Computer Forensik

Seminar Analyse & Angriffe auf Netzwerke

Version 0.1

Zürcher Hochschule für Angewandte Wissenschaften

Daniel Brun

xx. Juni 2015

Eigenständigkeitserklärung

Hiermit bestätigt ich, dass vorliegende Seminararbeit zum Thema „Evaluation einer Mini ERP Lösung für einen Verein“ gemäss freigegebener Aufgabenstellung ohne jede fremde Hilfe und unter Benutzung der angegebenen Quellen im Rahmen der gültigen Reglemente selbständig verfasst wurde.

Thalwil, 11. Februar 2015 Daniel Brun

Inhaltsverzeichnis

1 Einleitung **1**
1.1 Hintergrund . . . 1
1.2 Aufgabenstellung . . . 1
1.3 Abgrenzung . . . 1
1.4 Motivation . . . 2
1.4.1 Computerkriminalität . . . 2
1.5 Struktur . . . 2

2 Angriffe **5**
2.1 Angriffstypen . . . 5
2.2 Kategorien von Schwachstellen . . . 6
2.3 Komplexität . . . 6
2.4 Täter-Typen . . . 6
2.5 Typischer Ablauf . . . 7
2.5.1 Survey (Untersuchung) . . . 8
2.5.2 Delivery (Positionierung) . . . 8
2.5.3 Breach (Ausnutzung) . . . 8
2.5.4 Affect (Beeinträchtigung / Infizierung) . . . 8
2.5.5 Clean Up (Aufräumen) . . . 8

3 Incident Detection & Incident Response **9**
3.1 Incident Detection (Erkennung eines Vorfalls) . . . 9
3.1.1 Hinweise Netzwerkseitig . . . 9
3.1.2 Hinweise Serverseitig . . . 9
3.1.3 Hinweise durch Intrusion-Detection-Systeme . . . 10
3.1.4 Weitere Hinweise . . . 10
3.1.5 Meldung eines Vorfalles . . . 10
3.2 Incident Response Team . . . 11
3.3 Incident Response . . . 12
3.3.1 Organisatorische Vorbereitung . . . 12
3.3.2 Incident Response Prozess . . . 13
3.4 Ablauf . . . 13

4 Computer Forensik **17**
4.1 Einbettung und Definition . . . 17
4.1.1 Forensik . . . 17
Ursprung . . . 17
Bedeutung . . . 17
Teilbereiche . . . 18
4.1.2 IT- / Digitale Forensik . . . 18
Teilbereiche . . . 18
4.1.3 Computer Forensik . . . 19
4.2 Einführung . . . 19
4.3 Anwendungsbereich . . . 19
4.4 Kategorien von Daten . . . 20
4.5 Anti-Forensik und Anti-Detection . . . 20
4.6 Ausbildung & Zertifizierung . . . 21
4.7 Hinweise für die juristische Verwertbarkeit . . . 21
4.7.1 Methoden, Techniken und Programme . . . 21
4.7.2 Glaubwürdigkeit und Reproduzierbarkeit . . . 22
4.7.3 Integrität . . . 22
4.7.4 Präsentation und Dokumentation . . . 22
4.7.5 Beweiskraft . . . 22
4.8 Hinweise zum Datenschutz . . . 22

5 Forensische Analyse **23**
5.1 Einführung . . . 23
5.1.1 Ein guter Prozess . . . 24
5.2 Phasen . . . 24
5.2.1 Readiness (Vorbereitung) . . . 24
5.2.2 Secure (Sicherstellen) . . . 25
Environment (Umgebung) . . . 26
Identify (Identifizieren) . . . 26
Asses and Decide (Beurteilen und Entscheiden) . . . 26
Collect (Sammeln) and Preserve (Aufbewahren) . . . 27
5.2.3 Analysis (Analyse) . . . 29
Preparation (Vorbereitung) . . . 29
Analysis (Analyse) . . . 30
5.2.4 Reporting (Dokumentation) . . . 31
5.2.5 Present (Präsentation) . . . 32
5.2.6 Review (Rückblick) . . . 32
5.3 Beweiskette und Beweissicherung . . . 33
5.4 Hinweise zur forensischen Analyse . . . 33

6 Tools und Techniken 35
6.1 Readiness . . . 35
6.1.1 Datenträger löschen . . . 35
dd . . . 35
Weitere Tools . . . 36
6.2 Secure . . . 36
6.2.1 Auslesen der Zeitkonfiguration . . . 36
6.2.2 Bestimmung der Linux-Distribution . . . 36
6.2.3 Shutdown eines Systemes . . . 38
6.2.4 Erstellen von Hashes . . . 38
md5sum . . . 38
md5deep . . . 39
6.2.5 Sicherung des Arbeitsspeicher-Inhaltes . . . 39
6.2.6 Sicherung des Arbeitsspeicher-Inhaltes ohne Zugriff auf das Betriebssystem . . . 39
6.2.7 Sicherung des Arbeitsspeicher-Inhaltes mit Zugriff auf das Betriebssystem 40
Linux (alte Kernel-Versionen) . . . 40
fmem . . . 40
lime . . . 40
Weitere Tools . . . 40
6.2.8 Forensische Duplikation . . . 41
Einsatz von Standard-Linux Tools . . . 41
dcfldd . . . 42
dc3dd . . . 42
Weitere Tools . . . 42
6.2.9 Verifikation eines forensischen Duplikates oder eines Beweisstückes . . . 43
md5sum und md5deep . . . 43
dcfldd . . . 43
Weitere Tools . . . 43
6.2.10 Sicherung der Binärdatei von ausgeführten Prozessen . . . 43
Linux . . . 43
6.2.11 Sicherung flüchtiger Daten . . . 43
Linux . . . 44
6.3 Analysis . . . 46
6.3.1 Image mounten . . . 46
6.3.2 Linux (dd) . . . 46
6.3.3 ewfmount . . . 46
6.3.4 xmount . . . 47
6.3.5 Gelöschte Datenträger . . . 47
6.3.6 Gelöschte Partitionstabelle . . . 47
Tools . . . 47
6.3.7 Analyse von gelöschten Dateien . . . 47
Sleuth Kit . . . 48
Weitere Tools . . . 48

6.3.8 Analyse von versteckten Dateien . . . 48
6.3.9 Dateien oder Fragemente wiederherstellen . . . 49
foremost . . . 49
Fatback . . . 49
unrm und lazarus . . . 49
Weitere Tools . . . 49
6.3.10 Entpacken von Dateien . . . 49
Linux . . . 50
7-Zip . . . 50
6.3.11 Suche nach Dateien / Filterung von Dateien . . . 50
Linux . . . 50
6.3.12 Analyse des File Slacks . . . 50
6.3.13 Timeline-Analyse . . . 51
Sleuth Kit . . . 51
log2timeline . . . 52
6.3.14 Analyse von Auslagerungsdateien . . . 52
6.3.15 Suche nach Rootkits . . . 52
chkrootkit . . . 53
rkhunter . . . 53
6.3.16 Systemprotokolle . . . 53
6.3.17 Untersuchung der Shell (Bash) . . . 53
6.3.18 Untersuchung der Druckerjobs und der Druckerqueue . . . 53
6.3.19 Untersuchung der Dateien / Dateiendungen . . . 54
Linux . . . 54
6.3.20 Datei- und Verzeichnisrechte . . . 54
6.3.21 Analyse des RAM-Dumps . . . 54
Volatility . . . 55
6.3.22 Konvertierung von Images . . . 55
ewfacquire . . . 55
xmount . . . 55
Weitere Tools . . . 56
6.3.23 Analyse des Master Boot Records . . . 56
Sicherung des Master Boot Records . . . 56
Sicherung des Master Boot Records . . . 56
Extraktion des Master Boot Records aus einem image . . . 56
6.3.24 Weitere Analyse-Möglichkeiten . . . 56
6.4 Reporting . . . 57
6.5 Hinweise zu Tooleinsatz . . . 57
6.6 Tool-Suiten und Toolsammlungen . . . 58

7 Schlusswort **61**
7.1 Fazit . . . 61
7.2 Reflexion . . . 61

Quellenverzeichnis **63**

Anhang **69**

A Vorlage: Formular Incident-Meldung **69**

B Vorlage Formular Ermittlung **71**

C Vorlage: Protokoll **73**

D Vorlage: Beweiszettel **75**

E Ablauf einer forensischen Analyse **77**

Liste der noch zu erledigenden Punkte **79**

KAPITEL 1

Einleitung

1.1 Hintergrund

Im Rahmen meines Bachelor-Studiums in Informatik an der Zürcher Hochschule für Angewandte Wissenschaften (ZHAW) muss im 6. Semester eine Seminararbeit zu einem vorgegebenen Themenbereich erarbeitet werden. Ich habe mich für den Themenbereich „Analyse und Angriffe auf Netzwerke“ entschieden.

Aus einem Themenkatalog konnte ein spezifisches Thema im Bereich „Analyse und Angriffe auf Netzwerke“ ausgewählt werden. Ich habe mich für das Thema „Computer Forensik“ entschieden.

Für die Arbeit sollen circa 50 Arbeitsstunden aufgewendet werden. Dies entspricht etwa einem Umfang von 15 bis 20 Seiten. Zusätzlich gelten die Rahmenbedingungen gemäss dem Reglement zur Verfassung einer Seminararbeit ([Ste12])

1.2 Aufgabenstellung

In dieser Arbeit soll ein Überblick über das Themengebiet der „Computer Forensik“ erarbeitet werden. Es soll gezeigt werden was für Themenbereiche es gibt und was für Werkzeuge und Tools eingesetzt werden können. Das Ganze soll mit einem Ablauf einer forensischen Untersuchung und entsprechenden Beispielen illustriert werden.

1.3 Abgrenzung

Aufgrund des grossen Themengebietes können nicht alle Detail-Aspekte der Computer Forensik berücksichtigt werden. Daher werden in dieser Arbeit nur einige Kernaspekte betrachtet.

Folgende Themengebiete werden im Detail erläutert:

- Analyse von normalen Einzelplatz Unix-Systemen

Explizit ausgeschlossen werden folgende Themenbereiche:

- Detaillierte rechtliche Aspekte (zum Beispiel Strafrechtliches Vorgehen, Strafantrag, Tatortprinzip, etc.)
- Remote-Analyse
- Analyse von RAID-Systemen
- Analyse von Windows und Mac OS X Systemen.

1.4 Motivation

Die forensischen Wissenschaften haben mich seit jeher fasziniert. Zusammen mit meinem berufsbedingten Interesse für Informatik, Computer und andere elektronische Geräte hat sich mit der Zeit das Interesse an der Computer Forensik herauskristallisiert. Ich hatte bereits vor längerer Zeit ein Buch zu diesem Thema gekauft, bin jedoch nie dazu gekommen, mich vertieft damit auseinanderzusetzen. Dieses Seminar hat mir nun ermöglicht, mich vertieft mit diesem Themenkomplex auseinanderzusetzen und erste Einblicke zu erhalten und Erfahrungen zu sammeln.

1.4.1 Computerkriminalität

Unter Computerkriminalität (auch als Cybercrime oder e-Crime bezeichnet) werden heute alle Straftaten zusammengefasst, welche mit Hilfe oder mit Unterstützung von informationsverarbeitenden Systemen durchgeführt wurden. Dazu zählen zum Beispiel: Betrug mit Zugangsberechtigungen, Betrug mit Konto- oder EC-karten mit PIN, Softwarepiraterie, Datenveränderung und Computersabotage oder Ausspähen von Daten. Angreifer können entweder Cyberkriminelle, Konkurrenten, Nachrichtendienste, Hacker, Hacktivisten oder auch Mitarbeiter sein.

Durch die starke Zunahme an Computerkriminalität in den letzten Jahren und die zunehmende Verbreitung von Informationstechnologien werden ich immer mehr Fachkräfte benötigt, welche in der Lage sind entsprechende Untersuchungen durchzuführen.

1.5 Struktur

Diese Arbeit gliedert sich in folgende Hauptteile:

- Einleitung
- Angriffe
- Incident Detection & Incident Response
- Computer Forensik
- Forensische Analyse

- Tools und Techniken
- Schlusswort

Im ersten Kapitel werden die Details zur Ausgangslage und die Hintergründe der Arbeit aufgezeigt. Im darauffolgenden Kapitel wird zum besseren Verständnis die Kategorien und Phasen eines Angriffes aufgezeigt. Anschliessend wird der Ablauf einer Incident Response erklärt. Die darauffolgenden Kapitel beschäftigen sich mit dem Kernbereich der Arbeit, der Computer Forensik. Zuerst werden allgemeine Informationen zur Computer Forensik vermittelt, bevor die Forensische Analyse im Detail betrachtet wird. Im Kapitel 6 werden dann verschiedene Tools und Techniken vorgestellt, welche im Rahmen der forensischen Analyse eingesetzt werden können. Am Ende folgt noch das Schlusswort mit einem Fazit und einer Reflexion über die gesamte Arbeit.

KAPITEL 2

Angriffe

Möchte man jemanden oder etwas besser verstehen, sollte man sich in ihn hineinversetzen und versuchen so zu denken wie er. Dieses Konzept lässt sich auf viele Bereich des Lebens und der Arbeit im Umfeld von Kriminalistik und Strafuntersuchungen anwenden. Ebenfalls lässt sich dieses Konzept im allgemeinen auf die Computer Forensik und im speziellen auf die Incident Response anwenden.

Um auf Angriffe korrekt reagieren zu können und anschliessend die hinterlassenen Spuren zu finden und korrekt auszuwerten ist ein vertieftes Verständnis der eingesetzten Angriffsmethoden und -techniken von Vorteil. Da sich diese Arbeit schwerpunktmässig mit dem Themenbereich „Computer Forensik“ beschäftigt, wird in diesem Kapitel ein grober Überblick über Angriffe auf Computer-Systeme vermittelt.

2.1 Angriffstypen

Grundsätzlich können zwei Angriffstypen unterschieden werden. Auf der einen Seite stehen Massenangriffe, so genannte „un-targeted attacks“, deren Ziel es ist so viele Geräte oder Services als möglich zu treffen. Das einzelne Opfer spielt dabei eine untergeordnete Rolle. Phising und Malware sind zwei Beispiele für solche Massenangriffe. Ausgenutzt wird hier grundsätzlich immer die Offenheit des Internets.

Auf der anderen Seite stehen gezielte Angriffe, so genannte „targeted attacks“. Diese Attacken sind in der Regel auf das Ziel oder das spezifische Szenario, massgeschneidert. Solche Angriffe werden über mehrere Monate hinweg geplant und vorbereitet. Oft sind diese Codes spezifisch entwickelt worden und können somit von Intrusion-Detection-Systemen und Anti-Viren-Software nicht oder nur sehr schwer erkannt werden. Ein Beispielt für eine solche Attacke wäre Spear-Phising.

Bei den gezielten Angriffen hat sich in den letzten Jahren eine neue Unterkategorie, die Kategorie der „advanced persistent threats“. Ziel dieser Angriffe ist es, möglichst lange unerkannt zu bleiben und den Einbruch zu vertuschen. Dabei werden gerade so viele Daten gesammelt, bzw. Aktionen durchgeführt, dass der Täter noch unerkannt bleibt. Ein

solcher Angriff wird über mehrere Monate, wenn nicht sogar Jahre, hinweg vorbereitet und anschliessend Schritt für Schritt umgesetzt. Auch der eingesetzte Schadcode wird so gebaut, dass dieser möglichst lange unterkannt bliebt, aber trotzdem so viel Nutzen als möglich erbringen kann.

2.2 Kategorien von Schwachstellen

Bei einem Angriff werden immer vorhandene Schwachstellen ausgenutzt. Diese Schwachstellen können in drei Kategorien unterteilt werden.

- **Flaws (Fehler / Mängel)**
  Bei einem Flaw handelt es sich um eine unbeabsichtigte Funktionalität der Anwendung. Dieser kann entweder durch schlechtes Design oder simpel und einfach durch einen Implementierungsfehler entstehen.
- **Features (Funktionalitäten)**
  Hier wird eine vorhandene Funktionalität für andere Zwecke missbraucht. Dabei handelt es sich um keinen Fehler in der Anwendungen, sondern um eine Funktionalität, welche entsprechend spezifiziert wurde.
- **User Errors (Benutzer Fehler)**
  User Errors werden durch den Benutzer verursacht. Zum Beispiel könnte ein unerfahrener Systemadministrator unwissentlich Schwachstellen im System freischalten.

2.3 Komplexität

Durch die vorherrschende Monokultur von Betriebsystemen (Windows, MAC OS X), Anwendungen (zum Beispiel: Internet Explorer von Microsoft) und Komponenten werden die Anforderungen an Hacker immer grösser. Der Grund dafür liegt, darin, dass durch die vielen Anwender die meisten Sicherheitslücken und Schwachstellen gefunden werden und anschliessend vom Hersteller behoben werden. Zusätzlich gibt es immer mehr Drittprodukte, welche zusätzlichen Schutz versprechen, beziehungsweise anbieten. Die Angreifer sind gezwungen immer ausgeklügeltere und komplexere Angriffsverfahren zu entwickeln, um einen Weg in das System zu finden. Mit den steigenden Anforderungen werden auch die Angriffe und die Angriffstechniken immer komplexer.

Bild CF Seite 13

2.4 Täter-Typen

Die Motive der Täter sind sehr unterschiedlich. Diese reichen von sozielen, politischen, finanziellen, staatlich-politischen Motiven über technische Ambitionen bis hin zu Regierungen oder Gruppierungen wie Anonymous. Neben der Motviation können die Täter auch nach Aussen- und Innentätern unterschieden werden. Innentäter verfügen über Insider-Wissen und arbeiten in der Regel für das angegriffene Unternehmen oder die angegriffene Organisation. Der Anteil an Innentätern am gesamten Tätervolumen ist sehr hoch und

wächst stetig. Unternehmen und Organisationen sind sich dessen aber nicht immer bewusst und wähnen sich in falscher Sicherheit.

Die „Berufsbezeichnungen“ der Täter sind sehr unterschiedlich und vielfältig. Nachfolgend sind einige der gängigsten Bezeichnungen und deren Bedeutung aufgelistet. Alle diese Berufsbezeichnungen stellen eine spezielle Ausprägung von Hackern dar. Hacker gibt es nicht nur im Infromatik-Bereich, sondern auch in anderen Bereichen wo Technik allgegenwärtig ist. Hacker sind Personen, welche gezielt Schwachstellen (zum Beispiel in einem Computer-System) suchen und diese anschliessend ausnutzen.

- **White-Hat oder „Ethical-Hacker“**
  Ein White-Hat oder Ethical-Hacker führt seine Tätigkeiten nur mit Ausdrücklicher Genehmigung durch und verhalten sich immer nach der Hackerethik. Sie sind meist in der Sicherheitsabteilung einer Organisation oder für eine spezialisierte Unternehmung im Security-Bereich tätig. Ihre Aufgabe ist es die Systeme und Netzwerke mit Penetrationstests zu prüfen, Schwachstellen zu finden und anschliessend entsprechende Massnahmen zu definieren.
- **Black-Hats**
  Black-Hats nutzen Schwachstellen in der Regel für die eigene Bereicherung oder zur Erlangung von ansehen aus. Im Gegensatz zu White-Hats haben sie keine Genehmigung, um diese Aktivitäten durchzuführen, diese sind somit illegal und können von einer Strafverfolgungsbehörde verfolgt werden.
- **Gray-Hats**
  Gray-Hats bewegen sich zwischen den Welten von Black-Hats und White-Hats. Auch sie verschaffen sich unter Ausnutzung von Schwachstellen unautorisierten Zugang zu Computer-Systemen. Im Gegensatz zu Black-Hats verlassen Gray-Hats das System / Netzwerk wieder, sobald sie sich Zugang verschafft haben. Anschliessend benachrichtigen Sie den Besitzer oder Administrator des gehackten Systemes, um diesen auf die Schwachstelle aufmerksam zu machen.
- **Elite Hacker**
  Elite Hacker ist nicht eine direkte Bezeichnung, sondern eher ein sozialer Status für sehr versierte / fähige Hacker.
- **Script Kiddie**
  Ein Script Kiddie hat im Gegensatz zu einem Black-Hat wenig bis gar kein fachliches Know-How und verwendet für seine Attacken vorwiegend vorgefertigte Tools und Scripts.

2.5 Typischer Ablauf

Ein Angriff kann in die nachfolgenden Phasen gegliedert werden. Diese können je nach Angriff in unterschiedlichen Ausprägungen vorkommen.

2.5.1 Survey (Untersuchung)

In dieser Phase werden so viele Informationen wie möglich gesammelt. Dazu gehören Informationen über die Organisation, die eingesetzte Hard- und Software und Prozesse. Anschliessend wird versucht so viele Schwachstellen wie möglich zu ermitteln. Zum einen wird ein Footprinting durchgeführt, welches so viele Informationen wie möglich über die Systeme zu Tage befördern soll. Zum Footprinting gehören unter anderem Port- und Protokollscanns und DNS- und WHOIS-Abfragen. Zum anderen werden mit Hilfe von Social Engeineering und Commodity-Toolkits und -Techniken weitere Schwachstellen ermittelt.

2.5.2 Delivery (Positionierung)

Diese Phase beschäftigt sich mit den expliziten Vorbereitungen für die Ausnutzung der Schwachstellen. Der Angreifer versucht das für dieses Szenario am besten geeignete Vorgehen zu ermitteln und bringt sich anschliessend in Position um die Schwachstellen auszunutzen. Eine typische Aktion in dieser Phase wäre zum Beispiel der Versand einer infizierten E-Mail oder das Unterjubeln eines inifizierten USB-Sticks.

2.5.3 Breach (Ausnutzung)

In dieser Phase wird die Schwachstelle ausgenutzt, um dem Angreifer Zugang zum gewünschten System zu verschaffen.

2.5.4 Affect (Beeinträchtigung / Infizierung)

Nach dem der Angreifer Zugang zum System erlangt hat, unternimmt er weitere Schritte um sein eigentliches Ziel zu erreichen. Dies kann zum Beispiel die Erweiterung seiner Zugriffsrechte, die Einrichtung von Hintertüren, die Sammlung von Daten oder der Angriff eines weiteren Systemes sein.

2.5.5 Clean Up (Aufräumen)

Je nach Ziel und Zweck des Angreifers verwischt er seine Spuren und räumt auf, damit er unerkannt bleibt oder allenfalls zu einem späteren Zeitpunkt nochmals zurückkehren kann.

KAPITEL 3

Incident Detection & Incident Response

Dieses Kapitel beschäftigt sich zum einen mit der Incident Detection, also die Erkennung eines Sicherheitsvorfalles, und zum anderen mit der Incident Response, der Reaktion auf einen Sicherheitsvorfall.

3.1 Incident Detection (Erkennung eines Vorfalls)

Bevor auf einen Angriff, beziehungsweise auf einen Sicherheitsvorfall, reagiert werden kann muss dieser zuerst bemerkt werden. Bleibt der Vorfall unerkannt, wird es nie zu einer Untersuchung kommen. Ein Angriff kann durch verschiedenste Indikatoren erkannt und zum Teil sogar vorausgesagt werden. Nachfolgend werden einige dieser Indikatoren aufgelistet.

3.1.1 Hinweise Netzwerkseitig

- Ungewöhnlich hohe Netzwerklast
- Ungewöhnliche Anzahl Firewall-Regelverstösse

3.1.2 Hinweise Serverseitig

- Unbekannte Prozesse
- Unbekannte / Neue User
- Unbekannte Dateien
- Ungewöhnliche Systemlast
- Dienste laufen nicht mehr
- Ungewöhnliche Systemanmeldungen
- Systemabsturz

- Kleiner werdende Log-Files
- Bestehende Dateien werden grösser (Beispiel: Ausführbare Datei wächst um mehrere kB)
- Versuch Berechtigungen zu verändern
- Schlechte Performance

3.1.3 Hinweise durch Intrusion-Detection-Systeme

Intrusion-Detection-Systeme sind dazu da Angriffe möglichst früh zu erkennen und die entsprechenden Stellen zu informieren. Ist das Intrusion-Detection-System gut konfiguriert, kann dieses Angriffe anhand von Strategien und Mustern erkennen.

3.1.4 Weitere Hinweise

Weitere Hinweise können durch Kunden, Partner, Mitarbeiter, Strafverfolgungsbehörden oder die Presse erfolgen.

3.1.5 Meldung eines Vorfalles

Wurde ein möglicher Sicherheitsvorfall oder ein Angriff gemeldet, ist es wichtig, dass die Person, welche die Meldung entgegen nimmt korrekt und schnell reagiert. Personen welche solche Meldungen entgegen nehmen könnten (z.B. Mitarbeiter des Service Desks) sollten geschult und mit einem entsprechenden Merkblatt und einer Checkliste / Formular ausgestattet werden. Die entgegenehmende Person muss vom Melder so viele Informationen wie möglich erfragen, damit anschliessend schnellere und effizientere Entscheidungen getroffen werden können. Dabei sind sowohl Informationen zum Melder, als auch über die Symptome und den Zustand des Systemes von Interesse. Ein Beispiel für ein solches Formular ist im Anhang A Vorlage: Formular Incident-Meldung zu finden.

Sollte die Meldung des Vorfalles nicht direkt an das Incident Response Team gelangt sein, muss der Vorfall unverzüglich dem zuständigen Incident Response Team gemeldet werden. Ist kein ständiges Incident Response Team vorhanden, muss dieses entsprechend aufgeboten werden. Gibt es in der Organisation kein Incident Response Team und keinen Incident Response Plan ist das weitere Vorgehen mit dem Vorgesetzten und allenfalls einem Mitglied des höheren Managements abzustimmen. Übereilte Reaktionen sollten vermieden werden, da dadurch Beweisspuren verwischt oder vernichtet werden können.

3.2 Incident Response Team

Das Incident Response Team ist die Eingreiftruppe beim Eintreten eines Sicherheitsvorfalles. Die Aufgabe dieses Team ist es im Falle eines Incidents auf Basis der vorhandenen Informationen eine Lagebeurteilung und Risikoeinschätzung durchzuführen und anschliessend entsprechende Massnahmen einzuleiten.

In einem Incident Response Team sollten folgende Rollen besetzt werden.

- **Kern-Team**
  - Koordinator / Leiter mit direktem Zugang zum Management
  - Kontaktstelle zur Entgegennahme von Verdachtsmeldungen
  - Incident-Spezialist oder einen Ermittler aus dem Bereich der Computer Forensik
- **Erweitertes Team**
  - Juristischer Berater
  - Auditor
  - Mitarbeiter der physikalischen Sicherheit
  - HR-Mitarbeiter
  - Fachspezialisten (z.B. Netzwerk-, Sicherheits- oder Datenbankadministratoren)

Die Mitarbeiter dieses Teams sollten über längere Erfahrung in ihrem Tätigkeitsbereich verfügen, gute Kommunikationsfähigkeiten besitzen, teamfähig sein und gut integriert und zuverlässig sein. Darüber hinaus müssen sie in der Lage sein unter Stress effiziente und akzeptable Entscheide zu treffen, sich an vorgegebene Regeln und Prozeduren zu halten und in sicherheitsrelevanten Aspekten als Vorbild dienen. Sie müssen in der Lage sein sich unter Stress an vorgegebene Regeln und Prozeduren zu halten.

Bei grossen Organisationen kann das Incident Response Team als Dauerhaftes Team vorhanden ist, welches auch noch andere Aufgaben im Sicherheitsbereich wahrnimmt. Bei kleineren Organisationen kann es sich um ein Team mit Mitgliedern aus mehreren Organisationseinheiten handeln, welche im Notfall zusammengerufen werden können. Denkbar ist es auch, dass das ganze Incident Response Team oder einen Teil davon (z.B. den Incident-Spezialisten) durch eine externe spezialisierte Unternehmung wahrgenommen wird.

3.3 Incident Response

Die Incident Response hat zum Ziel bei einem Sicherheitsvorfall so rasch als möglich den entstandenen Schaden zu beurteilen, die verwendeten Angriffsmethoden und die Auswirkungen für die Organisation zu bestimmen und anschliessend entsprechende Massnahmen zu planen und umzusetzen. Als Ansatzpunkt sollte immer zuerst die Ursache und die ausgenutzte Schwachstelle ermittelt werden. Ausgehend von diesen Informationen können weitere Schritte unternommen werden.

Die Computer Forensik ist ein essentieller Bestandteil des Incident Response Prozesses. Sie stellt die Methoden, Techniken und Werkzeuge zur Auffindung, Analyse und Auswertung der Spuren zur Verfügung. Es ist dabei notwendig die Massnahmen zur Beweissicherung fest im Prozess zu integrieren und zu etablieren. Nicht korrekt sichergestellte Spuren und Hinweise können unter Umständen juristisch nicht mehr verwertet werden. Ein guter und erfolgreicher Incident Response Prozess ist eine gute Grundlage für eine juristische Verfolgung des Angreifers.

3.3.1 Organisatorische Vorbereitung

Um schnell, effizient und korrekt auf einen Sicherheitsvorfall reagieren zu können ist es empfehlenswert einige Vorbereitungen auf organisatorischer Ebene zu treffen. Nachfolgend werden die wichtigsten Punkte aufgelistet, welche als Vorbereitung durchgeführt werden sollten. Diese Punkte können in einem Incident Response Plan festgehalten werden.

- Incident Awareness
  Bewusstsein für mögliche Sicherheitsvorfälle bei Mitarbeitern fördern.
- Konzept / Prozess für Monitoring und Alarmierung (zum Beispiel: zentralisierte Logs, Server-Auditing)
- Umsetzung des Konzeptes / Prozesses für Monitoring und Alarmierung im Rahmen des System Life Cycles.
- Weiterbildungen / Schulungen im Bereich Incident Detection und Incident Response
- Einholen der notwendigen Autorisierungen für die Einleitung der notwendigen Massnahmen.
- Festlegung der Rollen und Verantwortlichkeiten (inkl. Eskalations- / Alarmierungsregelung und Weisungskompetenzen)
- Konzept / Prozess für die Behandlung eines Sicherheitsvorfalles (Incident Response Prozess)
- Aufbau einer Datenbank mit den File-Hashes von bekannten / installierten und als ungefährlich eingestuften Betriebssystemen und Anwendungen.
- Verfassung und Etablierung von entsprechenden Policies, Guidelines und Procedures

Auch sollte der Kontakt zur Ermittlungsbehörde bereits im Vorfeld hergestellt werden, damit im Ernstfall ein entsprechender Kontakt bereits vorhanden ist und rasch reagiert werden kann. Gegebenenfalls ist es auch sinnvoll den Kontakt zu einem externen Security-Spezialisten herzustellen, falls nicht ausreichend Know-How vorhanden ist.

3.3.2 Incident Response Prozess

Wurde ein Vorfall gemeldet gilt es zuerst zu beurteilen, ob es sich um einen wirklichen Sicherheitsvorfalle handelt, oder ob es sich um eine Betriebsstörung handelt.

Handelt es sich um einen Sicherheitsvorfall muss auf Basis der vorhandenen Informationen eine erste Einschätzung durchgeführt werden. Um für die Einschätzung alle relevanten Informationen zur Verfügung zu haben, ist es essentiell, dass bei der Entgegennahme der Meldung die entsprechenden Informationen erfragt werden (Siehe dazu Kapitel **??**). Sind zu wenig Informationen vorhanden, kann bereits eine erste Analyse durchgeführt werden. Es ist jedoch darauf zu achten, dass keine Beweise durch unbedachtes / übereiltes handeln zerstört werden. Ist kein polizeilicher Ermittler oder ein entsprechend ausgebildeter Spezialist vor Ort, sollte auf voreilige Aktionen verzichtet werden, da diese oft mehr Schaden als Nutzen anrichtet.

3.4 Ablauf

Der Ablauf einer Incident Response ist immer stark von der jeweiligen Situation abhängig. Bei einem nicht kritischen System kann es unter Umständen sinnvoll sein, den Angreifer weitgehendst ungestört zu lassen und ihn zu beobachten. So können allenfalls wichtige Erkenntnisse und Hinweise zum Täter gesammelt werden, welche für die Identifizierung hilfreich sein könnten.

Bei einem kritischen System würde der Angriff wahrscheinlich so rasch als möglich unterbunden, das System gehärtet und anschliessend wieder in Betrieb genommen werden.

Eine weiteren Einfluss auf den Ablauf hat auch der Zeitpunkt des Angriffes. Je nach dem, wann die Meldung über den Sicherheitsvorfall eingegangen ist, kann der Angriff im vollen Gange oder aber schon vorbei sein. Es kann auch vorkommen, dass der eigentliche Angriff selbst noch gar nicht stattgefunden hat, aber zum Beispiel durch das Monitoring oder ein Intrusion Detection System Hinweise auf einen bevorstehenden Angriff erkannt wurden.

1. **Identify (Identifizierung)**

   a) Eingang eines Hinweises für einen Verdachtsmoment
   (Siehe dazu Kapitel 3.1.5 Meldung eines Vorfalles)

2. **Assess (Beurteilung)**

   a) Identifizierung der betroffenen Systeme

b) Durchführen einer ersten Analyse / Sicherstellung von Spuren

c) Einschätzung der Situation auf Basis der vorhandenen Informationen

d) Handelt es sich um einen Sicherheitsvorfall oder eine Betriebsstörung? Bestätigung / Wiederlegung des Verdachtes.

e) Information des Managements und weiteren zu involvierende Stellen.

3. **Respond (Reagieren)**

a) Klassifizierung des Vorfalles
Mögliche Klassifizierungen:

- Probing
- Portscanning
- Denial-of-Service Angriff
- Unberechtigter Zugriff auf User-Account / Admin-Account
- Datendiebstahl
- Datenmanipulation
- ...

b) Auswahl einer Response-Strategie
Zu berücksichtigende Faktoren:

- Kritikalität des betroffenen Systems in Bezug auf die Unternehmensprozesse
- Kritikalität / Wichtigkeit der gestohlenen Daten.
- Täter-Vermutung
- Erforderliches Wissen / Fähigkeiten beim Täter
- Wie weit ist der Täter gekommen?
- Ist eine Downtime verkraftbar?
- Geschätzter finanzieller Schaden.
- Ist der Vorfall an die Öffentlichkeit gelangt?

c) Entscheid über Umsetzung der gewählten Strategie durch Management der Systemeigentümer.

d) Vermeidung von unüberlegten Aktionen und Gegenangriffe

e) Vorbereitung und Durchführung einer forensischen Analyse.
(Siehe dazu die Kapitel 4, 5 und 6)

f) Muss der Sicherheitsvorfall veröffentlicht werden ? (Abwägung der Vor- / Nachteile, Eventuell muss der Vorfall aufgrund einer bindenden Vereinbarung gemeldet werden.)

g) Gibt es eine Versicherung für diese Art von Vorfall?
Wenn Ja: Einbezug der Versicherung

h) Meldung des Vorfalles an die Strafverfolgungsbehörde (falls notwendig)

4. **Report (Bericht)**

a) Aufzeigen der Kennzahlen: Reaktionszeit, Wirksamkeit, Kosten, etc.

b) Verfassung eines detaillierten Berichtes über den Vorfall und die forensische Analyse.

5. **Review (Rückblick)**

a) Analyse Ermittlungsablauf

b) Optimierung / Verbesserung Incident Response Prozess

c) Festlegung von permantenten Massnahmen.

6. **Measures (Massnahmen)**
Die aufgelisteten Massnahmen können je nach Situation bereits während den Schritten 3, 4 oder 5 durchgeführt werden.

a) Überprüfung / Update / Wiederherstellung der kompromittierten Systeme

b) Vorläufige Sperrung von verwendeten Accounts / Erzwingung Passwort-Wechsel für die betroffenen Accounts.

c) Umsetzung von permanenten Massnahmen.

KAPITEL 4

Computer Forensik

Dieses Kapitel definiert den Begriff der Computer Forensik und beschreibt das Themengebiet im Allgemeinen.

4.1 Einbettung und Definition

4.1.1 Forensik

Ursprung

Der Begriff „Forensik“ stammt aus den Zeiten des antiken Roms. Damals wurden Gerichtsverfahren, Untersuchungen, Urteilsverkündungen und der Vollzug von Strafen öffentlich auf dem Marktplatz abgehalten. Marktplatz (oder auch Forum) wird im lateinischen mit *forum* bezeichnet. Die Plural-Form von *forum* ist *foren*. Aus dieser Plural-Form hat sich der Begriff „Forensik“ entwickelt.

Bedeutung

Die Forensik ist ein Wissenschaftszweig, welche sich mit dem Nachweis, Beweis und der Aufklärung von kriminellen, oder allgemein strafbaren, Handlungen beschäftigt. Die forensische Untersuchung ist eine systematische Analyse mit dem Ziel strafbare Handlungen zu identifizieren, analysieren und rekonstruieren.

Der „Guide to Integrating Forensic Techniques into Incident Response“ des National Institute of Standards and Technology (NIST) beinhaltet eine kurze und prägnante Definition für den Begriff der „Forensik“.

„Forensic science is generally defined as the application of science to the law“
[E20d, S. ES-1]

Übersetzt bedeutet dies so viel wie „Forensische Wissenschaft ist allgemein definiert, als die Anwendung der Wissenschaft für das Gesetz“.

Teilbereiche

Wie in der vorangehenden Definition bereits angedeutet, gibt es grundsätzlich für jeden Wissenschaftszweig einen entsprechenden Wissenschaftszweig in der Forensik. Nachfolgend sind einige für die Strafverfolgung bedeutendsten Teilbereiche der Forensik aufgelistet.

- Forensische Pathologie
- Forensische Kriminaltechnik
- Forensische Phsychiatrie und Psychologie
- Forensische Toxikologie
- Ballistik
- Computer-Forensik

4.1.2 IT- / Digitale Forensik

Die IT-, bzw. Digitale, Forensik beschäftigt sich mit der Auffindung, Untersuchung und Wiederherstellung von Material, bzw. Daten, auf elektronischen, bzw. digitalen, Geräten. Dabei kann es sich zum Beispiel sowohl um verlorene Daten, als auch um explizites oder nicht explizites Beweismaterial handeln.

Teilbereiche

Die Unterteilung der IT- / Digitalen Forensik in ihre Teilgebiete ist nicht offiziell definiert. Nachfolgend wird eine mögliche Unterteilung aufgezeigt. Diese Unterteilung ist nicht vollständig und nicht abschliessend.

- Computer Forensik
- Forensische Datananalyse
- Datenbank Forensik
- Mobile Device Forensik
- Netzwerk Forensik
- Forensische Videoanalyse
- Forensische Audioanalyse

4.1.3 Computer Forensik

Für die Definition der Computer Forensik existieren zum heutigen zwei verschiedene Ansätze. Ein Ansatz sieht die Computer Forensik als Teilgebiet der IT-, bzw. der Digitalen Forensik. Der andere Ansatz betrachtet den Begriff Computer Forensik als Synonym zu den Begriffen IT- und Digitale Forensik.

Diese Arbeit richtet sich nach dem ersten Ansatz, bei dem die Computer Forensik ein Teilgebiet der Digitalen Forensik ist.

Die Computer Forensik ist ein Teilgebiet der Digitalen Forensik und beschäftigt sich mit der Analyse von Computer-Systemen mit Fokus auf Einzelplatzsysteme und Server.

4.2 Einführung

Die Computer Forensik kann in verschiedenen Kontexten zum Einsatz kommen. Zum einen erfolgt während, bzw. nach einem Sicherheitsvorfall (Incident), z.B. Systeminbruch eine forensische Untersuchung (Mehr dazu im Kapitel **??**).

Im Kontext der Incident Response ist es das Ziel der Computer Forensik die ausgenutzte Schwachstelle zu finden, den Schaden zu beziffern, den Angreifer zu Identifizieren und die Beweise für allfällige juristische Schritte zu sichern. Im Kontext der Untersuchung von Straftaten oder ähnlich ist es das Ziel, aus dem System so viele Informationen wie möglich zu extrahieren und diese anschliessend zu analysieren. Aus den analysierten Daten werden Beweise gewonnen, welche entweder eine bestimmte Theorie unterstützen, widerlegen oder keine der beiden Aussage unterstützen.

4.3 Anwendungsbereich

Die Computer Forensik findet unter anderem in folgenden Bereichen Anwendung:

- Strafuntersuchungen
- Incident Response / Incident Handlung
- Log Monitoring
- Datenwiederherstellung
- Datenbeschaffung

4.4 Kategorien von Daten

In der Computer Forensik können grundsätzlich zwei Kategorien von Daten unterschieden werden. Auf der einen Seite stehen flüchtige Daten. Diese Daten stehen in der Regel nur temporär zur Verfügung und sind spätestens mit einem normalen Shutdown des Systems unwiderruflich verloren. Als Beispiel seien her der Inhalt des Arbeitspseichers oder die Liste der aktiven Prozesse genannt. Auf der anderen Seite stehen nichtflüchtige Daten. Diese Daten sind auch nach einem Shutdown des Systemes noch vorhanden und können ausgelesen werden. Dazu zählen zum Beispiel Systemdateien, Programme oder Daten des Benutzers (Fotos, Videos, Dokumente, etc.)

Flüchtige Daten können weiter in Unterkategorien aufgeteilt werden. Zum einen gibt es die Flüchtigen Daten selbst, welche bei einem normalen Shutdown verloren gehen (Zum Beispiel: Cache-Inhalte, Inhalt des Hauptspeichers, Status der Netzwerkveribndungen). Zum anderen gibt es noch fragile Daten. Fragile Daten sind grundsätzlich gespeichert und stehen theoretisch auch nach einem Shutdown weiter zur Verfügung. Bei fragilen Daten besteht jedoch die Gefahr, dass diese sich bei einem Zugriff ändern können (zum Beispiel die Zeit des letzten Dateizugriffes unter Unix). Die dritte Unterkategorie beinhaltet die temporären Daten. Diese Daten stehen nur zu einem bestimmten Zeitpunkt zur Verfügung.

Flüchtige Daten sind meist von sehr hohem Interesse und sollten als erstes gesichert werden. Die Sicherung dieser Daten erfordert jedoch ein besonnenes und koordiniertes Vorgehen.

4.5 Anti-Forensik und Anti-Detection

Straftäter und Angreifer auf Computer Systeme werden sich immer mehr bewusst, dass sie Spuren auf dem System hinterlassen. Diese versuchen dann entweder keine oder so wenig Spuren wie möglich zu hinterlassen, Spuren und Beweise zu verändern oder gar zu löschen oder falsche Fährten zu legen. Dies kann entweder manuell oder mit Hilfe von Anti-Forensik und Anti-Detection Tools erfolgen.

Das primäre Ziel dabei ist, zu verhindern, dass das Eindringen oder die verdächtige Handlung entdeckt wird. Dies wird eigentlich eher dem Themenbereich der Anti-Detection, also dem „Unbemerkt bleiben“, zugeordnet. Bei der Anti-Detection versuchen die Täter unerkannt und unbemerkt zu bleiben. Zusätzlich wird versucht die Ermittler zu behindern, abzulenken oder die Datensammlung zu stören oder zu unterbinden. Zum Teil wird auch versucht den Umstand ausgenutzt, dass für eine Ermittlung nur eine beschränktes Zeitkontingent und Budget vorhanden ist. Dies kann dazu führen, dass der Ermittler nur die Beweise findet, die er soll und sich dann aus zeitlichen und budgettechnischen Gründen damit zufrieden gibt und die Untersuchung abschliesst.

Kennt der Angreifer die eingesetzten Werkzeuge oder kann diese ermitteln, kann er Schwachstellen und Sicherheitslücken in diesen ausnutzen und gezielt angreifen. im schlimmsten Fall kann der Angreifer die Ermittlungen gezielt manipulieren, ohne dass der Ermittler dies bemerkt. Daher sollte die Analyse zum einen in einer geschützten Umgebung durchgeführt werden und die verwendete regelmässig upgedatet werden.

4.6 Ausbildung & Zertifizierung

In der Schweiz gibt es aktuell nur wenige Ausbildungsprogramme im Bereich der Computer Forensik. Einige Hochschulen und Universitäten bieten aktuell nur einzelne Kurse zu diesem Themenbereich an. Für die Absolvierung von international anerkannten Zertifizierungen gibt es nur wenige Anbieter in der Schweiz. Nachfolgend werden die gängigsten Zertifizierungen aus dem Bereich der Computer Forensik aufgelistet. Neben diesen Zertifizierungen gibt es noch einige weitere, welche von den Herstellern spezieller Hard- und Software angeboten werden.

- Certified Computer Examiner (CCE)
  The International Society of Forensic Coputer Examiners
- Computer Hacking Forensic Investigator (CHFI)
  International Council of E-Commerce Consultants
- Certified Computer Forensics Examiner (CCFE)
  International Association of Computer Investigative Specialists
- Certified Forensic Analyst (GCFA)
  Global Information Assurance Certification
- Certified Forensic Examiner (GCFE)
  Global Information Assurance Certification
- Certified Network Forensic Analyst (GNFA)
  Global Information Assurance Certification
- Reverse Engineering Malware (GREM)
  Global Information Assurance Certification
- Professional Certified Investigator (PCI)
  ASIS International

4.7 Hinweise für die juristische Verwertbarkeit

Sollen die sichergestellten Daten und Informationen juristisch verwertbar sein, zum Beispiel als Beweise in einem Strafprozess müssen einige zusätzliche Punkte beachtet werden. Grundsätzlich ist es sinnvoll die folgenden Punkte bei jeder Untersuchung zu berücksichtigen.

4.7.1 Methoden, Techniken und Programme

Die angewendeten Methoden und eingesetzten Techniken und Programme sollten in der Fachwelt akzeptiert und beschrieben sein. Neue Tools und Verfahren haben in der Regel einen schweren Stand, bis diese allgemein akzeptiert wurden.

4.7.2 Glaubwürdigkeit und Reproduzierbarkeit

Um die Glaubwürdigkeit der Ergebnisse sicherzustellen müssen sämtliche Schritte und die resultierenden Ergebnisse von Laien nachvollzogen werden können. Zusätzlich müssen die Ergebnisse durch einen anderen Experten reproduziert werden können. Der Ermittler, bzw. die Person, welche die forensische Untersuchung durchgeführt hat, muss in der Lage sein den gesamten Ablauf im Detail zu erklären. Erklärungen im Stiel von „Diese Information wurde vom eingesetzten Analyse-Programm automatisch gefunden“ sind nicht gern gesehen und können die Glaubwürkdigkeit der gesamten Untersuschung in Frage stellen.

4.7.3 Integrität

Während der gesamten Ermittlung (und auch darüber) hinaus muss die Integrität der untersuchten Daten und gefundenen Informationen, Daten und Beweise lückenlos sichergestellt werden. Die Integrität muss jederzeit vollständig belegt werden können.

4.7.4 Präsentation und Dokumentation

Die Ergebnisse müssen angemessen dokumentiert und präsentiert werden. Am geeignetsten ist es, wenn die Ergebnisse in Form von Ursache - Wirkung aufgezeigt werden. Die Beweisspuren, Ereignisse und Personen sollen möglichst logisch und nachvollziehbar in Relation zu einander gebracht werden.

4.7.5 Beweiskraft

Die gefundenen Informationen und Daten (zum Beispiel: Einträge in Log-Dateien) haben für sich alleine keinerlei Beweiskraft. Es handelt sich dabei um Sachbeweise. Die Beweiskraft ergibt sich erst durch den Kontext, bzw. durch die Person, welche den Beweis in Zusammenhang mit der Tat bringt. Der Sachbeweis ist somit eng mit dem Personenbeweis verbunden.

Ein Beweis verliert relativ rasch seine Beweiskraft, wenn dieser unrichtig, bzw. falsch, dargelegt wurde. Es ist daher zwingend erforderlich, denn Beweis sachlich zu präsentieren. Auch die Integrität und Glaubwürdigkeit der präsentierenden Person ist ebenfalls von hoher Wichtigkeit.

4.8 Hinweise zum Datenschutz

Auch im Rahmen einer Computer forensischen Analyse hat der Datenschutz weiterhin seine Gültigkeit und die Analyse und Auswertung von personenbezogenen Daten muss entsprechend genehmigt / autorisiert werden. Im Zweifelsfall sollte die zuständige Stelle in der Organisation (zum Beispiel der Datenschutzbeauftrage) hinzugezogen werden. Im Rahmen einer strafrechtlichen Untersuchung kann bei überwiegendem öffentlichen Interesse der Datenschutz jedoch ausser Kraft gesetzt werden.

KAPITEL 5

Forensische Analyse

In diesem Kapitel wird der Ablauf der Computer forensischen Analyse im Detail aufgezeigt und erklärt. Die Techniken und Tools zur Unterstützung dieses Prozesses werden im Kapitel 6 Tools und Techniken erläutert.

5.1 Einführung

Der Prozess der forensischen Analyse lässt sich grundsätzlich in die nachfolgenden Phasen unterteilen werden. Die Erläuterung der einzelnen Phasen erfolgt in den nachfolgenden Kapiteln.

1. Readiness (Vorbereitung)
2. Secure (Sicherstellung)
3. Analysis (Analyse)
4. Documentation (Dokumentation)
5. Present (Präsentation)
6. Review (Rückblick)

Ziel ist es auf den sichergestellten Datenträgern Beweise zu finden. Dazu werden zuerst die Datenträger gesichert und anschliessend die Daten extrahiert. Aus den extrahierten Daten werden dann Informationen gewonnen, welche allenfalls als Beweise im Zusammenhang mit anderen Informationen verwendet werden können.

5.1.1 Ein guter Prozess

Die Grundlage für eine saubere Beweisaufnahme und eine mögliche juristische Verwertbarkeit dieser Beweise ist ein guter forensischer Prozess. Ein guter Forensischer Analyse Prozess zeichnet sich durch folgende Punkte aus:

- Kreuzvalidierung von essentiellen Ergebnissen mit anderen Tools
- Sauberer und Korrekter Umgang mit Beweismaterial
- Untersuchung wird vollständig durchgeführt (Berücksichtigung aller Aspekte)
- Case-Management für die Verwaltung der Untersuchung
- Dokumentation und Archivierung der Beweismittel und der Ergebnisse
- Dokumentierte und geprüfte Arbeitsprozesse
- Konformität zu gesetzlichen Vorgaben und Restriktionen
- Flexibilität

5.2 Phasen

5.2.1 Readiness (Vorbereitung)

Um während der Untersuchung Fehler zu verhindern und wertvolle Zeit zu sparen, ist es sinnvoll gewisse Vorbereitungsarbeiten vor jedem Einsatz, beziehungsweise vor jeder Untersuchung, durchzuführen.

Vorbereitungsarbeiten

- Sterilisieren / Formatieren von Datenträgern für die Sicherung des Beweismaterials
- Formulare und Protokolle vorbereiten und ausdrucken
- Vorbereitung und Verpackung der notwendigen technischen Ausrüstung
  - Kleines Werkzeugset
  - Digitalkamera
  - Handschuhe
  - Notizblock und Stifte
  - Wasserfeste Filzstifte und Etiketten
  - Antistatische Beutel und Verpackungsmaterial
  - Messer, Schere und Zange
  - Taschenlampe

  - Erdungs- / Anti-Statik-Armband
  - Dokumente (Manuals, Anleitungen, Abläufe, etc.)
  - Writeblocker
  - Datenträger / Speichermedien
  - (Mobiles) Analysesystem + Zubehör (Adapter, USB-Hubs, Card-Reader, Multi-Card-Reader, CD/DVD/Blue-Ray Leser / Brenner, Drucker)
  - ...
- Vorbereitung und Verpackung der notwendigen Tools und Programme
  - Tool für Datensicherungen
  - Tool zum Auffinden von (versteckten) Dateien
  - Tool zur Sicherung und Auswertung der Internet-History und des Caches der verschiedenen Browser
  - Tool zum Öffnen von Multimediainhalten
  - Tool zum Öffnen von E-Mail-Nachrichten
  - Tool zum Öffnen der gängigen Dokumente (PDF, XML, Microsoft-Office Dokumente, ...)
  - Tool zum Knacken von Passwörtern.
  - Tool zur Untersuchung von mobilen Geräten
  - Tool um grosse Datenmengen zu analysieren
  - ...

5.2.2 Secure (Sicherstellen)

Die Phase „Secure“ lässt sich weiter in die Phasen „Environment (Umgebung)“, „Identify (Identifizieren)“, „Asses and Decide (Beurteilen und Entscheiden)“, „Collect (Sammeln)“ und „Preserve (Aufbewahren)“ unterteilen.

Environment (Umgebung)

Diese Phase muss grundsätzlich nur berücksichtigt werden, wenn es sich bei der Untersuchung um eine Ermittlung im Rahmen einer Incident Response oder einer Tatortssicherung handelt.

Bei Ankunft des Ermittlers am „Tatort“ sollte er sogleich sicherstellen, dass nur noch berechtigte Personen Zugang zum Tatort und der näheren Umgebung haben. Bevor die Personen den Tatort verlassen, sind zum einen die Kontaktdaten für spätere Rückfragen und zum anderen weitere Informationen (zum Beispiel: Passwörter, Besonderheiten des Systems) zu protokollieren. Sofern noch nicht erfolgt, sollte der Tatort isoliert und dokumentiert werden. Für die Tatortdokumentation sind Fotos und Skizzen sehr gut geeignet.

Identify (Identifizieren)

Zuerst müssen sämtliche Datenquellen am Tatort und in der näheren Umgebung identifiziert werden. Dazu zählt zum Beispiel das zu untersuchende System, externe Festplatten, USB-Sticks, Digitalkameras, MP3-Player, Wechseldatenträger, etc. Zu den möglichen Datenquellen zählen auch entfernte Systeme wie Firewalls, Router, Internet-Provider, etc. und physische Dokumente, Zettel, Notizen, etc.

Falls gestattet, sollte in der näheren Umgebung, zum Beispiel im Aktenschrank oder im Korpus, nach weiteren Datenquellen gesucht werden. Eine mehr oder weniger gut versteckte Passwortliste kann die Arbeit erheblich erleichtern. Wurden nicht an ein System angeschlossene Datenträger, Medien und Dokumente gefunden ist dies sofort zu dokumentieren und der Beweisgegenstand fachgerecht zu sichern. Mehr dazu im nächsten Kapitel.

Asses and Decide (Beurteilen und Entscheiden)

Mit dem aktuellen Wissensstand muss der Ermittler nun eine Lagebeurteilung durchführen und eine Strategie für das weitere Vorgehen auswählen. Je nach Situation muss an dieser Stelle das Incident Response Team, ein Mitglied des Managements oder der Ermittlungsleiter involviert werden, um das weitere Vorgehen abzusprechen und zu genehmigen.

Der Ermittler kann entweder eine Live-Analyse oder eine Post-Mortem-Analyse durchführen. Bei einer Live-Analyse wird das System nicht heruntergefahren und es wird versucht so viele Daten als möglich zu sichern, ohne den Zustand des Systemes allzu stark zu verändern, beziehungsweise eigene Spuren zu hinterlassen. Eine Live-Analyse wird typischerweise durchgeführt, wenn ein unternehmenskritisches System betroffen ist, welches nicht heruntergefahren werden kann. Hier gilt es sorgfältig das Kosten / Nutzen-Verhältnis abzuwägen. Handelt es sich um ein kritisches System, der Schaden ist vergleichsweise klein und die Wahrscheinlichkeit den Täter ausfindig zu machen sehr gering, lohnt sich ein herunterfahren aus wirtschaftlicher Sicht oft nicht. Für eine Live-Analyse muss zwingend ein Zugang zum System bestehen.

Bei einer Post-Mortem-Analyse wird der Grossteil der Sicherungs- und Analyse-Arbeiten erst durchgeführt, wenn das System ausser Betrieb genommen wurde. So ist gewährleistet, dass der Datenstand erhalten bleibt und so wenig Beweise als möglich vernichtet oder beschädigt werden. Einer Post-Mortem-Analyse kann auch eine Live-Analyse vorausgehen. In der Regel wird bei einem laufenden System versucht so viele flüchtige Daten als möglich zu sammeln. Sobald diese gesichert wurden, wird das System ausser Betrieb genommen und die weiteren Sicherungs- und Analyseschritte durchgeführt.

Collect (Sammeln) and Preserve (Aufbewahren)

Nachdem sicher der Ermittler für eine Strategie entschieden hat, sollte auf Basis der identifizierten Datenquellen und der gewählten Strategie eine Planung zur Sicherung der Daten ausgearbeitet werden. Dieser Plan sollte die Sicherungsreihenfolge der Datenquellen anhand ihrer Priorität und die dazu zu verwendenden Tools und Techniken beinhalten. Häufig ist es nicht möglich, sämtliche Datenquellen zu sichern, daher müssen diese auf Basis der Halbwertszeit der Daten, der Erfahrung des Ermittlers, der Einschätzung der Situation und des Aufwandes (Zeit / Kosten / Ausrüstung) um die Daten zu sichern, priorisiert werden.

Bei der Sicherung ist zu beachten, dass auf dem System allenfalls trojanisierte Systemprogramme installiert wurden. Es sollten daher für sämtliche Befehle statisch vorkompilierte Programme verwendet werden. Viele Forensik-Kits bieten einen guten Grundstock von statisch vorkompilierten Anwendungen für die Sicherung und Analyse.

System ist eingeschaltet
Dieser Schritt gilt grundsätzlich für beide Analyse-Strategien. Bei einer Live-Analyse sind die Zustände entsprechend zu dokumentieren, jedoch ohne einen Shutdown durchzuführen.

1. Nähere Umgebung und Zustand des Systems dokumentieren
2. Befindet sich das System im Standby?
   Befindet sich das System im Standby ist abzuwägen, ob das System aufgeweckt oder ein harter Shutdown gemacht werden soll. In dieser Situation ist in der Regel ein harter Shutdown zu empfehlen.
3. Ist der Screensaver aktiv?
   Ist auf dem System ein Screensaver ist abzuwägen, ob dieser „deaktiviert“ werden soll oder ein harter Shutdown gemacht werden soll. Dies ist stark situationsabhängig und sollte von Fall zu Fall entschieden werden. Ist anzunehmen, dass die Freischaltung des Screens mit einem Passwort erfolgt, welches nicht bekannt ist, ist auch hier der harte Shutdown zu empfehlen. Wird der Screensaver „deaktiviert“ muss dies entsprechend mit der exakten Uhrzeit dokumentiert werden.
4. Ist das System durch ein Passwort geschützt?
   Ist das System durch ein unbekanntes Passwort geschützt, ist in der Regel ein harter Shutdown zu empfehlen. In Ausnahmefällen kann durchaus auch ein Versuch

unternommen werden, dass Passwort mit Hilfe von entsprechenden Werkzeugen zu knacken.

Ist der Zugang zum System hergestellt, kann mit der Sicherung der flüchtigen Daten begonnen werden.

1. Festhalten des Bildschirminhalts und der geöffneten Anwendungen
2. Festhalten der aktuellen Systemzeit und einer Referenzzeit, sowie deren Abweichung
3. Erstellen eines Abbildes des Hauptspeichers
4. Sicherung des Hauptspeichers pro Prozess-ID
5. Sicherung der Cache- und Auslagerungsdateien
6. Liste der aktiven Prozesse
7. Pro Prozess: Umgebungsvariablen, Übergabeparameter, geladene Bibliotheken, Offene Dateideskriptoren, etc.
8. Liste der geöffneten Sockets
9. Liste der Anwendungen, die auf geöffnete Sockets hören
10. Liste der geöffneten Ports
11. Liste der angemeldeten User
12. Status und Statistik der Netzwerkverbindungen
13. Informationen über das verwendete Betriebssystem

Ist die Sicherung der flüchtigen Daten abgeschlossen sollte das System mit einem harten Shutdown heruntergefahren werden.

System ist nicht eingeschaltet
Ist das System ausgeschaltet oder wurde es heruntergefahren wird als erstes das System von der Stromversorgung getrennt. Anschliessend wird das Innen- und Aussenleben dokumentiert und sämtliche Anschlüsse entfernt. Sämtliche Datenträger werden ausgebaut, beziehungsweise entfernt, und beschriftet.

Im Anschluss wird von sämtlichen Datenträgern ein forensisches Duplikat erzeugt. Ein forensisches Duplikat ist eine exakte Kopie (Bitweise 1:1 Kopie) des Quelldatenträgers und sollte immer auf einen Datenträger gesichert werden, der zuvor formatiert wurde. Die Erstellung eines Duplikates ist grundsätzlich immer sinnvoll da dies bessere Analysemöglichkeiten bietet. Ist die Untersuchung ein Teil einer Strafuntersuchung muss in jedem Fall ein Duplikat angefertig werden.

Sind sämtliche Datenträger entfernt, sollte das System gestartet und direkt das BIOS aufgerufen werden. Der entsprechende Key um ins BIOS zu wechseln sollte vorab ermittelt / recherchiert werden. Anschliessend sind sämtliche BIOS-Informationen zu dokumentieren.

Besonders wichtig sind hier die im BIOS eingestellte Systemzeit und das Datum. Besteht eine Diskrepanz zur aktuellen Zeit oder zur notierten Zeit des Betriebssystemes ist dies entsprechend zu notieren und später bei der Analyse zu berücksichtigen.

5.2.3 Analysis (Analyse)

Die Analyse Phase unterteilt sich zum einen in die Phase „Preparation (Vorberietung)“ und zum anderen in die eigentliche Analyse-Phase. Der Übergang zwischen diesen beiden Phasen ist fliessend und lässt sich nicht immer klar trennen. Mit Abschluss der Secure-Phase kann der Ermittler sämtliche weiteren Arbeiten im Labor durchführen. Die Anwesenheit am Tatort ist nicht mehr zwingend, da sämtliches Material eingesammelt und gesichert wurde.

Die anzuwendenden Techniken und durchzuführenden Schritte bei der Analyse sind stark von der Situation abhängig. Bei der Untersuchung einer Cyber-Attacke sind oft andere Aspekte relevant, als bei der Untersuchung einer „normalen“ Straftat. Im nachfolgenden Abschnitt werden einige Ansatzpunkte für eine Untersuchung aufgezeigt.

Preparation (Vorbereitung)

Bevor die Analyse der Daten beginnen kann, muss das forensiche Image gemountet werden. Dafür gibt es verschiedene Vorgehensweisen. In jedem Fall ist jedoch sicherzustellen, dass das Image ReadOnly gemountet wird. Es sollte vor und nach der Analyse jeweils ein Hash des Images erstellt, um zu verifizieren, dass das Image nicht manipuliert wurde.

Nachdem das Image gemountet wurde, wird empfohlen einige weitere Vorbereitungsarbeiten durchzuführen. Zuerst wird versucht so viele Daten wie möglich wiederherzustellen, beziehungsweise sichtbar zu machen, und diese für die anschliessende Analyse zu indexieren.

1. Wiederherstellung des File-Systems / der Beweisspuren (gelöschte, umbenannte, versteckte, verschlüsselte Dateien)
2. Generierung von Hashes für alle Dateien.
3. Abgleich der generierten Hashes.
   Die generierten Hashes können mit Datenbanken abgeglichen werden, welche Hashes von zahlreichen System- / Programmdateien von verschiedenen Betriebssystemen und Programmen enthalten. Die so als irrelevant identifizierten Dateien können für die weitere Analyse ausgeblendet werden.
4. Suchindex über sämtliches Material erstellen
   Es wird empfohlen ein Suchindex aus lesbaren Zeichen zu bilden. Bei der Index-Erstellung ist darauf zu achten, dass auch der File-Slack, alle belegten und unbelegten Bereiche auf dem Datenträger und die Metadaten des Dateisystems indexiert werden.
5. Kategorisierung der Dateien (zum Beispiel nach Typ)

Analysis (Analyse)

Bei der Analyse werden die Rohdaten nach Informationen durchsucht, welche den aktuellen Fall unterstützten. Die gefundenen Informationen werden analysiert und in ein der folgenden drei Gruppen eingeteilt.

- Beweise untermauern eine bestimmte Theorie.
- Beweise widerlegen eine bestimmte Theorie.
- Beweise unterstützen keine bestimmte Theorie.

Die Schwierigkeit bei der Analyse der Informationen besteht darin, diese in einen kausalen und zeitlichen Zusammenhang zu setzen. Diese Zusammenhänge müssen zum einen plausibel und nachvollziehbar sein und zum anderen mit anderen Ereignissen korrelieren. Die Herstellung eines Zusammenhanges ist nicht immer einfach oder offensichtlich. Zum Beispiel kann es notwendig sein, die Grammatik eines Verdächtigen zu analysieren, um eine Verbindung zwischen E-Mails, Dokumenten und Chat-Protokollen herzustellen.

Nachfolgend werden einige der gängigen Analysetechniken aufgelistet. Je nach System sind andere Analyse-Techniken erforderlich. Zum Beispiel kann unter Windows, beziehungsweise NTFS-Dateisystemen, eine Analyse des Alternate Datastreams oder der Windows Registry vorgenommen werden.

- Analyse des File Slacks
- Timeline-Analyse
- Analyse der Auslagerungsdateien
- Analyse der versteckten Dateien
- Analyse unbekannter Binärdateien
- Analyse der Systemprotokolle
- Analyse der Netzwerkschnittstellen
- Analyse der Shell
- Analyse der Druckerjobs und der Druckerqueue
- Analyse der Dateien / Dateiendungen
- Analyse von User Aktivitäten
- Analyse der eingerichteten Jobs
- Bei Verdacht auf Einsatz von Anti-Forensik-Techniken: Vertiefte Analyse
- Weitere Untersuchungen auf Anwendungsebene (zum Beispiel: E-Mail, Browser)
- ...

Unterstützende Fragestellungen Für die Analyse können folgende Fragestellungen hilfreich sein:

- War ein physischer Zugang zum System notwendig?
  Ja: Kontrolle der physischen Überwachung und Zutrittskontrolle.
- Hatten andere Personen Zugang zum System?
  Ja: Kontrolle der physischen Überwachung und Zutrittskontrolle.
- Was für Computerkenntnisse hat der Verdächtige? Was für Computerkenntnisse waren notwendig?
- Was hat der Angreifer für Tools hinterlassen?
- Wie wurden Tools und Befehle aufterufen? (Manuell / von Hand, via Copy & Paste„ via Script)

5.2.4 Reporting (Dokumentation)

Die gesamte forensische Untersuchung muss im Detail protokolliert und dokumentiert werden. Es müssen sämtliche Arbeitsschritte nachvollzogen und gegebenenfalls durch einen anderen Experten reproduziert werden können. Die eingesetzten Tools (inkl. Version) und Techniken sollten kurz beschrieben werden Im Rahmen einer Strafuntersuchung müssen die durchgeführten Schritte und angewandten Techniken so erläutert werden, dass diese von Laien verstanden und nachvollzogen werden können.

Die Dokumentation der Untersuchung sollte soweit als möglich und praktikabel sofort bei der Durchführung erstellt werden, da ansonsten wichtige Informationen, Gedanken und Arbeitsschritte verloren gehen. Sie sollte klar, einfach und auf Fakten aufbauend geschrieben und mit Visualisierungen, Screenshots oder gegebenenfalls Fotos, welche mit der Digitalkamera aufgenommen wurden, unterlegt werden.

- Verwendete Tools (inkl. Versionsnummer)
- Verwendete Hardware (zum Beispiel FastBloc Write Blocker)
- Angewendete Techniken
- Prüfsummen von Dokumenten, Protokollen und Beweisen
- Erläuterung der Evaluation der Tools und Techniken

Nachfolgend werden kurz einige verschiedene Typen von Berichten vorgestellt und beschrieben:

- Interne Berichte Der interne Bericht ist der häufigste, der erstellt werden muss. Er ist nicht formal, hat aber trotzdem Einfluss und Auswirkungen und sollte vor der Herausgabe unbedingt einem Review unterzogen werden. Der Bericht wird allenfalls als Grundlage für eine Erklärung, eine Eidesstattliche Erklärung oder einen Experten-Bericht verwendet. Der Bericht setzt sich aus folgenden Teilen zusammen:

Zusammenfassung / Abstract, Resultate und Erkentnisse. Es sollte aufgezeigt werden, was gemacht wurde und was herauskam und was für weitere Schritte beauftragt wurden. Auch sind sämtliche Beweise aufzulisten und aufzuzeigen wieso diese relevant sind.

- Erklärungen Nachdem ein interner Bericht erstellt wurde und weitere juristische Schritte unternommen werden, wird meistens als erstes eine Erklärung (oder auch Deklaration) erstellt. Diese Erklärung muss von Antwälten, Richtern und Beratern verstanden werden und auch für Leute ohne technischen Hintergrund nachvollziehbar sein. Der Bericht setzt sich aus folgenden Teilen zusammen: Erklärung, Hintergrundinformationen zum Ermittler (Qualifikation, etc.), Hauptteil mit Informationen, Meinungen und Schlussvolgerungen und einem Fazit am Ende. Für den Bericht dürfen nur Beweise aus erster Hand verwendet werden.
- Eidesstattliche Erklärungen Eine Eidesstattliche Erklärung entspricht einer Erklärung, welche notariell Beglaubigt wurde.
- Experten-Berichte Bei Experten Berichten wird sehr hoher Wert auf Formalität gelegt und entspricht praktisch einer Dissertation über den vorliegenden Fall. Der Experte darf auf Basis der Beweise und der Untersuchung seine Meinungen und Schlussfolgerungen äussern. Es ist jedoch darauf zu achten, keine Spekulationen oder Annahmen zu treffen. Der Aufbau des Berichtes ist stark vom Experten abhängig, sollte aber mindestens volgende Elemente aufweisen: Titelseite, Überblick, Qualifikationen des Experten, Was wurde gemacht?, Analyse, Schlussvolgerung und Fazit.

5.2.5 Present (Präsentation)

Bei der Präsentation werden die Ergebnisse der Untersuchung aufgezeigt und erläutert. Das Zielpublikum kann entweder zum Beispiel der Ermittlungsleiter, der Auftraggeber, das Management der Unternehmung oder ein Gericht sein. Hierbei ist zu beachten, dass auch Laien den Ausführungen folgen können und diese verstehen (analog der Dokumentation).

5.2.6 Review (Rückblick)

In der Review-Phase wird der Ablauf der vergangene Ermittlung betrachtet und analysiert, was gut gelaufen ist, beziehungsweise wo es noch Verbesserungspotenzial gibt. Ziel dieser Phase ist es, dass der eingesetzte Prozess für die forensische Analyse kontinuierlich verbessert und den neusten Erkenntnissen angepasst wird. Gegebenenfalls sind Massnahmen zu treffen, um langfristige Optimierungen und Verbesserungen zu erreichen. Mit dem Aufkommen von neuen Technologien und Trends (zum Beispiel Geräte aus dem Bereich Internet of Things) kann es notwendig sein, entsprechende Analyse-Werkzeuge zu erstellen oder zu evaluieren und im Labor erste Erfahrungen zu sammeln.

5.3 Beweiskette und Beweissicherung

Die korrekte Beweissicherung und die Einhaltung der Beweiskette sind essentiell für eine spätere Verwertung der Beweise bei einer Strafuntersuchung. Wird ein Beweisstück gefunden muss dieses korrekt sichergestellt werden. Jeder Beweis muss beschriftet werden und anschliessend ein Beweiszettel ausgefüllt werden. Ein Beispiel, beziehungsweise eine Vorlage, für einen Beweiszettel ist im Anhang im kapitel D Vorlage: Beweiszettel zu finden. Dadurch wird sichergestellt, dass kein zweifel an der Herkunft und dem Besitztum des Beweises entsteht. Entweder im Beweiszettel oder separat wird ein Protokoll geführt, wer, wann, was mit dem Beweis gemacht hat. Dadurch kann der Weg des Beweises lückenlos zurückverfolgt werden.

Handelt es sich beim Beweis um ein elektronisches Gerät oder Bauteil, ist dieses in einem antistatischen Sack aufzubewahren. Je nach Situation wird bei Datenträgern das Original nicht mitgenommen, da dieses für den Weiterbetrieb benötigt wird. In diesen Fällen ist dies entsprechend zu dokumentieren und sicherzustellen, dass ein entsprechendes forensisches Duplikat erstellt wurde. Bei Datengrägern ist auf dem Beweiszettel zusätzlich der Hash des Datenträgers zu vermerken, sodass die Unversehrtheit jederzeit geprüft werden kann.

5.4 Hinweise zur forensischen Analyse

Bei einer forensischen Analyse sind folgende wichtige Aspekte zu berücksichtigen.

- **Zeuge / Zweitperson**
  Während der Untersuchung sollte eine Zweitperson, bzw. ein Zeuge anwesend sein.
- **Protokollierung**
  Sämtliche durchgeführten Arbeitsschritte müssen protokolliert werden. Am Ende der Untersuchung sollte das Protokoll durch den Zeugen, die Zweitperson abgenommen und von beiden unterschrieben werden.
- **Schutz der eigenen Umgebung**
  Die eigene Analyseumgebung sollte gut gegen Angriffe geschützt sein und nicht direkt mit dem angegriffenen System oder Netzwerk verbunden werden. Sollte sich der Angreifer noch im Netzwerk oder auf dem System befinden, könnte er das Analysesystem angreifen und weiteren Schaden anrichten.
- **Schutz der Beweismittel**
  Sämtliche Beweismittel müssen sichergestellt und anschliessend geschützt werden. Eine Veränderung der Daten nach der Sicherung darf nicht mehr möglich sein beziehungsweise muss zweifelsfrei festgestellt werden können.
- **Verwendung von Systembefehlen**
  Zur Sammlung und Sicherung von Daten sollten niemals Systembefehle verwendet werden. Die Systemprogramme könnten vom Angreifer durch modifizierte Programme ausgetauscht werden sein. Der Ermittler sollte immer statisch vorkompilierte Programme verwenden.

- **Einsatz Grafischer Programme**
  Bei der Untersuchung eines Live-Systems sollte soweit als möglich auf den EInsatz von Programmen mit einer grafischen Oberfläche verzichtet werden. Grund dafür ist, dass diese eine Vielzahl an Binärdateien und Konfigurationen benötigen. Zum einen werden dadurch viele Zeitstempel geändert und zum anderen benötigen diese mehr RAM als Konsolenanwendungen.
- **Patches und Updates**
  Eher nicht, wenn kritisch, nach Rücksprache, Vernichtung Beweise
- **Remote Untersuchung**
  Bei einer forensischen Analyse ist nicht immer ein direkter Zugriff auf das System vorhanden. Gewisse Forensik-Tools erlauben den Einsatz via Netzwerk. Diese benötigt jedoch auf dem zu untersuchenden System einen entsprechenden Forensik-Client, mit welchem sich der Server anschliessend verbinden kann. Dieser sollte bereits vorgängig (vor dem Sicherheitsvorfall) auf dem System installiert worden sein.

KAPITEL 6

Tools und Techniken

In diesem Kapitel werden die grundlegenden Tools und Techniken einer forensischen Analyse vorgestellt. Die Tools und Techniken werden dabei nach der Phase, in der diese eingesetzt und angewendet werden, gegliedert.

6.1 Readiness

In diesem Kapitel werden einige Techniken und Tools aufgezeigt, welche während der Phase „Readiness“ von Bedeutung sind.

6.1.1 Datenträger löschen

Digitale Beweise und Kopien von Images sollten immer auf leere, beziehungsweise zuvor komplett gelöschte Datenträger gesichert werden. Wird dies nicht gemacht, kann es sein, dass Rückstände / Daten von vorherigen Untersuchungen die Analyse beeinflussen.

dd

Unter Linux kann für das Löschen, genauer gesagt das Überschreiben mit Zufallswerten (1) oder Nullen (2) das Tool *dd* verwendet werden. Mit dem Tool *dd* können sowohl ganze Datenträger, als auch einzelne Partitionen überschrieben werden.

```
1 > dd if=/dev/urandom of=/dev/<DriveToWipe>
2 > dd if=/dev/zero of=/dev/<DriveToWipe>
```

Weitere Tools

Die nachfolgenden Tools verfügen ebenfalls über entsprechende Funktionalitäten:

- EnCase
- FTK
- X-Ways

6.2 Secure

In diesem Kapitel werden einige Techniken und Tools aufgezeigt, welche während der Phase „Secure“ von Bedeutung sind.

6.2.1 Auslesen der Zeitkonfiguration

Die Zeitkonfiguration des Systemes wird je nach Betriebssystem und zum Teil sogar je nach Distribution an einem anderen Ort gespeichert.

Unter Ubuntu, beziehungsweise Debian-Systemen ist diese unter */etc/timezone* zu finden. Bei Red-Hat-Distributionen unter */etc/sysconfig/clock*. Beim Auslesen der Zeitkonfiguration ist auch zu notieren, ob das System automatisch zwischen Sommer- und Winterzeit umstellt.

```
1 > cat /etc/timezone
2 Europe/Zurich
```

Das Datum und die Uhrzeit kann über nachfolgenden Befehl ausgelesen werden. Zu beachten ist hier, dass eine statisch vorkompilierte Version des Befehsl verwendet wird.

```
1 > date
2 Fre Mai 29 08:00:00 CEST 2015
```

6.2.2 Bestimmung der Linux-Distribution

Jede Linux-Distribution hat ihre Eigenheiten. Dazu zählen unter anderem Befehle, Speicherorte, Methoden für Tracking und Auditierung der User Aktivität und das Logging von System Events.

Unter vielen Distributionen befindet sich unterhalb des Ordners */etc* ein File, welches die Versions-Informationen enthält.

```
> ls /etc | grep -E "`release|version"'
debian_version
lsb-release
lsb-release.dpkg-dist
os-release
upstream-release

> cat /etc/debian_version
jessie/sid

> cat /etc/os-release
NAME="Ubuntu"
VERSION="14.04.2 LTS, Trusty Tahr"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 14.04.2 LTS"
VERSION_ID="14.04"
HOME_URL="http://www.ubuntu.com/"
SUPPORT_URL="http://help.ubuntu.com/"
BUG_REPORT_URL="http://bugs.launchpad.net/ubuntu/"
```

Ist kein entsprechendes File vorhanden kann das Logon-Banner oder die Log-Einträge unter */var/log/dmesg* oder */var/log/messages* allenfalls Hinweise zur Distribution enthalten:

```
> cat /etc/issue
Linux Mint 17 Qiana \n \l

> cat /var/log/dmesg
[    0.000000] Initializing cgroup subsys cpuset
[    0.000000] Initializing cgroup subsys cpu
[    0.000000] Initializing cgroup subsys cpuacct
[    0.000000] Linux version 3.13.0-49-generic (buildd@akateko) (gcc version
4.8.2 (Ubuntu 4.8.2-19ubuntu1) ) #83-Ubuntu SMP Fri Apr 10 08:00:00 UTC 2015 (
Ubuntu 3.13.0-49.83-generic 3.13.11-ckt17)
[    0.000000] Command line: BOOT_IMAGE=/boot/vmlinuz-3.13.0-49-generic root=UUID
=ee905b77-1264-477b-9942-23bd448df95c ro quiet splash acpi_backlight=vendor
acpi_osi=Linux vt.handoff=7
[    0.000000] KERNEL supported cpus:
[    0.000000]   Intel GenuineIntel
[    0.000000]   AMD AuthenticAMD
[    0.000000]   Centaur CentaurHauls
[    0.000000] e820: BIOS-provided physical RAM map:
```

Oft gibt auch der verwendete Package-Manager ein Hinweis auf die verwendete Distribution:

- Ubuntu Linux: APT, Synaptic
- Red Hat / Fedora Linux: RPM

- Gentoo Linux: Portage / Emerge
- SUSE Linux: YaST
- Debian Linux: APT

6.2.3 Shutdown eines Systemes

Ein laufendes System kann grundsätzlich auf zwei verschiedene Arten heruntergefahren werden. Zum einen kann ein normaler, regulärer Shutdown des Systemes durchgeführt werden. Dabei werden sämtliche Datei gespeichert und temporäre Daten gelöscht / aufgeräumt. Das System befindet sich anschliessend in einem sauberen / lauffähigen Zustand. Während dem Shutdown werden jedoch bei zahlreichen Dateien die Zeitstempel verändert und sämtliche Daten welche sich im Arbeitsspeicher befinden gehen verloren. Dies kann unter Umständen die Analysearbeiten erschweren oder im schlimmsten Fall wichtige Beweise vernichten.

Die andere Methode ein System herunterzufahren ist der harte Shutdown. Beim harten Shutdown wird das System von der Stromversorgung getrennt, ohne das dieses vorher heruntergefahren wurde. Mit diesem Vorgehen wird sichergestellt, dass die Zeitstempel unverändert bleiben und die temporären Daten erhalten bleiben. Auch ist die Wahrscheinlichkeit da, dass auf der Festplatte eine Auslagerungsdatei vorhanden ist, welche analysiert werden kann. Die Extraktion und Analyse dieser Daten ist jedoch sehr aufwändig. Diese Methode des Shutdowns kann bei gewissen Dateisystemen zu irreparablen Schäden führen. Daher ist vorgängig abzuwägen, ob ein harter Shutdown sinnvoll und verkraftbar ist.

Bei beiden Varianten ist die Zeit der Durchführung und die Art des Shutdowns zu protokollieren.

Die Möglichkeiten den Inhalt des Arbeitsspeichers zu sichern, werden im nachfolgenden Kapitel vorgestellt.

6.2.4 Erstellen von Hashes

Für die Sicherung von Daten ist die Erstellung von Hashes von essentieller Bedeutung. Jede Datei, Partition oder ähnlich sollte vor und nach der Sicherung gehasht werden, um sicherzustellen, dass diese durch den Sicherungsvorgang nicht verändert wurde. Auch Protokolle, Dokumentation, Ergebnisse der Analyse, etc. sollten immer mit einem Hash versehen und geprüft werden.

md5sum

Mit folgendem Befehl können MD5-Hashes von einzelnen Dateien erstellt werden. Es stehen unter anderm MD2, MD4, MD5, SHA und SHA1 als Algorithmen zur Verfügung. Der Befehl entspricht jeweils dem Algorithmus. Zum Beispiel für SHA1 wird der Befehl *sha1sum* verwendet.

```
1 > md5sum <FileName>
```

md5deep

Mit dem Tool „md5deppe“ können Hashes von Ordnerinhalten rekursiv erstellt werden. Es können dabei Hashes mit dem MD5, dem SHA1, dem SHA256, dem Tiger und Whirlpool Algorithmus erstellt werden.

```
1 > md5depp -r <PathToDirectory>
```

6.2.5 Sicherung des Arbeitsspeicher-Inhaltes

Bei der Sicherung von Arbeitsspeicher-Inhalten ist grundsätzlich zu beachten, dass die verwendeten Tools und Techniken einen kleinen Memory-Footprint aufweisen. Je mehr Arbeitsspeicher ein Tool verwendet, desto mehr wichtige Daten und somit auch Beweise gehen verloren. Ein weiterer wichtiger Punkt der beachtet werden soll ist, dass die Sicherung nicht im User-Mode, sondern im Kernel-Mode erfolgt. So ist sichergestellt, dass der gesamte Inhalte des Arbeitsspeichers gesichert werden kann. Andernfalls können aufgrund von Berechtigungseinschränkungen nicht alle Daten gesichert werden.

6.2.6 Sicherung des Arbeitsspeicher-Inhaltes ohne Zugriff auf das Betriebssystem

Grundsätzlich gilt, dass der Inhalt des Arbeitsspeichers beim herunterfahren des Systems verloren geht. Besteht kein Zugriff auf das Betriebssystem und es besteht keine Möglichkeit das Passwort zu erhalten, muss gegbenenfalles eine der nachfolgend beschriebenen Techniken eingesetzt werden. Studien haben gezeigt, dass die Inhalte des Arbeitsspeichers noch bis zu mehreren Minuten nach dem Verlust der Stromversorgung im Arbeitsspeicher vorhanden sind. Bei einigen BIOS-Varianten wird der Arbeitsspeicher beim Herunterfahren des Systemes vollständig gelöscht. Bei diesen Systemen ist ein Auslesen nach dem Herunterfahren nicht mehr möglich.

- Cold Boot
  Nach dem harten Shutdown des Systemes wird ein speziell präpariertes USB-Boot-Medium eingesetzt und das System sofort neu gestartet. Dabei ist im BIOS die Bootreihenfolge so zu ändern, dass das USB-Medium als primäres Boot-Medium verwendet wird. Das USB-Medium wurde vorgängig mit einem Minibetriebssystem, einem Spezialtool und einer separaten Partition zur Speicherung des Arbeitsspeicher-Abbildes. Als Tool zur Sicherung kann entweder *Msramdmp* oder der *USB Memory Scraper* verwendet werden.
- Cold Boot mit Kühlung des physikalischen Bausteins
  Bevor das System mit einem harten Shutdown heruntergefahren wird, wird das Gehäuse des Systems geöffnet und der Arbeitsspeicher mit einem Stickstoffspray auf -50 Grad Celsius herutner gekühlt. Nach dem harten Shutdown wird der Arbeitsspeicher ausgebaut und in einem Analyse-System eingebaut. Das Analyse-System wird mit einer Spezialsoftware gestartet, welche den gesamten Arbeitsspeicher auf einen anderen Datenträger sichert.

- Firewire-Attacke
  Bei der Firewire-Attacke wird eine Schwachstelle der Firewire-Schnittstelle ausgenutzt, um Inhalte des Arbeitsspeichers zu sichern. Die Schwachstelle kann jedoch nur unter bestimmten Bedingungen ausgenutzt werden und es kann nur eine begrenzte Menge (aktuell max. 4 GB) an Arbeitsspeicher ausgelesen werden. Als unterstützendes Tool zur Realisierung dieses Angriffes sei an dieser Stelle das Tool *PythonRaw1394* genannt. Weitere Informationen zur Firewire-Attacke sind hier zu finden: https://blogs.gnome.org/muelli/2010/04/reading-ram-using-firewire/, http://link.springer.com/chapter/10.1007%2F978-3-642-23602-0_14#page-1

6.2.7 Sicherung des Arbeitsspeicher-Inhaltes mit Zugriff auf das Betriebssystem

Für die Sicherung gibt es einige verschiedene Tools. Unter früheren Linux-Kerneles konnte direkt auf das Arbeitsspeicher-Device zugegriffen werden. In neuer Versionen wurde dies aus Sicherheitsgründen unterbunden.

Linux (alte Kernel-Versionen)

```
1 > dd if=/dev/mem of=ram.dd
```

fmem

Nachdem *fmem* (vorgängig) kompiliert wurde, muss das Kernel-Modul zuerst geladen werden. Dazu wird am besten das mitgelieferte Shell-Script *run.sh* verwendet werden. Nach dem das Kernel-Modul geladen wurde, steht unter */dev/fmem* ein neues Device zur Verfügung. Dieses Device kann mit den im Kapitel 6.2.8 Forensische Duplikation beschriebenen Tools gesichert werden.

lime

Das Tool *lime* ist ein weiteres Kernel-Modul, welches für die Akquisition von Arbeitsspeicher-Inhalten unter Linux und Linux-basierten Systemen verwendet werden kann.

```
1 > insmod <PathToLime>/lime.ko "path=<PathToDumpFile> format=<raw|padded|lime>"
```

Weitere Tools

Die nachfolgenden Tools verfügen ebenfalls über entsprechende Funktionalitäten:

- Win32dd und Win64dd
- DumpIt
- Winen und Winen64
- Mdd
- FTK Imager

6.2.8 Forensische Duplikation

Bei einer forensischen Duplikation sollte der Quelldatenträger immer Bit-Weise kopiert werden. Zusätzlich ist sicherzustellen, dass der Quelldatenträger Read-Only gemountet oder über einen Writeblocker angeschlossen ist. Ein Writeblocker verhindert physisch, dass Daten auf einen Datenträger geschrieben werden können. Es ist zu empfehlen grundsätzlich immer einen Writeblocker auf Hardware-Basis zu verwenden.

Die Sicherung kann auf folgende Arten durchgeführt werden:

- Auf einem Live-System
  Die Duplikation des Datenträgers erfolgt unter Einsatz eines Software-Writeblockers auf dem Live-System
- Auf dem zu untersuchenden System
  Die Duplikation erfolgt über eine Live-CD, nachdem das System heruntergefahren wurde. Für die Duplikation wird ein Writeblocker auf Software- oder Hardware-Basis eingesetzt.
- Auf dem Analyse-System
  Die Duplikation erfolgt auf dem Analyse-System. Für die Duplikation wird ein Writeblocker auf Software- oder Hardware-Basis eingesetzt.
- Hardware-Imager
  Die Duplikation erfolgt durch ein auf Imaging spezialisierte externe Hardware.

Das Image des Quelldatenträgers kann entweder auf einem separaten, sterilen Datenträger oder über ein Netzwerk auf einen Server gesichert werden. Bei der Duplikation ist sicherzustellen, dass auch versteckte Bereiche des Datenträgers, wie Host Protected Areas, Device configuration Overlays gesichert werden.

Einsatz von Standard-Linux Tools

1. Mounten des Zieldatenträgers

```
> mount -t <Dateisystem> <PathToDisk> <MountPoint>
```

2. Sicherung der Partitionstabelle

```
> fdisk -l <PathToDisk> > <MountPoint>/<FileName.fdisk>

> parted -l <PathToDisk> > <MountPoint>/<FileName.fdisk>
```

3. Generierung eines Hashes
   Vor der Duplikation wird ein Hash des zu sichernden Datenträgers / Partition gemäss Kapitel 6.2.4 Erstellen von Hashes erstellt und dokumentiert.

4. Duplikation

```
> dd conv=noerror bs=512k if=<PathToDisk> of=<MountPoint>/<FileName.dd>
```

5. Generierung eines Hashes
   Nach der Duplikation wird ein Hash des zu sichernden Datenträgers / Partition gemäss Kapitel 6.2.4 Erstellen von Hashes erstellt und mit dem dokumentierten Hash verglichen. Dies stellt sicher, dass der Datenträger während der Duplikation nicht unbemerkt verändert wurde. Anschliessend ist vom erstellten Image ebenfalls ein Hash zu erstellten, welcher mit dem dokumentierten verglichen werden muss. Ist dieser mit dem dokumentierten Hash identisch, wurde der Datenträger / die Partition korrekt dupliziert.

   Im Anschluss muss das Verfahren für jede verfügbare Partition wiederholt werden.

6. Transfer des Images via Netzwerk

```
1 > nc -l -p 8000 |dd of=<MountPoint>/<PathToImage>
```

7. Transfer des Images via Netzwerk (Verschlüsselt)

```
1 > cryptcat -k <Pssword> -l -p 8000 |dd of=<MountPoint>/<PathToImage>
```

dcfldd

Das Tool *dcfldd* erweitert das Tool *dd* um einige nützliche Funktionen. So können automatisch Hashes für den zu sichernden Datenträger oder die zu sichernde Partition generiert.

```
1 > dcfldd if=<PathToDisk> of=<MountPoint>/<FileName.dd> hash=sha256 sha256log=<
FileName.sha256> bs=1MB count=1000
```

dc3dd

Das Tool *dc3dd* ist ein Erweiterung, beziehungsweise eine gepatchte Version des Tools *dd*.

```
1 > dc3dd if=<PathToDisk> of=<MountPoint>/<FileName.dd> hash=md5 progress=on
```

Weitere Tools

Die nachfolgenden Tools verfügen ebenfalls über entsprechende Funktionalitäten:

- Adepto
- Guymager
- ewfacquire
- Raptor Toolbox

6.2.9 Verifikation eines forensischen Duplikates oder eines Beweisstückes

Vor und nach der Verwendung eines forensischen Duplikates oder eines Beweisstückes sollte jeweils die Integrität verifiziert werden. Dazu wird ein neuer Hash erzeugt (Siehe Kapitel 6.2.4 Erstellen von Hashes). Dies kann entweder manuell oder durch den Einsatz eines Tools oder einer Tool-Suite sichergestellt werden.

md5sum und md5deep

Mit diesen beiden Tools wird manuell ein neuer Hash generiert, welcher mit dem dokumentierten Hash des Duplikates oder des Beweisstückes verglichen wird.

dcfldd

Mit *tcfldd* kann geprüft werden, ob der Hash eines Quelldatenträgers und eines bereits erstellten Images identisch sind.

```
1 > dcfldd if=<PathToSourceDevice> vf=<PathToImage.dd> verifylog=<PathToLog.log>
```

Weitere Tools

Die nachfolgenden Tools verfügen ebenfalls über entsprechende Funktionalitäten:

- EnCase
- FreeHelix
- FTKImager

6.2.10 Sicherung der Binärdatei von ausgeführten Prozessen

Unter Umständen wurde auf dem System ein Prozess gestartet und die dazugehörige Binärdatei anschliessend gelöscht. Der Prozess ist dann solange aktiv, bis das System neu gestartet oder heruntergefahren wird. Solange das System noch läuft kann eine Kopie der Binärdatei um Proc-Dateisystem gefunden werden.

Linux

```
1 > cat /proc/<PDI>/exe > /<OutputPath>
```

6.2.11 Sicherung flüchtiger Daten

Die Sicherung von flüchtigen Daten muss effizient, schnell und korrekt erfolgen. Nachfolgend werden die wichtigsten zu sichernden Informationen mit den dazugehörigen Befehlen aufgelistet.

Linux

- Informationen über den Prozessor

```
> cat /proc/cpuinfo
```

- Grösse und Anzahl der eingebundenen Partitionen und deren Füllungsgrad

```
> df -h
```

- Informationen über den physischen Datengräger und die Partitionierung

```
> fdisk -lu <PathToHarddisk>
> mmls <PathToHarddisk> (The Sleuth Kit)
> mmls -o <StartSector> <PathToBSDHarddiskSlice> (The Sleuth Kit, FreeBSD)
```

- Information über den aktiven Kernel, Compiler-Version und Kommpilierdatum, etc.

```
> cat /proc/version
```

- Anzeige der aktiven Boot-Parameter

```
> cat /proc/cmdline
```

- Anzeige der Shell-Umgebungsvariablen

```
> env
```

- Anzeige der angemeldeten User

```
> who
```

- Liste der alufenden Prozesse

```
> ps -efl
```

- Informationen über die konfigurierten Netzwerk-Interfaces und deren Statistik

```
> ifconfig -a, ifconfig -s
```

- Anzeige der Einträge der Arp-Table

```
> arp -n
```

- Anzeige des Inhalts des Host-File

```
> cat /etc/hosts
```

- Anzeige der DNS-Konfiguration

```
> cat /etc/resolv.conf
```

- Inhalt der Passwortdatei

```
> cat /etc/passwd
```

- Inhalt der Shadow-Datei

```
> cat /etc/shadow
```

- Anzeige der aktiven Netzwerkverbindungen

```
> netstat -anp
```

- Anzeige der aktiven Netzwerkverbindungen und der dazugehörigen Programmej

```
> netstat -tunp
```

- Anzeige der Routing-Tabelle

```
> netstat -rn
> route
```

- Anzeige der geöffneten und aktiven Ports

```
> lsof -P -i -n
```

- Komplette Ausgabe aller durch Prozesse geöffneten Dateien

```
> lsof
```

- Informationen über den Hauptspeicher

```
> cat /proc/meminfo
```

- Information über aktive Module

```
> cat /proc/modules
```

- Informationen über die gemounteten Dateisysteme

```
> cat /proc/mounts
```

- Informationen über die Swap-Konfiguration

```
> cat /proc/swap
```

- Konfiguration der Mountpoints

```
> cat /etc/fastab
```

- Geöffnete Dateisysteme

```
> mount -v
```

- Installierte Software (via Packet-Manager installiert)

```
> dpkg -l
> rpm -qa
```

- Installierte Software (manuell installiert, Eintrag im *$PATH*)

```
> echo $PATH
> whereis <NameOfProgram>
> which <NameOfProgram>
> locate <NameOfProgram>
```

- Pro Prozess: Umgebungsvariablen, verwendete Speicherbereiche, etc.

```
> ls /proc | sort -n | grep -v [a-z,A-Z] | while read PID
  do
    echo "`Prozess ID $PID:"`
    cat /proc/$PID/cmdline
    cat /proc/$PID/environ
    cat /proc/$PID/maps
    cat /proc/$PID/stat
    cat /proc/$PID/statm
    cat /proc/$PID/status
    cat /proc/$PID/mem
    ls -ld /proc/$PID/root
    ls -ld /proc/$PID/cwd
    ls -ld /proc/$PID/exe
    ls -lrta /proc/$PID/fd
    echo "`--------------------"'
  done > <OutputPath> 2>&1
```

6.3 Analysis

In diesem Kapitel werden einige Techniken und Tools aufgezeigt, welche während der Phase „Analysis“ von Bedeutung sind.

6.3.1 Image mounten

Ein forensisches Duplikat, beziehungsweise ein Image, sollte immer Read-Only gemountet werden. Um sicherzustellen, dass das Image nicht verändert wird, sollte vor und nach dem Mounten jeweils ein Hash des Images erzeugt werden.

6.3.2 Linux (dd)

```
> mount -r -o loop <PathToImage> <MountPoint>
```

6.3.3 ewfmount

```
> ewfmount /<PathToImage>.E01 <MountPoint>
```

6.3.4 xmount

```
> xmount -in <InFormat> --out <OutFormat> <PathToImage> <
MountPointForConvertedImage>
> mount -o ro <MountPointForConvertedImage> <MountPoint>
```

6.3.5 Gelöschte Datenträger

Ist ein gesicherter Datenträger leer, beziehungsweise hat den Anschein, als wäre dieser leer, könnte eine Löschsoftware zur Verwischung von Spuren eingesetzt worden sein. Ist dies der Fall sollte zuerst versucht werden herauszufinden, welche Löschsoftware eingesetzt wurde. Allenfalls wurden dazu Hinweise oder Spuren auf anderen Datenträgern hinterlassen. Gewisse Produkte arbeiten beim Löschen nicht mit reinen Zufallswerten, sondern verwenden Löschpattern. Dies kann einen weiteren Hinweis auf die verwendete Löschsoftware liefern. Oft arbeiten auch die Produkte nicht zuverlässig, und es bleiben gewisse temporäre Dateien, Registry-Einträge oder Protokolle übrig, welche für eine Analyse herangezogen werden können. Wurde der Datenträger mehrfach mit echten zufälligen Bitmustern oder Nullen überschrieben, besteht in der Regel praktisch keine Möglichkeit zur Datenwiederherstellung.

6.3.6 Gelöschte Partitionstabelle

Hat ein Datenträger den Anschein gelöscht worden zu sein, sollte diese in jedem Fall trotzdem untersucht werden. Befinden sich auf dem Datenträger noch Daten kann ein Versuch unternommen werden, das Dateisystem zu rekonstruieren. Im Falle eines ext-Dateisystems muss der Datenträger nach einer bestimmten Signatur (0xef53) durchsucht werden. Diese Signatur kennzeichnet den Superblock des Dateisystems. Ein Backup dieses Superblocks wird in der Regel redundant an mehreren Stellen des Dateisystems abgelegt. Der Superblock ermöglicht die Rekonstruktion des ganzen oder eines grossen Teiles des Dateisystems.

Tools

Die nachfolgenden Tools bieten gewisse Funktionalitäten, um gelöschte Partitionstabellen zu rekonstruieren.

- SMART
- findsuper
- PartitionMagic

6.3.7 Analyse von gelöschten Dateien

Wurden Dateien vom Dateien vom System gelöscht besteht ein gewisse Wahrscheinlichkeit, dass diese mit einem entsprechenden Tool wiederhergestellt werden können.

Sleuth Kit

Die Wiederherstellung von gelöschten Daten mit Hilfe der Tools aus dem Sleuth Kit erfolgt nach folgendem Ablauf:

1. Identifizierung von gelöschten Daten

```
> fls -rd <MountPoint>/<PathToImage.dd>
```

2. Wiederherstellung ausgewählter Dateien

```
> istat <MountPoint>/<PathToImage.dd> <Inode>
> icat <MountPoint>/<PathToImage.dd> <Inode> > <OutputPath>
```

Weitere Tools

Die nachfolgenden Tools verfügen ebenfalls über entsprechende Funktionalitäten:

- TASK
- SMART (für Ext2 und Ext3)

6.3.8 Analyse von versteckten Dateien

Auf einer Festplatte können an einigen Stellen Daten versteckt werden. Aufgrund des Aufbaus und der Aufteilung einer Festplatte in Sektoren fester Grösse über die gesamte Fläche, können am Rand Abstände zwischen den Sektoren entstehen. In diesen sogenannten Gaps können Daten versteckt werden. Eine weitere Möglichkeit Daten zu verstekcen sind die Partition-Gaps (Abstände zwischen Partitionen) und unpartitionierte Bereiche der Festplatte. Eine etwas andere Methode ist die Verwendung von Blöcken, welche vom Betriebssystem als unbrauchbar markiert wurden (Bad Blocks). Die einfachste Art um Dateien zu verstecken, ist die Verwendung des Hidden-Attributes des Betriebssystems. Dieses existiert in der einen oder anderen Form in den meisten gängigen Betriebssystemen.

Auch mit Hilfe von Rootkits lassen sich Dateien und Verzeichnisse verstecken. Darüber hinaus können unter anderem auch Prozesse und Netzwerkverbindungen verborgen werden. Normale Rootkits verwenden dafür trojanisierte (manipulierte) Systemprogramme, welche die entsprechenden Dateien, Verzeichnise, etc. herausfiltern. kernel-Level-Rootkits verwenden keine trojanisierten Systemprogramme, sondern manipulieren die Syscall-Tabelle. Einige Kernel-Level-Rootkits verändern sogar direkt den Syscall-Code. Diese Art von Rootkits sind auf dem System sehr schwer aufzuspüren. Um diese zu finden, müssen die Strukturen analysiert, das System beobachtet und mit einem Referenz-System verglichen werden.

6.3.9 Dateien oder Fragemente wiederherstellen

Wurden Dateien auf einem System gelöscht, bleiben möglicherweise gewisse Dateifragmente über längere Zeit erhalten. Können die einzelnen Fragmente im File-Slack oder unallozierten Bereich gefunden werden, können diese zusammengesetzt werden. Der Prozess der Suche und Wiederherstellung von Dateien oder Dateifragementen wird auch als File Carving bezeichnet.

foremost

Foremost ist ein File-Carving-Tool welches speziell für Unix entwickelt wurde. Die Widerherstellung erfolgt aufgrund der Analyse von Header- und Footer-Informationen.

```
> foremost -v -c <PathToForemostConfig> <PathToImage.dd>
```

Fatback

File-Carving-Tool um FAT-Dateisysteme unter Unix zu untersuchen.

```
> fatback <PathToImage.dd>
```

unrm und lazarus

Mit den Tools unrm und lazarus können Dateien aus dem unallozierten Bereich der Festplatte wiederhergestellt werden. Bei diesem Vorgang fällt eine sehr grosse Menge an Rohdaten an, es sollte daher entsprechend Speicherplatz zur Verfügung stehen.

```
> unrm <PathToDisk> > <OutputPath>
> lazarus -h <OutputPath>
```

Weitere Tools

Die nachfolgenden Tools verfügen ebenfalls über entsprechende Funktionalitäten:

- Scalpel
- Foregone

6.3.10 Entpacken von Dateien

Dateien können auf einem System in Archiven verpackt werden. Wird bei einer Analyse ein File gesucht, welches sich in einem Archiv befindet, kann dies nicht gefunden werden. Daher sollte man in Erwägung ziehen, die Archive zu Beginn einer Analyse zu entpacken. Dabei können jedoch sehr grosse, zusätzliche Datenmengen anfallen.

Linux

```
> find <PathToImage> -iname "*.tar.gz" -exec tar xvzf {} -C <OutputPath> \;
> find <PathToImage> -iname "*.zip" -exec unzip {} -d <OutputPath> \;
```

7-Zip

Mit 7-Zip können weitere Archivtypen entpackt werden.

```
> find <PathToImage> -iname "*.rar" -exec 7z e {} -C <OutputPath> \;
```

6.3.11 Suche nach Dateien / Filterung von Dateien

Muss auf einem System eine bestimmte Datei gefunden werden oder sollen als ungefährlich eingestufte Dateien aus der Gesamtheit der zu analysierenden Daten herausfiltriert werden, kann dies über Hash-Listen gemacht werden.

Für den ersten Fall wird für jede gesuchte Datei ein Hash erzeugt und in einer Datei abgelegt. Die zu suchende Datei muss zwingend bis auf das letzte Bit identisch mit der zu suchenden Datei sein. Andernfalls kann diese nicht gefunden werden. Für solche Fälle könnte die Datei-Blockweise gehasht und anschliessend nach diesen Teil-Hashes gesucht werden.

Linux

```
> find . -type f -print0 | xargs -o md5sum > <PathToHashlist>
> find <PathToFilesToSearch> -type f -print0 | xargs -o md5sum | awk '{
print $1}' > <PathToHashesOfFilesToSearch>
> grep -f <PathToHashlist> <PathToHashesOfFilesToSearch>
```

6.3.12 Analyse des File Slacks

Aufgrund der Besonderheiten einiger Dateisysteme können auf Datenträgern mit einem solchen Dateisystem Dateien versteckt werden. Der Grund dafür liegt, darin das eine Datei auf dem Datenträger in sogenannten Dateiblöcken mit einer festen Länge (Sektoren) gespeichert werden. Die kleinste beschreibbare Einheit besteht dabei aus einer Gruppe (typischerweise 8) von Sektoren. Eine solcher Gruppe von Sektoren wird als Cluster bezeichnet. Füllt eine Datei nicht den gesamten Dateiblock aus entsteht ein ungenutzter Bereich. Dieser Bereich wird File Slack genannt und wird vom Betriebssystem mit zufälligen Daten aufgefüllt. Dieser Slack unterteilt sich zum einen in RAM-Slack, welcher früher unter Windows mit Inhalten aus dem Arbeitsspeicher aufgefüllt wurde (heute mit Nullen), und zum anderen in Drive-Slack. Der RAM-Slack erstreckt sich vom letzten Byte der Datei bis zur nächsten Sektorgrenze. Der Drive-Slack (auch Sektor- oder Cluster-Slack genannt) ist der restliche Bereich zwischen dem Ende des RAM-Slacks und der Cluster-Grenze.

Im Drive-Slack können sicht unter Umständen wertvolle Informationen für die weitere Analyse befinden. Wichtig zu wissen ist hier, das ein Beweis eindeutig einer Person zugeordnet werdene können muss. Dies ist beim Drive-Slack nicht der Fall.

Neben dem File-Slack gibt es auf einem Computer-System noch weitere Slack-Bereiche wie zum Beispiel der Partition-Slack oder Disk-Slack.

Die File Slack Analyse ermöglicht dem Ermittler unter Umständen auf Inhalte zuzugreifen, welcher der Täter bereits gelöscht hat.

6.3.13 Timeline-Analyse

Bei der Timeline-Analyse werden die letzten Aktivitäten, welche auf dem System durchgeführt wurden, in ein zeitlich logische Abfolge gebracht. Bei der Herstellung einer Verbindung zu anderen Beweisen ist eine allfällige Abweichung der Systemzeit von der Referenzzeit zu beachten. Die Timeline-Analyse ist häufig der erste Ansatzpunkt einer Ermittlung. Durch die Analyse kann festgestellt werden, was undder Täter auf dem System gemacht / verändert / installiert hat. Ausgehend von dem entstehenden Zeitstrahl können weitere, vertiefte Analysen durchgeführt werden. Ein wichtiger Baustein der Timeline-Analyse ist die Auswertung der MAC-Time der Dateien und Ordner. Die MAC-Time setzt sich aus folgenden Elementen zusammen:

- Modification Time
  Zeitpunkt der letzten Modifikation (Schreiben), Der Zeitstempel ändert sich bei folgenden Aktionen nicht: Kopieren, Verschieben, Umbenennen, Veränderung Dateiattribute
- Access-Time
  Zeitpunkt des letzten Zugriffes (Lesen / Ausführen), Der Zeitstempel wird auch verändert, wenn Metadaten oder Dateiinhalte angezeigt werden
- Creation-Time (Windows)
  Zeitpunkt der Erstellung der Datei, Der Zeitstempel wird bei Erstellung einer Kopie aktualisiert. Beim Verschieben einer Datei / Ordner wird der Zeitstempel nicht aktualisiert.
- Change-Time (Unix)
  Zeitpunkt der Veränderung bestimmter Metadaten der Datei

Nicht jede Aktion löst auf der Datei selbst eine Veränderung der MAC-Time aus. Unter Linux bewirkt die Verschiebung einer Datei eine Veränderung der MAC-Time des Verzeichnises, aber nicht der Datei selbst. Die MAC-Time kann jedoch relativ einfach durch den Angreifer, beziehungsweise durch Anti-Forensik-Tools manipuliert oder unbrauchbar gemacht werden. Eine weitere Schwierigkeit besteht darin, dass die MAC-Time sich je nach Betriebssystem anders verhält.

Sleuth Kit

1. Dateinformationen sammeln

```
1 fls -f <Filesystem> -m <Path> -r <MountPoint>/<PathToimage> > body.fls
2 fls -f linux-ext2 -m / -r /mnt/images/hda7.dd > body.fls
```

2. Metadaten sammeln

```
ils -f <Filesystem> -m <MountPoint>/<PathToImage> >> body.fls
ils -f linux-ext2 -m /mnt/images/hda7.dd >>body.fls
```

3. Gemeinsame Auswertung

```
mactime -b body.fls
```

log2timeline

Das Tool *log2timeline* ist in der Lage Zeitlinien von verschiedensten Log-Dateien zu erstellen. Darunter die Logs des Apache WebServers, Firefox, Chrome, Windows-Ereignisprotokolle, Mactime-formatierte Dateien und viele mehr.

```
> log2timeline -z CET -f apache_access -o sqlite -w /case/1234/mactime.sqlite /
mnt/image/avar/log/apache2
> log2timeline -z CET -f <InputModule> -o <OutputModule> -w <OutputPath>
```

Das Tool *log2timeline* besitzt viele Optionen und Parameter. Eine Vereinfachung des Tools stellt *log2timeline-sift* dar. Eine Weiterentwicklung und Optimierung von *log2timeline* wird unter dem Namen *plaso* entwickelt. Die gesammelten Daten können entweder mit einem Forensik-Tool oder aber einer Anwendung zum Erstellen von Diagrammen und Statistiken grafisch dargestellt werden. Unter Linux können die Daten zum Beispiel mit Hilfe von *GnuPlot* visualisiert werden.

6.3.14 Analyse von Auslagerungsdateien

Fast alle Betriebssysteme kennen das Prinzip der Auslagerungsdatei, bzw. Swap-Datei. Diese Auslagerungsdatei erweitert entweder den physisch nutzbaren Arbeitsspeicherbereich oder dient der Auslagerung von kurzfristig nicht benötigten Speicherinhalten. Dieser Swap-Bereich ist entweder eine Datei oder ein eigenes physisches oder virtuelles Dateisystem. Unter Windows wird das Page-File (früher Swap-File) beim Shutdown nicht gelöscht und kann somit bei einer Analyse interessante Informationen beinhalten. Wird ein System in den Ruhezustand, beziehungsweise Suspend-to-Disk-Modus versetzt, wird der gesamte Arbeitsspeicherinhalt auf die Festplatte gesichert. Dieser wird bei Reaktivierung wieder geladen, um den Zustand des Systemes wiederherzustellen. Unter Windows heisst diese Datei *hyberfil.sys* und liegt im Wurzelverzeichnis des Systemlaufwerkes.

6.3.15 Suche nach Rootkits

Die Suche nach Rootkits auf einem System kann sich unter Umständen als sehr schwierig und zeitintensiv herausstellen. Handelt es sich um ein Standard-Rootkit gibt es einige Tools, welche die Rootkits anhand von File- und Hash-Analysen aufspüren können.

chkrootkit

```
> chkrootkit
```

rkhunter

```
> rkhunter --check
```

6.3.16 Systemprotokolle

Systemprotokolle bieten in der Regel gute Hinweise auf einen zukünftigen oder bereits erfolgten Angriff. Die Analyse der Systemprotokolle ist meistens nicht Teil der Post-Mortem-Analyse, da der Angriff erst durch die Systemprotokolle entdeckt wird. In den Systemprotokollen lassen sich die Meldungen grundsätzlich in drei Kategorien einteilen. Zum einen die normalen Meldungen aus dem Tagesbetrieb und zum anderen kritische und unbekannte Meldungen, welche einen Hinweis auf einen Angriff darstellen können. Weitere Informationen zu diesem Thema sind im Kapitel 3.1 Incident Detection (Erkennung eines Vorfalls) aufgezeigt.

6.3.17 Untersuchung der Shell (Bash)

Die Untersuchung der Shell kann weitere wertvolle Informationen liefern. Zum einen kann die MAC-Time der Shell selbst untersucht werden. Zum anderen können die von der Shell verwendeten Dateien untersucht werden. Dazu gehören die Dateien *.bash_profile* und *.bashrc*, welche die Befehle enthalten, welche bei einem Start der Shell ausgeführt werden. Die interessantere Datei ist die Datei *.bash_history*. Diese Datei enthält den Audit-Trail (Histroy) der eingegebenen Befehle. Angreifer verändern diese Datei oft so, dass diese einen Link auf die Datei */dev/null* darstellt. Alles was unter Linux nach */dev/null* geschrieben wird, ist unwiderruflich verloren. Allenfalls weitere Hinweise könnte die Datei *.bash_logout* liefern, welche die Befehle enthält, welche beim Verlassen der Shell ausgeführt werden.

Es kann sich auch lohnen das Verzeichnis */etc/skel* einer genaueren Analyse zu unterziehen, da die Standard-Dateien enthält, welche beim Anlegen eines neuen Benutzers kopiert werden.

6.3.18 Untersuchung der Druckerjobs und der Druckerqueue

Die vom System aufgegebenen Druckerjobs können interessante Informationen enthalten. Auf älterne Unix-Systemen sind die entsprechenden Log-Dateien unter */vaR/log/lpr.log* zu finden. Auf neueren Systemen unter */var/log/cups*. Der Ort der Log-Datei kann allenfalls auch durch ein Überschreiben der Konfiguration im Verzeichnis */etc* geändert worden sein. Die Kopien der gedruckten Dateien können im Verzeichnis */var/spool*, beziehungsweise */var/spool/cups* gefunden werden. Die Analyse der Druckjobs und der Druckerqueue kann einen guten Ansatzpunkt in der Timeline-Analyse darstellen. Die Log-Einträge können zum Beispiel Hinweise auf eine gelöschte Datei beinhalten und bieten somit einen ersten Anhaltspunkt für die Suche der Datei im Speicherbereich.

6.3.19 Untersuchung der Dateien / Dateiendungen

Oft wird eine Datei über deren Dateiendung identifiziert. Wurde diese absichtlich verändert, kann der Dateityp weiterhin bestimmt werden. Dies wird durch File-Signatur, beziehungsweise den File-Header, ermöglicht. Durch eine Signaturanalyse können auf einem System Dateien gefunden werden, bei denen Dia Signatur nicht mit der Dateiendung übereinstimmt. Die Ergebnisse einer solchen Analyse sind jedoch mit Vorsicht zu geniessen, da der File-Header auch falsch sein kann. Ein Beispiel wäre eine Text (txt) Datei mit dem Inhalt „FLV ist ein cooles Format“. Die Signatur würde dann nahelegen, dass es sich um ein File im „FLV“ Format handelt. In Wirklichkeit handelt es sich jedoch um ein txt-File.

Linux

Der File Befehl ermittelt gestützt auf das *magic* File den File-Header der Datei. Diese Datei befindet sich oft in einem der folgenden Verzeichnisse: */usr/magic*, */etc/gnome-vfs-mime-magic*, */usr/share/mimelnk/magic*.

```
> file <PathToFile>

> file ./Test.txt
Test.txt: Macromedia Flash Video
```

6.3.20 Datei- und Verzeichnisrechte

Auch die Untersuchung der Datei-Verzeichnisrechte, beziehungsweise deren Veränderung der MAC-Time, kann wichtige Hinweise für die Analyse liefern.

Linux

```
>  -lrta /etc/
> ls -lrta /bin
> ls -lrta /sbin
> ls -Rlrta /usr
> ls -Rlrta /var
> ls -Rlrta /dev
> ls -Rlrta /home
> ls -Rlrta /lib
```

6.3.21 Analyse des RAM-Dumps

Die Auswertung von RAM-Dumps kann sehr zeitaufwändig sein, da sich die Struktur des RAMs grundlegend von der Struktur von normalen Filesystemen unterscheidet. Hinzu kommt, dass im RAM oft nur Fragemente von Dateien abgelegt sind.

Volatility

Volatility ist ein Framework zur Analyse von RAM-Dumps. Das Framework bietet unzählige Optionen und Einstellmöglichkeiten. Nachfolgend werden einige grundlegende Befehle aufgezeigt.

- Allgemeine Informationen

```
> python vol.py -f <PathToImage> imageinfo
```

- Liste aller Prozesse

```
> python vol.py --profile=<Profile> -f <PathToImage> pslist
```

- Prozesse als Baumstruktur

```
> python vol.py --profile=<Profile> -f <PathToImage> pstree
```

- Liste von Dateiobjekten

```
> python vol.py --profile=<Profile> -f <PathToImage> filescan
```

6.3.22 Konvertierung von Images

Ab und zu kann es notwendig sein, die erstellten Images in ein anders Format zu konvertieren.

ewfacquire

- Konvertierung von DD nach E01

```
> ewfacquire -t <PathToSourceImage.dd> <PathToTargetImage.E01>
```

- Konvertierung von E01 nach DD

```
> ewfeport <PathToSourceImage.E01>
```

xmount

- Konvertierung von vmdk nach E01

```
> xmount --in ewf --out vmdk <PathToSourceImage.E01> <DestinationPath>
```

- Konvertierung von E01 nach vmdk

```
> xmount --in vmdk --out ewf <PathToSourceDisk.vmdk> <DestinationPath>
```

Weitere Tools

Die nachfolgenden Tools verfügen ebenfalls über entsprechende Funktionalitäten:

- FTK Imager
- VHD Tool

6.3.23 Analyse des Master Boot Records

Der Master Boot Record ist für den Bootvorgang von essentieller Bedeutung und kann auch bei einer forensischen Analyse hilfreiche Informationen bezüglich Partitionierung und verwendeten Dateisystemen beinhalten. Der Master Boot Record ist wie folgt aufgebaut:

- Bootcode (Bytes 0-439)
- Disksignatur (Bytes 440-443)
- Reservierter Bereich (Bytes 444-445)
- Partitionstabelle (Bytes 446-509)
- Signatur *0x55 0xAA* (Bytes 510-511)

Sicherung des Master Boot Records

Mit Hilfe des Tools *dd* oder eines ähnlichen Tools kann der Master Boot Record wie folgt gesichert werden:

```
1 > dd if=<PathToSource> of=<PathToTargetFile.dd> bs=512 count=1 skip=0
```

Sicherung des Master Boot Records

Mit Hilfe des Tools *dd* oder eines ähnlichen Tools kann der Master Boot Record wie folgt gesichert werden:

```
1 > dd if=<PathToSource> of=<PathToTargetFile.dd> bs=512 count=1 skip=0
```

Extraktion des Master Boot Records aus einem image

```
1 > dd if=<PathToSourceImage.dd> of=<PathToTargetFile.dd> bs=512 count=1 skip=0
```

6.3.24 Weitere Analyse-Möglichkeiten

Nachfolgend werden weitere Analysemöglichkeiten aufgezeigt, welche in diesem Kapitel nicht näher beschrieben werden.

- Analyse unbekannter Binärdateien
- Analyse von User Aktivitäten

- Analyse bei Verdacht auf Anti-Forensik-Techniken
- File-Carving
- Block-Hashing
- Suche nach Regulären Ausdrücken (Regex)
- Erstellen von Datei-Listen
- Sammeln von bestimmten Dateitypen
- Suche und Auswertung von Log-Files
- Suche nach SUID und GID-Dateien
- Analyse der Master File Table
- Analyse von NTFS-Streams
- Analyse von NTFS TxF
- Analyse der Windows-Registry
- Analyse der Windows UserAssist Keys
- Analyse der Windows Prefetch-Dateien
- Analyse von Netzwerkmitschnitten
- NTFS-Volumen-Schattenkopien

6.4 Reporting

Gewisse Tools und Tool-Suiten bieten eine integrierte Möglichkeit um Berichte und Dokumentationen zu generieren.

6.5 Hinweise zu Tooleinsatz

Nachfolgend einige Hinweise zum Einsatz von Tools bei einer forensischen Analyse:

- Auswahl des Tools kann Grundsatzdiskussionen zu den Ermittlungsmethoden auswählen.
- Verständnis der unterschiedlichen Ansätze verschiedener Forensik-Tools und -Tool-Suiten.
- Open Source vs. Kommerzielle Tools: Klassische ideologische Diskussion. Für die juristische Verwendbarkeit sind beide Ansätze legitim, sofern die gewählten Tools akzeptiert und etabliert sind.

- Einsatz von Script-Sammlungen (Konsolen-Befehle) vs. GUI-Anwendungen. Der Einsatz von GUI-Anwendungen bietet erheblich mehr komfort als der Einsatz von Script-Sammlungen, beziehungsweise dem Absetzen von Konsolen-Befehlen. Der Nachteil der GUI-Anwendungen ist jedoch, dass diese einen höheren RAM-Bedarf haben. Daher sollte bei einer Live-Analyse auf den Einsatz von grafischen Tools wann immer möglich verzichtet werden.
- Die einzusetzenden Tools sind stark von der jeweiligen Situation und dem zu untersuchenden System abhängig. Die Toolsammlung sollte daher jeweils den Umständen entsprechend angepasst werden.
- Der Umgang mit den verschiedenen Tools sollte regelmässig geübt werden, damit bei der Untersuchung schnell, effizient und korrekt gearbeitet werden kann.
- Bei der Live-Analyse sollten immer nur statisch vorkompilierte Systembefehle verwendet werden. Andernfalls ist die Korrektheit der so gesammelten Daten nicht gewährleistet.
- Das Analyse-System des Ermittlers sollten immer abgesichert sein.
- Die Ergebnisse der Analyse sollten verschlüsselt abgespeichert werden.

6.6 Tool-Suiten und Toolsammlungen

Nachfolgend werden enige Tool-Suiten und Toolsammlungen für die forensische Analyse aufgelistet:

- F.R.E.D. (First Responder's Evidence Disk)
  Schnelle Sammlung von Statusinformationen auf Live-Systemen, statisch vorkompilierte Vefehle, anpassbares und erweiterbares Batch- / Shell-Script
- Incident Response Collection Report (IRCR)
  Sammlung von Werkzeugen zur Sammlung kritischer Systemdaten, Tool für Windows, Verwendung von Systembefehlen zur Sammlung der Daten (hinterlässt Spuren auf dem System)
- Windows Forensic Toolchest (WFT)
  Tool-Sammlung zur Analyse von Live-Systemen unter Windows, Hinterlässt wenige Spuren, Flexible Anpassung der generierten Reports
- Live View
  Java-Anwendung zur Generierung einer read-only VM-Ware Virtual Machiene aus einem dd Image oder einem physischen Datenträger unter Windows.
- EnCase
  Windows-Anwendung zur Analyse von verschiedenen Dateisystemen (Windows, Mac OS, Linux, Solaris, AIX, HP UX, ...), Automatisierung von verschiedenen Aufgaben

via EnScript, Breite Palette an Tools, Analyse von verschiedensten Anwendungsdaten, Mehrsprachig, erhältlich als Enterprise-Version

- F.I.R.E
  Freie Toolsammlung zur forensischen Analyse und Datenrettung. Beinhaltet das Sleuth Kit, den Autopsy Forensic Browser und viele mehr, geeignet für die Incident Response an Live-Systemen und Penetrationstests, Erhältlich für Windows, Linux und Solaris, Keine Wartung / Weiterentwicklung
- Knoppix Security Tools Distribution
  Sammlung von Security-Tools für Penetrations-Tests, Forensische Analysen und Incident-Response, Keine Wartung / Weiterentwicklung
- Helix Forensisches Toolkit inklusive Case-Management (Enterprise-Version), statisch vorkompilierte Dateien für LInux, Solaris und Windows, Support durch Hersteller, hoher Integrationsgrad, Live-Response für Windows
- ForensiX-CD
  Toolsammlung für Windows und Linux, Statisch vorkompilierte Systemprogramme
- C.A.I.N.E und WinTaylor
  Live-CD für Windows (WinTaylor) und Linux (C.A.I.N.E), Sammlung von Incident-Response-Tools, Benötigt als Basis Visual Basic 6 (entsprechende Bibliotheken werden auf dem System benötigt)
- DEFT und DEFT-Extra
  Werkzeug-Sammlung für die Live-Response unter Linux und Windows.
- Forensic Acquisition Utilities
  Tool-Sammlung für die forensische Analyse, beziehungsweise Datenesammlung
- AccessData Forensic Toolkit
  Komplette Analyseumgebung für Unix- und Windows-Dateisysteme, Mehrbenutzerfähig (via Datenbank-Server)
- The Coroner's Toolkit and TCTUtils
  Tool-Sammlung für die Post-Mortem Analyse von Unix-Systemen.
- The Sleuth Kit
  Sammlung von diversen Tools zur forensischen Analyse, Unterstützt eine breite Palette an Formaten.
- Autopsy Forensic Browser
  Grafische Oberfläche für „The Leuth Kit“, Neuste Version nur für Windows verfügbar

KAPITEL 7

Schlusswort

7.1 Fazit

7.2 Reflexion

Quellenverzeichnis

[E20a] 2015. URL: https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/400106/Common_Cyber_Attacks-Reducing_The_Impact.pdf.

[E20b] 2015. URL: http://de.wikipedia.org/wiki/Hacker_(Computersicherheit).

[E20c] *Cyber Incident Response Guide*. EN. Multi-State Information Sharing und Analysis Center (MS-ISAC). 2010. URL: http://msisac.cisecurity.org/members/local-government/documents/finalincidentresponseguide.pdf.

[E20d] *Guide to Integrating Forensic Techniques into Incident Response*. NIST. 2006. URL: http://csrc.nist.gov/publications/nistpubs/800-86/SP800-86.pdf (siehe S. 17).

[Sil] Siller, Prof. (FH) Mag. Dr. Helmut: *Forensik*. DE. Wirtschaftslexikon Gabler. URL: http://wirtschaftslexikon.gabler.de/Archiv/1408495/forensik-v3.html.

[Ste12] Stern, Olaf: *Reglement: Seminararbeit*. Deutsch. ZHAW. 2012. URL: https://ebs.zhaw.ch/files/documents/informatik/Reglemente/Bachelor/Seminararbeit/a_Reglement-Seminar_Studiengang-Informatik_V2.1.docx (siehe S. 1).

[Web11] Websense: *Advanced persistent threats and other advanced attacks*. EN. Rev2. Websense. 2011. URL: https://www.websense.com/assets/white-papers/whitepaper-websense-advanced-persistent-threats-and-other-advanced-attacks-en.pdf.

Abbildungsverzeichnis

Tabellenverzeichnis

ANHANG A

Vorlage: Formular Incident-Meldung

Informationen zur Meldung

Datum: _____ Uhrzeit: _____

Meldung via: ☐ Telefon Bearbeiter: _____
☐ E-Mail
☐ Monitoring
☐ _____

Informationen zum Melder

Meldertyp: ☐ Person
☐ System

Name: _____ Vorname: _____
Organisation: _____ Kurzzeichen: _____
Telefon: _____ Ort: _____
Sprache: _____
Büro-Adresse: _____
_____

Weiteres: _____
_____

Informationen zum Incident

Datum: _____ Uhrzeit: _____

Ort: _____

Beschreibung Vorfall: _____

_____

_____

_____

Betroffene Systeme: _____

_____

Festgestellte Auswirkungen: _____

_____

Vermuteter Schaden: _____

_____

Aktueller Zustand: _____

_____

Details zum System: _____

_____

Vermutung zur Ursache: _____

_____

Informierte Personen: _____

_____

Getroffene Massnahmen: _____

_____

Klassifizierung Daten:

- ☐ Öffentlich
- ☐ Nur für internen Gebrauch
- ☐ Vertraulich
- ☐ Streng vertraulich

Weitere Informationen: _____

_____

_____

_____

_____

_____

_____

_____

ANHANG B

Vorlage Formular Ermittlung

-Fallnummer -Datum / Zeit -Wieso forensische untersuchung (Verdacht?) -Grund -Ermittlungsleiter -Ermittlungsteam -Betroffene Systeme / Geräte / Anwendungen 8(Seriennummern und interne Bezeichnung) -Verantwortliche Administratoren -Protokolle, Incident Meldung, Beweiszettel

Täterprofil -Was waren / sind mögliche Ziele -Was ist der Grund für den Angriff / Einbruch? -(Interne) komplizen? -Tools / Techniken? -Spuren? ....

ANHANG C

Vorlage: Protokoll

Tabelle mit : Laufnummer, Zeit, Befehl / Aktion, Hash Ergebnisdatei, Kommentar CF Seite 84

ANHANG D

Vorlage: Beweiszettel

Buch CF: Seite 85

Beweiskette:

Laufwerke: Manufacturer, Model, Serial Number, Evidence Description (Name of suspect, Technologie: SATA, IDE, ...)

ANHANG E

Ablauf einer forensischen Analyse

Liste der noch zu erledigenden Punkte

Bild CF Seite 13 6