Geräten extrahieren, die TouchXperience nicht liefern kann. Dazu gehören all-

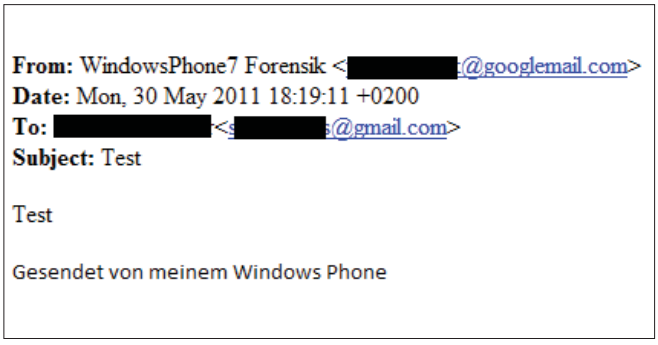


Abbildung 14. *Ausgelesene Email*

gemeine Geräteinformationen (z.B. Hersteller oder Geräte ID), laufende Prozesse (Prozessname, Prozess ID) und die Registry (allgemeine Windows Keys sowie herstellerspezifische Keys). Abbildung 7 zeigt einen Screenshot der Una Client App, die auf einem Smartphone läuft.

Mittels der beiden Apps können nun verschiedene System- und Anwendungsdaten von WP7-Smartphones gesichert werden. Im Fallbeispiel wurde ein HTC Trophy 7 ausgewertet. Hier lieferte die Methode das allgemeine Dateisystem, die Registry, aktive Tasks, Geräteinformationen, sowie anwendungsspezifische Daten wie SMSs, Emails, Kontakte, Anruflisten, GSM / WiFi-Daten, Daten der Maps App, des Internet-Explorers oder der Facebook App, private Dokumente, Notes sowie Bilder und Videos. Im Folgenden werden einige der Daten vorgestellt.

Dateisystem: Das WP7-Dateisystem gleicht in seiner Darstellung einem normalen Desktop-Dateisystem, wie man es von Windows XP, Windows Vista oder Windows 7 her kennt. Es ist aus den üblichen Verzeichnissen und Dateien aufgebaut. Über ein Uni-

Abbildung 15. *Angesehene Fotos aus den Profilen der Freunde*

```
0x20: 01 30 00 00 00 00 45 48 61 6C 6C 6F 2C 0A 0A 68   .0....EHallo,..h
0x30: 65 75 74 65 20 41 62 65 6E 64 20 74 72 65 66 66   eute Abend treff
0x40: 65 6E 20 77 69 72 20 75 6E 73 21 20 0A 0A 31 39   en wir uns! ..19
0x50: 20 55 68 72 20 69 6E 20 41 61 63 68 65 6E 0A 0A    Uhr in Aachen..
0x60: 47 72 75 C3 9F 0A 54 68 6F 6D 61 73 0F 31 30 30   GruÃ▮.[redacted]100
0x70: 30 30 31 34 36 35 39 35 38 30 31 33 0E 28 6B 65   001465958013.(ke
```

Abbildung 16. *Ausgehende Facebook-Nachricht*

```
0x01040: 24 00 00 00 68 00 74 00 74 00 70 00 3A 00 2F 00   $...h.t.t.p.:./.
0x01050: 2F 00 6D 00 2E 00 74 00 65 00 73 00 74 00 2E 00   /.m...t.e.s.t...
0x01060: 64 00 65 00 2F 00 00 00 6F 00 6D 00 2F 00 77 00   d.e./...o.m./.w.
0x01070: 65 00 6C 00 63 00 6F 00 6D 00 65 00 2F 00 00 00   e.l.c.o.m.e./...
0x01080: 02 00 00 00 00 00 74 00 74 00 70 00 3A 00 2F 00   ......t.t.p.:./.
0x01090: 2F 00 6D 00 2E 00 74 00 65 00 73 00 74 00 2E 00   /.m...t.e.s.t...
0x010A0: 64 00 65 00 2F 00 00 00 00 00 00 00 00 00 00 00   d.e./...........
```

Abbildung 17. *Link einer mit Internet Explorer angesehenen Webseite*

```
0x140: 00 00 02 00 8F 00 00 00 00 00 03 00 02 00 02 00   .... ..........
0x150: 04 00 4A E8 C7 E8 02 00 FE 00 10 80 32 30 20 4B   ..JèÇè..þ..[redacted]K
0x160: 61 6D 70 65 72 20 53 74 72 61 DF 65 02 00 FE 00   amper Straße..þ.
0x170: 02 80 4E 57 02 00 FE 00 00 80 02 00 FE 00 0B 80   .▮NW..þ..▮..þ..▮
0x180: 44 65 75 74 73 63 68 6C 61 6E 64 02 00 FE 00 00   Deutschland..þ..
0x190: 80 02 00 FE 00 1E 80 4B 61 6D 70 65 72 20 53 74   ▮..þ..▮Kamper St
0x1A0: 72 61 DF 65 20 32 30 2C 20 35 32 30 36 34 20 41   raße [redacted] 52064 A
0x1B0: 61 63 68 65 6E 02 00 FE 00 06 80 41 61 63 68 65   achen..þ..▮Aache
0x1C0: 6E 02 00 FE 00 05 80 35 32 30 36 34 02 00 FE 00   n..þ..▮52064..þ.
```

Abbildung 18. *Die letzte von Maps gespeicherte Position*

fied Storage System wird die Komplexität mehrerer Partitionen auf verschiedenen internen und externen Speichermedien verborgen und nur eine einzelne Verzeichnisstruktur ist sichtbar. Aus Sicht eines Ermittlers sind die Verzeichnisse „Application Data", „Applications", „My Documents" und „Windows" besonders interessant. Der Ordner „Appliciation Data" enthält Daten der auf dem Telefon vorinstallierten Apps, inklusive Outlook, Maps und Internet Explorer. Apps, die vom Benutzer installiert wurden, finden sich im Ordner „Applications". Dort befindet sich auch der isolierte Speicherbereich der jeweiligen Apps. Das Verzeichnis „My Documents" enthält verschiedene Office-Dokumente, z.B. Word, Excel oder PowerPoint. Hier finden sich auch Konfigurationsdateien sowie Musik- und Videodateien. Im Windows-Ordner liegen die Dateien des WP7-Betriebssystems.

Registry: Die Windows Registry, eine Datenbank in der das Betriebssystem und Anwendungen z.B. Umgebungsvariablen speichert, existiert auch auf WP7-Smartphones. Abbildung 8 zeigt beispielhaft eine mit Una extrahierte Registry eines WP7-Telefons.

Wie der Screenshot verdeutlich, ähnelt die Registry der ganz normalen Windows Desktop Registry. Sie enthält umfangreiche Informationen, z.B. über die Anwendungen die der Benutzer installiert hat und kann mit forensischen Registry-Tools weiter analysiert werden.

Active Tasks: Die aktiven Tasks sind die laufenden Prozesse auf einem WP7-Smartphone. Anhand der Liste der Tasks lässt sich zum Beispiel feststellen, ob unbekannte Prozesse auf dem Gerät laufen (z.B. Malware). Abbildung 9 zeigt den Screenshot der aktiven Tasks, die mittels Una ermittelt wurden.

SMS: WP7 speichert alle eingehenden und ausgehenden Kurznachrichten in der Datei „store.vol", welche im Verzeichnis „\Application Data\Microsoft\Outlook\Stores\DeviceStore" liegt. Die Datei kann nicht direkt kopiert werden, was vermutlich daran liegt, dass sie permanent in Benutzung ist. Benennt man die Datei jedoch um, so erzeugt WP7 automatisch eine Kopie der Datei, welche dann mit einem normalen Text-Editor analysiert werden kann. Abbildung 10 zeigt die umbenannte und kopierte Datei „store.vol.txt", Abbildung 11 den Inhalt der Datei.

Für sämtliche ein- und ausgehenden SMSs können die Telefonnummern und Nachrichteninhalte in der Textdatei gefunden werden. Im vorliegenden Fall wurde

Literatur

[1] IDC Press Release, „Samsung Takes Top Spot as Smartphone Market Grows 42.6% in the Third Quarter", *http://www.idc.com/getdoc.jsp?containerId=prUS23123911*

[2] Eric Katz, Richard Mislan, Marcus Rogers, „Results of Field Testing Mobile Phone Shielding Devices", Proceedings of the Third ICST International Conference on Digital Forensics and Cyber Crime, Dublin, Ireland, October 2011

[3] UFED 1.1.8.6 Release Notes, *http://cellebrite.com/releases/1186/release-note-1186-november-2011.pdf*

[4] ElcomSoft Breaks iPhone iOS4 Encryption, *http://www.brightsideofnews.com/news/2011/5/24/elcomsoft-breaks-iphone-ios4--encryption.aspx*

[5] Felix Freiling, Sven Schmitt, Michael Spreitzenbarth, „Forensic Analysis of Smartphones: The Android Data Extractor Lite (ADEL)", Proceedings of the Conference on Digital Forensics, Security and Law, Richmond, Virginia, USA, 2011

[6] Stefan Maus, Hans Höfken, Marko Schuba, „Forensic Analysis of Geodata in Android Smartphones", Proceedings of Cyberforensics 2011, Glasgow, Schottland 2011

[7] Cellebrite, *http://www.cellebrite.com/*

[8] MOBILedit!, *http://www.mobiledit.com/*

[9] Guidance Software, *http://www.guidancesoftware.com/*

[10] Katana Forensics, *http://katanaforensics.com/*

[11] Micro Systemation, *http://www.msab.com/*

[12] Paraben Coorporation, *http://www.paraben.com/index.html*

[13] viaForensics, *http://viaforensics.com/*

[14] Thomas Schaefer, Hans Höfken, Marko Schuba, „Windows Phone 7 from a Digital Forensics' Perspective", Proceedings of the Third ICST International Conference on Digital Forensics and Cyber Crime 2011, Dublin, Ireland, October 2011

[15] Microsoft App Hub, „how it works – create", *http://create.msdn.com/en-US/home/about/how_it_works_create,* 2011

[16] Microsoft MSDN Library, „Application Platform Overview for Windows Phone", *http://msdn.microsoft.com/de-de/library/ff402531.aspx,* 2011

[17] Microsoft, „Windows Phone 7 security model", *http://go.microsoft.com/fwlink/?LinkId=207799, 2010*

[18] Microsoft MSDN Library, „Isolated Storage Overview for Windows Phone", *http://msdn.microsoft.com/en-us/library/ff402541%28v=vs.92%29.aspx,* 2011

[19] Thomas Hounsell, „Avoiding Reflection: Adding the InteropServices library to the WP7 SDK", *http://thounsell.co.uk/2010/11/avoiding-reflection-adding-the-interopservices-library-to-the-wp7-sdk/,* 2010

[20] Rafael Rivera, Chris Walsh, Long Zheng, „Pursuing the future of homebrew on Windows Phone 7", *http://www.chevronwp7.com/post/2057541126/pursuing-the-future-ofhomebrew-on-windows-phone-7,* 2010

[21] TouchXperience – Project Website, http://www.touchxperience.com/, 2011

[22] Microsoft, „Getting to know the Zune software", *http://www.microsoft.com/windowsphone/en-sg/howto/wp7/start/get-to--know-the-zune-software.aspx,* 2011

diese Auswertung manuell durchgeführt, ließe sich aber problemlos automatisieren.

Emails: Der Standard Email-Client von WP7 ist Outlook. Er ermöglicht es dem Benutzer seine Emails mit verschiedenen Email-Diensten zu synchronisieren (z.B. Googlemail). Alle Outlookdaten (inklusive der Bilder der Kontakte, mit denen gemailt wird) werden im Verzeichnis „\Application Data\Microsoft\Outlook\Stores\DeviceStore\data“ gespeichert. Das Verzeichnis enthält eine Reihe von nummerierten Ordnern, welche unterschiedliche Inhalte speichern: Ordner „3“ speichert Bilder der Kontakte des Benutzers in Dateien mit Extension „.dat“. Durch File Carving, also Analyse der binären Datei-Header zeigt sich, dass es sich dabei eigentlich um JPEG-kodierte Bilder handelt. Ein einfaches Umbenennen von „.dat“ zu „.jpg“ macht die Bilder sichtbar. Ordner „4“ enthält Emails, die – im Fall von Googlemail – als html-Dateien gespeichert sind. Abbildung 12 zeigt ein Beispiel einer extrahierten Email.

Facebook: Hat ein Benutzer eine Facebook-App auf seinem WP7-Telefon installiert, so können auch hier Daten ausgewertet werden. Die App generiert eine Anzahl von Verzeichnissen, z.B. „Cache“, „History“ und „IsolatedStore“. Die Ordner enthalten allgemeine Konfigurationsdaten der App sowie Unterordner wie „Images“ oder „DataCache.%userid%“, wobei „%userid%“ in diesem Fall mit der Benutzer ID ersetzt werden muss. Letzterer Ordner enthält alle Daten der Facebook-App in Klartext. Die Ordner sind für Ermittler durchaus interessant, weil sie eine große Menge an Daten speichern. Die gefundene Datei „%userid%.settings“ enthält den Profilnamen des Benutzers, einen Link zu seinem Profil sowie das Profilbild. Jedes Bild, welches der Benutzer mit seiner Facebook-App angeschaut hat, wird im Ordner „Images“ des Verzeichnisses „IsolatedStore“ abgelegt. Das schließt auch Bilder ein, die Freunde über Ihr Profil veröffentlich, aber als privat gekennzeichnet haben (siehe Abbildung 13).

Die Datei „DataCache.%userid%“ enthält Informationen über die Homepage des Benutzers inklusive seiner letzten Position (falls die Ortsbestimmung eingeschaltet war), eingehender und ausgehender Nachrichten (siehe Beispiel in Abbildung 14). Zudem findet sich hier eine Liste der Freunde des Benutzers, mitsamt ihrer Geburtstage, ihrer Benutzer IDs und Links zu ihren Profilbildern.

Internet Explorer: Der standardmäßige Webbrowser auf einem WP7-Smartphone ist der Internet Explorer. Für einen Ermittler könnten besuchte Webseiten wichtig sein. Alle relevanten Daten befinden sich in der Datei „TabsStorage.bin“ im Verzeichnis „Application Data\ Microsoft\Internet Explorer“. Ein Auszug dieser Datei wird in Abbildung 15 gezeigt.

Maps: Die Microsoft-App „Maps“ erlaubt es WP7-Benutzern ihre Position zu bestimmen und Routen zu berechnen. Die dazugehörenden Anwendungsdaten werden im Verzeichnis „Application Data\Maps“ gespeichert. Die Datei „MapsSetData.dat“ enthält die letzte Position des Geräts in Klartext (siehe Abbildung 16). Interessanterweise wird die Position als Adresse gespeichert, nicht als GPS-Koordinaten. Eigene Tests zeigten, dass diese Positionen durchaus akkurat sind, mit einer Abweichung von nicht mehr als zwei Häusern.

Ausblick WP7: Auch mit der vorgestellten Analysemethode unterliegt man den Herausforderungen der Smartphone-Forensik. Das Verfahren wurde Anfang 2011 entwickelt. Mittlerweile hat Microsoft zwei neue Updates für WP7 veröffentlicht (NoDo und Mango/WP7 Version 7.5), bei denen das Verfahren zunächst einmal nicht mehr funktioniert. Das liegt zum einen an neuen Sicherheitsmechanismen, die die Ausführung von nativem Code weiter einschränken, zum anderen an Veränderungen des Prozesses, neue Apps auf das Telefon zu laden. Doch die weltweite Community ist schnell. Für beide sind inzwischen Verfahren entwickelt worden, die die Mechanismen umgehen können. Die Integration in das vorliegende Tool ist in Arbeit.

MARKO SCHUBA, HANS HÖFKEN, THOMAS SCHAEFER

Marko Schuba ist Professor an der FH Aachen. Davor war er über zehn Jahre in der Telekommunikations-Industrie tätig. Er lehrt und forscht in den Bereichen Datennetze und IT-Sicherheit sowie - als einer der wenigen Professoren in Deutschland - im Bereich IT-Forensik. Darüber hinaus ist er geschäftsführender Gesellschafter der schuba & höfken Gbr.
schuba@fh-aachen.de

Hans Höfken ist Leiter der Rechenzentrale und wissenschaftlicher Mitarbeiter im Fachbereich Elektrotechnik und Informationstechnik der FH Aachen. Er arbeitet auf dem Gebiet IT-Sicherheit und IT-Management und ist verantwortlicher Trainer der Cisco Academy der FH Aachen und Kursleiter von Hacking- und IT- Forensikkursen.
hoefken@fh-aachen.de

Thomas Schaefer ist Student der FH Aachen. Im Sommer 2011 machte er seinen Bachelor-Abschluss im Bereich Informatik mit dem Thema „Windows Phone 7 aus forensischer Sicht“. Momentan erweitert er dieses Studium, ebenfalls an der FH Aachen, um einen Master-Abschluss im Bereich „Information System Engineering“.
sch.thomas@gmail.com