Alle Übungen im Buch zum **Download**

Marc Brandt

Mac OS Hacking

Professionelle Werkzeuge und Methoden zur forensischen Analyse des Apple-Betriebssystems

- Sicherungs- und Analysetechniken für digitale Spuren
- Integrierte Mac-OS-Sicherheitssysteme angreifen und überwinden
- Forensische Analysestrategien zu Spotlight, Time Machine und iCloud

FRANZIS

Marc Brandt

Mac OS Hacking

Marc Brandt

Mac OS Hacking

Professionelle Werkzeuge und Methoden zur forensischen Analyse des Apple-Betriebssystems

- Sicherungs- und Analysetechniken für digitale Spuren
- Integrierte Mac-OS-Sicherheitssysteme angreifen und überwinden
- Forensische Analysestrategien zu Spotlight, Time Machine und iCloud

Bibliografische Information der Deutschen Bibliothek

Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie;
detaillierte Daten sind im Internet über http://dnb.ddb.de abrufbar.

Alle Angaben in diesem Buch wurden vom Autor mit größter Sorgfalt erarbeitet bzw. zusammengestellt und unter Einschaltung wirksamer Kontrollmaßnahmen reproduziert. Trotzdem sind Fehler nicht ganz auszuschließen. Der Verlag und der Autor sehen sich deshalb gezwungen, darauf hinzuweisen, dass sie weder eine Garantie noch die juristische Verantwortung oder irgendeine Haftung für Folgen, die auf fehlerhafte Angaben zurückgehen, übernehmen können. Für die Mitteilung etwaiger Fehler sind Verlag und Autor jederzeit dankbar. Internetadressen oder Versionsnummern stellen den bei Redaktionsschluss verfügbaren Informationsstand dar. Verlag und Autor übernehmen keinerlei Verantwortung oder Haftung für Veränderungen, die sich aus nicht von ihnen zu vertretenden Umständen ergeben. Evtl. beigefügte oder zum Download angebotene Dateien und Informationen dienen ausschließlich der nicht gewerblichen Nutzung. Eine gewerbliche Nutzung ist nur mit Zustimmung des Lizenzinhabers möglich.

© 2017 Franzis Verlag GmbH, 85540 Haar bei München

Alle Rechte vorbehalten, auch die der fotomechanischen Wiedergabe und der Speicherung in elektronischen Medien. Das Erstellen und Verbreiten von Kopien auf Papier, auf Datenträgern oder im Internet, insbesondere als PDF, ist nur mit ausdrücklicher Genehmigung des Verlags gestattet und wird widrigenfalls strafrechtlich verfolgt.

Die meisten Produktbezeichnungen von Hard- und Software sowie Firmennamen und Firmenlogos, die in diesem Werk genannt werden, sind in der Regel gleichzeitig auch eingetragene Warenzeichen und sollten als solche betrachtet werden. Der Verlag folgt bei den Produktbezeichnungen im Wesentlichen den Schreibweisen der Hersteller.

Satz: DTP-Satz A. Kugge, München
art & design: www.ideehoch2.de
Druck: M.P. Media-Print Informationstechnologie GmbH, 33100 Paderborn
Printed in Germany

ISBN 978-3-645-60551-9

Über den Autor

Der Autor Marc Brandt ist an der Hochschule für Polizei Baden-Württemberg als Dozent tätig. Hier hält er schwerpunktmäßig Seminare zu den Themengebieten Mac, Mobile Devices und Network Forensics.

Marc Brandt begann seine Laufbahn als Streifenpolizist. Ein Jahr später wechselte er zur Kriminalpolizei. Bereits während seiner Tätigkeit bei einer Spezialdienststelle der Polizei Baden-Württemberg konnte er seine technische Begeisterung einbringen und vertiefen. Nach seinem Wechsel zum Polizeipräsidium Mannheim ergriff er 2007 die Chance auf eine Beschäftigung in der IT-Forensik. Seit diesem Zeitpunkt bildet er sich stetig weiter und krönte seine Leidenschaft für digitale Spuren im Jahr 2015 mit dem Abschluss M. Sc. in Digitaler Forensik an der Hochschule Albstadt-Sigmaringen.

Im Rahmen seiner Tätigkeit als Dozent erstellte er bereits mehrere Skripte und Papers für IT-Experten innerhalb der Polizei und führte entsprechende Seminare durch. Das Buch »Mac OS Hacking« ist ein Ergebnis seiner praktischen Erfahrungen und seiner lehrenden Tätigkeit.

Danksagungen

Dieses Buch widme ich meiner Frau Nina, die mich seit Jahren begleitet. Mit ihrer Expertise ist sie meine größte Inspiration und zugleich größte Kritikerin.

Danken möchte ich Daniel Jud von der Kantonspolizei Zürich für seine wertvollen Tipps und Hilfen.

Mein Dank gebührt zudem dem Franzis Verlag für die freundliche, motivierende und professionelle Begleitung bei der Erstellung bis hin zur Veröffentlichung des Buchs, insbesondere Herrn Ulrich Dorn für sein wertvolles fachliches Lektorat.

Vorwort

Das Buch »Mac OS Hacking« ist eine umfangreiche Anleitung und Hilfe für IT-Forensiker, Analysten und interessierte Apple-Nutzer. Es bietet ein profundes Grundlagenwissen zum Thema Mac OS und führt durch alle wichtigen Prozesse einer forensischen Analyse eines Mac-Computersystems. Ein aufmerksames Studium des Buchs macht Sie zu einem Spezialisten für das Betriebssystem Mac OS.

Neben der Beschreibung von Funktionalitäten und Besonderheiten des Betriebssystems werden insbesondere digitale Spuren lokalisiert, kategorisiert und praktische Methoden für eine Analyse aufgezeigt. Das Auffinden von Informationen in Plist- oder SQLite-Dateien kann bei der Arbeit von forensischen Analysten u. a. von Ermittlungsbehörden zur Beantwortung bedeutsamer Fragestellungen führen und damit zur Aufklärung von Straftaten beitragen. Ebenso kann eine Sicherung und Analyse des Hauptspeichers beispielsweise zur Identifizierung von Passwörtern oder zur Lokalisierung von Malware verhelfen.

Das Buch entstand unter anderem auf Grundlage meiner Abschlussarbeit »Forensische Analyse von Mac OS X« im Rahmen des Studiums der Digitalen Forensik an der Hochschule Albstadt-Sigmaringen. »Mac OS Hacking« richtet sich an forensische Analysten und interessierte Leser, die einen tieferen Einblick in das Betriebssystem Mac OS erhalten möchten.

Im Lauf des Buchs lernt der Leser entsprechende Sicherungs- und Analysetechniken für digitale Spuren kennen. Darüber hinaus enthält das Buch weitere neu hinzugekommene Themenbereiche wie ein umfangreiches Kapitel über das Dateisystem HFS+ oder Analysemöglichkeiten von Mac-Technologien wie Spotlight, Time Machine oder iCloud. Ebenso neu eingeflossen sind Themen wie die fortgeschrittene analytische Nutzung des Terminals, ein Kapitel zu AppleScript und zu Mac OS X Server.

Das Buch lebt in erster Linie davon, gelesen und benutzt zu werden. Es würde mich freuen, wenn Ihnen das Lesen Spaß bereitet und Sie viele Informationen mitnehmen und einsetzen können. Gerne freue ich mich über Ihr Feedback und Ihre Ideen zur Verbesserung des Buchs und der zugehörigen Übungen.

Marc Brandt, Calw im Sommer 2017

Motivation und Aufbau des Buchs

Apple-Computersysteme mit dem Betriebssystem Mac OS spielen heute eine immer größere Rolle. Dennoch befindet sich die Mac-Forensik als Teildisziplin der Computerforensik noch immer in einer Nischenrolle. Es existieren vergleichsweise wenige Fortbildungsangebote und nur wenige verlässliche Dokumentationen – und diese wiederum meist nicht in deutscher Sprache. Apple selbst ist recht zurückhaltend und liefert nur zu bestimmten Bereichen technische Informationen (sogenannte Technical Notes), bei-

spielsweise dem XNU-Kernel. Viele proprietäre Funktionalitäten bzw. Frameworks sind hingegen wenig bis überhaupt nicht dokumentiert.

Bekannte Bücher wie Amit Singhs »OS X Internals: A Systems Approach« oder das exzellente Werk »Mac OS X and iOS Internals« von Jonathan Levin bieten zwar profunde Informationen zur OS-X-Systemarchitektur, zu Objective-C, Frameworks und APIs, sind aber derzeit nicht auf aktuellem Stand oder legen keine Schwerpunkte auf die forensische Analyse. Das Buch »OS X Internals: A Systems Approach« beispielsweise ist PowerPC-orientiert und wurde 2006 (bis zur OS-X-Version 10.4 Tiger) zuletzt aktualisiert. Das über 800 Seiten schwere Buch »Mac OS X and iOS Internals« wurde zuletzt 2013 aufgelegt (bis zur OS-X-Version 10.7.4 Lion und iOS 5.1.1).

Zur Zeit befindet sich Mac OS in der Version 10.12.2 mit der Bezeichnung macOS Sierra, die mobile Variante des Betriebssystems iOS in der Version 10.2.1. Aus diesem Bedürfnis nach Informationen heraus entstand die Idee für dieses Buch. Es möchte auf die Besonderheiten des Betriebssystems Mac OS eingehen, das mehr und mehr Menschen in seinen Bann zieht. Es möchte hinter die Kulissen von Mac OS und der damit korrespondierenden Apple-Geräte schauen, möchte eingesetzte Technologien und Mechanismen transparent erklären, um Analysten, Forensikern und interessierten Lesern als Anleitung bzw. Hilfestellung zur Verfügung zu stehen.

Das Buch ist in die folgenden Teilbereiche gegliedert:

- **Mac OS Lab**: Das Kapitel führt durch die Einrichtung eines Übungs-Accounts auf Ihrem Mac-Computer. Mit dem eingerichteten Account können Sie die begleitenden Übungen zum Buch durchführen.
- **Hintergrund**: Das Kapitel gibt eine Einführung in die Grundkonzepte und die Systemarchitektur von Mac OS. Es bietet einen Blick unter die Oberfläche des Betriebssystems und zeigt Mac-spezifische Technologien auf.
- **Mac-Datenträger, -Partitionen und -Dateisystem**: Das Kapitel bietet umfangreiche Informationen zu eingesetzten Partitionierungs-Schemata und den Dateisystemen HFS+ und Apple File System unter forensischen Gesichtspunkten.
- **Forensische Analyse von Mac OS**: Das Kapitel zeigt Möglichkeiten der forensischen Sicherung und Analyse von Mac-Computern und Mac OS auf. Dabei liefert es einen Einblick in den aktuellen Stand der Forschung.
- **Kategorisierung von Spuren**: Das Kapitel enthält eine mögliche Kategorisierung von digitalen Spuren. Dies erleichtert die Suche und Bewertung entsprechender Spuren und dient als Grundlage für eine weitere forensische Analyse.
- **Persistente Spuren**: Das Kapitel betrachtet nicht flüchtig gespeicherte digitale Spuren eines Mac-OS-Betriebssystems und bietet eine weitere Kategorisierungsmöglichkeit an. In diesem Kapitel werden systemnahe und anwendungsorientierte Artefakte, Analysemöglichkeiten und ihre Bewertung besprochen.
- **Mac-OS-Log-Dateien**: Mac OS besitzt mächtige Logging-Funktionalitäten, die eine Fundgrube für Informationen darstellen. Das Kapitel thematisiert die von Mac OS verwendeten Technologien, unter anderem auch das mit macOS Sierra neu eingeführte Unified-Logging-System.

- **Hack the Mac**: Mac OS ist als sicheres Betriebssystem bekannt. Viele Anwender nutzen die sehr intuitiv einzurichtenden Zugangsbeschränkungen wie Kennwörter, Schlüsselbund-Dateien oder die Verschlüsselungstechnologie FileVault. Das Kapitel zeigt praktisch auf, wann und wie sie überwunden bzw. konkret angegriffen werden können.
- **Übung Analyse und Cracking**: In diesem Kapitel haben Sie die Möglichkeit, ein kriminalistisches Szenario zu bearbeiten. Das Szenario umfasst Inhalte und Techniken aus den Kapiteln »Persistente Spuren«, »Mac OS Log-Dateien« und »Hack the Mac«.
- **Anwendungsanalyse unter Mac OS**: Nach der Vorstellung eines allgemeinen Modells zur Analyse von Programmen führt das Kapitel am Beispiel der Nachrichten-Applikation durch eine Anwendungsanalyse unter Mac OS.
- **Nicht-persistente Spuren**: Als äußerer Rahmen einer Kategorisierung von digitalen Spuren beinhalten nicht persistente Spuren im weitesten Sinne flüchtige Spuren. Dazu zählen beispielsweise der Hauptspeicher, Auslagerungsdateien oder Cloud Storage.
- **Random-Access-Memory-Analyse**: Die Sicherung und Analyse von Hauptspeicherinhalten wird in einem eigenen Kapitel besprochen. Das Buch legt hierbei einen Schwerpunkt auf Analysemöglichkeiten mit der Open-Source-Software Volatility.
- **Mac-Technologien unter forensischer Betrachtung**: Das Kapitel bespricht praxisrelevante Mac-spezifische Technologien wie Versions, Spotlight, Time Machine und iCloud. Neben einem möglichst tiefen Einblick in die Funktionsweisen bietet das Kapitel forensische Analysestrategien zu den jeweiligen Themenbereichen.
- **Advanced Terminal**: Das Mac-Terminal ist ein äußerst mächtiges und vielfältiges Werkzeug auch für Analysten. Das Kapitel gibt einen tieferen Einblick in die Möglichkeiten, das Potenzial der Kommandozeile für forensische Zwecke zu nutzen.
- **AppleScript**: Das Kapitel beinhaltet eine Einführung in die Skriptsprache AppleScript und das Werkzeug Automator. Anhand einiger praktischer Beispiele werden einfach umzusetzende Automatisierungsmöglichkeiten aufgezeigt.
- **OS X Server**: Das Kapitel betrachtet das Serversystem OS X Server und gibt eine Einführung in forensische Analysemöglichkeiten.

Übungen und Downloads
Das Buch wurde so verfasst, dass sich ein chronologisches Lesen, angefangen bei der Beschreibung der Betriebssystemarchitektur über die Kategorisierung der digitalen Spuren bis hin zur forensischen Analyse bzw. der RAM-Analyse und den weiteren Kapiteln des Buchs, anbietet. Nichtsdestotrotz können die einzelnen Kapitel des Buchs auch für sich allein stehend gelesen werden.

Zum optimalen Lesen und Verstehen der Inhalte dieses Buchs benötigen Sie einen Mac-Computer mit einem aktuellen Betriebssystem, idealerweise OS X 10.11 El Capitan oder macOS 10.12 Sierra. Obwohl das Vorhandensein eines Mac-Computers nicht unbedingt eine Voraussetzung ist, um das Buch zu verstehen, empfehle ich sehr, die Übungsbeispiele auch praktisch durchzuführen.

Die im Buch vorgestellten Übungsbeispiele stehen auf der Website des Franzis Verlags unter der URL *www.buch.cd* zum Download zur Verfügung.

Typografische Konventionen
Das Buch nutzt unterschiedliche typografische Konventionen und didaktische Elemente, um den Lesefluss zu erleichtern. Im Folgenden sind einige der genutzten Stilelemente beschrieben:

Programmcode, Listings und Terminaleingaben werden im Fließtext formatiert dargestellt. Funktionen und Anweisungen sind entsprechend eingerückt.

```
def example():
print(„Hallo Welt“)
```

Eingaben im Terminal sind durch das Prompt `$` gekennzeichnet. In diesem Buch werden Terminaleingaben und korrespondierende Ausgaben in vielen Fällen in Form von beispielhaften Screenshots aufgezeigt.

```
$ ls -lai
```

Datei- und Pfadangaben werden in *kursiver* Schrift dargestellt.

/Users/marc/Desktop/example.txt

~/Desktop/example.txt

Hyperlinks zu externen Quellen im Internet werden ebenfalls in kursiver Schrift dargestellt.

http://www.franzis.de

Inhaltsverzeichnis

1 Ein Mac OS Lab aufsetzen ... 19
1.1 System Integrity Protection deaktivieren ... 19
1.2 Einen Übungs-Account einrichten ... 19
1.2.1 Einen neuen Benutzer-Account erstellen ... 20
1.2.2 Gatekeeper-Funktionalität deaktivieren ... 21
1.3 Programme im Mac OS Lab installieren ... 22
1.3.1 Java SE ... 22
1.3.2 Xcode ... 22
1.3.3 Fuse ... 23
1.3.4 Mac Ports ... 23
1.3.5 Homebrew ... 23
1.3.6 Libewf ... 24
1.3.7 Xmount ... 24
1.3.8 Sleuth Kit ... 25
1.3.9 SQLite-Browser ... 25
1.3.10 Hex-Editoren ... 25
1.3.11 Github ... 25
1.3.12 Python ... 25
1.4 Den Übungs-Account wieder löschen ... 26

2 Wichtige Hintergrundinformationen ... 27
2.1 Das Betriebssystem Mac OS ... 27
2.2 Applelution in Cupertino ... 28
2.2.1 Apple-Betriebssystem-Modelle ... 31
2.3 Mac OS X intern ... 31
2.3.1 Historie der Mac-OS-X-Versionen ... 32
2.3.2 Darwin – das Grundgerüst von Mac OS ... 37
2.4 Die Mac-OS-Architektur ... 38
2.4.1 HFS+ ... 39
2.4.2 HFS+ case-sensitive ... 39
2.4.3 /bin ... 40
2.4.4 /sbin ... 40
2.4.5 /usr ... 41
2.4.6 /etc ... 41
2.4.7 /dev ... 41
2.4.8 /tmp ... 41
2.4.9 /var ... 41
2.4.10 /Applications ... 42
2.4.11 /Developer ... 42
2.4.12 /Library ... 42

2.4.13 /Network 42
2.4.14 /System 43
2.4.15 /Users 43
2.4.16 /Volumes 43
2.4.17 /Cores 43
2.4.18 Apple EFI 46
2.4.19 Launchd 50
2.4.20 Prozesse und Threads 51
2.4.21 Mach-O-Binaries 53
2.4.22 Bundles und Packages 55
2.4.23 Applications 56
2.4.24 Frameworks 56
2.5 Mac-OS-Sicherheitskonzepte 57
2.5.1 Code Signing 57
2.5.2 Sandboxing 57
2.5.3 Gatekeeper 58
2.5.4 File Quarantine 59
2.5.5 System Integrity Protection 59
2.5.6 XPC 60
2.6 Zusammenfassung 61
2.7 Übung: Mac-OS-Handling 62

3 Das Mac-OS-Dateisystem im Fokus 67
3.1 Solid State Disks 68
3.2 GUID-Partitionsschema 70
3.2.1 GUID Partition Table 70
3.2.2 Analyse der GUID Partition Table 75
3.2.3 Zusammensetzen von Fusion-Drive-Laufwerken 77
3.3 Hierarchical File System Plus (HFS+) 78
3.3.1 Speichersystematik 80
3.3.2 HFS+ Special Files 81
3.3.3 Extraktion von HFS+ Special Files 82
3.3.4 Volume Header 83
3.3.5 Allocation File 86
3.3.6 B-Baum-Struktur 87
3.3.7 Catalog File 95
3.3.8 Extents Overflow File 100
3.3.9 Attributes File 100
3.3.10 Journal 103
3.3.11 Dateikomprimierung 103
3.3.12 Hardlinks 105
3.3.13 Mac-OS-Zeitstempel 107
3.4 Apple File System (APFS) 108
3.4.1 Flexible Partitionen 109
3.4.2 Dateisystem-Snapshots 111
3.4.3 Dateien und Verzeichnisse klonen 112
3.4.4 Verschlüsselung 112
3.4.5 Eine APFS-Volume erstellen 113

3.4.6 Partitionsschema ... 115
3.4.7 Container-Superblock ... 117
3.4.8 Volume Header ... 117
3.4.9 Forensische Ansätze ... 120
3.4.10 Ausblick auf das APFS ... 123
3.5 Übung: Partitionen und Dateisystem ... 124

4 Forensische Analyse von Mac OS ... 131
4.1 Stand der Forschung ... 131
4.2 Modelle der Digitalen Forensik ... 132
4.3 Der investigative Prozess nach Casey ... 132
4.3.1 Der investigative Prozess für Mac-Computer ... 135
4.4 Live Response ... 135
4.4.1 Maßnahmen bei eingeschalteten Mac-Computern ... 136
4.4.2 Vertrauenswürdige Binaries ... 137
4.4.3 Sammlung volatiler Daten (Triage) ... 138
4.4.4 Virtuelle Maschinen ... 139
4.5 Übung: Live Response ... 140
4.5.1 RAM-Sicherung ... 145
4.5.2 Logische Sicherung ... 147
4.6 Post-Mortem-Analyse ... 147
4.6.1 Forensische Abbilder von Datenträgern ... 148
4.6.2 Live-CD/-DVD oder bootbarer USB-Stick ... 150
4.6.3 Sicherung von MacBooks mit NVMe-Controllern ... 152
4.6.4 Sicherung über die Recovery-Partition ... 154
4.6.5 Target Disk Mode ... 155
4.6.6 FileVault 2 und Fusion Drive ... 155
4.6.7 Open-Firmware-Passwort ... 156
4.6.8 Disk Arbitration ... 156
4.7 Sicherungsstrategien für Mac-Computer ... 157
4.8 Übung: Sicherung erstellen ... 158
4.8.1 Sicherung mit dd/dcfldd ... 158
4.8.2 Sicherung mit ewfacquire ... 159
4.8.3 Sicherung mit dem FTK Imager ... 159

5 Kategorisierung digitaler Spuren ... 161
5.1 Persistente Spuren ... 161
5.2 Mac-spezifische Formate ... 162
5.2.1 Property List Files ... 162
5.2.2 NSKeyedArchiver-Format ... 163
5.2.3 SQLite ... 166
5.2.4 Analyse von SQLite ... 169
5.2.5 Disk Images ... 178
5.2.6 Forensische Abbilder mounten ... 180
5.3 System- und lokale Domäne ... 181
5.3.1 Systeminformationen ... 181
5.3.2 Nutzerkonten ... 184
5.3.3 Netzwerkeinstellungen ... 185

5.3.4 Software-Installationen ... 186
5.3.5 Drucker ... 189
5.3.6 Keychains ... 190
5.3.7 Firewall ... 192
5.3.8 Launch Agents ... 193
5.3.9 Launch Daemons ... 194
5.3.10 Freigaben ... 195
5.4 Nutzer-Domäne ... 201
5.4.1 Nutzer-Account-Informationen ... 202
5.4.2 Papierkorb ... 204
5.4.3 Zuletzt genutzte Objekte ... 205
5.4.4 Dock ... 206
5.4.5 Spaces ... 207
5.4.6 Anmeldeobjekte von Nutzern ... 209
5.4.7 SSH ... 210
5.4.8 Apps ... 211
5.4.9 Kontakte ... 211
5.4.10 Kalender ... 213
5.4.11 Mail ... 214
5.4.12 Safari ... 217
5.4.13 Fotos ... 218
5.4.14 Nachrichten ... 219
5.4.15 FaceTime ... 221
5.4.16 Notizen ... 221
5.4.17 Continuity ... 223
5.4.18 Siri ... 224
5.4.19 Applikationen von Drittanbietern ... 225
5.5 Netzwerk-Domäne ... 226
5.6 Zusammenfassung ... 226

6 Informationen aus Log-Dateien ... 227
6.1 Log-Dateien des Betriebssystems ... 227
6.1.1 Nutzer-/Account-Informationen ... 227
6.1.2 Software-Installationen ... 228
6.1.3 Filesystem Check ... 228
6.1.4 Storage Manager ... 228
6.1.5 WiFi.log ... 229
6.1.6 System.log ... 229
6.1.7 Periodische Log-Dateien ... 231
6.1.8 Apple System Logs ... 232
6.1.9 Audit-Logs ... 233
6.1.10 Unified Logging ... 235
6.2 Log-Dateien der Nutzer-Domäne ... 245
6.2.1 Verbundene iOS-Geräte ... 245
6.2.2 FaceTime-Verbindungen ... 246
6.2.3 Übersicht ... 246

7 Hack the Mac ... 249
7.1 Mac-OS-Nutzerpasswörter ... 249
7.1.1 Cracking des Nutzerpassworts mit Dave Grohl ... 251
7.1.2 Cracking des Nutzerpassworts mit Hashcat ... 252
7.2 FileVault 2 – Full Disk Encryption ... 256
7.2.1 FileVault2-Cracking mit JtR – EncryptedRoot.plist.wipekey ... 259
7.2.2 FileVault2-Cracking mit JtR – Image-Datei ... 264
7.3 Mac-OS-Keychains cracken ... 266
7.3.1 Angriff auf den Nutzerschlüsselbund mit JtR ... 266
7.4 Verschlüsselte Disk Images ... 267
7.4.1 Angriff auf eine verschlüsselte DMG-Datei mit JtR ... 268
7.4.2 Angriff auf eine verschlüsselte Sparsebundle-Datei mit JtR ... 268
7.5 Übung: Analyse und Cracking – Teil 1 ... 268
7.5.1 Szenario ... 268
7.5.2 Lösung: Szenario ... 270
7.5.3 Fortsetzung des Szenarios ... 273
7.5.4 Lösung: Fortsetzung des Szenarios ... 273

8 Anwendungsanalyse unter Mac OS ... 275
8.1 Tools zur Anwendungsanalyse ... 275
8.1.1 Mac OS: Aktivitätsanzeige ... 275
8.1.2 List open Files: lsof ... 277
8.1.3 Fs_usage ... 277
8.1.4 Xcode: Instruments ... 278
8.1.5 DTrace ... 280
8.1.6 FSmonitor ... 280
8.1.7 DaemonFS ... 281
8.2 Modell zur Analyse von Applikationen unter Mac OS ... 281
8.3 Anwendungsanalyse der Nachrichten-App ... 283
8.3.1 Analyseumgebung ... 283
8.3.2 Anwendungsanalyse der Nachrichten-App ... 283
8.3.3 Ansätze für eine forensische Analyse ... 304
8.3.4 Zusammenfassung ... 305

9 Random-Access-Memory-Analyse ... 307
9.1 Stand der Forschung ... 308
9.2 Struktur des RAM-Speichers ... 308
9.3 Tools zur Sicherung und Analyse ... 311
9.4 RAM-Analyse mit Volatility ... 312
9.5 Volatility-Plugin vol_logkext.py ... 314
9.6 Zusammenfassung ... 317

10 Forensische Betrachtung der Mac-Technologien ... 319
10.1 Versions ... 319
10.2 Spotlight ... 326
10.2.1 Analyse von Spotlight ... 327
10.2.2 Spotlight als Werkzeug ... 329
10.3 Time Machine ... 333
10.3.1 Time-Machine-Spuren auf zu sichernden Rechnern ... 334

10.3.2 Allgemeine Struktur eines gemounteten Time-Machine-Backups 337
10.3.3 Struktur eines lokalen Backups 344
10.3.4 Analyse von Time-Machine-Backups 345
10.4 iCloud 350
10.4.1 iCloud-Spuren unter Mac OS 353
10.4.2 iCloud-Daten sichern 360
10.5 iOS-Backups 361
10.6 Übung: Cracken eines verschlüsselten iOS-Backups 368
10.7 Übung: Angriff auf die Manifest.plist 368
10.8 Übung: Analyse und Cracking – Teil 2 369
10.8.1 Suchen mit Spotlight 369
10.8.2 Fortsetzung des Szenarios 371
10.8.3 Lösung: Suchen mit Spotlight 371
10.8.4 Lösung: Fortsetzung des Szenarios 372

11 Advanced Terminal im forensischen Umfeld 373
11.1 Basiskommandos 373
11.2 Tastaturfunktionen 374
11.3 Spezielle Kommandos 374
11.3.1 Suche nach Dateien: locate 374
11.3.2 Suche nach Dateien: find 375
11.3.3 Grep 376
11.4 Mac-OS-Kommandos 377
11.4.1 Anzeige von erweiterten Metadaten 377
11.4.2 Anzeige und Konvertierung von Plist-Dateien 378
11.5 Scripting-Grundlagen 379
11.6 Übung: Advanced Terminal 380

12 AppleScript, Automator, OS X Server 381
12.1 Ein kurze Einführung in AppleScript 381
12.2 Automator und relevante Arbeitsabläufe 383
12.2.1 Workflow: Copy Files 386
12.2.2 Workflow: Kalenderdaten parsen 387
12.2.3 Workflow: Kontakte parsen 388
12.2.4 Dienst: Dateiliste erstellen 389
12.2.5 Dienst: MD5-Hashliste erstellen 391
12.2.6 Programme: Versteckte Dateien anzeigen und ausblenden 391
12.2.7 Programme: Diskarbitration Daemon aktivieren und deaktivieren 392
12.3 Mac OS als vollwertiges Serversystem 393
12.3.1 OS-X-Server-Upgrade über den App Store 393
12.3.2 Grundlegende OS-X-Server-Einstellungen 394
12.3.3 Dateifreigaben innerhalb eines Mac-Netzwerks 395
12.3.4 Digitale Spuren zu eingerichteten Diensten 395

Literaturverzeichnis 397
Kapitel »Wichtige Hintergrundinformationen« 397
Kapitel »Das Mac-OS-Dateisystem im Fokus« 400
Kapitel »Forensische Analyse von Mac OS« 401
Kapitel »Kategorisierung digitaler Spuren« 402
Kapitel »Informationen aus Log-Dateien« 404
Kapitel »Hack the Mac« 405
Kapitel »Anwendungsanalyse unter Mac OS« 405
Kapitel »Random-Access-Memory-Analyse« 406
Kapitel »Forensische Betrachtung der Mac-Technologien« 407
Kapitel »AppleScript, Automator, OS X Server« 407

Stichwortverzeichnis 409

1 Ein Mac OS Lab aufsetzen

Besitzen Sie einen Apple-Computer mit Mac OS 10.11 El Capitan oder macOS 10.12. Sierra? Gerne möchte ich Sie dazu einladen, Ihren Computer so einzurichten, dass Sie in der Lage sind, die einzelnen Kapitel des Buchs und die zugehörigen Übungen praktisch zu bearbeiten und nachzuvollziehen. Das folgende Kapitel thematisiert das Einrichten eines eigenen Nutzer-Accounts unter Mac OS. Folgen Sie hierzu der schrittweisen Anleitung und richten Sie Ihren Mac so ein, dass sämtliche benötigte Funktionalitäten und Programme, die in diesem Buch aufgezeigt werden, zur Verfügung stehen. Im Rahmen des Buchs finden zur Durchführung von Übungen ausschließlich frei verfügbare Open-Source-Programme Verwendung. Nach dem Lesen des Buchs und Bearbeiten der Übungsbeispiele können Sie den Übungs-Account wieder aus Ihrem Mac-OS-System löschen und damit den Ausgangszustand wiederherstellen.

1.1 System Integrity Protection deaktivieren

Die Sicherheitseinstellung System Integrity Protection verhindert unter Umständen die Installation von Drittanbieterprogrammen unter Mac OS. Um eine Installation der Programme zu ermöglichen, sollte die Funktion deaktiviert werden. Führen Sie dazu folgende Schritte durch:

1. Führen Sie mit der Tastenkombination cmd + R einen Neustart des Mac im Recovery-Modus durch.
2. Öffnen Sie das Dienstprogramm Terminal und geben Sie im Terminal-Fenster den Befehl `csrutil disable` ein.
3. Damit der Befehl wirksam wird, führen Sie einen Neustart des Mac durch.

Das Deaktivieren der System Integrity Protection (abgekürzt SIP) erfolgt systemweit und beeinflusst somit sämtliche eingerichtete Benutzer.

Nach der Arbeit mit dem Buch kann die System Integrity Protection selbstverständlich wieder aktiviert werden. Hierzu werden die oben angeführten Schritte wiederholt, im Terminal wird dann allerdings der Befehl `csrutil enable` eingegeben. Der SIP-Status kann mit dem Befehl `csrutil status` abgefragt werden.

1.2 Einen Übungs-Account einrichten

Um auf Ihrem Mac-OS-System einen Übungs-Account einzurichten, sollten Sie auf Ihrem System Volume mindestens 10 GB freien Speicherplatz zur Verfügung haben. Überprüfen Sie den zur Verfügung stehenden Speicherplatz über die Menüleiste Ihres Desktops. Öffnen Sie hierzu das *Apfel*-Symbol-Menü und wählen Sie dort *Über diesen Mac*.

Bild 1.1: Das geöffnete *Apfel*-Symbol-Menü.

1.2.1 Einen neuen Benutzer-Account erstellen

Erstellen Sie einen neuen Benutzer-Account, indem Sie in den *Systemeinstellungen* Ihres Mac OS *Benutzer & Gruppen* wählen. Authentifizieren Sie sich dort mit Ihrem Administrator-Passwort und erstellen Sie den neuen Benutzer-Account *MacOS* im Administrator-Modus. Wählen Sie ein Passwort und bestätigen Sie mit *Benutzer erstellen*.

Bild 1.2: Einen neuen Benutzer erstellen.

Nach dem erfolgreichen Erstellen wird der Übungs-Account *MacOS* in der Benutzerverwaltung aufgeführt. Um ein schnelles Wechseln zwischen den Accounts Ihres Systems zu ermöglichen, aktivieren Sie unter *Anmeldeoptionen* das Optionsfeld *Menü für schnellen Benutzerwechsel*.

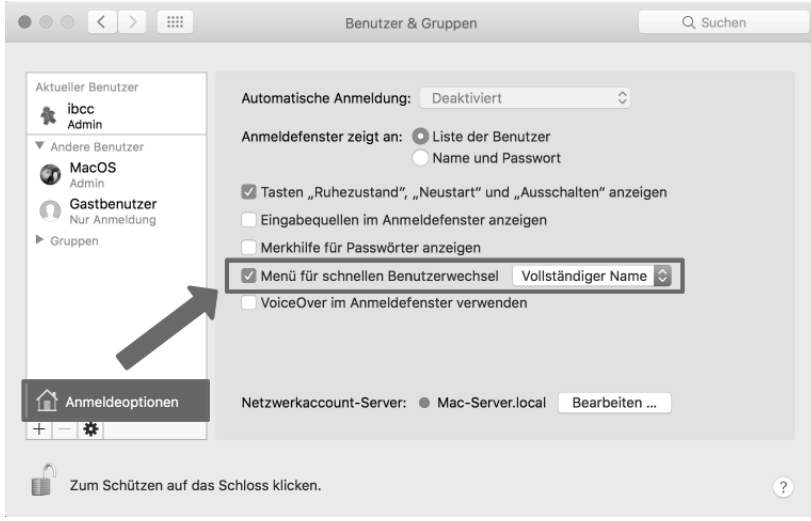

Bild 1.3: Menü für schnellen Benutzerwechsel aktivieren.

In der Mac-OS-Menüleiste sehen Sie jetzt auf der rechten Seite einen neu hinzugekommene Eintrag mit dem Namen des aktuell auf Ihrem Apple-Computer angemeldeten Benutzers – hier *ibcc*. Öffnen Sie das Menü der Schaltfläche und wechseln Sie zum neuen Benutzer *MacOS*.

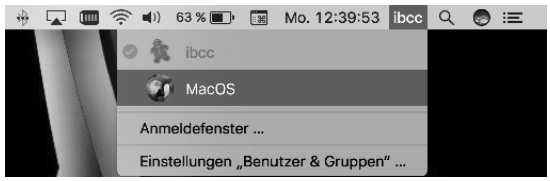

Bild 1.4: In den Übungs-Account *MacOS* wechseln.

Nach dem Wechsel in den Übungs-Account *MacOS* können Sie mit der Installation weiterer Software fortfahren. Über die Menüleiste können Sie jederzeit zurück in Ihren ursprünglichen Benutzer-Account wechseln.

1.2.2 Gatekeeper-Funktionalität deaktivieren

Die Gatekeeper-Funktionalität von Mac OS sollte ebenfalls deaktiviert werden. Unter Mac OS X in der Version 10.11 oder kleiner sollte in den Gatekeeper-Einstellungen das Optionsfeld *Keine Einschränkungen* ausgewählt werden. Unter macOS 10.12 ist es nicht mehr möglich, die Gatekeeper-Einstellungen entsprechend einzuschränken. In diesem Fall muss die Ausführung von nicht signiertem Programmcode dezidiert jedes Mal erlaubt werden.

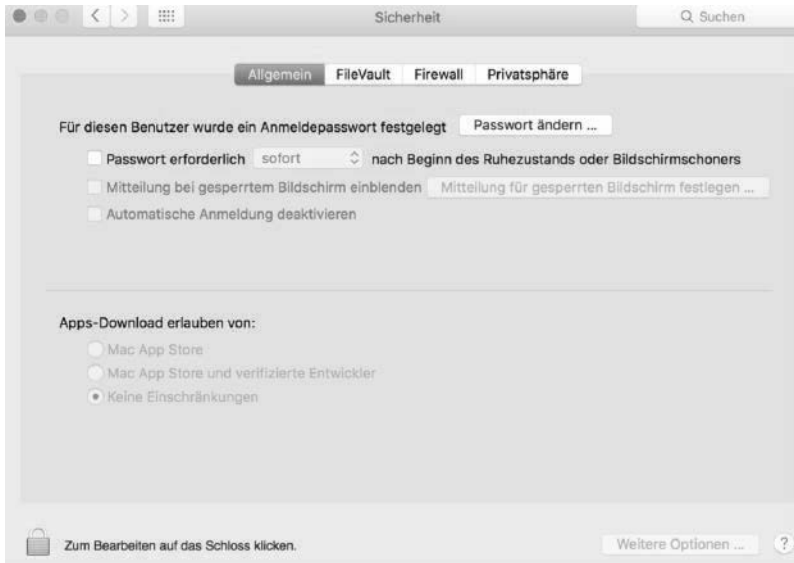

Bild 1.5: Mit der Option *Keine Einschränkungen* wird die Gatekeeper-Funktionalität deaktiviert.

1.3 Programme im Mac OS Lab installieren

Nachdem Sie den neuen Benutzer-Account angelegt haben, installieren Sie die folgenden Programme und Tools innerhalb Ihres eingerichteten Übungs-Accounts *MacOS*.

1.3.1 Java SE

Laden Sie das aktuelle Java SE Development Kit unter der URL *http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html* herunter und installieren Sie das Programm anschließend durch Ausführung der DMG-Installationsdatei.

1.3.2 Xcode

Installieren Sie die Entwicklungsumgebung Xcode für Mac OS. Registrieren Sie sich hierfür kostenfrei bei Apple und erstellen Sie einen Apple-Entwickler-Account (*https://developer.apple.com/register*). Hierüber haben Sie Zugriff auf umfangreiche kostenfreie Downloads und Informationen zu technischen Spezifikationen von Apple-Produkten und -Technologien.

Nach erfolgreicher Registrierung können Sie Xcode und die aktuellen Command Line Tools unter *https://developer.apple.com/downloads* herunterladen. Installieren Sie die beiden Programme in Ihrem Übungs-Account.

Alternativ können Sie sich im Übungs-Account mit Ihrer Apple ID anmelden und Xcode über den App Store beziehen.

1.3.3 Fuse

Laden Sie Fuse für Mac OS über die URL *http://osxfuse.github.io* herunter und installieren Sie die DMG-Installationsdatei. Fuse erweitert die nativen Fähigkeiten Ihres Mac-OS-Systems zur Verwaltung von Dateisystemen und erlaubt es Ihnen, weitere Drittanbieter-Dateisysteme zu nutzen. Fuse ist eine Voraussetzung für das Programm Xmount.

1.3.4 Mac Ports

Zur Verwendung von Linux Binaries unter Mac OS existiert die Portierungslösung Mac Ports. Mac Ports kann von der Webseite *www.macports.org* kostenfrei bezogen werden. Die Installation erfolgt durch die Ausführung der DMG-Installationsdatei. Nach Durchführung der Installation sollte Mac Ports mit dem Terminalkommando `$ sudo port -v selfupdate` aktualisiert werden. Die Bedienung von Mac Ports erfolgt über das Terminal. Mac Ports ermöglicht das Herunterladen von Linux-Programmen, die für Mac OS portiert wurden (Ports). Die Programme werden von Mac Ports heruntergeladen, Abhängigkeiten werden geprüft, kompiliert und installiert. Das Installationsverzeichnis von Mac Ports unter Mac OS ist */opt/local/var/macports*.

Mac Ports kann mit den folgenden Terminalkommandos gesteuert werden:

Terminalkommandos: Mac Ports	
`$ sudo port selfupdate`	Aktualisiert Mac Ports
`$ sudo port search Suchbegriff`	Suche nach Ports
`$ sudo port install Paketname`	Installation eines Ports
`$ sudo port uninstall Paketname`	Löschen eines Ports
`$ sudo port upgrade outdated`	Aktualisierung von Ports
`$ sudo port uninstall inactive`	Deinstalliert inaktive Ports

1.3.5 Homebrew

Eine Installation von weiteren freien Programmen für Mac OS ermöglicht Homebrew. Der Paketmanager kann von der Webseite *http://brew.sh* kostenfrei bezogen werden. Die Installation erfolgt im Terminal mit dem Befehl:

```
$ ruby -e "$(curl -fsSL https://raw.github.com/mxcl/homebrew/go)"
```

Homebrew ermöglicht das Herunterladen von Programmen sowie deren Kompilierung und Installation. Dabei verzichtet Homebrew auf Root-Rechte. Programme werden in das Verzeichnis */usr/local/Cellar/* installiert, von wo aus jeweils ein symbolischer Link nach */usr/local/bin* erzeugt wird. Sind Programme bereits im System installiert, wird kein Link erzeugt. Überschneidungen zu bereits vorhandenen Installationen werden dadurch verhindert. Die Bedienung von Homebrew erfolgt ebenfalls über das Terminal.

Homebrew kann mit den folgenden Terminalkommandos gesteuert werden:

Terminalkommandos: Homebrew	
`$ brew update`	Aktualisiert Homebrew
`$ brew search Suchbegriff`	Sucht nach einem Programm
`$ brew install Paketname`	Installiert ein Programm
`$ brew options Paketname`	Zeigt Optionen zur Installation
`$ brew uninstall Paketname`	Deinstalliert ein Programm
`$ brew outdated`	Zeigt veraltete Programme an
`$ brew upgrade Paketname`	Aktualisiert Programme
`$ brew cleanup`	Deinstalliert veraltete Programmversionen

Nach der Installation von Homebrew ist es empfehlenswert, die Umgebungsvariable von Mac OS anzupassen und das Homebrew-Verzeichnis mit aufzunehmen. Dies kann durch Hinzufügen des folgenden Eintrags am Ende der Datei *~/.profile* erfolgen:

```
$ export PATH=/usr/local/bin:/usr/local/sbin:$PATH
```

1.3.6 Libewf

Die Libewf-Bibliothek bietet Zugriff auf das EWF-Format (Expert Witness Compression), das von vielen forensischen Produkten eingesetzt wird, um Sicherungen von Datenträgern anzufertigen. Beispielsweise enthält die Bibliothek das Programm ewfacquire zur Erstellung von forensischen Abbildern von Datenträgern oder ewfmount zum Einbinden von forensischen Abbildern im EFW-Format in Mac OS. Eine Installation von libewf kann über Mac Ports erfolgen:

```
$ sudo port install libewf
```

1.3.7 Xmount

Das Programm Xmount ermöglicht das Konvertieren von verschiedenen Formaten von Festplattenabbildern untereinander. Es nutzt ein mit Fuse erzeugtes virtuelles Dateisystem, unter dem das konvertierte Datenträgerabbild verfügbar ist. Xmount kann so forensische Abbilder von Datenträgern im RAW- oder EWF-Format in das DMG-Format konvertieren, so dass sie im System eingebunden werden können. Ein anderer Einsatzbereich ist das Konvertieren in Formate wie VMDK oder VHD, um die Abbilder im Rahmen einer Virtualisierung mit Produkten wie VMware, VirtualBox, QEMU und anderen verwenden zu können. Mit Xmount konvertierte Abbilder von Datenträgern können unter Nutzung einer zusätzlichen Cache-Datei zudem beschreibbar eingebunden werden. Alle Schreibvorgänge werden dann virtuell im Cache File verarbeitet. Xmount kann über die URL *https://www.pinguin.lu/xmount* als 64-Bit-Package heruntergeladen und die DMG-Installationsdatei anschließend installiert werden.

1.3.8 Sleuth Kit

Sleuth Kit ist eine mächtige Sammlung von Werkzeugen zur Analyse von Datenträgern oder Abbildern von Datenträgern. Sleuth Kit gehört in das Repertoire eines jeden IT-Forensikers und ermöglicht eine profunde und manuelle Analyse von Partitionierungsschemata und Dateisystemen. Die Werkzeugsammlung kann über Mac Ports bezogen werden.

```
$ sudo port install sleuthkit
```

1.3.9 SQLite-Browser

Zur Analyse von SQLite-Datenbanken unter Mac OS kann beispielsweise der kostenfreie SQLite-Browser genutzt werden. Die DMG-Installationsdatei des Programms ist unter der URL *http://sqlitebrowser.org* verfügbar.

1.3.10 Hex-Editoren

Installieren Sie einen der beiden kostenfreien Hexadezimaleditoren Hex Fiend oder 0XED. Hex Fiend kann als ZIP-Datei über die URL *http://ridiculousfish.com/hexfiend* heruntergeladen, entpackt und in das *Programme*-Verzeichnis von Mac OS verschoben werden. 0XED ist ebenfalls als ZIP-Datei über die URL *http://suavetech.com/0xed* erhältlich.

1.3.11 Github

Github ist ein Onlinedienst zur Bereitstellung von Software-Projekten. Entwickler können diese über sogenannte Repositories (von Github verwaltete Verzeichnisse) zur Verfügung stellen. Zur Installation von Programmen über Github installieren Sie das Programm Git über Homebrew.

```
$ brew install git
```

1.3.12 Python

Mac OS enthält bereits als Standard die Installation von Python 2.7. Eine direkte Nutzung von Python ist somit möglich. Es empfiehlt sich allerdings, zur Nutzung und zur Entwicklung von Python-Programmen zusätzlich die offiziellen Python-Releases zu installieren. Diese können über die URL *https://www.python.org/downloads/mac-osx* bezogen werden. Die offiziellen Stables bieten jeweils den neuesten Versionsstand von Python.

Eine Installation von Python 2.7 und/oder Python 3 kann sehr einfach über das Terminal unter Nutzung des Paketmanagers Homebrew durchgeführt werden:

```
$ brew install python
$ brew install python3
```

Die Installation von Python über Homebrew beinhaltet automatisch auch die Installation der Python-Paketmanager pip und Setuptools, die eine einfache Installation von Drittanbieter-Libraries für Python ermöglichen. Im Terminal kann die installierte Version für Python 2.7 mit folgendem Befehl ermittelt werden:

```
$ python --V
```

Die Überprüfung einer Installation von Python 3 erfolgt analog mit dem Befehl:

```
$ python3 --V
```

Weitere Software
Mit der Installation der beschriebenen Programme haben Sie die Basis-Einrichtung des Übungs-Accounts *MacOS* abgeschlossen. Im weiteren Verlauf des Buchs werden Sie zur Durchführung von Übungen aufgefordert, weitere Software hinzuzufügen. Die Installation wird dann in den entsprechenden Kapiteln gesondert beschrieben.

1.4 Den Übungs-Account wieder löschen

Falls Sie das Buch gelesen und die Übungen abgeschlossen haben, können Sie den Übungs-Account *MacOS* wieder von Ihrem System löschen. Hierzu wählen Sie in den *Systemeinstellungen* den Punkt *Benutzer & Gruppen*. Dort können Sie den Übungs-Account nach der Authentifizierung mit Ihrem Administratorkennwort durch Auswahl der *Minus*-Schaltfläche löschen.

2 Wichtige Hintergrundinformationen

Das Kapitel gibt wichtige Hintergrundinformationen zum Betriebssystem Mac OS. Im ersten Abschnitt wird die Entwicklung hin zur aktuellen Betriebssystemversion macOS 10.12 Sierra und die damit verbundene Entwicklung der Apple-Produktfamilie vorgestellt. Die folgenden Abschnitte »Mac OS intern« und »Mac-OS-Architektur« beschäftigen sich mit wichtigen Grundfunktionalitäten des Betriebssystems und internen Mechanismen. Anschließend gibt der Abschnitt »Mac-OS-Sicherheitskonzepte« einen Einblick in implementierte Sicherheitsmechanismen. Am Ende des Kapitels werden die wichtigsten Erkenntnisse nochmals zusammengefasst.

2.1 Das Betriebssystem Mac OS

Mac OS hat sich seit seiner Einführung im Jahr 1984 durch den kalifornischen Computerhersteller Apple von einem Nischen-Betriebssystem zu einem der erfolgreichsten Betriebssysteme überhaupt entwickelt. Produkte wie MacBook, MacBook Pro, MacBook Air oder iMac erfreuen sich insbesondere in den letzten Jahren einer explosionsartigen Popularität.

Apple konnte als einer der wenigen Computerhersteller, die Hard- und Software in Produkteinheit vertreiben, sein Mac OS zum weltweit erfolgreichsten UNIX-basierten Betriebssystem aufsteigen lassen. Mit der Einführung des iPhones im Jahr 2007 hat die mobile Variante iOS den Siegeszug der Smartphones eingeleitet und eine völlig neue Ära für mobile Endgeräte bzw. Computer eröffnet. Heute ist Apple das wirtschaftlich erfolgreichste Unternehmen weltweit. Im November 2015 überschritt der Marktwert des Unternehmens erstmals die Marke von 700 Milliarden Dollar. Damit ist Apple das wertvollste Unternehmen aller Zeiten und lässt Konkurrenten wie Microsoft (394 Milliarden Dollar) weit hinter sich. Apple ist mit seinen Produkten ein Trendsetter für technische Innovationen und steht für hochwertige, moderne und leistungsfähige Geräte.

Diese Entwicklung ist kein Zufall. Apple verbesserte sein Betriebssystem Mac OS und die damit verbundenen Computer in einem jahrelangen Prozess ständig weiter. Ein Quantensprung in der Entwicklung war die Einführung von Mac OS X im Jahr 2001, welches das bis dahin nicht mehr konkurrenzfähige Mac OS Classic ablöste. Mac OS X war eine völlige Neuentwicklung. Begonnen mit dem Kernel, wurde die Betriebssystemarchitektur komplett neu entworfen und damit die Basis für die bis heute andauernde Entwicklung von Mac OS X gelegt. Die endgültige Befreiung als Nischenprodukt gelang Apple mit dem Wechsel seiner Prozessorarchitektur von PowerPC- zu Intel-Prozessoren im Jahr 2006. Die derzeit aktuelle Mac-OS-X-Version 10.12, auch als macOS Sierra bezeichnet, wurde am 20.09.2016 veröffentlicht. Die mobile Variante des Betriebssystems iOS 10

wurde am 13.09.2016 veröffentlicht. Die Funktionalitäten und Besonderheiten von OS X werden in den folgenden Abschnitten thematisiert.

Damit Sie Mac OS X und die besonderen Features des Betriebssystems besser verstehen, will das Buch zu einer kleinen Reise in die Vergangenheit einladen, die zeigt, wie sich Mac OS X über die Jahre entwickeln konnte. Dabei geht es nicht darum, jede technische Neuerung aufzuzählen. Apple führt teilweise von Version zu Version Hunderte neuer Features ein, insbesondere in der Mac OS X GUI(Graphical User Interface) oder den Anwendungsframeworks. Diese können bei Bedarf den technischen Spezifikationen der einzelnen Versionen bzw. Wikipedia entnommen werden. Vielmehr geht es darum, globaler auf die Entwicklung der entscheidenden Eckpunkte und Technologien von Mac OS X zu blicken.

2.2 Applelution in Cupertino

Die Firma Apple Computer wurde 1976 von Steve Jobs, Steve Wozniak und Ronald Wayne in Cupertino (USA, Kalifornien) gegründet. Ronald Wayne verließ das Gründungstrio einige Tage später. Der erste Computer wurde im gleichen Jahr entworfen und in sehr geringer Stückzahl (200 Computer) unter dem Namen Apple I für 666,66 US-$ veräußert. Im Juni 1977 entwickelten Jobs und Wozniak das Nachfolgemodell Apple II. Die Architektur des Apple II war für die damaligen Verhältnisse sensationell. Jobs und Wozniak gelang es, die benötigte Hardware in ein recht flaches Gehäuse zu integrieren. Der Apple II konnte auf einem entsprechenden Monitor farbige Grafiken darstellen und mit einer Erweiterungskarte an einen Fernseher angeschlossen werden. Als Massenspeichermedium nutzte er bereits damals ein Floppy-Laufwerk mit 5 1/4-Zoll-Disketten. Der Apple-II-Computer war einer der ersten 8-Bit-Computer, die eine weite Verbreitung fanden.

Als Betriebssystem kam das von Wozniak entwickelte kommandozeilenbasierte Apple Integer Basic zum Einsatz. Der Rechner war aufgrund seines Designs, seines modularen Aufbaus und seiner Erweiterungsmöglichkeiten eine Innovation. Das verbesserte Modell Apple II+ wurde 1979 entwickelt und fand erstmals auch in Europa Verbreitung. Im Apple II+ kam das von Microsoft entwickelte Betriebssystem Applesoft BASIC zum Einsatz. Es bot erweiterte Funktionalitäten wie das Rechnen mit Gleitkommazahlen oder Befehle zur Darstellung hochauflösender Grafiken.

Bild 2.1: Apple II

Die von Microsoft entwickelte Codebasis von Applesoft BASIC wurde 1982 auch im Commodore BASIC des Commodore 64 benutzt. Zur Verbreitung des Computers trug außerdem maßgeblich das 1979 von Daniel Bricklin und Bob Frankston an der Harvard Business School erfundene Programm VisiCal bei. VisiCal war die erste Tabellenkalkulation überhaupt und konnte damals exklusiv auf Apple-II-Computern genutzt werden. Die Grundideen von VisiCal wurden später beispielsweise von Lotus und Microsofts Excel übernommen.

Zwar war der Apple-II-Computer bereits für eine größere Käuferschicht interessant, dennoch erforderte die Bedienung des Rechners durch das kommandozeilenorientierte BASIC ein gewisses Maß an technischem Verständnis. Die Vision von Jobs war es, einen Computer zu entwickeln, der von jedermann bedient werden kann. Ein erster Schritt in diese Richtung war das 1983 entwickelte Apple LISA OS, eines der ersten grafischen Betriebssysteme. LISA OS war allerdings zu teuer, um kommerziell Erfolg zu haben. Unter dem Begriff Macintosh wurde schließlich 1984 im Rahmen eines Werbespots zum Superbowl der erste kommerziell erfolgreiche Personal Computer mit einem grafischen Betriebssystem eingeführt.

Bild 2.2:
Blick zurück auf LISA OS

Der Begriff Macintosh (im Weiteren auch verkürzt als Mac bezeichnet) sollte in Folge für die Apple-Betriebssysteme gelten. Das 1984 eingeführte Mac OS oder Mac OS Classic unterstützte die Bedienung per Maus und besaß als zentrales Element den bis heute bekannten Dateibetrachter Finder.

Ebenfalls im Jahr 1984 stellte Apple Computer John Sculley als Konzernchef ein. Die anfänglich gute Beziehung von Steve Jobs und John Sculley wandelte sich mehr und mehr zu einer Auseinandersetzung, die schließlich im Austritt von Steve Jobs aus der Firma Apple Computer 1985 ihr Ende fand.

Apple Computer konnte den anfänglichen technologischen Vorsprung in der Folge nicht halten. Das Mac-OS-Betriebssystem konnte nicht schnell genug fortentwickelt werden und verlor im Vergleich zu den mittlerweile vorhandenen Betriebssystemen Linux, BSD und vor allem Microsoft Windows (3.0 bis NT) den Anschluss. Eklatante Schwächen von Mac OS waren insbesondere die nur schwach entwickelten Multitasking-Fähigkei-

ten (sog. kooperatives Multitasking) und der nicht vorhandene Speicherschutz bei Programmausführungen.

Die kurz vor der Insolvenz stehende Firma Apple Computer entschloss sich 1997 zu einem radikalen Wandel, stellte Steve Jobs wieder ein und kaufte zudem die zwischenzeitlich von Steve Jobs aufgebaute Firma NeXT auf. Mit der Wiedereinstellung von Jobs änderte sich die Unternehmenskultur dramatisch. Jobs reduzierte das Unternehmensportfolio auf einige wenige Produkte und forcierte die Entwicklung eines neuen Betriebssystems Mac OS X. Mac OS X ist seit 2002 das alleinige Betriebssystem für Apple-Computer.

Bild 2.3:
Blick zurück auf Mac OS 10.0

Die neue Produktstrategie sah vor, den Computer mehr und mehr in den Lebensmittelpunkt zu stellen (digital hub) und in alltägliche Abläufe zu integrieren.

Die Entwicklung von Software wie iTunes zur Medienwiedergabe, iPhoto zur Verwaltung und Bearbeitung von Fotos oder iMovie für die Bearbeitung von Filmen ließen Mac-Computer und Mac OS X für Heimanwender immer interessanter werden. Marktinnovationen waren der 2001 vorgestellte MP3-Player iPod und der iTunes Music Store, in dem bis heute online Musik eingekauft werden kann.

Im Zuge der Umstellung der Prozessorarchitektur von PowerPC- auf Intel-Prozessoren erfuhr auch die Mac-Produktfamilie eine Neusortierung. Apple Computer führte 2006 die bis heute gebräuchlichen Modelle MacBook Pro, iMac, MacBook, Mac Pro und Mac mini ein. Ein weiterer Meilenstein war die Vorstellung des iPhones und des Apple TV im Jahr 2007. Beide nutzten als Betriebssystem iOS, eine im Kern mobile Variante von Mac OS X für ARM-Architekturen. Apple Computer erweiterte sein Portfolio damit um weitere Produktlinien außerhalb des eigentlichen Computer-Kernsegments. Symbolisch verdeutlichte die Firma dies durch eine Umbenennung von Apple Computer ins schlichtere Apple. Als weitere neue Produktlinien folgten schließlich 2010 das iPad (Betriebssystem iOS) und 2014 die Apple Watch (Betriebssystem watchOS, basierend auf iOS).

2.2.1 Apple-Betriebssystem-Modelle

Die Produktlinie von Apple besteht aus Mac-Systemen, die prinzipiell auf drei unterschiedlichen Betriebssystem-Modellen basieren:

- Das Betriebssystem Mac OS X findet sich in den Mac-Desktop-Rechnern und Laptops wieder. Als Desktop-Rechner sind die Modelle Mac Pro, Mac mini und iMac erhältlich, im Laptop-Bereich die Geräte MacBook, MacBook Air und MacBook Pro. Sämtliche Desktop- und Laptop-Modelle basieren derzeit auf Intel-Core-i- oder Core-M-Prozessoren (Broadwell/Skylake/Kaby Lake) in unterschiedlichen Ausführungen.
- Das mobile Betriebssystem iOS läuft in den Produktfamilien iPhone, iPod Touch, iPad und dem Apple TV. Das auf Mac OS X basierende iOS ist eine für mobile Geräte angepasste Betriebssystemversion für ARM-Architekturen. Dabei werden die Geräte von einem von Apple entwickelten SoC (System on a Chip) angetrieben. Diese SoCs kombinieren ARM-CPU, Grafikprozessor und Hauptspeicher auf einem Chip. Die erste SoC-Generation Apple A4 wurde im iPhone 4 und dem iPad der ersten Generation verbaut. Die derzeit aktuellen SoCs A10 bzw. A10X kommen im iPhone 7, 7 plus und im iPad Pro zum Einsatz.
- Eine Besonderheit ist die Apple Watch, deren Betriebssystem watchOS wiederum ein stark verschlanktes und funktionseingeschränktes iOS als Basis hat. Die dritte Produktlinie besitzt ein proprietäres embedded OS und besteht aus den Geräten iPod Shuffle, iPod Classic und iPod Nano.

2.3 Mac OS X intern

Als Steve Jobs 1985 Apple Computer verlassen hatte, gründete er die Firma NeXT. Unter dem Label NeXT entstand das Betriebssystem NeXTSTEP. NeXTSTEP basierte auf dem Mach Microkernel und unterstützte bereits Speicherschutz, Threads, Multitasking und Interprozesskommunikation durch Versenden sogenannter Mach-Messages. Das Betriebssystem war zudem objektorientiert. Es beinhaltete diverse Frameworks, unter anderem zur Entwicklung von Programmen unter der grafischen Oberfläche und zur Unterstützung von Gerätetreibern (DriverKit).

Als Programmiersprache zur Entwicklung von Programmen konnte das ebenfalls objektorientierte Objective-C eingesetzt werden. NeXTSTEP nutzte bereits das Konzept von sogenannten Bundles für Anwendungsprogramme und Bibliotheken. Bundles bestehen aus einer Verzeichnisstruktur und entsprechenden Dateien und können zu einer Bundle-Datei (Paketdatei) gepackt werden. NeXTSTEP nutzte Bundles beispielsweise, um Software und entsprechende Abhängigkeiten durch einfaches Verschieben der Bundles zu installieren bzw. zu deinstallieren.

Als Apple 1997 die Firma NeXT übernahm, erwarb sie vor allem deren technologische Entwicklungen. Das Betriebssystem NeXTSTEP selbst wurde nicht weiterentwickelt, die NeXTSTEP-Architektur hingegen nutzte Apple als Grundlage für die Konzeption eines gänzlich neuen Betriebssystems. Das angestaubte Mac OS mit seiner Finder-basierten GUI wurde kombiniert mit dem technisch fortschrittlichen, aber weniger schönen

Stichwortverzeichnis

Symbole

0XED 25

A

Address Space Layout Randomisation 310
AFP 112, 199, 230, 394, 395
Agents 51
Aktivitätsanzeige 275
Allocation File 81, 82, 84, 86, 94
Allokationsblock 80, 81, 85, 86
Anmeldeobjekte 209
Anwendungsanalyse 9, 225, 275, 280, 305
APFS-Container 74, 109, 115, 117
Apple 27
Apple Bonjour 33
Apple File System 8, 108, 109, 126, 228, 246, 400
Apple File System Guide 112
Apple I 28
Apple II 28
Apple II+ 28
Apple Integer Basic 28
Apple LISA OS 29
Apple Script 33
AppleScript 9, 33, 381
AppleScript Studio 33
Applesoft BASIC 28
Apple System Logs 232
Apple TV 30
Apple Watch 30, 37
Applikationen 56
Applikationsschicht 46
App Store 34
  Einstellungen 187
Aqua 33
ASLR 34, 310
ATS 35
Attributes File 79, 81, 82, 84, 87, 90, 100, 101, 102, 326
Audio 33
Audit-Logs 233
Auto Login 182
Automatic Termination 310
Automator 9, 33, 383, 384, 385

B

Bash History 202
B-Baum 87, 89, 93, 94, 95
Befehle für Datenträger und Partitionen 75
Betriebssystem
  iOS 31
  Mac OS X 31
  watchOS 31
Big-Endian 73, 80
Bildschirmfreigabe 144, 196, 197, 198, 200
Bitweise Kopie 134
BlackBag 146, 150, 366, 401, 402, 403
Block Devices 149
Bluetooth-Geräte 185
Bluetooth-Verbindungen 203
Bob Frankston 29
BoM-Datei 186
Bootbare USB-Medien 150
Boot Camp 34, 49
boot.efi 47, 48
Boot.efi 50
bootstrap 51
Bricklin, Daniel 29
Brute-Force 191, 250, 251, 252, 253, 255, 368
B-Tree 87
Bundles 31, 53, 55, 56, 178, 211, 225, 284

C

Case-Anweisung 177
Casey 132, 133, 135, 401
Catalog File 79, 81, 82, 84, 87, 90, 95, 96, 326
Catalog Node ID 82, 94, 95, 328, 341
Chain of Custody 133
Cheetah 33, 79
Clump 81, 84, 85, 93
Cocoa 32, 44, 46, 107, 240, 404
Code Signing 43, 57
Commodore BASIC 29
Compressed Memory 35, 146, 307, 310, 317
Container-Superblock 117
Continuity 35, 223
Copy-On-Write 112
Core Data 33
Core Dumps 43
Core-OS-Schicht 43
Core Services 44
Core Storage 34, 69, 74, 75, 77, 78, 109, 123, 149, 155, 257, 258, 260, 264, 265
Core-Storage-Volume-Header 258, 260
Cracking 9, 251, 252, 257, 265, 266, 369, 407
Cross Clipboard 36
CROSS JOIN 173
crowbarDMG 267
Cupertino 28
CUPS 33

D

DaemonFS 281, 292, 405
Daemons 51
Darwin 37
Data Fork 78, 79, 81, 97
Dateifreigabe 199, 326
Dateikomprimierung 103
Dateisystem-Snapshots 108, 111, 112, 123
Dateisystem-Support 39
datetime() 176
dcfldd 148
dd 72, 73, 75, 85, 148, 153, 154
Device Tree 48
diagnosticd 236
Disk Arbitration 43, 155, 156, 157, 179
Disk Identifier 75
Disk Images 113, 178, 179, 259, 267, 268, 326, 329
diskutil 69, 75, 110, 113, 258
DMG-Images 178
Dock 33, 62, 63, 64, 152, 206, 209, 293
Download 57, 59, 153, 169, 182, 197, 203, 217, 368
Drucker 189, 203
DTrace 278, 280, 292, 298

E

EFI 46, 50
EFI-Boot-Services 47
EFI-Runtime-Services 47
egrep 376
El Capitan 19, 35, 60, 222
Elcomsoft 305, 353, 361, 365, 368
EnCase 113, 132
EncryptedRoot.plist.wipekey 260
Entitlements 57
Erweiterte Metadaten 100, 337, 338, 374
ewfacquire 24, 150, 159
ewfmount 24, 76, 180
ExFAT 40, 178
Exposé 33
Extended File Allocation Table 40
Extensible Firmware Interface 46
Extension Data Attribute 100, 101
Extent 79, 85, 94, 97
Extents Overflow File 81, 82, 87, 90, 97, 100

F

Facebook 35
FaceTime 219, 220, 221, 247
FAT 40, 79, 178
Fat Binaries 34
File Allocation Table 40
File Quarantine 59
FileVault 33
FileVault 2 34, 69, 112, 123, 149, 155, 249, 256, 257, 258, 260, 305
find 61, 375
Finder 29, 35, 37, 41, 42, 56, 62, 63, 64, 84, 96, 105, 141, 163, 179, 205, 344, 384
Firewall 143, 192, 193, 404

Firmware-Passwort 47, 146, 150, 153, 156
Fork Data Attribute 100, 101
Fotos 218
Frameworks 56, 57
FreeBSD Kernel 37
Freigaben 63, 142, 144, 195, 196, 226
FSeventer 281
FSEvents 34
FSmonitor 280, 281
fsstat 75, 85
fs_usage 277
Full Disk Encryption 34
Full-Screen-Modus 34
Fuse für Mac OS 23
Fusion Drive 69, 77, 78, 123, 148, 149, 155, 156, 257, 304, 401

G

Garbage Collection 69
Gatekeeper 21, 43, 58, 59
GCP 52
Github 25
Global Unique ID 288
Grand Central Dispatch 44, 52, 399
grep 140, 227, 228, 230, 376
Grohl, Dave 184, 251ff., 403
GUI 28
GUID-Partitionsschema 70

H

Handoff 35
Hardening 57
Hardlinks 105, 106
Hashcat 184, 251, 252, 253, 365, 369
hdiutil 75, 76, 78, 85, 124, 126, 179, 180, 264
Header Node 89, 91, 92
Hex-Editoren 25
Hex Fiend 25
HFS+ 39
HFS+ case-sensitive 39
HFS+ Special Files 79, 85
HFSX 39
HFS+-Zeitstempel 107
Hierachical File System 39
Hierarchical File System 78
Hintergrund-Systemprozesse 193, 194
Homebrew 23

I

iBooks 35
iChat 33
iCloud 9, 34, 35, 36, 48, 142, 143, 188, 190, 219, 221, 225, 244, 245, 256, 259, 288, 290, 292, 302, 305, 320, 350, 352, 353, 360, 368, 407
iMac 27, 30
Image 33
iMessage 35
Index Node 91
Inline Data Attribute 100, 103
INNER JOIN 173, 174, 176
Installations-DVD 34
Installationszeitpunkt 181
Installation von Software 186
install.log 228
Instruments 278, 279, 280
Intel 33
interaktive Tastaturübersicht 66
Interprozesskommunikation 31, 37, 38, 52
I/O-Kit 37, 48
IOKit 32
ioreg 49
iOS 27, 30, 31
iOS Backups 369
iPad 30
iPhone 30
iPhoto 30
iPod 30
istat 104, 347
iTunes 30
iTunes-Backup 305, 368

J

Jaguar 33
Java SE 22
John the Ripper 184, 191, 251, 256, 259, 263, 265, 268, 403
Journal 84, 103

K

Kalender 213
Kernel- und Treiber-Schicht 39
Keychain 35, 45, 145, 190, 191, 256, 266, 267, 297, 300, 307, 353
kMDItemDownloadedDate 121, 299
kMD-Items 326, 377

kMDItemWhereFroms 121, 123, 299, 331, 372, 380
Kontakte 211
Kooperatives Multitasking 30

L

Laufzeitanalyse 292
Launch Agents 193, 194
Launchd 50
Launch Daemons 194, 195, 231
Launchpad 34, 64
Leaf Nodes 89, 95
Leopard 34
Libewf 24
Linux-Live-Distribution 151, 153
Lion 34, 405
Live-CD 154
Live-Forensik 135, 141, 202
Live Log Stream 237
Live Response 135, 137, 138, 140, 145, 150, 226, 311
locate 374
log 181, 186, 189, 190, 224, 231, 238, 241, 242, 243, 244, 395
logd 236
Log-Dateien 227
Logical Block Addressing 68
logische Sicherung 136, 147, 155
Logische Volumes 69
logKext 314, 315, 316, 407
Lokale Domäne 161, 181
lsof 277, 292, 311, 315

M

Mac 29
Mac App Store 34
MacBook 27, 30
MacBook Air 27
MacBook Pro 27, 30
Mac Epoch 107, 108, 217, 223
Mach 32
Mach-Messages 31
Mach Microkernel 31
Mach-O-Binaries 53
Mach Task 52
Macintosh 29
maclife.d 280, 292, 298
Mac mini 30
macOS 32
Mac OS 27, 29, 32
  Zeitstempel 107
Mac-OS-Architektur 27, 38, 39
Mac OS Classic 29
Mac OS Integer 107
Mac OS Lab
  Fuse 23
  Github 25
  Hex-Editoren 25
  Homebrew 23
  Java SE 22
  Libewf 24
  Mac Ports 23
  Python 25
  Sleuth Kit 25
  SQLite-Browser 25
  Xcode 22
  Xmount 24
Mac OS Layer 39
Mac-OS-Notationen 37
macOS Sierra 8, 19, 27, 35, 37, 58, 59, 62, 112, 113, 143, 211, 213, 219, 222, 224, 225, 228, 235, 247, 250, 264, 275, 280, 283, 286, 307
Mac OS X 27, 31, 32
  GUI 28
  Versionen 32
Mac Ports 23
Mac Pro 30
MacQuisition 146, 150, 156, 311, 402
Mail 33, 214
Map Nodes 89
Maps 35
Mavericks 35
MBR 67, 70, 71, 125
mdfind 331, 371
mdls 203, 326
mdutil 179
Media-Schicht 45
Metal Framework 35
mount 141, 147, 154, 180, 230, 313
Mountain Lion 34
Mounten 44, 156, 179, 180
mount_hfs 180
MSAB XRY 366
MTMFS

Dateisystem 345
Multi-Touch-Gesten 34

N

Nachrichten 219
Netzwerk-Domäne 226
NeXT 30, 31
NeXTSTEP 31
Nicht-persistente Spuren 9, 311
Notizen 221
NSKeyedArchiver Format 163, 164, 206
NTFS 40
Nutzer-Domäne 161, 188, 201, 226, 245
Nutzerpasswörter 184, 249, 253
NVMe-Controller 152, 153
NVM Express 152
NVRAM 47, 48

O

Objective-C 31, 34
Open Directory 394
opensnoop 280
OS X Server 9, 393
OUTER JOIN 174

P

Packages 55
Panther 33
Papierkorb 204
Partitionstabelle 70, 71, 72, 73, 74, 115, 125
Partitionstypen und GUIDs 74
Passware 184, 251, 257, 305, 361, 365
PBKDF2 184, 250, 257, 403
Periodische Log-Dateien 231
Persistente Spuren 8, 9, 161, 226
Physikalische Volumes 69
plutil 162, 164, 212, 292, 372, 378
POSIX-Signale 53
Post-Mortem-Analyse 147, 162, 226, 264, 305, 353
Power Napping 135
PowerPC 33
PRAM 47, 156
Predicate-Filter 239, 240
Preemptives Multitasking 38
Preemtives Scheduling 52
Primary-GPT-Header 70
Primary Partition Table 71
Property List Files 162
Protective MBR 70
Proximity 37
Prozesse 51
Puma 33
Python 25, 34

Q

Quartz Extreme 33
Quick Look 34

R

RAM-Sicherung 136, 145, 307
Recon Imager 311
Recovery-Modus 19, 60, 154
Recovery-Partition 153, 154, 156, 256, 257, 258
rEFIND 50
rEFIT 50
Resource Fork 78, 79, 97
root-Benutzer 202
Ruby 34
rwsnoop 280, 292, 293, 298

S

Safari 33, 36, 217
Sandbox 57, 58
Sandboxing 43
Schloss-Icon 179
Schlüsselbund 190, 256, 266, 352
Schreibtisch & Bildschirmschoner
aktive Ecken 143
Schützender Master Boot Record 70
Sculley, John 29
Secondary Partition Table 71
Secure Virtual Memory 309
select-Abfragen 171
Seriennummer 142, 225, 230, 243, 363
Server.app 393, 394
Single User Mode 150, 156
SIP 19
SIP-Status 19
Siri 36, 224
Sleuth Kit 25, 75, 82, 106, 113, 203, 260, 373
SMB 44, 199, 200, 230, 394, 395
Snow Leopard 34
Solid State Disks 68, 161

Soziale Medien 35
Spaces 34, 64, 65, 136, 140, 207, 208
Sparse Bundle 178
Sparse Disk Image 178
Split Screen 65
Spotlight 9, 33, 34, 100, 102, 179, 180, 203, 275, 281, 319, 326, 332
Spotlight Indexing 179
sqlite3 169, 170, 173, 176, 291
SQLite-Browser 25
SQLite-Datenbank 167
SQLite Joins 173
SSD 68
SSH 144, 210
Startup File 81, 82, 84
Steve Jobs 28
Steve Wozniak 28
strings 54, 145, 190, 191, 218, 266, 301, 309, 313
Structured Query Language 169
Sumuri Recon 156, 180, 191, 366
Sumuri Recon Imager 146
SWAP Dateien 309, 310
SWIFT 35
syslog 232
System-Domäne 181
Systeminformationen 137, 138, 311
System Integrity Protection 19, 35, 59, 60, 400
System.log 229
Szenario 268

T

Tabs 35
Tags 35
Tapbacks 37
Target Disk Mode 47, 155, 156
Threads 51
Tiger 33, 383
Time Machine 9, 34, 35, 112, 142, 144, 190, 240, 241, 242, 319, 333, 342, 345
tmutil 345
Triage 136, 138, 145
TSK Befehle 75
Twitter 35

U

Ubiquity 350
Übungs-Account 19
UDF 40
UEFI 46
UFED Cellebrite 366
Unified Logging 235
Unified-Logging-System 8, 236, 240
Unique Device ID 363
Unix Epoch 107, 203, 230, 324
User Data Record 92, 93

V

Verbundene iOS Geräte 245
Verschlüsselung 39, 50, 108, 112, 120, 123, 136, 143, 147, 155, 256, 257, 305, 309, 365
  TLS 35
  vollständige 34
Versionen 32
Versions 9, 34, 319, 323, 324
Vertrauenswürdige Binaries 137
Verzeichniseintrag 96
VFS 40
Video 33
Virtuelle Maschinen 139, 140, 149
VisiCal 29
VNC 144, 196, 197
VNC-Passwort 197
Volatility 9, 308, 311, 312, 314, 315, 316, 317, 407
Volatility Plugins 312
Volume Header 81, 83, 85, 117, 119, 260

W

watchOS 31
Wayne, Ronald 28
Wear Leveling 69
Webkit Integer 107
Whatis-Datenbank 231
Widgets 33
WLAN-Netzwerke 185
Workflow 385
Write Ahead Logdatei 168
Wurzelknoten 89

X

xattr 203, 377
Xcode 22, 42, 48, 60, 292
xmount 75, 76, 180, 264
Xmount 23, 24
XNU-Kernel 37, 38, 39, 47, 48, 50, 60, 62, 309
XPC 43, 60, 61
XProtect 59
X-Ways 104, 113, 132, 322, 349
xxd 163, 372

Y

Yosemite 35

Z

Zeitzone 182
zuletzt im System eingeloggte Nutzer 182
Zwei-Faktor-Authentifizierung 352

Marc Brandt

Mac OS Hacking

Dieses Buch ist eine umfangreiche Anleitung und Hilfe für alle IT-Forensiker, Daten-Analysten und interessierten Mac-Nutzer, die erfahren wollen, welche unentdeckten Möglichkeiten im Apple-Betriebssystem schlummern. Es liefert fundiertes Grundlagenwissen und führt durch alle wichtigen Prozesse einer forensischen Betriebssystem-Analyse.

Im Verlauf des Buchs lernen die Leser verschiedene Sicherungs- und Analysetechniken für digitale Spuren kennen: Neben der Lokalisierung und Kategorisierung digitaler Spuren beschreibt es praktische Analyse-Methoden wie das Aufspüren von kritischen Informationen in Plist- oder SQLite-Dateien oder die Identifikation von Passwörtern durch Sicherung und Analyse des Hauptspeichers.

Mit den richtigen Konsolen-Befehlen finden Sie versteckte Informationen in Mac OS

In seinen 12 umfangreichen Kapiteln widmet sich das Buch auch den Dateisystemen APFS und HFS+, der Skriptsprache AppleScript, Mac OS X Server sowie der fortgeschrittenen analytischen Nutzung des Terminals. In zahlreichen Übungen, die auch zum Download bereitstehen, können die Leserinnen und Leser das neu gewonnene Wissen direkt in praxisnahen Szenarien anwenden.

Aus dem Inhalt:

- Mac OS Lab: Richten Sie einen Übungs-Account auf Ihrem Mac ein
- Hintergrund: Lernen Sie Architektur und Sicherheitskonzepte in Mac OS X kennen
- Mac-OS-Log-Dateien: Nutzen Sie die mächtigen Logging-Funktionen als Fundgrube für Informationen
- Hack the Mac: Alles über Passwörter, Disk Encryption und Keychains
- Flüchtige Spuren finden: Untersuchen Sie RAM, Auslagerungsdateien und Cloud Storage.
- Advanced Terminal: Entdecken Sie das wahre Potenzial der Kommandozeile

Über den Autor:

Marc Brandt ist Dozent an der Hochschule für Polizei Baden-Württemberg. Hier hält er schwerpunktmäßig Seminare zu den Themengebieten Mac, Mobile Devices und Network Forensics. Sein technisches Wissen baute er im Einsatz bei einer Spezialdienststelle der Polizei Baden-Württemberg auf. Seit 2007 arbeitet er als IT-Forensiker am Polizeipräsidium Mannheim. Im Jahr 2015 krönte er seine Leidenschaft für digitale Spuren mit dem Abschluss Master of Science in Digitaler Forensik an der Hochschule Albstadt-Sigmaringen. Im Rahmen seiner Tätigkeit als Dozent erstellte er bereits zahlreiche Skripte und Papers für IT-Experten innerhalb der Polizei und führt entsprechende Seminare durch. Das Buch »Mac OS Hacking« ist ein Ergebnis seiner praktischen Erfahrungen und seiner lehrenden Tätigkeit.

9 783645 605519

40,– EUR [D]
ISBN 978-3-645-60551-9

Besuchen Sie unsere Website
www.franzis.de

FRANZIS