Grundlagenpraktikum Netz- und Datensicherheit

Thema:

Forensische Untersuchung eines Ransomware-Angriffs

Lehrstuhl für Netz- und Datensicherheit
Ruhr-Universität Bochum

Versuchsdurchführung: Raum ID 2/168

Betreuung: Dominik Noß
Zusammengestellt von: Dominik Noß
Stand: 13. Dezember 2021
Version: 2.1

Inhaltsverzeichnis

1 Versuchsbericht **2**

2 Szenario **2**

3 Plausibilität **3**
3.1 Hackazon . . . 3
3.2 Ransomware . . . 3

4 Abbild-Dateien von Datenträger **4**

5 Löschen von Dateien **5**

6 Durchführung **6**
6.1 Erstellung des Hashes . . . 6
6.2 Lesen der Partitionierung . . . 6
6.3 Mounten des Images . . . 6
6.4 Auslesen des Home-Verzeichnis . . . 7
6.5 Versuchte Wiederherstellung der Benutzer-Dateien . . . 7
6.6 Herleitung des Angriffverlaufs . . . 7
6.7 Wiederherstellung der Log-Dateien . . . 8
6.8 Analyse der Ransomware . . . 8
6.9 Verständnis der Verschlüsselung . . . 8
6.10 Umkehrung der Verschlüsselung . . . 8
6.11 Rückblick . . . 9

7 Kontrollfragen **9**

1 Versuchsbericht

Beantworten Sie in Ihrem Bericht alle Fragen, die im Abschnitt 6 gestellt werden. Dokumentieren Sie Ihr Vorgehen mit Screenshots und Listings.

2 Szenario

Mr. Smith führt einen Online-Versandhandel namens „Hackazon“. Sie wurden Mr. Smith mit folgender Nachricht kontaktiert.

```
1  Hallo!
2
3  Unser Server wurde gehackt und alle meine Daten wurden verschlüsselt!
4  Bitte helfen Sie mir, die Dateien wieder zu bekommen!
5  Ich habe einen Erpresserbrief erhalten. Wenn ich nicht innerhalb einer
6  Woche 10000 euro in Bitcoin zahle, wird der Entschlüsselungs-Schlüssel
7  gelöscht!
8  Auf dem Server ist mein Web Shop installiert. Außerdem lagere ich dort
9  wichtige Finanzdaten und zwei wichtige Bilder.
10
11 --- Weitergeleitete Nachricht ---
```

```
12 Gestern um 12:59 schrieb Oskar Evil:
13
14 Sie wurden gehackt!
15
16 Übertragen Sie 10000 Euro in Bitcoins mit dem Formular auf
17 http://et5oskarevilinx.onion!
18 Geben sie die Kundenreferenznummer 74630 an, um Ihren Schlüssel zur
19 Entschlüsselung zu erhalten. Er wird ihnen dann 100%ig per Email
20 zugesandt.
21
22 Versuchen Sie nicht, die Dateien selber zu recovern! Das ist unmöglich,
23 weil sie mit military grade enryption verschlüsselt sind!
24 Außerdem sind ganz viele Viren auf dem Server, also besser nicht
25 anfassen!
26
27 Toodaloo,
28 Oskar
```

3 Plausibilität

3.1 Hackazon

Hackazon ist ein bewusst verwundbar gestalterer Web Shop. Hergestellt von Rapid7[1] dient es als Lehrmaterial im Bereich Web-Sicherheit. Es können Cross Site Scripting, SQL Injections, Command Execution und viele weitere Varianten von Schwachstellen getestet werden. Diese Software in einer echten Produktionsumgebung einzusetzen, wäre fahrlässig und unsicher. Der Einsatz hier steht stellvertretend für einen regulären Web Shop, in welchem die oder der Angreifende eine Schwachstelle identifiziert hat.

3.2 Ransomware

Ransomware wird eine Gruppe von Malware genannt, welche bei Ausführung die persönlichen Dateien auf dem infizierten Rechner verschlüsselt. In einer Nachricht wird der Nutzer aufgefordert, ein Lösegeld zu zahlen, um den zur Entschlüsselung benötigten Schlüssel zu erhalten. Auf dem Zahlungsweg werden oft Maßnahmen zur Anonymisierung des Empfängers eingesetzt, wie etwa Bitcoin und das TOR-Netzwerk.

Tatsächliche Ransomware ist in der Realität leider wesentlich besser geworden als die hier verwendete. Oft ist es nicht möglich, die genutzte Kryptographie zu brechen. Die Dateien können dann nicht ohne den Schlüssel wiederhergestellt werden.
Die einzige Möglichkeit bleibt dann, durch Zahlung des Lösegeldes den kryptographischen Schlüssel zu erhalten. Doch selbst bei Zahlung ist nicht sichergestellt, dass der Schlüssel übermittelt wird. In diesem Fall sind sowohl die Daten als auch das Lösegeld verloren.

Eine effektive Methode um Verluste zu vermeiden, ist Vorbeugung. Mit einer durchdachten Backup-Strategie kann ein von Ransomware befallener Host ohne großen Datenverlust einfach gelöscht und ausgetauscht werden. Sichere Backups finden regelmäßg und häufig statt, sind redundant ausgelegt (z.B. RAIDs, Cloud-Storage), befinden sich dezentral an mehreren, geographisch getrennten Orten (verschiedene Gebäude oder Länder) und setzen mehr als eine Technologie ein (z.B. Festplatten, Bandlaufwerke, DVDs, Cloud).
Weiterhin ist es wichtig, eine Recover-Prozedur zu haben. Ein Backup hat wenig Nutzen, wenn das Wiederherstellen zeitaufwendig oder gar nicht implementiert ist.

[1] https://github.com/rapid7/hackazon

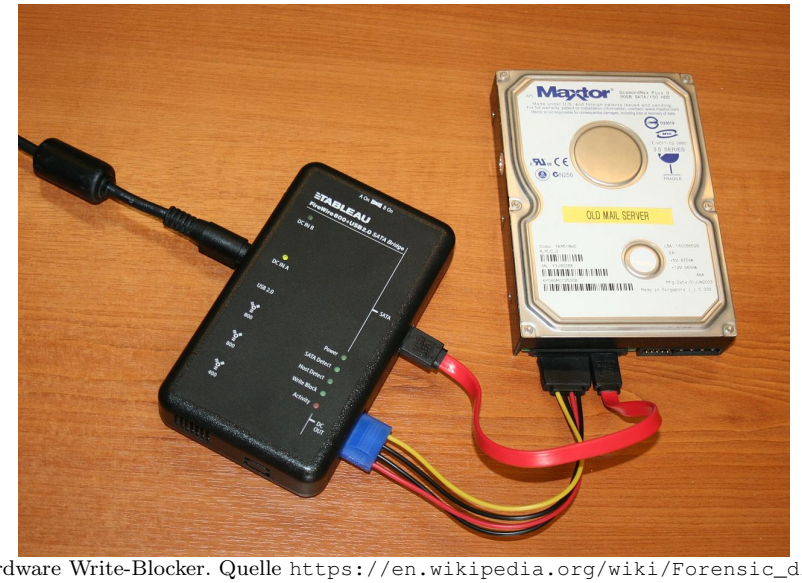

Abbildung 1: Hardware Write-Blocker. Quelle `https://en.wikipedia.org/wiki/Forensic_disk_controller`

Außerdem muss durch eine sichere IT-Strategie einem Befall von Ransomware vorgebeugt werden. Updates einspielen, Systeme sicher konfigurieren und ordentliche Schulung der Nutzer ist extrem wichtig.

4 Abbild-Dateien von Datenträger

Wichtig bei forensischen Untersuchungen ist es, den Datenträger nicht zu verändern. Die Spurensicherung im Rahmen polizeilicher Ermittlungen resultiert nur dann in rechtlich nutzbare Beweise, wenn eine Veränderung der Datenträger ausgeschlossen werden kann. Dafür werden Write-Blocker[2] eingesetzt. Das sind Hardware-Adapter oder Software für den Anschluss von Festplatten via IDE/-SATA/SAS. Dabei wird ausschließlich Lese-Zugriff erlaubt, Schreibzugriff ist deaktiviert. Damit wird sichergestellt, dass beim Auslesen des Datenträgers kein einziges Bit verändert wird. Die über den Write-Blocker angeschlossene Festplatte wird nun Bit-weise ausgelesen und in eine Datei geschrieben. Dies ist nicht zu verwechseln dem simplen Kopieren aller Dateien! Eine Eins-zu-Eins-Kopie ist wesentlich genauer. Beispielsweise die Sicherung einer 1TB (1000.000.000 Byte) resultiert in eine 1TB (1000.000.000 Byte) große Datei. Selbst dann, wenn der Datenträger nur eine 500GB große Partition enthält, auf welcher sich nur 100GB Dateien befinden[3].

Eine Grund dafür, eine Bit-genaue Kopie dem Kopieren aller Dateien vorzuziehen, ist es, dass Dateien und auch Partitionen versteckt sein können. Zum Beispiel erscheinen mit Veracrypt[4] angelegte Partitionen als zufällige Daten, bis man sie entschlüsselt.
Eine Bit-genaue Kopie ermöglicht auch das Wiederherstellen von „gelöschten“ Dateien. Das werden Sie im Rahmen des Versuches durchführen.
Weiterhin gibt es Dateisysteme, die nicht Datei-basiert arbeiten. Beispiel ist die Swap Partition auf Linux-Systemen. Diese Partitionen können aber wertvolle Informationen enthalten, wie etwa kryptographische Schlüssel, die bei der Auslagerung von Prozessen aus dem Arbeitsspeicher auf die Festplatte geschrieben wurden.

[2] `http://www.forensicswiki.org/wiki/Write_Blockers`
[3] Anschließend können die Image-Dateien komprimiert werden, um Platz zu sparen.
[4] `https://www.veracrypt.fr`

Die Inspektion der Daten findet nicht direkt auf dem Datenträger statt, sondern auf dem Abbild.

5 Löschen von Dateien

Das Löschen von Dateien bewirkt, dass der von ihnen belegte Speicherplatz freigegeben und für andere Zwecke verwendet werden kann. Dabei gibt es einiges zu beachten.

Abhängig von der Art der Löschung sind Dateien nicht komplett verloren. Das „Verschieben in den Papierkorb“, unabhängig vom verwendeten Betriebssystem, etwa ist auf Ebene des Dateisystems nicht mehr als das Verschieben in einen Ordner und somit keine tatsächliche Löschung. Die Dateien bleiben dabei erhalten und belegen weiterhin Speicherplatz. Erst durch die Leerung des Papierkorbes wird dieser freigegeben.

Das Löschen einer Datei unter Linux Systemen kann durch die Benutzung des **rm** Befehls umgesetzt werden. Der belegte Speicherplatz ist damit sofort freigeben.

Sobald die Datei gelöscht ist, gibt es keine einfache und universelle Methode, die Löschung Umzukehren. Soll die Datei wiederhergestellt werden, ist abhängig vom verwendeten Dateisystem zusätzliche Software notwendig.
Auf EXT4 Dateisystemen etwa ist dies mit **extundelete** möglich. Das Programm liest das **Journal** aus, welches als Teil von **EXT4** einen Teil der vergangenen Aktionen loggt. Findet es darin die Aktion, mit der eine Datei gelöscht wurde, kann es den damals damit verbundenen Speicherplatz auslesen und den aktuellen Wert ausgeben. Damit kann es gelingen, die Datei wiederherzustellen.
Das Wiederherstellen einer Datei, deren Speicher unangetastet ist und deren EXT4 Journal-Eintrag verfügbar ist, dauert meist nur wenige Millisekunden (oder mehr bei großen Dateien).

Je mehr das Dateisystem benutzt wurde, desto unwahrscheinlicher ist es, dass eine gelöschte Datei rekonstruierbar ist. Das kann dadurch geschehen, dass der freigegebene Speicherplatz, an welchem sich der Inhalt der ehemaligen Datei befindet, mit neuen Daten überschrieben wird.
Auch loggt das EXT4 Journal nur eine begrenzte Anzahl der zeitlich letzten Aktionen. So kann der Eintrag, welcher die Löschung der Datei beschreibt, nicht mehr verfügbar sein.

In der Situation eines fehlenden Journal-Eintrages ist es immer noch möglich, dass der Dateiinhalt nicht überschrieben wurde. Dann kann mit Programmen wie **photorec** eine heuristische Methode versucht werden. Dabei wird jede Einheit des Dateisystem ausgelesen und geprüft, ob damit ein bekannter Dateityp, etwa JPEG oder MP4, beginnen könnte. Es wird dann geprüft, ob die darauffolgenden Daten den angenommenen Dateityp vervollständigen würden. Bei Erfolg schreibt es die Datei in ein Verzeichnis.
Erheblicher Nachteil der Methode ist die Unzuverlässigkeit und Zeitaufwendigkeit. Sie erfordert Expertenwissen über jeden gesuchten Dateityp und unterstützt nur eine begrenzte Menge. Exotischere Dateitypen sind nicht findbar, ohne selbst einen Suchalgorithmus zu programmieren. Auch dauert es bei großen Datenträgern extrem Lange, alle Daten auszulesen und zu verarbeiten. Bei Hard Disks, also Festplatten mit rotierender Magnetscheibe, kann dies viele Stunden dauern.

Interessanterweise kann die Wahrscheinlichkeit, eine spezielle Datei mit **photorec** zu rekonstruieren recht gering sein, doch die Wahrscheinlichkeit, mindestend eine beliebige Datei zu finden sehr hoch.
Versuchen Sie doch einmal außerhalb des Praktikums, Photorec auf eines Ihrer privaten Speichermedien (USB-Stick, Festplatte, SD-Karte) anzuwenden und lassen Sie sich vom Ergebnis überraschen. Vielleicht finden Sie darauf jahrelang verschollen geglaubte Dateien.

Das Ergebnis sollte Ihnen auch verdeutlichen, wie kritisch der Weiterverkauf oder Verlust eines Datenträgers ist. Selbst, wenn alle Dateien gelöscht wurden, oder der Datenträger formatiert wurde, kann eine Vielzahl von Dateien wiederherstellbar sein.

Das **sichere Löschen** von Dateien ist nur möglich, wenn der von ihnen belegte Speicherplatz sicher überschrieben wurde. Auch das ist kompliziert, da etwa die darunterliegende Technologie dies verhindern könnte. SSD Festplatten können dies beispielsweise ungewollt durch **Wear Leveling** verhindern. Um die Daten dann aber noch wiederherstellen zu können, ist es notwendig, die Festplatten-Hardware zu untersuchen, wie etwa die auf der Platine verbauten Speicherzellen auszulesen.

6 Durchführung

6.1 Erstellung des Hashes

Entpacken Sie die komprimierte Abbildatei. Warten Sie, bis der Befehl von sich aus beendet und unterbrechen Sie das Programm nicht. Das entpackte Image ist etwa 4GB groß. Deshalb nimmt das Entpacken etwas Zeit in Anspruch.

```
1 tar xvf image.tar.gz
```

Es befindet sich nun eine neue Datei `image.img` im Arbeitsberzeichnis. Berechnen Sie von dieser den SHA256 Hash mit dem Programm **sha256sum**. Das Ergebnis muss wie folgt aussehen.

```
1 f17b5ffe45356427183c0bd003946f41e8c108e5c2d74df1b079d1660d76b68e  image.img
```

Interpretieren Sie das Ergebnis. Was bedeutet Gleichheit, bzw. Ungleichheit des gegebenen und des gerade errechneten Hashs?

6.2 Lesen der Partitionierung

Lesen Sie die Partitionstabelle des Festplattenabbildes aus.

```
1 parted image.img print
```

Welche Partitionen gibt es? Welche Dateisysteme befinden sich darauf? Welche Partitionen sind für Sie von Interesse?

6.3 Mounten des Images

Legen Sie Loop Devices für die einzelnen Partitionen an. Vergessen Sie nicht den read-only Parameter `-r`. Der Parameter bewirkt, dass über die Devices nicht in die Datei geschrieben werden darf. Andernfalls würden Sie eine Manipulation der Abbilddatei riskieren.

```
1 sudo kpartx -v -a -r image.img
```

Mit **lsblk** können Sie sich Block Devices anzeigen lassen. Die Ausgabe enthält sowohl als Hardware existierende als auch virtuelle, also gemappte, Geräte.

Mounten Sie das Loopback-Device der Hauptpartition des Images aus dem Ordner '/dev/mapper/' als das Verzeichnis **/mnt/**. Vervollständigen Sie den folgenden Befehl:

```
1 sudo mount /dev/mapper/..... /mnt/
```

Warum ist es wichtig, das Image als Read-Only zu mounten? Was verhindern Sie dadurch?

6.4 Auslesen des Home-Verzeichnis

Von Mr. Smith erfahren Sie, dass die vermissten Dateien im Benutzerverzeichnis des Benutzers **webstoreadmin** liegen. Geben Sie den Inhalt des `/home/webstoreadmin` Verzeichnis aus. Denken Sie daran, dass alle absoluten Pfade des Servers nun relativ zu `/mnt` befinden.

Welche Dateien und Ordner befinden sich dort? Analysieren Sie die Dateien mit dem **file** Befehl. Welche Informationen erhalten Sie über den Datei-Inhalt? Bringen Sie in Erfahrung, was die Ausgabe „**data**“ bedeutet. Welchen Zweck erfüllt der **file** Befehl? Wie lauten die Dateiendungen?
Welchen Schluss ziehen Sie daraus?

6.5 Versuchte Wiederherstellung der Benutzer-Dateien

Sie vermuten, dass die Malware die Dateien in verschlüsselter Variante auf die Festplatte geschrieben hat und die Originale gelöscht hat. Benutzen Sie **extundelete**, um alle Dateien im Ordner **/home/webstoreadmin** zu rekonstruieren.

```
1 sudo extundelete --restore-directory /home/webstoreadmin /dev/mapper/loop0p1
```

extundelete erstellt einen Ordner **RECOVERED_FILES** im aktuellen Arbeitsverzeichnis.

Was schließen Sie aus der Programmausgabe? Welche gelöschten inodes („Dateien“) wurden von extundelete identifiziert? Welche konnten rekonstruiert werden? Aus welchem Grund konnten sie nicht wiederhergestellt werden?

6.6 Herleitung des Angriffverlaufs

Ihr Ziel ist es nun, den Angriff nachzuvollziehen.

Sie wissen von Mr. Smith, dass auf dem Server neben dem Web Server keine weiteren Dienste aktiviert sind. Daraus schließen Sie, dass der Angriff über diesen Service stattgefunden hat.

Es ist nicht immer so eindeutig, welcher Angriffsvektor benutzt wurde. Häufig müssen Sie über weitere Wege Informationen sammeln, wie der Angriff stattgefunden hat. Beispielsweise durch Befragung der Nutzer, Auswertung von Firewall-Logs oder anderweitiger Überwachungsmaßnamen.

Beschreiben Sie in wenigen (3-6) Sätzen drei der folgenden Angriffspunkte. Wie könnte ein Angriff darüber vonstatten gehen? Hier ist ihre Kreativität gefragt. Angriffs-Ziel muss nicht zwingend der hier gegebene Server sein.

1. Email-Anhänge
2. Vom Internet aus zugänglicher Telnet-Dienst
3. Vom Internet aus zugänglicher SSH-Dienst
4. Windows Remote Desktop Verbindungen
5. Download einer Installations-Datei für Windows von einer Website
6. Ein auf dem Parkplatz gefundener USB-Stick
7. Ein über USB angeschlossenes Smartphone
8. Eine Postscript-Datei
9. Ein bösartiger, transparenter HTTP MITM proxy
10. Ein über mehrere Websites identisches Passwort

6.7 Wiederherstellung der Log-Dateien

Da Sie abgesehen von Mr. Smiths Aussage und der Abbild-Datei keine weiteren Informationen haben, müssen Sie mit den Logs des Servers arbeiten. In welchem Ordner befinden sich bei Ubuntu standardmäßig die Log-Dateien der System-Dienste? Welche Logs befinden sich im dortigen **apache2** Ordner? Welche Gründe könnte es für Ihre Beobachtung geben?

Sie vermuten, dass Oskar die Log-Daten entfernt hat. Da der Server nach Entdeckung des Angriffs sofort heruntergefahren wurde und es sich um ein EXT4-Dateisystem handelt, besteht die Möglichkeit, die Logs wiederherstellen zu können.

Benutzen Sie das Programm **extundelete**, um alle Dateien im Log-Verzeichnis von Apache2 zu rekonstruieren. Was darf nicht passieren, damit gelöschte Dateien auf einem EXT4 Dateisystem wiederherstellbar bleiben?

Glücklicherweise finden Sie die Log-Daten. Durchsuchen Sie sie nach verdächtigen Informationen. In welchem Ordner liegt die Ransomware?

6.8 Analyse der Ransomware

Sie möchten die Ransomware analysieren, um eine Entschlüsselung der Dateien vornehmen zu können. Glücklicherweise ist die Malware keine ausführbare Binärdatei, sondern ein Bash-Skript. Das erleichtert das Verständnis. Ausführbare Binärdateien, wie ELF oder PE Dateien, müssten erst mittels Reverse Engineering untersucht werden. Öffnen Sie die gefundene Ransomware-Datei im Texteditor.

Beschreiben Sie den Aufbau des Programms. Wo ist der Startpunkt? An welcher Stelle geschieht Rekursion? Wie durchläuft das Programm das Dateisystem? Was passiert dabei mit Ordnern, und was mit Dateien?

An welchen Stellen löscht das Skript Dateien? Wie macht es das? Warum konnten die Dateien im Home-Verzeichnis in Schritt 6.5 nicht wiederhergestellt werden, die Log-Dateien in Schritt 6.7 aber schon?

6.9 Verständnis der Verschlüsselung

Welche „military grade encryption“ hat Oskar eingesetzt? Welche Software wird benutzt, um die Verschlüsselung durchzuführen? Hat Oskar die Verschlüsselung selber programmiert?

Beschreiben Sie die angewandte Kryptographie. Welche Chiffre wird benutzt? Asymmetrisch oder symmetrisch? Blockchiffre oder Stromchiffre? In welchem Modus? Wie viel Bit enthält das Schlüsselmaterial?

Gibt es kryptographische Angriffe gegen diese Chiffre? Wurde ein sicherer Schlüssel gewählt? Warum müssen Sie die Chiffre nicht kryptographisch brechen, um die gekidnappten Dateien zu entschlüsseln? Wie würde sich der Einsatz einer Verschlüsselung mit anderer Symmetrie auf die Chancen auswirken, die Dateien wiederherstellen zu können?

6.10 Umkehrung der Verschlüsselung

Beschreiben Sie grob, wie ein Programm vorgehen müsste, um alle von der Ransomware befallenen Dateien entschlüsseln zu können. Erkennen Sie dabei Ähnlichkeiten zu dem Ablauf der bereits bestehenden Malware?

Sie können die Ransomware umprogrammieren, sich selbst umzukehren! Wie praktisch. An welchen Stellen müssen Sie eine Änderung vornehmen?

Kopieren Sie die Malware aus dem Image in einen Orner auf ihren Computer. Öffnen Sie die Datei und ändern Sie die Zeilen. Stellen Sie sicher, dass die neue Datei *executable* ist. Falls nicht, können Sie dies wie folgt beheben:

```
chmod +x <filename>
```

Kopieren Sie die verschlüsselten Dateien von Mr. Smith in denselben Ordner. Ändern Sie die Malware, sodass Sie in dem gewählten Ordner rekursiert.
Führen Sie die modifizierte Malware aus und entschlüsseln Sie die Dateien.

Mr. Smith hat Ihnen im Vorfeld das schriftliche Einverständnis gegeben, die Dateien zu öffnen, um die korrekte Entschlüsselung zu verifizieren. Was enthalten die Dateien?

Häufig gemachter Fehler "Decrypt Error": Wenn Sie `openssl` **dieselbe** Datei übergeben als Eingabe und als Ausgabe, entsteht ein `Decrypt Error`. Das liegt an folgendem: openssl öffnet die Datei und liest Daten in den RAM. Dann entschlüsselt es das Chiffrat. Der entstandene Klartext wird in die Ausgabedatei geschrieben. Dann versucht openssl mehr Chiffrat-Daten aus der Eingabe-Datei zu lesen... doch diese hat sich verändert! Openssl überschreibt die Datei, während es sie liest - und zerstört sie folgedessen. Sollte Ihnen dies passieren, müssen Sie leider die nun kaputten Dateien löschen und es erneut versuchen mit einem reparierten Skript und neuen Kopien der originalen Chriffraten.

6.11 Rückblick

Beschreiben Sie, wie der Angriff seitens des Online-Shop Betreibers Mr. Smith hätte verhindert werden können.
Welchen Fehler begang Oskar, sodass die Erpressung erfolglos blieb?

7 Kontrollfragen

Die Kontrollfragen sollen nicht in den Auswertungen der Teilnehmer beantwortet werden, sondern dienen der Überprüfung, ob die Teilnehmer vorbereitet sind, um ihr Eingangstestat zu erhalten.

1. Was ist ein *disk image* im Zusammenhang mit der forensischen Untersuchung von Datenträgern?
2. Warum werden Datenträger nicht direkt untersucht, sondern deren Bit-genaue Kopie?
3. Wie groß ist ein *disk image* im Vergleich des Ursprung-Datenträgers?
4. Warum lassen sich *disk images* in der Regel gut komprimieren?
5. Was ist ein *Write-Blocker*?
6. Wofür benutzt man den Befehl *mount*?
7. Wofür wird *diskdump* (kurz *dd*) verwendet?
8. Wie funktioniert die heuristische Erkennung von Dateien, beispielsweise mit *PhotoRec*?
9. Wie funktioniert das Wiederherstellen (die Umkehrung einer Löschung) einer Datei mithilfe des *EXT Journals*?
10. Wann ist eine Datei *sicher* gelöscht?

Literatur

[1] Beispiel Autor. Beispiel Titel mit Umlauten Ääöü, 2014. `http://example.org/example.html`.