- **Mobilgeräte-Forensik:** Auswertung der Informationen, die durch Smartphones oder Tablets teilweise automatisch generiert werden.
  - Kommunikationsdaten (Mail, Messenger …)
  - Standortdaten (Funkzellen, Positionssysteme …)
- **Multimedia-Forensik:** Untersuchung von Fotos und Videos daraufhin, ob Manipulationen vorliegen oder ob versteckte Informationen enthalten sind.
  - Analyse von Fotos und Videoaufnahmen (Manipulationen, Zuordnung …)
  - Möglichkeit von verschleierter Kommunikation — *Steganografie*
- **Cloud-Forensik:** Für forensische Untersuchungen in Cloud-Umgebungen müssen spezielle Methoden angewandt werden.
  - eingeschränkte Möglichkeiten der Untersuchung
  - Zugriffe über Schnittstellen und Analyse der Daten
- **IoT-Forensik:** Geräte mit integrierten Rechnersystemen sammeln kontinuierlich Daten, die für eine Analyse von Interesse sind.
  - Drucker, Network-Attached Storage und Automobile erzeugen relevante Daten.
  - IoT und Smart Home werden immer interessanter.
- **Webforensik:** Immer mehr Dienste werden ausschließlich als Webanwendung bereitgestellt, gleichzeitig werden immer neue Funktionen in Webbrowser integriert.
  - entstandene Spuren im Webbrowser (teilweise für den Anwender unsichtbar)
  - Daten in Datenbanken oder Logfiles auf Webservern

Analyse von Sicherheitsvorfällen

Die Analysemethoden der IT-Forensik lassen sich in zwei große Bereiche unterteilen:

- **Post-Mortem-Analyse:** Beschreibt die Untersuchung eines Rechnersystems im ausgeschalteten Zustand.
- **Live-Analyse:** Hierbei wird ein System im eingeschalteten Zustand untersucht, um auch flüchtige Spuren zu sichern.

7.2 Post-Mortem-Untersuchung

Bei der Post-Mortem-Analyse findet die Untersuchung statt, nachdem ein Vorfall erkannt wurde. Diese Analyse wird durchgeführt, wenn der flüchtige Speicher für den zu klärenden Vorfall nicht relevant ist oder dieser Vorfall schon längere Zeit zurückliegt. Der Vorteil dabei ist, dass Daten nicht aus Versehen zerstört werden können

und der gesamte Analyseprozess bzw. der Einsatz von Tools planbar ist, da keine Informationen verlorengehen können. Dies geschieht im Wesentlichen durch die Untersuchung von Datenträgern der betroffenen Rechnersysteme. Zugleich kann die Untersuchung auf mehrere Personen aufgeteilt werden.

Forensische Sicherung eines Speichers

Der erste Schritt einer Post-Mortem-Untersuchung ist die forensische Sicherung eines Datenspeichers. Dabei spielt es im ersten Moment keine Rolle, ob es sich um eine Speicherkarte, einen USB-Stick oder eine klassische Festplatte handelt. Wichtig ist, dass dabei keine Veränderungen auf dem Datenspeicher vorgenommen werden dürfen, da dies das Untersuchungsergebnis verfälschen würde.

Um eine Modifikation der Daten oder gar das versehentliche Überschreiben zu verhindern, müssen Sie daher einen Write-Blocker verwenden. Diese Methode verhindert, dass auf den zu sichernden Datenspeicher schreibend zugegriffen wird. Per Software kann in verschiedenen Betriebssystemen ein Schreibschutz realisiert werden. Dies hat den Vorteil, dass ohne weitere Hardware und zusätzliche Kosten ein Schutz vor Veränderung schnell realisiert werden kann. Die sicherere Alternative ist ein Hardware-Schutz in Form eines Hardware-Write-Blockers, den Sie zwischen dem zu untersuchenden Datenspeicher und Ihrem Rechner anschließen.

Am Beispiel des Betriebssystems Kali Linux zeigen wir die Einrichtung eines Schreibschutzes auf Software-Ebene. Dazu müssen Sie als Erstes das automatische Einbinden von neu erkannten Laufwerken deaktivieren, bevor Sie den Datenspeicher anschließen. Zu diesem Zweck stoppen Sie den Service `udisks2`, der für das Einbinden zuständig ist:

```
sudo systemctl stop udisks2.service
```

Beachten Sie allerdings, dass damit nur der Automatismus deaktiviert wurde. Sie können weiterhin den Datenspeicher als Laufwerk manuell einbinden oder sogar die Daten überschreiben oder löschen. Schließen Sie nun den Datenspeicher an, und suchen Sie mittels `dmesg` (Kernelmeldungen) oder `fdisk -l` (Übersicht der Laufwerke) die genaue Bezeichnung des Datenträgers. Im Beispiel wird hierfür `/dev/sdb` verwendet.

Für die forensische Sicherung werden nicht wie üblich Dateien einfach kopiert, sondern es wird eine 1:1-Kopie auf niedrigster Ebene erstellt -– also bitweise ein Abbild erstellt. Mit diesem Vorgang kann garantiert werden, dass wirklich alle Informationen gesichert wurden. Für solch einen Kopiervorgang bietet sich das Tool *dd* an. Von diesem Tool existieren mehrere Derivate mit erweiterten Funktionen. Speziell für forensische Sicherungen wurde vom US-amerikanischen *Department of Defense Cyber Crime Center* (DC3) das Tool `dc3dd` entwickelt.

Die wichtigsten Optionen zur Steuerung von `dc3dd` sind:

- `if=DEVICE`: Laufwerk, das gesichert werden soll
- `hof=FILE`: Ausgabe der Sicherung in einer Datei inklusive Hash
- `hash=ALGORITHM`: verwendet einen Hash-Algorithmus (MD5, SHA1, SHA256 oder SHA512)
- `hlog`: Logdatei mit den generierten Hashes

Wie Sie an der kleinen Liste oben schon sehen, geht es bei der forensischen Sicherung um Hashes. Die Integrität ist ein zentrales Element der IT-Forensik, und mittels Generierung von Hashes vom ursprünglichen Datenspeicher und von dem erstellten Abbild wird die Integrität nachgewiesen. Mit dem folgenden Befehl starten Sie die Sicherung:

```
sudo dc3dd if=/dev/sdb hof=image.dd hash=sha512 hlog=image.hash
```

Die Validierung wird von `dc3dd` automatisch vorgenommen, und der Vergleich der Hashes wird in der Datei `image.hash` gespeichert. Nachdem Sie eine forensische 1:1-Sicherung erstellt haben, erstellen Sie eine Kopie der Sicherung. Dadurch kann sichergestellt werden, dass bei einer Veränderung noch eine korrekte Kopie vorhanden ist.

```
cp image.dd image2.dd
```

Anschließend binden Sie das Image mit dem Befehl `losetup` als Loop-Device ein:

```
sudo losetup -f -P image2.dd
```

Bevor Sie den eigentlichen Mount-Befehl einsetzen können, müssen Sie einen Ordner anlegen. Dazu verwenden wir hier den Unterordner `image` im Verzeichnis `mnt`. Abschließend kann das Image eingebunden werden. Die Option `ro` (*read only*) sorgt dafür, dass Sie keine Schreibrechte bekommen und die Daten nicht verändern können.

```
sudo mkdir /mnt/image/
sudo mount -o ro /dev/loop0p1 /mnt/imag
```

Jetzt können Sie die Dateien im File Manager oder im Terminal betrachten und mit der Analyse beginnen.

Gelöschte Dateien wiederherstellen per File-Carving

Angreifer versuchen zum Teil gezielt, ihre Spuren zu verwischen, indem etwa Logdateien gelöscht werden. Aber auch Anwendungen erstellen temporäre Dateien, die nach dem Beenden automatisch gelöscht werden. Daher lohnt es sich häufig, nach gelöschten Dateien zu suchen, um weitere Spuren zu sichern.

Der Hintergrund dieser Analyse ist, dass Dateisysteme Verzeichnisse nutzen, um den Speicherort von Dateien zu organisieren. Darin sind unter anderem der Dateiname und der genaue Speicherort auf dem Datenspeicher festgehalten. Damit muss das Betriebssystem, um eine Datei zu finden, nur auf dieses Verzeichnis zugreifen und nicht den gesamten Datenspeicher durchsuchen.

Wenn über das Betriebssystem eine Datei gelöscht wird, wird typischerweise nur der Eintrag im Verzeichnis entfernt, um den Speicherplatz für neue Dateiinhalte freizugeben. Diese Strategie verfolgen viele Betriebssysteme, da diese Methode viel schneller ist als das komplette Überschreiben der Daten mit Nullen. Dadurch sind die eigentlichen Daten jedoch noch immer auf der Festplatte vorhanden und können wiederhergestellt werden.

Daher können Sie mit entsprechenden Tools gezielt nach gelöschten Dateien suchen, was als *File-Carving* bezeichnet wird. Beim File-Carving – oder einfach nur Carving – handelt es sich um die Suche nach Dateien auf Datenspeichern auf Basis einer inhaltlichen Analyse der Datenblöcke anhand von Mustern (Header, Signaturen u. a.). Diese Muster leiten sich aus den Vorgaben ab, mit denen die unterschiedlichen Dateiformate strukturiert, genauer gesagt im Standard definiert sind. Diese Muster enthalten neben der eigentlichen Bytefolge auch sogenannte *Marker*, die etwa den Beginn und das Ende einer Datei beschreiben. Da diese Marker fest definiert und bekannt sind, können sie auch als Signaturen verwendet werden.

Für die Wiederherstellung gelöschter Dateien steht eine Reihe von verschiedenen Hilfswerkzeugen zur Verfügung. Unter Linux können mit Hilfe des Befehls `file` unbekannte Dateien oder Datentypen anhand ihrer Signatur erkannt werden. Deutlich mehr Möglichkeiten stehen Ihnen mit der Anwendung *Foremost* (verfügbar für Windows, macOS und Linux) zur Verfügung, das häufig in der IT-Forensik eingesetzt wird. Das folgende Beispiel zeigt, wie Sie die Wiederherstellung von gelöschten Dateien mit Foremost unter Kali Linux durchführen.

Als Erstes installieren Sie das Tool Foremost und legen ein Verzeichnis an, in dem die wiederhergestellten Dateien gespeichert werden:

```
sudo apt install foremost
mkdir file-carve
```

Die wichtigsten Optionen zur Steuerung von Foremost sind:

- `-i`: Image, das untersucht werden soll
- `-o`: Ordner, in dem die gefundenen Dateien gespeichert werden
- `-t`: Dateityp, der gesucht werden soll; mit `-t all` wird nach allen Dateien gesucht.
- `-v`: aktiviert den Verbose-Modus

Mit dem folgenden Aufruf starten Sie die Suche nach gelöschten Dateien. Grundsätzlich ist der Vorgang aufwendig, da alle Daten eines Speichers analysiert werden.

Zusätzlich ist Dauer des Vorgangs abhängig von der Größe und der Geschwindigkeit des Datenspeichers.

```
foremost -i image2.dd -o file-carve -t all -v
```

Als Ergebnis wird für jeden gefundenen Dateityp ein Unterordner angelegt, in dem alle gefundenen Dateien abgelegt werden. Die Datei `audit.txt` wird auf der Hauptebene angelegt und enthält das Protokoll und eine Liste der Funde.

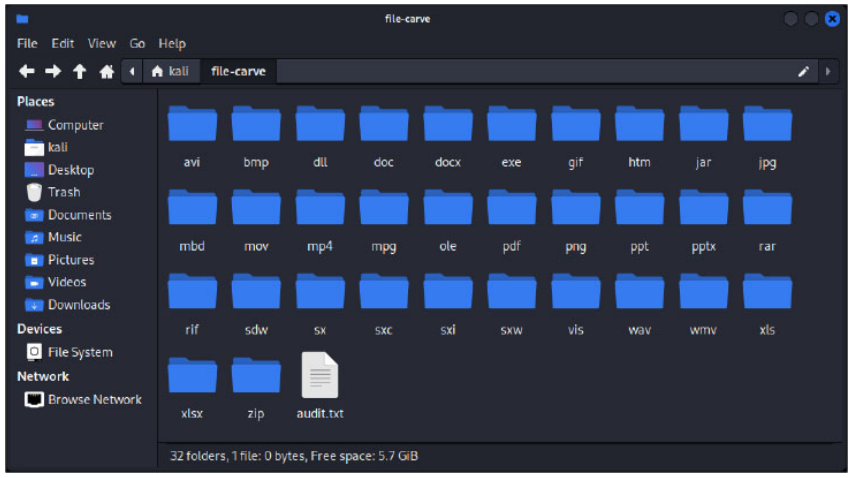

Abbildung 7.1 Durch Foremost erstellte Ordner

Sehr nützlich bei Foremost ist, dass weitere Dateitypen einfach hinzugefügt werden können. Bei einem Standardaufruf wird die Konfigurationsdatei `/etc/foremost.conf` verwendet. Kopieren Sie sie in Ihr Arbeitsverzeichnis, und übergeben Sie bei einem neuen Aufruf den Pfad zu Ihrer Konfigurationsdatei mit dem Parameter `-c`.

```
foremost -i image2.dd -o file-carve -t all -c foremost.conf -v
```

Um zum Beispiel `.webp`-Dateitypen hinzuzufügen, öffnen Sie eine solche Datei mit einem Hex-Editor und betrachten den ersten Block:

```
52 49 46 46 78 56 01 00 57 45 42 50 56 50 38 20  RIFFxV..WEBPVP8
```

Hier ist der Start des ersten Blocks von Interesse. In diesem Fall wird für das WebP-Bildformat das Containerformat *RIFF* verwendet (wie auch bei AVI- und WAV-Dateien). Dazu müssen für die Signatur in der Konfigurationsdatei die entsprechenden Hex- oder ASCII-Werte für die Bezeichner `RIFF` und `WEBPVP8` übernommen werden. Die vier Zeichen dazwischen können variieren und werden daher ausgelassen, also durch Fragezeichen ersetzt. Fügen Sie nun den folgenden Eintrag in die Konfigurationsdatei ein. Der Wert `30000000` legt eine maximale Dateigröße von 30 MB fest:

```
webp    y    30000000 RIFF????WEBPVP8
```

Damit haben Sie Foremost so angepasst, dass jetzt auch `webp`-Dateien wiederhergestellt werden können. Wie bei anderen Dateiformaten wird ein Extraordner angelegt, in dem die gefundenen Dateien abgelegt werden.

Analyse von Metadaten und Dateien

Bei der Post-Mortem-Analyse spielt die Untersuchung der Zeitstempel und der Metadaten eine wesentliche Rolle, um eine Nutzung zu rekonstruieren. Hier wird auch von der Erstellung einer *Timeline* gesprochen. Als Erstes überprüfen Sie die Zeitstempel (*Timestamps*) von relevanten Dateien. Anhand des Zeitstempels einer Datei oder eines Ordners kann festgestellt werden, wann die letzten Änderungen vorgenommen wurden. Dazu können Sie unter Kali Linux das Tool `stat` verwenden.

```
stat logfile.log
```

```
  File: logfile.log
  Size: 5           Blocks: 8      IO Block: 4096, reg. file
Device: 801h/2049d  Inode: 912561  Links: 1
Access: 0644 / -rw-r--r--  Uid: 1000 / kali  Gid: 1000 / kali
Access: 2022-08-05 01:18:04.384491301 -0400
Modify: 2022-08-05 01:17:51.201903247 -0400
Change: 2022-08-05 01:17:51.213909244 -0400
 Birth: 2022-08-05 01:17:51.201903247 -0400
```

Es werden daraufhin die vier Zeitstempel *Access Time*, *Modify Time*, *Change Time* und *Birth Time* angezeigt. Beachten Sie die angezeigte Zeitzone für eine korrekte Zuordnung.

- **Access Time:** Die *Access Time* wird jedes Mal aktualisiert, wenn auf den Inhalt der Datei zugegriffen wurde. Mit ihr wird also der letzte Zugriff auf den Inhalt festgehalten. Dabei wird jede Art von Zugriff protokolliert, auch ein Kopiervorgang, weil dabei auf die Inhalte der Datei zugegriffen werden muss. Ausgenommen sind nur reine Schreibvorgänge, die am Ende der Datei weitere Informationen anhängen. Das Verschieben der Datei oder Änderungen der Dateiattribute, wie der Zugriffsrechte, haben ebenfalls keine Auswirkungen, da sie nicht den Inhalt betreffen.
- **Modify Time:** Der Zeitstempel *Modify Time* wird immer aktualisiert, wenn der Inhalt der Datei verändert wurde. Damit können Sie also die letzte Veränderung der Datei feststellen. Dies ist z. B. sehr interessant, wenn es eine Konfigurationsdatei gibt, da Sie so einen Hinweis bekommen, wann der Angreifer seine Konfiguration abgeschlossen hat.
- **Change Time:** Der Zeitstempel *Change Time* wird aktualisiert, sobald sich ein Dateiattribut verändert hat, d. h., wenn die Datei umbenannt wird oder sich die Berechtigungen ändern. Allerdings wirkt sich auch eine Änderung des Inhalts auf

den *Change Time*-Zeitstempel aus, da die Dateigröße aktualisiert werden muss. Der Zeitstempel wird nur dann nicht aktualisiert, wenn ein reiner Lesezugriff erfolgt.

- **Birth Time:** Der Zeitstempel *Birth Time* wird gesetzt, wenn die Datei erstellt wird. Eine alternative Bezeichnung ist *Creation Time*. Da dieser Zeitstempel nicht von allen Dateisystemen unterstützt wird, zeigen ihn nicht alle Tools an.

Neben den Zeitstempeln können Metadaten, die direkt in die Dateien integriert sind, relevante Informationen enthalten. Sie sind strukturierte Zusatzinformationen zu den eigentlichen Dateiinhalten, die zum Teil automatisiert integriert werden. Je nach Dateityp unterscheiden sich die Anzahl und der Umfang der Informationen. Dort verstecken sich Hinweise darauf, wer der Urheber einer Datei ist oder mit welchem Programm sie erstellt wurde. Es können sich jedoch noch andere Informationen dort verbergen. Die meisten Dateiformate speichern Metadaten; es gibt nur wenige Dateitypen, wie einfache `.txt`-Dateien, die keine zusätzlichen Informationen enthalten.

So speichern etwa die Office-Programme von Microsoft Metadaten in ihren Dokumenten ab. Den größten Teil dieser Informationen können Sie selbst direkt in Word, Excel und Co. unter DATEI • INFORMATIONEN einsehen. Unter Kali Linux nutzen Sie das Tool `mat2`, um diese Informationen auszulesen oder zu löschen. Es ist eigentlich dazu gedacht, um Metadaten zu entfernen. Sie können es aber auch zur reinen Ausgabe der Metadaten nutzen. Dazu müssen Sie als Erstes das Software-Paket `mat` installieren, das die ausführbare Datei `mat2` enthält:

```
sudo apt install mat
```

Anschließend rufen Sie `mat2` mit dem Parameter `-s` (`--show`) und dem Dateinamen auf, um eine Liste der Metadaten auszugeben:

```
mat2 -s text.docx

  ...
  Metadata for docProps/core.xml:
  cp:lastModifiedBy: mustermann
  ..
  dc:creator: Max Mustermann
```

Bei einer Word-Datei ist zum Beispiel zu erkennen, welche Word-Version und welche Vorlage verwendet wurde. Zusätzlich werden der Ersteller, der Benutzer der Datei mit den letzten Änderungen sowie der Zeitpunkt der letzten Änderungen angezeigt.

Das Tool `mat2` kann grundsätzlich auch die Metainformationen von Bilddateien auslesen. Mehr Informationen erhalten Sie jedoch, wenn Sie die Anwendung `exiftool` verwenden. Um sich die Metainformationen anzeigen zu lassen, müssen Sie nur den Dateinamen ohne weitere Parameter übergeben:

```
exiftool test.jpg
```