Information Security

Network Forensics

State of the Art

Diplomarbeit

zur Erlangung des akademischen Grades

Diplom-Ingenieur/in

eingereicht von

Florian Scheiber

is191846

im Rahmen des

Studienganges Information Security an der Fachhochschule St. Pölten

Betreuung

Betreuer/in: FH-Prof. Dipl.-Ing. Dipl.-Ing. Christoph Lang-Muhr, BSc

Mitwirkung: -

St. Pölten, 11. Oktober 2021

(Unterschrift Verfasser/in) (Unterschrift Betreuer/in)

Ehrenwörtliche Erklärung

Ich versichere, dass

- ich diese Diplomarbeit selbständig verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und mich sonst keiner unerlaubten Hilfe bedient habe.
- ich dieses Diplomarbeitsthema bisher weder im Inland noch im Ausland einem Begutachter/einer Begutachterin zur Beurteilung oder in irgendeiner Form als Prüfungsarbeit vorgelegt habe.
- diese Arbeit mit der vom Begutachter/von der Begutachterin beurteilten Arbeit übereinstimmt.

Der Studierende/Absolvent räumt der FH St. Pölten das Recht ein, die Diplomarbeit für Lehre- und Forschungstätigkeiten zu verwenden und damit zu werben (z.B. bei der Projektevernissage, in Publikationen, auf der Homepage), wobei der Absolvent als Urheber zu nennen ist. Jegliche kommerzielle Verwertung/Nutzung bedarf einer weiteren Vereinbarung zwischen dem Studierenden/Absolventen und der FH St. Pölten.

Ort, Datum *Unterschrift*

Kurzfassung

Cyberkriminalität macht weder vor verschlossenen Türen noch an Staatsgrenzen halt. Sie ist ein weltweites Phänomen, das stetig wächst. Überall dort, wo vernetzte Geräte zum Einsatz kommen, kann Cyberkriminalität stattfinden. Um Cyberkriminelle zur Verantwortung zu ziehen, müssen diese ausfindig gemacht und die digitalen Beweise gesichert werden. Dabei kommt die digitale Forensik zum Einsatz. Durch die fortschreitende Vernetzung von Geräten und Dingen kommt immer öfter die Netzwerk-Forensik als Teilgebiet der digitalen Forensik zur Anwendung. Sie befasst sich mit der Entdeckung und dem Erlangen von beweiskräftigen Informationen über ein Verbrechen in einer vernetzten Umgebung. Rechenzentren dienen heutzutage als Basis für viele Services. Seit einigen Jahren veränderten sich zunehmend die Netzwerke der Rechenzentren, von traditionellen zu virtuellen Netzen, um mehr Flexibilität zu erreichen und den Verwaltungsaufwand zu reduzieren. Dieser Umstand und das Aufkommen von sicheren Verschlüsselungsprotokollen wie TLS 1.3 stellen die Netzwerk-Forensik vor neuen Herausforderungen. Das Ziel dieser Arbeit ist es, einen umfassenden Überblick über die Netzwerk-Forensik zu geben, sowie den Ist-Zustand festzuhalten. Es werden wichtige Standards und Guidelines beschrieben und gegenübergestellt. Verschiedene Tools, die bei einer netzwerkforensischen Untersuchung unterstützen können, werden beschrieben und drei dieser Tools in einem Praxistest analysiert. Zudem werden die Auswirkungen von virtuellen Netzen und dem Verschlüsselungsprotokoll TLS 1.3 auf die Netzwerk-Forensik vermittelt und aktuelle Frameworks und Konzepte evaluiert.

Abstract

Cybercrime does not stop at closed doors or national borders. It is a worldwide phenomenon that is constantly growing. Wherever networked devices are used, cybercrime can take place. To hold cybercriminals accountable, they must be tracked down and the digital evidence secured. This is where digital forensics comes in. Due to the progressive networking of devices and things, network forensics is increasingly used as a subfield of digital forensics. It deals with the discovery and obtaining of conclusive information about a crime in a networked environment. Nowadays, data centres serve as the basis for many services. Over the past few years, data centre networks have increasingly changed from traditional to virtual networks in order to achieve more flexibility and reduce management overhead. This circumstance and the emergence of secure encryption protocols such as TLS 1.3 pose new challenges to network forensics. The aim of this paper is to provide a comprehensive overview of network forensics, as well as to capture the current state. Important standards and guidelines are described and compared. Various tools that can support a network forensic investigation are described and three of these tools are analysed in a practical test. In addition, the effects of virtual networks and the TLS 1.3 encryption protocol on network forensics are discussed and current frameworks and concepts are evaluated.

Inhaltsverzeichnis

1 Einleitung

Das Internet hat zwar tiefgreifende Auswirkungen auf alle Aspekte des modernen Lebens, doch seine Nutzung macht Unternehmen und Einzelpersonen anfällig für Cyberkriminalität.[1] Mit den Fortschritten in der Informationstechnologie und der rasanten Digitalisierung der Welt wächst auch die Zahl der Geräte, die an ein Netzwerk angeschlossen werden können.[2] Dadurch ergibt sich eine immer größer werdende Angriffsfläche für Cyberkriminelle. Das Resultat daraus ist, dass die Cyberkriminalität von Jahr zu Jahr zunimmt.[1] Im ersten Quartal 2021 wurden mehr als 87 Millionen neue Malware-Instanzen entdeckt und die Gesamtzahl der Malware hat über 1,5 Milliarden erreicht.[1] Die Folgen sind unter anderem finanzielle Schäden, Verlust der Privatsphäre, Verletzung des geistigen Eigentums und Rufschädigung.[1]
Um Cyberkriminelle zur Verantwortung zu ziehen, müssen diese ausfindig gemacht und die digitalen Beweismittel der Straftat gesichert werden. Dabei kommt die digitale Forensik zum Einsatz. Sie wurde von der Strafverfolgung eingeführt und hat viele Leitprinzipien aus der investigativen Methodik des Justizsystems.[3] Als Reaktion auf die Hacker-Community entwickelte sich die Netzwerk-Forensik[4] und bildet heute ein wichtiges Teilgebiet der digitalen Forensik. Durch die fortschreitende Vernetzung von Geräten und Dingen, wird sie bei digitalen Ermittlungen immer wichtiger.[5] Die Netzwerk-Forensik kann allgemein als Wissenschaft definiert werden, die sich mit der Entdeckung und dem Erlangen von beweiskräftigen Informationen über ein Verbrechen in einer vernetzten Umgebung befasst und zwar so, dass sie vor Gericht zugelassen werden können.[6]

Die Entwicklung von Virtualisierungstechniken verändert die Betriebsprinzipien in den heutigen Rechenzentren. Die Virtualisierung von Servern, Netzwerken und Speichern erhöht die Flexibilität und Dynamik der Umgebung, indem sie den Verwaltungsaufwand reduziert. Ausgehend von einem physischen Underlay-Netzwerk werden verschiedene logische Netzwerke mit neuen Protokollen implementiert. Neue Paradigmen wie Software-Defined-Networks oder Network Function Virtualization bieten neue Möglichkeiten zur Neugestaltung der gesamten Netzwerkinfrastruktur. Dieser Trend schafft neue Herausforderungen für die

[1] https://www.mcafee.com/enterprise/en-us/assets/reports/rp-threats-jun-2021.pdf [Zugriff am: 12.09.2021]

digitale Ermittlung von Vorfällen.[7] Auch neue Verschlüsselungsprotokolle, die den Netzwerkverkehr verschlüsseln und so die Analyse dessen verhindern, stellen eine weitere Herausforderung für die Netzwerk-Forensik dar. Um diesen Herausforderungen auf den Grund zu gehen, ergibt sich folgende Forschungsfrage:

Welche Standards und Guidelines gibt es? Welche Tools können zur netzwerkforensischen Untersuchung verwendet werden? Wie funktioniert die Netzwerk-Forensik in virtuellen Netzen? Wie wirkt sich TLS 1.3 auf die Netzwerk-Forensik aus?

Das Ziel dieser Arbeit ist es, einen umfassenden Überblick über die Netzwerk-Forensik zu geben, sowie den Ist-Zustand festzuhalten. Es werden werden verschiedene Standards und Guidelines die international beziehungsweise europaweit relevant und anerkannt sind zusammengetragen und allgemein beschrieben. Im Anschluss werden die Eigenheiten dieser hervorgehoben. Verschiedene Tools, die bei einer netzwerkforensischen Untersuchung unterstützen können, werden kategorisiert und beschrieben. Drei dieser Tools werden in einem Praxistest auf deren Fähigkeiten analysiert. Dabei werden Testdaten im PCAP(NG)-Format ausgewählt und verschiedene Kriterien zur Bewertung festgelegt. Zudem werden die Auswirkungen von virtuellen Netzen und dem Verschlüsselungsprotokoll TLS 1.3 auf die Netzwerk-Forensik vermittelt und aktuelle Frameworks und Konzepte evaluiert.

1.1 Struktur der Arbeit

Dieses Dokument ist in mehrere Teile gegliedert. Kapitel 1 führt in das Thema, die Probleme, die Herausforderungen und die Motivation ein. Kapitel 2 bietet einen umfassenden Einblick in die Grundlagen der Netzwerk-Forensik Kapitel 3 listet wichtige wissenschaftliche Arbeiten für diese Diplomarbeit auf, zudem wird ein Prozessmodell der Netzwerk-Forensik vorgestellt.
Kapitel 4 beschreibt die geleistete Arbeit und die Ergebnisse. Dieses Kapitel ist in fünf Bereiche unterteilt. Abschnitt 4.1 listet und beschreibt wichtige Standards und Guidelines. Abschnitt 4.2 listet und beschreibt verschiedene Tools, die bei einer netzwerkforensischen Ermittlung eingesetzt werden können, auf. Im Abschnitt 4.4 werden drei Tools in der Praxis getestet und deren Fähigkeiten analysiert. Abschnitt 4.5 bietet einen Überblick über den Stand der Forschung der Netzwerk-Forensik in virtuellen Netzen. Abschnitt 4.6 beschreibt verschiedene Möglichkeiten, um Netzwerkverkehr der mit TLS 1.3 verschlüsselt wurde zu analysieren. Kapitel 5 fasst die Ergebnisse der Arbeit zusammen und gibt einen Ausblick auf die weiterführende Arbeit.

2 Grundlagen

2.1 Definition

Die digitale Forensik wurde von der Strafverfolgung eingeführt und hat viele Leitprinzipien aus der investigativen Methodik des Justizsystems.[3] Als Reaktion auf die Hacker-Community entwickelte sich die Netzwerk-Forensik[4] und bildet heute ein wichtiges Teilgebiet der digitalen Forensik. Durch die fortschreitende Vernetzung von Geräten und Dingen, wird sie bei digitalen Ermittlungen immer wichtiger.[5] Die Netzwerk-Forensik ist die Wissenschaft, die sich mit der Erfassung, Aufzeichnung und Analyse des Netzwerkverkehrs zur Erkennung von Eindringlingen und der Untersuchung dieser Vorfälle beschäftigt. Dabei werden Verkehrsdaten analysiert, die durch Firewalls, Intrusion-Detection-Systeme oder an Netzwerkgeräten wie Routern und Switches aufgezeichnet werden.[8]

Der Begriff Netzwerk-Forensik wurde erstmals 1997 vom Securityexperten Marcus Ranum als die Erfassung, Aufzeichnung und Analyse von Netzwerkereignissen, um die Quelle von Sicherheitsangriffen oder anderen problematischen Vorfällen zu entdecken, definiert.[9]

Der erste Digital Forensics Research Workshop (DFRW) wurde 2001 abgehalten und befasste sich mit dem Aufdecken von unautorisierten Aktivitäten und Wiederherstellungsmaßnahmen. Dabei wurde der Begriff Netzwerk-Forensik wie folgt definiert: Die Verwendung wissenschaftlich erprobter Techniken zum Sammeln, Sichern, Identifizieren, Untersuchen, Korrelieren, Analysieren und Dokumentieren digitaler Spuren, die aus mehreren aktiv verarbeitenden und übertragenden digitalen Quellen stammen. Das Ziel dabei ist das Aufdecken unbefugter Aktivitäten, die darauf abzielen, Systemkomponenten zu stören, zu beschädigen oder zu kompromittieren. Sowie Informationen bereitzustellen, die bei der Reaktion auf oder der Wiederherstellung nach diesen Aktivitäten helfen.[10]
Zudem entwickelte der DFRW ein Prozessmodell für die digitale Forensik. Dieses Prozessmodell ist die Grundlage für alle vorgeschlagenen Modelle, die bis heute verwendet werden.[11]

Bei der Netzwerk-Forensik wird der Netzwerkverkehr überwacht, um festzustellen, ob es eine Anomalie im Datenverkehr gibt und ob diese auf einen Angriff hinweist. Wenn ein Angriff erkannt wird, wird auch die Art des Angriffs bestimmt. Netzwerkforensische Techniken ermöglichen es den Ermittlerinnen/Ermittlern, die Angreiferinnen/Angreifer zurückzuverfolgen. Das ultimative Ziel ist es, genügend Beweise zu liefern, um die/den Täterin/Täter strafrechtlich verfolgen zu können.[12]

Netzwerk-Forensik ist kein anderer Begriff für Netzwerksicherheit. Sie kann als eine erweiterte Phase der Netzwerksicherheit gesehen werden, da die Daten für die forensische Analyse aus Sicherheitsprodukten gesammelt werden. Die defensiven Ansätze der Netzwerksicherheit wie Firewalls und Intrusion Detection Systeme können Angriffe nur aus der Perspektive der Prävention, Erkennung und Reaktion angehen. Der alternative Ansatz der Netzwerk-Forensik ist dahingehend sehr wichtig, da er auch die investigative Komponente beinhaltet.[9]

2.1.1 Digitale Beweise

Casey definiert digitale Beweise als alle Daten, die mit einem Computer gespeichert oder übertragen wurden und die eine Theorie darüber, wie eine Straftat begangen wurde, stützen beziehungsweise widerlegen oder die sich auf entscheidende Elemente der Straftat beziehen, wie Vorsatz oder Alibi.

Die Daten, auf die in dieser Definition Bezug genommen wird, sind im Wesentlichen eine Kombination von Zahlen, die Informationen verschiedener Art darstellen, einschließlich Text, Bilder, Audio und Video.[13]

2.1.2 Cloud-Forensik

Die Cloud-Forensik ist ein Teilgebiet der Netzwerk-Forensik. Cloud Computing basiert auf einem breiten Netzzugang. Daher folgt die Cloud-Forensik den Hauptphasen der Netzwerkforensik mit Techniken, die auf Cloud-Computing Umgebungen zugeschnitten sind.[14] Auf die Cloud-Forensik wird in dieser Arbeit jedoch nicht weiter eingegangen.

2.2 Motivation

Die große Anzahl von Sicherheitsvorfällen, von denen viele Organisationen betroffen sind, und die zunehmende Raffinesse dieser Cyberangriffe sind die Hauptantriebskraft für die Netzwerk-Forensik.[8] Erfolgreiche Angreiferinnen/Angreifer sorgen oft dafür, dass sie ihre Spuren verwischen. Erfolglose Angriffe bleiben

oft unbemerkt und selbst wenn sie bemerkt werden, sind nur wenige Informationen verfügbar die bei der Diagnose helfen.[15]
Infolgedessen werden Informationssicherheitsexpertinnen/Informationssicherheitsexperten aufgefordert, ihre Netzwerke als Beweisquelle zu kultivieren und Systeme zur Überwachung von Protokolldateien und Netzwerkverkehr auf verdächtige Aktivitäten zu installieren. Aufgezeichneter Netzwerkverkehr ist eine besonders überzeugende Form der digitalen Spuren, da er dazu verwendet werden kann, alle Aktionen der/des Täterin/Täters darzustellen. Vergleichbar mit einer Videoaufzeichnung von einem Überfall auf einen Supermarkt. Viele Unternehmen unterschätzen die Bedeutung von digitalen Beweisen und erkennen nicht, dass diese Daten die Grundlage für Schlussfolgerungen und Entscheidungen in Bezug auf einen Vorfall bilden. Schwache Beweise können zu ungenauen Schlussfolgerungen und schlechten Entscheidungen führen, die mehr Schaden und Haftung verursachen können als der Vorfall selbst.[16]

Die Netzwerk-Forensik hilft dabei, Cyberkriminelle aufzuspüren und digitale Spuren zu sammeln. Um diese Kriminellen zur Verantwortung zu ziehen, können diese Spuren in einem Gerichtsverfahren vorgelegt werden. Dies wird potenzielle Eindringlinge dazu bringen, zweimal nachzudenken, bevor sie in Zukunft ihre Angriffe durchführen.[17]

Im Falle eines digitalen Einbruchs kann die Netzwerkforensik helfen, Fragen zu beantworten, die die Organisation möglicherweise hat:[1]

- Wie lange dauert diese Aktivität schon an (d. h., wann hat der Einbruch begonnen)?
- Ist die Aktivität noch im Gange?
- Wie viele Systeme waren betroffen?
- Welche Daten wurden entwendet?
- Wurden sensible, geschützte oder vertrauliche Informationen entwendet?

Neben dem Erkennen von Netzwerkangriffen ist die Netzwerk-Forensik anwendbar, um Netzwerkprobleme von geschäftskritischen Systemen zu lösen. Wie etwa Geräte mit Sicherheitsschwachstellen zu identifizieren, Malware zu stoppen, die Performance des Netzwerks zu erhöhen und den Netzwerkfluss für den täglichen Betrieb zu überwachen.[17]

[1] https://www.fireeye.com/blog/executive-perspective/2014/07/network-forensics-use-cases-in-the-enterprise.html [Zugriff am: 21.04.2021]

Praktisch jede Organisation muss über eine gewisse Befähigung zur Durchführung von netzwerkforensischen Untersuchungen verfügen. Ohne eine solche Möglichkeit wird eine Organisation Schwierigkeiten haben, festzustellen, welche Ereignisse in ihren Systemen und Netzwerken stattgefunden haben, zum Beispiel die Offenlegung geschützter und sensibler Daten.[18] Wenn man an netzwerkforensische Untersuchungen denkt, denkt man oft an Ermittlungen von Strafverfolgungsbehörden. Aber neben diesen und privaten Ermittlungsstellen können jedoch auch Incident-Response-Teams, Sicherheitsexpertinnen/Sicherheitsexperten und Netzwerkadministratorinnen/Netzwerkadministratoren netzwerkforensische Untersuchungen durchführen.[19]

2.3 Herausforderungen

Der Netzwerkverkehr stellt eine Reihe von Herausforderungen als Beweisquelle dar. Der Netzwerkverkehr ist sehr flüchtig (dynamisch) und muss sofort erfasst und aufbewahrt werden, sonst ist er für immer verloren.[20] Es gibt in der Regel nur eine Möglichkeit, Daten auf ihrem Weg durch ein Netzwerk zu erfassen und unzureichende Systeme zur Beweissicherung können zu unwiederbringlichen Verlusten führen.
Darüber hinaus werden Daten in Pakete zerkleinert bevor sie übertragen werden, so dass es notwendig ist Pakete zusammenzusetzen, um Daten in ihrer ursprünglichen Form zu erhalten. Es kann sich auch als schwierig erweisen, bestimmte Elemente aus der großen Anzahl von Datenströmen in einem Netzwerk zu finden und zu extrahieren. Zudem kommt, dass der Netzwerkverkehr aus vielen verschiedenen Protokollen und Medientypen besteht, was die Komplexität einer ohnehin schon komplizierten Quelle für digitale Spuren noch erhöht.[16]
Außerdem sind die Logs von Sicherheitskomponenten oft nicht für eine gründliche Untersuchung gedacht. Es kann sein, dass die Logs nicht genügend bzw. eine Menge unbenötigter Details enthalten. Außerdem liegen sie in verschiedenen inkompatiblen Formaten und Abstraktionsebenen vor.[21]

Eine zentrale Herausforderung bei der Netzwerk-Forensik besteht darin, zunächst sicherzustellen, dass das Netzwerk forensisch bereit ist. Für eine erfolgreiche Netzwerkuntersuchung muss das Netzwerk selbst mit einer Infrastruktur ausgestattet sein, die diese Untersuchung vollständig unterstützt. Das Entwerfen einer netzwerkforensischen Infrastruktur ist aufgrund der vielen Möglichkeiten im Design eine anspruchsvolle Aufgabe.[10] [22][23][9]
Zudem stellen neue Technologien und die Verschlüsselung des Datenverkehrs die Ermittlerinnen/Ermittler vor Herausforderungen, die es gilt zu überwinden, wie etwa die stetige Weiterentwicklung des eigenen

Knowhows oder das Speichern von Session-Keys.

Die hohe Datenrate des Netzwerkverkehrs erschwert es der Netzwerk-Forensik, alle Netzwerkpakete zu erfassen und aufzubewahren, da Millionen von Paketen in kürzester Zeit über das Netzwerk gesendet werden.[24] Dabei wird eine gewaltige Menge an Daten erfasst, aus dieser die/der Ermittlerin/Ermittler die Spuren extrahieren muss. Da die Speicherkapazität auf Netzwerkgeräten limitiert ist, müssen die erfassten Daten auf geeignete Geräte mit ausreichend Speicherplatz gesichert werden.[25]

Die Datenintegrität spielt eine wichtige Rolle im Prozess der Netzwerk-Forensik. Datenintegrität ist die Fähigkeit, genaue, vollständige und konsistente Daten im Netzwerk zu halten. Die Sicherstellung der Integrität bei der Datenerfassung im Netzwerk ist eine kritische und anspruchsvolle Aufgabe für die Netzwerk-Forensik. Der Umfang, die Größe und die Geschwindigkeit der Daten machen es für die Ermittlerinnen/Ermittler zu einer Herausforderung, die Integrität der Daten zu wahren. Die Datenintegrität wird durch Hardware- und Software-Fehler, böswillige Angriffe, Fehlfunktionen des Systems und die häufige Mobilität der im Netzwerk vorhandenen Daten beeinträchtigt. Es wirkt sich nachteilig auf den forensischen Prozess aus, wenn die Integrität der Daten, durch absichtliches oder unabsichtliches verändern, nicht erhalten bleibt.[17] Um die Integrität überprüfen zu können, wird direkt nach der Erfassung der Hashwert der Daten generiert. Dieser bildet den Fingerabdruck der erfassten Daten. So können diese zu jedem Zeitpunkt auf Veränderungen und Manipulationen überprüft werden.

Der Datenschutz ist ein wichtiger Faktor in der Netzwerk-Forensik. Es ist zu erwarten, dass die gesammelten Daten sensible Informationen wie persönliche E-Mails und Dateien enthalten. Diese Tatsache ermöglicht es forensischen Ermittlern, die Privatsphäre der User zu verletzen. Daher ist der richtige Umgang mit diesen Daten entscheidend.[26]

Die interne Verwendung der gesammelten Daten innerhalb einer Organisation ist etwas anderes als der Präsentation der Daten vor einem Gericht. In letzterem Fall müssen die gesammelten Daten strenge rechtliche Verfahren durchlaufen um sich vor Gericht als Beweismittel zu qualifizieren. Sie müssen eine Zulässigkeitsprüfung bestehen, einen sogenannten Screening-Prozess durch das Gericht.[27][28]

2.4 Ziele und Anforderungen einer forensischen Untersuchung

Die Ziele einer forensischen Untersuchung nach einem Sicherheitsvorfall bzw. eines Systemeinbruchs sind in der Regel folgende:

- Das Erkennen der Schwachstelle oder der Methode, die zum Systemeinbruch geführt haben könnte.
- Den entstandenen Schaden nach einem Sicherheitsvorfall zu ermitteln.
- Die Identifikation der/des Angreiferin/Angreifers.
- Die Sicherung der Beweise um weitere juristische Aktionen durchführen zu können.[29]

Um diese Ziele erreichen zu können, müssen folgende Fragen[2] beantwortet werden:

- Was ist geschehen?
- Warum ist es geschehen?
- Wer war daran beteiligt?
- Wann hat es stattgefunden?
- Wo hat es stattgefunden?
- Wie ist es passiert?

Diese Fragen sind sowohl für digitale als auch für analoge forensische Fälle identisch.[19]

Damit die Beweismittel vor Gericht Bestand haben, ist es wichtig die gewählten Methoden und Hilfsmittel zur Beschaffung der Beweismittel auf ihre Robustheit und Sinnhaftigkeit zu hinterfragen. Denn eine/ein Dritte/Dritter, muss den Tätigkeiten, die während der forensischen Untersuchung durchgeführt wurden, Glauben schenken können. Dabei ist zu beachten, dass diese/dieser Dritte eventuell nicht dasselbe Fachwissen wie die/der forensische Ermittlerin/Ermittler besitzt. Deshalb ist es wichtig, folgende allgemeine Anforderungen an die Methoden und Hilfsmittel zu beachten[29]:

Akzeptanz

Die angewandten Methoden und Schritte müssen in der Fachwelt allgemein akzeptiert und beschrieben sein. Verfahren oder Werkzeuge sollten idealerweise bereits von anderen forensischen Ermittlerinnen/Ermittlern eingesetzt worden sein bzw. in einschlägigen Publikationen oder Konferenzen positive Erwähnung gefunden haben. Das Einsetzen von neuen Methoden und Hilfsmittel, die kaum beschrieben wurden, ist nicht ausgeschlossen, jedoch sollte ein Nachweis über die Korrektheit dieser

[2] Auch bekannt unter dem Synonym 5W1H - What, Why, Who, When, Where and How

erfolgen.

Glaubwürdigkeit

Die Funktionalität und Robustheit der angewendeten Methoden, sollte bei Bedarf nachgewiesen werden können. Dies ist besonders wichtig, wenn komplexe Methoden und Werkzeuge eingesetzt werden, deren Wirkungsweise nicht plausibel erklärt werden können.

Wiederholbarkeit

Die verwendeten Hilfsmittel und Methoden müssen bei Anwendung von Dritten wiederholbar sein. Das heißt, wenn eine dritte Person, dieselben Schritte durchführt, werden die gleichen Ergebnisse generiert.

Integrität

Die digitalen Spuren, die während der forensischen Untersuchung sichergestellt werden, dürfen nicht unbemerkt verändert werden können. Die Integrität der Beweise muss zu jedem Zeitpunkt bewiesen werden können.

Ursachen und Auswirkungen

Durch die ausgewählten Methoden muss es möglich sein, logisch und nachvollziehbare Verbindungen zwischen Ereignissen, Beweisspuren und idealerweise Personen herzustellen.

Dokumentation

Jeder Schritt der forensischen Ermittlung muss entsprechend dokumentiert werden.[29] Die Beweismittelkette (Chain of Custody) ist ein wesentlicher Bestandteil des digitalen Ermittlungsprozesses. Damit die Beweise vom Gericht als gültig akzeptiert werden, muss dokumentiert werden wer, wann, wo, warum und wie in jeder Phase des digitalen Ermittlungsprozesses mit den Beweisen in Kontakt gekommen ist.[30] Die Beweismittelkette soll sicherstellen, dass die Beweise, die dem Gericht vorgelegt werden, original und authentisch bleiben und dass keine Manipulationen stattgefunden haben.[31]

2.5 Arten von netzwerkbasierten Spuren

Es gibt verschiedene Arten von netzwerkbasierten Spuren, die alle Vor- und Nachteile im Hinblick auf die forensische Analyse haben. In diesem Abschnitt werden die verschiedenen Typen kurz vorgestellt.[19]

2.5.1 Full Content Data

Es handelt sich hierbei um jede einzelne Information, die über ein Netzwerk (oder Netzwerke) läuft. Es wird nichts gefiltert, es werden exakte Kopien des gesamten Datenverkehrs (oft "Packet Captures"genannt,

abgekürzt PCAP) gespeichert.[19]

Eine vollständige Paketerfassung ist zwingend erforderlich, wenn untersucht werden soll, was in einem Netzwerk zu einem bestimmten Zeitpunkt passiert ist und wer tatsächlich an einer Aktivität beteiligt war, da die IP-Adresse des Computers einer/eines Verdächtigen allein aufgrund der dynamischen Natur von IP-Adressen nicht als Grundlage für forensische Untersuchungen dienen kann.[32]

2.5.2 Session Data

Sitzungsdaten sind die Zusammenfassung der Kommunikation zwischen zwei Netzwerkgeräten. Diese zusammenfassenden Daten werden auch als Konversation oder Flow bezeichnet. Sie bieten zwar nicht die Detailtiefe, die in vollständigen Packet-Capture-Daten zu finden ist, haben aber einige einzigartige Stärken, die für Forensikerinnen/Forensiker von großem Wert sind. In Bezug auf die Netzwerkforensik können Sitzungsdaten der/dem Ermittlerin/Ermittler Aufschluss über Fragen wie z. B. wer mit wem, wann, wie lange gesprochen hat geben, ohne dass der Inhalt der Konversation überhaupt betrachtet wird.[19]

2.5.3 Alert Data

Immer wenn der Netzwerkverkehr ein vordefiniertes Element von Interesse auslöst (z. B. ein bestimmtes Muster von Bytes oder Zählungen von Aktivitäten oder andere Merkmale), hat es die/der Analystin/Analyst mit Alarmdaten zu tun. Alarme werden in der Regel von Network Intrusion Detection Systems (NIDS) erzeugt.

Ein häufiges Problem bei fast allen Untersuchungen, bei denen Alarmdaten verwendet werden, ist, dass die/der Analystin/Analyst häufig mit falschen Alarmen (allgemein als "False Positives"bezeichnet) konfrontiert wird und daher bei der Interpretation der Daten besonders vorsichtig sein muss. Außerdem reichen Alarmdaten oft nicht aus, um zu entscheiden, ob ein bestimmtes Muster des Netzwerkverkehrs bösartig oder gutartig ist. Die/Der Ermittlerin/Ermittler benötigt mehr Kontext, um zu einer Schlussfolgerung zu gelangen.[19]

2.5.4 Statistical Data

Es gibt viele verschiedene Arten von statistischen Daten (manchmal auch als Metadaten bezeichnet) und viele nützliche Tools, um diese verschiedenen Datentypen zu erzeugen. Statistische Daten liefern der/dem Analystin/Analysten netzwerkbezogene Aspekte wie die Anzahl der in einer Paketspur enthaltenen Bytes, Start- und Endzeiten von Netzwerkkonversationen, Anzahl der verwendeten Dienste und Protokolle, aktivste Netzwerkknoten, am wenigsten aktive Netzwerkknoten, Ausreißer in der Netzwerknutzung, durchschnittli-

che Paketgröße, durchschnittliche Paketrate und so weiter. Sie kann daher auch als nützliche Quelle für die Erkennung von Anomalien dienen.[19]

2.6 Quellen von digitalen Spuren im Netzwerk

2.6.1 Router

Mit Routern können Unternehmen mehrere LANs zu einem Wide Area Network (WAN) verbinden. Dadurch bewältigen sie eine umfangreiche Menge an Datenverkehr. Ein wichtiges Beweisstück das Router enthalten ist die Routing-Tabelle. Diese Tabelle enthält Informationen über die gelernten Netzwerke, auf welchem Port diese erreichbar sind und die Bewertung des Weges (Metrik) zu diesen. Router können auch so konfiguriert werden, dass sie bestimmten Datenverkehr zwischen Netzwerken verweigern und Logs über erlaubten Datenverkehr und Datenflüsse führen. Eine weitere wichtige Quelle von digitalen Spuren, die Router liefern können sind NetFlow-Daten. NetFlow liefert Daten zu IP-Adressen, Ports und Protokollen des Netzwerkverkehrs.[33] Diese Daten können verwendet werden, um den Verkehrsfluss von verschiedenen Segmenten des Netzwerks zu bestimmen.[19] Zudem werden je nach konfiguriertem Log-Level verschiedene Messages (wie etwa Interface und Line Protocol Up/Down, Systemneustart, fehlgeschlagene Anmeldeversuche, Konfigurationsänderungen und Access-List Verstöße) gespeichert. Des Weiteren können mittels verschiedener ßhow"Befehle weitere Informationen (aktuelle Konfiguration, Access-Listen, Routing-Table, ARP-Statistiken, angemeldete User, Firmware, TCP/UDP-Statistiken und andere) für die Ermittlung liefern. Die effektivste Methode den Router zu erfassen und zu analysieren, ist die Erstellung eines Core Dumps. Ein Core Dump enthält das vollständige Speicherabbild des Routers zum Zeitpunkt seiner Erstellung.[3]

2.6.2 Switches

Switches sind über das gesamte Netzwerk verteilt. Dabei unterscheidet man zwischen Core-Switches, die den Datenverkehr von mehreren Netzwerksegmenten verarbeiten, und Edge-Switches, die den Datenverkehr für einzelne Segmente verarbeiten. Folglich durchläuft der Verkehr, der von einem Host ausgeht und das interne Netzwerk verlässt, eine Reihe von Switches. Bei Switches gibt es mehrere wichtige Punkte, die beachtet werden sollten. Der erste ist die CAM-Tabelle (Content Addressable Memory). Diese CAM-Tabelle ordnet die physischen Ports des Switches der Netzwerkschnittstellenkarte (NIC) auf jedem mit dem Switch verbundenen Gerät zu. Dies kann bei der Identifizierung möglicher bösartiger Geräte helfen, wie z.

[3] https://www.sans.org/blog/cisco-router-forensics/ [Zugriff am: 04.08.2021]

B. Access Points oder Systeme, die von einer/einem Angreiferin/Angreifer an das interne Netzwerk angeschlossen wurden. Der zweite Weg, auf dem Switches eine Untersuchung unterstützen können, ist die Erfassung des Netzwerkverkehrs.[33] Dabei wird oft das Feature Port-Mirroring bzw. SPAN-Port verwendet. Dies ist eine Methode, mit der eine Kopie aller Pakete von einem Switch-Port an einen anderen Switch-Port gesendet wird.[19] Zudem gibt es auch die Möglichkeit den Netzwerkverkehr mittels eines Netwerk-TAPs. Dies ist ein speziell entwickeltes Hardware-Gerät, das den Netzwerkverkehr dupliziert. Dabei bleibt der Netzwerk-TAP völlig transparent, was zur Folge hat, dass der dahinter liegende Sniffer im Netzwerk unsichtbar und nicht erreichbar ist. Darüber hinaus bietet der Switch ähnliche Funktionen wie beim Router, um an verschiedene Informationen zu gelangen. So bietet er die Möglichkeit mittels ßhow"Befehle oder durch Erstellung eines Core Dumps weitere Informationen zu bekommen. Neben dem Router ist auch der Switch ein wichtiger Lieferant von NetFlow-Daten.

2.6.3 Firewalls

Obwohl das Hauptziel von Firewalls darin besteht, einen Teil oder den gesamten Netzwerkverkehr abzuschirmen oder zu filtern, können Firewall-Logs reichhaltige Verkehrsinformationen liefern, die als Beweismittel verwendet werden können.[34] Firewalls der nächsten Generation enthalten zudem eine Vielzahl von Funktionen wie Intrusion Detection und Prevention, Web-Filterung, Data Loss Prevention sowie detaillierte Logs über den Datenverkehr. Häufig dienen Firewalls als Erkennungsmechanismus, der das Sicherheitspersonal auf potenzielle Vorfälle aufmerksam macht.[33]

2.6.4 Network Intrusion Detection and Prevention Systems (NIDS/NIPS)

Diese Systeme wurden gezielt entwickelt, um Sicherheitspersonal und Incident-Responder mit Informationen über potenzielle bösartige Aktivitäten in der Netzwerkinfrastruktur zu versorgen. Diese Systeme verwenden eine Kombination aus Netzwerküberwachung und Regelsätzen, um festzustellen, ob eine bösartige Aktivität vorliegt. IDS sind oft so konfiguriert, dass sie vor einer bestimmten bösartigen Aktivität warnen, während ein IPS potenzielle bösartige Aktivitäten erkennen, aber auch blockieren kann. In jedem Fall sind beide Arten von Plattformen ein ausgezeichneter Ort für Ermittlerinnen/Ermittler, um spezifische Spuren für bösartige Aktivitäten zu finden.[33] Der Nachteil dieser Systeme ist, dass sie durch Verschlüsselungs- oder Verschleierungstechniken umgangen werden können. Auch können durch exzessiven Netzwerkverkehr Schwachstellen oder eine hohe CPU-Auslastung ausgelösen werden.[19]

2.6.5 Web Proxy Server

Während der Web-Proxy ursprünglich hauptsächlich zur Beschleunigung der Netzwerkleistung (durch Zwischenspeichern von Dateien, auf die viele Benutzerinnen/Benutzer zugreifen) und zur Inhaltsfilterung (durch Blockieren unangemessener oder verdächtiger Websites) eingeführt wurde, hat es sich heutzutage zu einem Werkzeug entwickelt, das bei forensischen Untersuchungen eingesetzt werden kann. Proxy-Server liefern nicht nur die Historie des Web-Browsing der gesamten Organisation, speichern Kopien einer Webseite (für begrenzte Zeit), sondern können auch HTTPS-Verkehr entschlüsseln.[19]

2.6.6 Domain Controller/Authentification Server

Authentifizierungsserver, die die gesamte Netzwerkdomäne bedienen, sind der primäre Ort, an dem Ermittlerinnen/Ermittler Details über erfolgreiche oder erfolglose Anmeldungen, Manipulation von Anmeldedaten oder andere Verwendungen von Anmeldedaten finden können.[33]

2.6.7 DHCP Server

Die Verwendung von DHCP (Dynamic Host Configuration Protocol) ermöglicht die dynamische Zuweisung von IP-Adressen an Systeme im LAN. DHCP-Server enthalten oft Logs über die Zuweisung von IP-Adressen, die der MAC-Adresse der NIC des Hosts zugeordnet sind.[33] Damit die/der Ermittlerin/Ermittler den Spuren nachgehen kann, benötigt er den Zugang zu den Logs des DHCP-Servers. Mittels diesen Logs kann das System ausfindig gemacht werden, von dem der verdächtige Netzwerkverkehr stammt.[19]

2.6.8 Application Server

Eine breite Palette von Anwendungen ist auf Netzwerkservern untergebracht, wie etwa E-Mail- und Web-Anwendungen. Jede dieser Anwendungen kann Logs liefern, die für den jeweiligen Anwendungstyp spezifisch sind. Bei einer Ermittlung sind auch alle Logs von Interesse, die sich auf Remote-Verbindungen beziehen. Angreiferinnen/Angreifer wechseln oft von einem kompromittierten System zu Servern, um Zugriff auf vertrauliche Daten zu erhalten oder für andere Folgeaktivitäten.[33]

2.6.9 SIEM/SOC

Das Ziel eines SOC (Security Operations Center) ist die Echtzeitüberwachung und Identifizierung sicherheitsrelevanter Ereignisse von IT-Ressourcen des Unternehmens, einschließlich des IT-Netzwerks, der Perimeter-Verteidigungssysteme wie Firewalls und Intrusion Prevention Systemen, Anwendungsserver, Datenbanken

und Benutzerkonten. Jedes Asset kann mit einer Vielzahl von Sensoren überwacht werden und Log-Dateien der Aktivitäten führen. Das SOC empfängt Ereignisinformationen von den Sensoren und den Log-Dateien und löst Warnungen aus, die auf mögliches bösartiges Verhalten hinweisen, sowohl am Rande des Netzwerks als auch im Unternehmen. Security Information and Event Management (SIEM) Systeme sind ein wichtiges Werkzeug, das in SOCs eingesetzt wird. Sie sammeln Sicherheitsereignisse aus vielen verschiedenen Quellen im Unternehmensnetzwerk, normalisieren die Ereignisse in ein gemeinsames Format, speichern die normalisierten Ereignisse für forensische Analysen und korrelieren die Ereignisse, um bösartige Aktivitäten in Echtzeit zu identifizieren.[35]

2.7 Typen von Netzwerkforensischen Systemen

Garfinkel klassifiziert die netzwerkforensischen Systeme in zwei Typen:[36]

Catch-it-as-you-can

Alle Pakete, die einen bestimmten Verkehrspunkt passieren, werden aufgezeichnet und in den Speicher geschrieben. Die Analyse wird anschließend im Batch-Modus durchgeführt. Dieser Ansatz erfordert große Mengen an Speicherplatz.

Stop-look-and-listen

Jedes Paket wird rudimentär im Speicher analysiert und nur bestimmte Informationen werden für zukünftige Analysen gespeichert. Dies erfordert einen schnelleren Prozessor, um mit dem Tempo des eingehenden Datenverkehrs Schritt zu halten.

2.8 Wichtige Formate von digitalen Spuren

Tools für die Netzwerksicherheit und -überwachung sind nicht für forensische Untersuchungen ausgelegt. Deshalb ist es wichtig die gesamten Datenpakete zu erfassen und sie im Detail zu analysieren. Es gibt zwei Möglichkeiten, den Netzwerkverkehr zu erfassen. Pakete können in PCAP-Dateien erfasst werden, indem ein Packet Sniffer ausgeführt wird oder über Netzwerkflussdaten, sie werden von Routern oder Switches gesammelt.[20]

2.8.1 Paketerfassung

Ein Paket-Capture, allgemein als PCAP bezeichnet, ist eine Sammlung von Netzwerkpaketen. PCAPs enthalten Steuerinformationen und Nutzdaten, die auch als Payload des Pakets bezeichnet werden können. Die Paketsteuerungsinformationen enthalten die Daten, die für die Übermittlung der Payload des Pakets benötigt werden und umfassen Details wie die Quell- und Ziel-IP-Adressen, Ports und einen Zeitstempel.[37]

PCAP

Das PCAP-Dateiformat wird in der Netzwerk- und Sicherheitsbranche häufig für die Paketerfassung verwendet.[19] Die grundlegende Bibliothek des Dateiformates, nennt sich Libpcap und wurde zum "de facto-Standard der Netzwerkerfassung auf UN*X Systemen.[4]

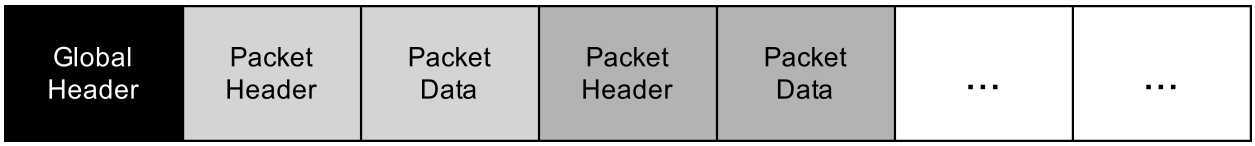

Abbildung 2.1: Aufbau einer PCAP-Datei[20]

Der allgemeine Aufbau einer Paketerfassung im PCAP-Format ist relativ einfach. Am Anfang der Datei befindet sich ein globaler Header, der Informationen für die gesamte Aufzeichnung enthält. Dieser Header kommt in der gesamten Aufnahme nur einmal vor. Danach folgt ein Paket-Header, der Informationen für das nachfolgende Paket enthält. Jeder Frame hat einen separaten Paket-Header. Die darauf folgenden Paketdaten sind eine Darstellung der Rohdaten die zum Zeitpunkt der Erfassung im Netzwerk vorhanden waren.[38] Abbildung 2.1 zeigt die generelle Struktur einer PCAP-Datei.

PCAPNG

Das PCAP-Dateiformat ist seit Jahren das Standard-Paketaufzeichnungsformat für viele Sicherheitstools, aber es ist bereits in die Jahre gekommen und es mangelt ihm an Funktionalität für modernere Anwendungen, insbesondere aus Sicht der Erweiterbarkeit. Das PCAP Next Generation (PCAPNG) Dateiformat ist derzeit als Internet-Engineering Task Force (IETF) Internet-Draft definiert und hat den Status „work in progress“, es soll das veraltete PCAP-Dateiformat ablösen.[39]

Die definierten Ziele des PCAPNG-Formats sind:

[4] https://wiki.wireshark.org/Development/LibpcapFileFormat [Zugriff am: 08.06.2021]

- **Erweiterbarkeit:** Es sollte möglich sein, neue Standardfunktionen dem Dateiformat im Laufe der Zeit hinzuzufügen. Dritte sollten in der Lage sein die in die Datei eingebetteten Informationen mit proprietären Erweiterungen zu ergänzen. Dabei sollten Tools, die neuere Erweiterungen nicht unterstützen, diese ignorieren können.
- **Portabilität:** Ein Paketmitschnitt muss alle Informationen enthalten, die zum Lesen von Daten notwendig sind - unabhängig von Netzwerk, Hardware und Betriebssystem des Rechners, der die Aufzeichnung durchführte.
- **Daten zusammenführen/anhängen:** Es sollte möglich sein, dass Daten am Ende einer gegebenen Datei hinzugefügt werden können. Dabei muss die resultierende Datei immer noch lesbar sein.[39]

PCAPNG ist derzeit das Standardformat für das weit verbreitete Tool, zur Analyse von Netzwerkprotokollen, Wireshark[5] und ist der Standard für Paketerfassungen. Die Unterstützung erstreckt sich auch auf mehrere der in Wireshark enthaltenen Tools.[38]
Ein Mitschnitt ist in Blöcken organisiert, die aneinander angehängt werden, um die Datei zu bilden. Alle Blöcke haben ein gemeinsames Format. Dies macht es einfach, eine Datei zu verarbeiten und nicht benötigte oder unbekannte Blöcke zu überspringen. Die Allgemeine Blockstruktur erlaubt es, bei Bedarf weitere Blöcke zu definieren. Ein Parser, der diese nicht versteht, kann deren Inhalt einfach ignorieren.[39] Der Aufbau einer PCAPNG-Datei wird in Abbildung 2.2 dargestellt.

Abbildung 2.2: Aufbau einer PCAPNG-Datei[20]

Der folgende Block muss mindestens einmal in jeder Datei erscheinen:

- **Section Header Block (SHB):** Er definiert die wichtigsten Merkmale der Aufzeichnungsdatei. Er ist vergleichbar mit dem Global Header der PCAP-Datei.[39]

Die folgenden Blöcke sind optionalen und können in einer Datei erscheinen:

- **Interface Description Block (ISB):** Er definiert die wichtigsten Merkmale der Schnittstelle(n), die für die Erfassung des Datenverkehrs verwendet werden.
- **Enhanced Packet Block (EPB):** Dieser Block enthält ein einzelnes aufgezeichnetes Paket oder

[5] https://www.wireshark.org/ [Zugriff am: 04.08.2021]

einen Teil davon. Er speichert neben dem aufgezeichneten Paket, Informationen wie Zeitstempel, Schnittstellen-ID, erfasste Länge und tatsächliche Paketlänge.

- **Simple Packet Block (SPB):** Er enthält ein einzelnes aufgezeichnetes Paket oder einen Teil davon mit nur minimalen Informationen darüber. Dies beschleunigt die Generierung der Aufzeichnungsdatei.
- **Name Resolution Block (NRB):** Definiert die Zuordnung von Adressen zu den dazugehörigen Namen. Dadurch wird eine DNS-Abfrage zu einem späteren Zeitpunkt vermieden, wenn sich diese geändert hat.
- **Interface Statistics Block (ISB):** Definiert, wie einige statistische Daten (z. B. verworfene Pakete usw.) zu speichern sind.
- **Custom Block (CB):** Er enthält herstellerspezifische Daten in einer portablen Form.[39]

2.8.2 Netzwerkflussdaten

Die Überwachung und Messung von Netzwerkverkehrsdaten auf der Flow-Ebene entstand aus der Anforderung, den Netzwerkverkehr mit einer Auflösung zu beobachten, die zwischen der Paketerfassung und der Paketzähler der Schnittstellen liegt. Netzwerkflussdaten sind heutzutage zu einem zentralen Bestandteil des Netzwerkmanagements und der Forschung geworden. Im Wesentlichen stellt ein Flow-Datensatz eine Verbindung zwischen zwei Sockets dar, wobei mindestens die Endpunktadressen, die Zeit und das Volumen der übertragenen Informationen erfasst werden. Dies bietet eine feinere Sicht auf den Datenverkehr als Zähler auf Schnittstellenebene, die vom Simple Network Management Protocol (SNMP) abgefragt werden und ermöglicht eine erhebliche Datenreduzierung im Vergleich zur Paketerfassung, was eine Skalierung auf große Netzwerke ermöglicht.[40]

Definition Flow

Ein Flow ist in RFC 7011[6] definiert als eine Menge von IP-Paketen, die während eines bestimmten Zeitintervalls einen Observationspunkt im Netzwerk passieren. Alle Pakete, die zu einem bestimmten Flow gehören, haben eine Reihe von gemeinsamen Eigenschaften. Jede Eigenschaft ist definiert als das Ergebnis der Anwendung einer Funktion auf die Werte von:

- Einem oder mehrerer Paket-Header (z. B. Ziel-IP-Adresse), Transport-Header (z. B. Ziel-Port-Nummer) oder Application-Header (z.B.: RTP-Header).
- Eine oder mehrere Eigenschaften des Pakets selbst (z. B. Anzahl der MPLS-Labels, etc...).

[6] https://datatracker.ietf.org/doc/html/rfc7011 [Zugriff am: 05.08.2021]

- Ein oder mehrere Felder aus der Paketbehandlung (z. B. Next Hop IP-Adresse oder Ausgangsschnittstelle)

Netflow

Das Konzept von NetFlow wurde erstmals 1996 bei Cisco entwickelt und wurde ursprünglich für die Performanceüberwachung, zum Debugging und zur Optimierung von Netzwerken eingesetzt. In diesen NetFlow-Aufzeichnungen befindet sich jedoch auch eine Fülle von Informationen, die für forensische Ermittlungen von großer Bedeutung sein können.[37]. Netzwerkelemente (Router und Switches) sammeln Flussdaten und exportieren sie an Kollektoren. Ein Flow (Fluss) ist definiert als eine unidirektionale Folge von Paketen mit einigen gemeinsamen Eigenschaften, die ein Netzwerkgerät durchlaufen.[41] Die grundlegende Ausgabe von NetFlow ist ein Flow-Record.[42] Diese Flow-Records liefern feinkörnige Messwerte und bieten eine Vielzahl an Einsatzmöglichkeiten, wie etwa zur Netzwerküberwachung, Intrusion Detection oder eben auch für forensische Ermittlungen.[41]
Es gibt ein paar Feinheiten, die Analystinnen/Analysten bei der Analyse von Flow-Records beachten müssen. Flussdaten sind unidirektional, das heißt sie enthalten einen Verkehrsfluss von der Quelle zum Ziel. Der "Rückweg", vom Ziel zurück zur Quelle (z. B. wenn eine/ein Benutzerin/Benutzer eine Webseite anfordert), wird als anderer Flow-Record dargestellt. Deshalb müssen bei der Analyse von bidirektionaler Kommunikation mehrere Flow-Records aggregiert werden.[37]

Die wichtigsten Daten die ein Flow-Record typischerweise beinhaltet, sind unter anderem:

- Zeitstempel
- Source und Destination IP-Adresse
- Source und Destination Port
- Ingress Interface
- Protokoll
- IP Type of Service
- Bytes- und Paket-Zähler[41]

Die Flow-Records können einer/einem forensischen Ermittlerin/Ermittler entscheidende Details zu einem Vorfall liefern. Bei einer Gesamtbetrachtung könnte sie/er anhand der Flussdaten eine einzige verdächtige IP-Adresse nehmen und schnell feststellen, wie viele Daten zwischen den Quellen und Zielen übertragen wurden. Darüber hinaus kann der Zeitstempel in den Flow-Records dabei helfen, den Zeitraum der Kom-

promittierung zu bestimmen.[37]

Die aktuelle Version von Netflow ist Version 9 und ist in dem RFC 3954[7] beschrieben. Diese Version basiert auf Templates, bietet Erweiterbarkeit und ist zukunftssicher.[20] Diese Version unterstützt neben anderen Funktionen IPv6, Virtual Local Area Networks (VLANs) und Multiprotocol Label Switching (MPLS).[43]

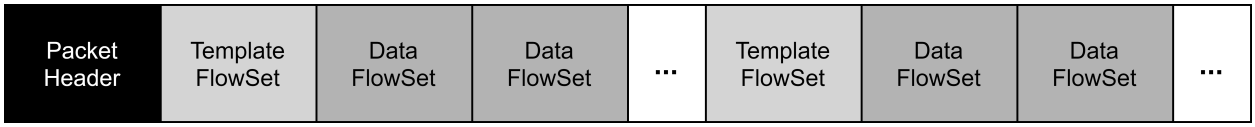

Abbildung 2.3: NetFlow Version 9 Exportpaket[20]

NetFlow besteht aus Exportpaketen, wie in Abbildung 2.3 dargestellt. Die Exportpakete werden von einem Gerät gebildet, das die NetFlow-Dienste aktiviert hat und werden an ein anderes Gerät (Netflow Kollektor) adressiert, welches diese sammelt und verarbeitet. Der erste Teil eines Exportpakets, der Packet Header, liefert grundlegende Informationen über das Paket, wie etwa die NetFlow-Version, die Anzahl der im Paket enthaltenen Datensätze und die Sequenznummerierung, wodurch verlorene Pakete erkannt werden können. Ein FlowSet ist ein allgemeiner Begriff für eine Sammlung von Datensätzen, die auf den Packet Header in einem Exportpaket folgen. Es gibt zwei verschiedene Arten von FlowSets:

- **Template FlowSet:** ist eine Sammlung von einem oder mehreren Template Records. Diese werden verwendet, um das Format der nachfolgenden Datensätze zu definieren, die in aktuellen oder zukünftigen Exportpaketen empfangen werden können.
- **Data FlowSet:** ist eine Sammlung von einem oder mehreren Data Records. Diese liefern Informationen über einen IP-Flow, der auf dem Gerät existiert, das ein Exportpaket erzeugt hat.[42]

IPFIX

Die Internet Engineering Task Force (IETF) hat die Arbeitsgruppe IP Flow Information Export (IPFIX) gegründet, um den Export von Flussdaten zu standardisieren und so die Interoperabilität bei der Messung von Netzwerkflüssen zu verbessern. IPFIX ist in RFC 7011[8] und 7012[9] beschrieben. Die Arbeitsgruppe wählte Cisco NetFlow v9 als die Basis von IPFIX. NetFlow v9, der Nachfolger des NetFlow v5-Protokoll, wurde mit den IPFIX-Anforderungen im Hinterkopf entwickelt.[40] Mehrere Verbesserungen zu NetFlow Version 9 wurden zu IPFIX hinzugefügt, darunter unternehmensdefinierte Felder[10], bidirektionale Flow-

[7] https://datatracker.ietf.org/doc/html/rfc3954

[8] https://datatracker.ietf.org/doc/html/rfc7011 [Zugriff am: 07.08.2021]

[9] https://datatracker.ietf.org/doc/html/rfc7012 [Zugriff am: 07.08.2021]

[10] https://datatracker.ietf.org/doc/html/rfc5610 [Zugriff am: 07.08.2021]

Exportfunktionen[11], einen Mechanismus für den effizienten Export von Flussdaten durch Zusammenfassen von Eigenschaften[12] und den Export von anonymisierten Daten mittels IPFIX (Status: Experimentell)[13][44][40]

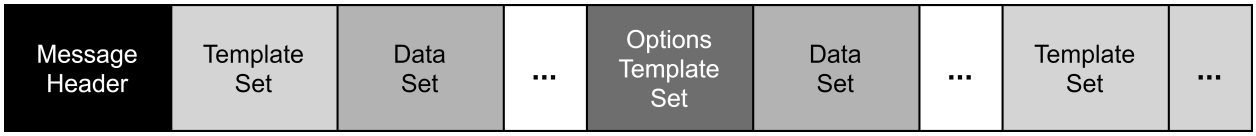

Abbildung 2.4: IPFIX Message[20]

In Abbildung 2.4 wird das Format einer IPFIX Nachricht dargestellt. Das Format ist selbstbeschreibend und ermöglicht Komprimierung, Indexierung und Suche, Fehlerbehebung, Authentifizierung, Vertraulichkeit, Integrität, Anonymisierung, Verschleierung, Auditierbarkeit von Sitzungen und Wiederholbarkeit.[20] Die ersten 16 Bytes einer Nachricht bildet der Message Header, dieser enthält eine Versionsnummer des Protokolls, Nachrichtenlänge, Exportzeit und eine Domänen-ID. Nach dem Header folgen ein oder mehrere Sets, die eine ID und eine variable Länge haben und von folgenden Typen sein können:[43]

- **Template Set:** enthält ein oder mehrere Templates, die das Layout von Data Sets beschreiben.
- **Data Set:** ist eine Sammlung von einem oder mehreren Data Records. Diese liefern Informationen über einen IP-Flow.
- **Options Template Set:** enthält Metadaten für die Flow-Kollektoren, wie zum Beispiel Daten der Steuerungsebene oder Daten, die für mehrere Datensätze gelten.[43]

Paketerfassung vs Netzwerkflussdaten

Während jedes Unternehmen mit einem Netzwerk technisch gesehen bereits Daten für netzwerkforensische Ermittlungen erzeugt, benötigen die meisten eine Möglichkeit, die Daten aus den Leitungen zu erfassen und zu speichern. Außerdem sollten Teams bei der Implementierung von Datenerfassungsplänen berücksichtigen, wie jeder Datentyp verwendet wird und wie lange sie die Daten aufbewahren können. Bestimmte Datentypen eignen sich beispielsweise nicht gut für die Paketerfassung, wie IP-Telefonie oder Streaming-Video.[37]

Netflow Protokolle eignen sich für den Einsatz in Hochgeschwindigkeitsnetzen und Bieten mehrere andere Vorteile im Vergleich zur regulären Paketerfassung. Sie sind weit verbreitet, hauptsächlich aufgrund ihrer

[11] https://datatracker.ietf.org/doc/html/rfc5103 [Zugriff am: 07.08.2021]
[12] https://datatracker.ietf.org/doc/html/rfc5474 [Zugriff am: 07.08.2021]
[13] https://datatracker.ietf.org/doc/html/draft-ietf-ipfix-anon [Zugriff am: 07.08.2021]

Integration in Geräte zur Paketweiterleitung, wie Router, Switches und Firewalls. Daher werden keine zusätzlichen Erfassungsgeräte benötigt, was die Flussüberwachung weniger kostspielig macht als die reguläre Paketerfassung.[43] Da bei der Paketerfassung in der Regel eigene Paketerfassungssysteme oder -software an Schlüsselstellen installiert werden müssen.[37]
Der Datenflussexport ist gut bekannt, da er unter anderem für Sicherheitsanalysen, Kapazitätsplanung, Troubleshooting und Netzwerküberwachung weit verbreitet ist. Der Export von Datenströmen ist in der Regel weniger datenschutzsensibel als der Export von Paketerfassungsdateien, da traditionell nur Paket-Header berücksichtigt werden.[43]
Netzwerkflussdaten können auch als Klartext-Log in ein SIEM exportiert werden, während PCAP-Dateien geparst werden müssen, um sie zu analysieren. Abhängig von der Dateiübertragung und der Netzwerkaktivität kann die Größe der PCAP-Dateien exponentiell ansteigen.[37] Trotz der Tatsache, dass der Export von Datenflüssen im Vergleich zur Paketerfassung die zu analysierende Datenmenge erheblich reduziert, kann die Größe von Flussdaten immer noch leicht mehrere Terabytes überschreiten.[43]

Einsatzgebiete

Es ist wichtig nicht nur die Stärken und Schwächen von NetFlow- und PCAP-Daten zu kennen, sondern auch, wie jedes Artefakt zur Analyse eingesetzt werden kann und sollte. Wie in Abbildung 2.5 dargestellt, ergänzen sich NetFlow- und PCAP-Daten nicht nur, sondern sind auch effektiver, wenn sie in einem zyklischen, iterativen Prozess verwendet werden, der das Verständnis für einen Vorfall weiter ausbaut.[37]

Indikatoren einer Kompromittierung des Netzwerks enthalten oft IP-Adressen und/oder Ports, was NetFlow zum perfekten Artefakt für die Erkennung bekannter böser Aktivitäten innerhalb eines Netzwerks macht. PCAP-Daten hingegen sollten verwendet werden, um eine Deep-Dive-Analyse des Netzwerkverkehrs durchzuführen. PCAP-Daten liefern die darin enthaltene Payload, die anschließend zur Malware-Analyse und/oder zur Generierung zusätzlicher Indikatoren verwendet werden kann. Diese Indikatoren sollten, wieder in die NetFlow-Analyse eingespeist werden. Dieser Prozess sollte wiederholt werden, bis die Untersuchung vollständig abgeschlossen ist. PCAPs können erneut abgespielt werden, um zusätzliche Erkenntnisse wie die Taktiken, Techniken und Prozeduren (TTPs) der/des Angreiferin/Angreifers zu erhalten.[37]

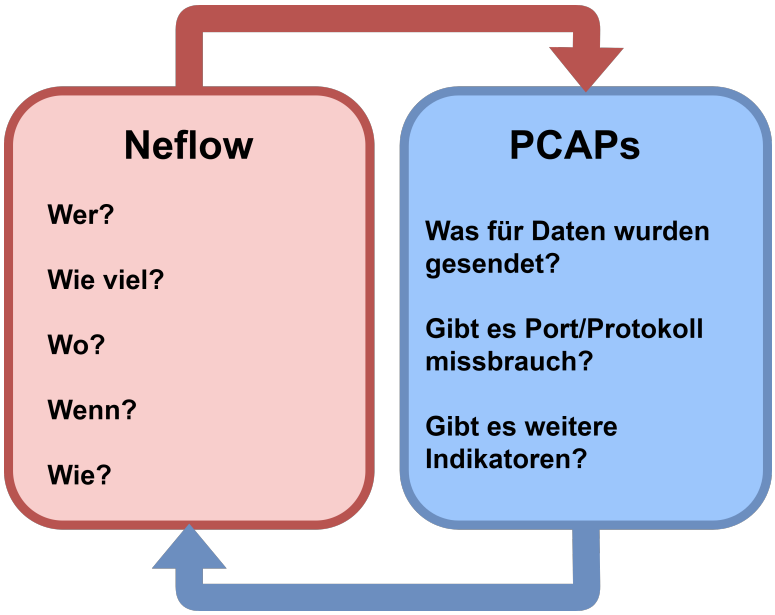

Abbildung 2.5: Komplementäre Aspekte von NetFlow und PCAP[37]

3 Stand der Forschung

3.1 Network-Forensik Prozessmodelle

Für den ordnungsgemäßen Umgang mit Beweismitteln und zur Minimierung von Fehlern bei forensischen Ermittlungen ist eine strukturierte Vorgehensweise bei der Durchführung erforderlich. Dieser strukturierte Ablauf wird als digitaler forensischer Prozess bezeichnet.[45] Dabei werden die notwendigen Schritte, meist getrennt in verschiedenen Phasen, definiert.[46] Bei forensischen Ermittlungen müssen nicht nur die digitalen Beweise selbst vor Gericht Bestand haben, sondern auch der angewandte Prozess und die verwendete Terminologie. Diese sollten gründlich sein und innerhalb der digitalen forensischen Gemeinschaft allgemein akzeptiert werden. Das primäre Ziel eines digitalen forensischen Prozessmodells ist es, der/dem Forensikerin/Forensiker bei der Erklärung zu helfen, wie bestimmte digitale Beweise beschafft und gefunden werden.[47]

Seit 2001 wurden verschiedene digitale forensische Modelle vorgeschlagen, die nicht nur einzelne Computer als Beweismittel betrachten, sondern auch vernetzte Umgebungen. Diese Modelle sind auf die digitale Forensik anwendbar und schließen die Netzwerk-Forensik in verallgemeinerter Form ein.[8]

Ren u. a. waren die ersten, die ein allgemeines Prozessmodell für die Netzwerk-Forensik vorschlugen. Das Prozessmodell besteht aus folgenden 6 Phasen:

- **Capture:** Erfassen der Daten aus der Datenquelle.
- **Copy:** Dabei wird eine Bit-zu-Bit Kopie der Originaldaten erstellt und auf ein Read-Only Medium oder Analysegerät kopiert.
- **Transport:** Die Daten werden auf ein forensisches Analysegerät übertragen.
- **Analysis:** Ist der umfassendste und anspruchsvollste Schritt im Prozess. Er beinhaltet Datenfilter, Meta-Analyse und integrierte Analyse.
- **Investigation:** Beschaffung von Informationen über die Quelle oder die Angreiferinnen/Angreifer.

- **Presentation:** Die Schlussfolgerung präsentieren und die Vorgehensweise bei der Erstellung der Schlussfolgerung darstellen.[48]

Pilli u. a. entwarfen das generische netzwerkforensische Prozessmodell, indem sie die Hauptmerkmale aus den bestehenden digitalen forensischen Prozessmodellen extrahierten und versuchten, diese in ihr vorgeschlagenes Modell zu integrieren. Sie haben die Incident-Response-Phase von Mandia u. a.[49] in ihrem Modell mit einer zweiseitigen Verknüpfung zwischen Detection- und Presentation-Phase integriert. Das Modell beinhaltet die neue Phase Detection, in der eine schnelle Auswertung erfolgt, um den mutmaßlichen Tatbestand zu überprüfen. In ihrer Arbeit wird auch ein Überblick über 33 Tools für die netzwerkforensische Analyse gegeben. Dabei wurden diese Kategorisiert und mit einem Satz beschrieben. Zudem wurde eine umfassende Übersicht über bestehende netzwerkforensische Prozessmodelle und Frameworks erstellt. Da diese wissenschaftliche Arbeit bereits 2010 veröffentlicht wurde sind viele der Tools veraltet beziehungsweise wurden diese umbenannt. Darüber hinaus bietet diese Arbeit keinen aktuellen Stand der bestehenden Prozessmodelle mehr.[8]

Kohn u. a. haben ein standardisiertes digitales forensisches Prozessmodell definiert, um den Untersuchungsprozess durch Befolgung der standardisierten Schritte zu unterstützen. Dabei haben sie mehrere bestehende Prozessmodelle analysiert und miteinander verglichen. Es wurden die wesentlichen Prozesse, die in einem digitalen forensischen Prozessmodell benötigt werden, identifiziert und abstrahiert. Zudem wurde die Terminologie bereinigt und standardisiert. Das Prozessmodell besteht aus folgenden Phasen: Preparation, Incident, Incident Response, Digital Forensic Investigation, Digital Forensic Investigation und Presentation.[47]

Liu u. a. beschreiben ein logikbasiertes netzwerkforensisches Prozessmodell unter Verwendung von Prolog (logische Programmiersprache), um die gesammelten Datenevidenzen zu analysieren und andere irrelevante Daten zu entfernen. Das Modell verwendet verschiedene Methoden, wie die Zuordnung von Spuren zu Systemschwachstellen, induktivem und abduktivem Schlussfolgern, um Angriffsszenarien zu rekonstruieren und so als Beweis vor Gericht präsentiert zu werden. Zudem werden Spuren wie False Positives, die unzulässig sind, ausgeschlossen und es werden Erklärungen für fehlende und zerstörte Beweise geliefert. Das auf Prolog basierende System verwendet bekannte Schwachstellen-Datenbanken und eine Anti-Forensik-Datenbank.[50]

Das vorgeschlagene netzwerkforensische Prozessmodell von Kaur u. a. zielt darauf ab, die/den Forensike-

rin/Forensikern zunächst zu autorisieren den Ermittlungsprozess durchzuführen. Nach erfolgreicher Autorisierung erfolgt das Sichern von Beweismitteln. Anschließend wird eine erste Bewertung vorgenommen. Hier gibt es eine Option, die Untersuchung abzubrechen, falls bestimmte Voraussetzungen nicht erfüllt sind, wie das Fehlen von vorinstallierten Sensoren, die den Netzwerkverkehr aufzeichnen. Für den Fall, dass eine weitere Untersuchung durchgeführt werden soll, wird eine Strategie geplant, um den gesammelten Netzwerkverkehr zu reduzieren und zu dokumentieren. Es wird eine weitere Analyse durchgeführt und eine Überprüfung auf weitere Verbesserungen vorgenommen.[51]

3.1.1 Zusammenfassung

Das Prozessmodell von Ren u. a. [48]gilt als erstes Netzwerk-Forensik Prozessmodell. Es ist sehr allgemein gehalten und beinhaltet nur die wichtigsten Phasen. Es sieht einem allgemeinen Prozessmodell der digitalen Forensik sehr ähnlich. Pilli u. a. [8] entwarfen ein Prozessmodell auf, das in der Wissenschaft oft referenziert wird. Neben der Incident-Response Phase und dem iterativen Prozess zwischen Analyse und Investigation Phase ist dieses Modell generisch, da es die Netzwerk-Forensik sowohl in Echtzeit als auch in Szenarien nach einem Angriff behandelt. Kohn u. a. [47] entwickelten ein Prozessmodell für die digitale Forensik, das zum Ziel hat, die Terminologie sowie den Prozess selbst zu standardisieren. Bei diesem Modell kommt die Incident-Response Phase vor der Investigation Phase und wird nicht etwa wie bei Pilli u. a. parallel abgearbeitet. Zudem gibt es keinen Prozess zur Verbesserung der Tools beziehungsweise des Vorgehens. Liu u. a. [50] entwarfen ein logikbasiertes Prozessmodell, das zum Hauptziel die Rekonstruktion von Angriffen hat. Dieses Modell kann nicht allgemein angewendet werden, da es auf das dazugehörige System, das auf Prolog basiert zugeschnitten ist. Das Prozessmodell von Kaur u. a. [51] ist vom Jahre 2018 und damit das Aktuellste. Jedoch haben sie bei ihrem Prozessmodell zwei essenzielle Punkte nicht beachtet. Zum einen gibt es keine Incident-Response Phase und zum anderen werden die Analyse und Investigation Phase nicht iterativ durchgeführt, was in der Praxis durchaus Sinn macht.

3.1.2 Generic process model for network forensics

In diesem Abschnitt wird auf das Prozessmodell von Pilli u. a. näher eingegangen, da es sich um ein allgemeingültiges Modell handelt und alle essenziellen Phasen einer netzwerkforensischen Ermittlung beinhaltet. Dieses Prozessmodell kann für netzwerkforensische Untersuchungen sowohl in Echtzeit als auch nach einem Angriff angewendet werden.[8]

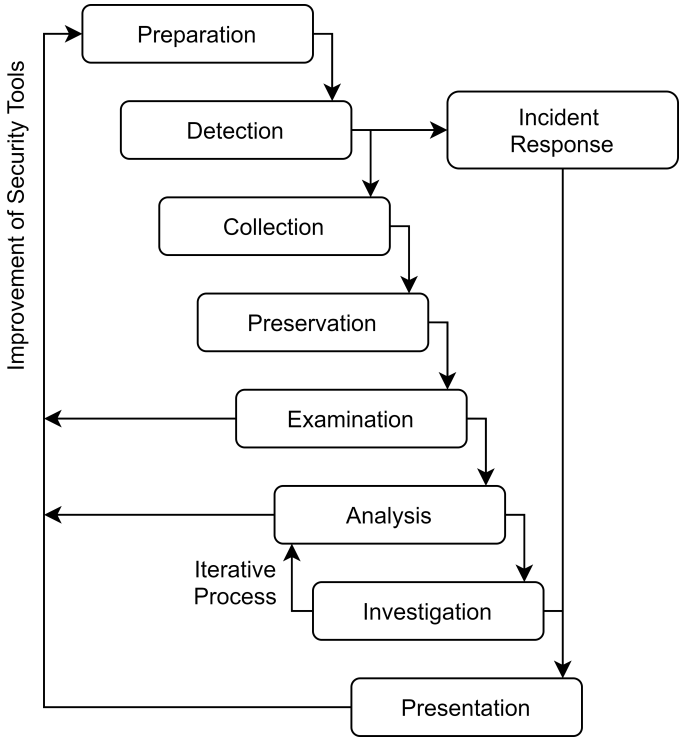

Abbildung 3.1: Generic process model for network forensics[8]

Preparation

Netzwerk-Forensik ist nur in Umgebungen anwendbar, in denen Netzwerksicherheits-Tools beziehungsweise Sensoren wie Intrusion Detection Systeme, Packet Analyzer, Firewalls, Software zur Messung des Datenverkehrsflusses an verschiedenen strategischen Punkten im Netzwerk eingesetzt werden. Daher wird in dieser Phase sichergestellt, dass dies der Fall ist. Zudem werden die erforderlichen Genehmigungen und rechtlichen Befugnisse eingeholt.

Detection

Die von verschiedenen Sicherheitstools generierten Warnungen, die auf eine Sicherheitsverletzung oder einen Richtlinienverstoß hinweisen, werden in dieser Phase beobachtet. Die festgestellten unberechtigten Ereignisse und Anomalien werden analysiert. Das Vorhandensein und die Art des Angriffs wird anhand verschiedener Parameter bestimmt. Eine schnelle Validierung wird durchgeführt, um den vermeintlichen Angriff zu bewerten und zu bestätigen. Dies erleichtert die wichtige Entscheidung, ob die Untersuchung fortgesetzt oder der Alarm als Fehlalarm zu ignorieren ist. Es sollten Vorsichtsmaßnahmen getroffen werden, damit die Beweise dabei nicht verändert werden. Diese Phase verzweigt sich in zwei Richtungen - Incident-Response und Collection.

Incident Response

Die Reaktion auf ein erkanntes Verbrechen oder Eindringen wird auf der Grundlage der gesammelten Informationen eingeleitet, um den Vorfall zu validieren und zu bewerten. Die eingeleitete Reaktion hängt von der Art des festgestellten Angriffs ab und richtet sich nach den Richtlinien der Organisation sowie den rechtlichen und geschäftlichen Einschränkungen. Es wird ein Maßnahmenplan zur Abwehr zukünftiger Angriffe und zur Behebung des bestehenden Schadens initiiert. Gleichzeitig wird die Entscheidung getroffen, ob die Untersuchung fortgesetzt und weitere Informationen gesammelt werden sollen. Eine ähnliche Reaktion ist nach der Investigation einzuleiten, wenn die gewonnenen Informationen bestimmte Maßnahmen zur Kontrolle und Abschwächung des Angriffs erfordern.

Collection

In dieser Phase wird der Datenverkehr von den Sensoren erfasst. Ein gut definiertes Verfahren mit zuverlässigen Hardware- und Software-Tools muss vorhanden sein, um ein Maximum an Spuren zu sammeln und dabei ein Minimum an Auswirkungen auf das Opfer zu verursachen. Diese Phase ist sehr wichtig, da sich die Verkehrsdaten schnell ändern und es nicht möglich ist, die gleiche Spur zu einem späteren Zeitpunkt zu erstellen. Die Menge der protokollierten Daten ist enorm und benötigt viel Speicherplatz und das System muss in der Lage sein, verschiedene Protokolldatenformate angemessen zu verarbeiten.

Preservation

Die in From von Logs und Protokollen gewonnenen Originaldaten werden auf einem Sicherungsgerät wie einem Nur-Lese-Medium gespeichert, dabei wird ein Hash aller Daten aufbewahrt. Eine Kopie der Daten wird analysiert, die ursprünglichen Netzwerkverkehrsdaten bleiben unangetastet. Dies geschieht, um rechtliche Anforderungen zu erfüllen. Damit wird gewährleistet, falls der forensische Prozess mit den Originaldaten wiederholt wird, er das gleiche Ergebnis liefert.

Examination

Die von verschiedenen Sensoren erhaltenen Spuren werden integriert und zu einem großen Datensatz zusammengesetzt, auf dem eine Analyse durchgeführt werden kann. Es gibt einige Probleme wie redundante Informationen und sich überschneidende Zeitzonen, die angepasst werden müssen. Es kann auch Fälle geben, in denen Alarme aus verschiedenen Quellen widersprüchlich sind. Dieser Prozess ist wichtig, damit keine wichtigen Informationen aus Quellen verloren gehen. Die gesammelten Spuren werden methodisch

durchsucht, um spezifische Indikatoren für das Verbrechen zu extrahieren. Es werden Angriffsattribute identifiziert, um möglichst genau nach relevanten Daten filtern zu können. Anschließend wird ein Feedback gegeben, um die Sicherheitstools zu verbessern.

Analysis

Die Indikatoren werden klassifiziert und korreliert, um mithilfe der vorhandenen Angriffsmuster wichtige Erkenntnisse abzuleiten. Statistische, Soft-Computing- und Data-Mining-Ansätze werden verwendet, um die Daten zu durchsuchen und Angriffsmuster abzugleichen. Einige der wichtigen Parameter beziehen sich auf Netzwerkverbindungsaufbau, DNS-Abfragen, Paketfragmentierung, Protokoll- und OS-Fingerprinting. Die Angriffsmuster werden zusammengesetzt, rekonstruiert und erneut abgespielt, um die Absicht und Methodik der/des Angreiferin/Angreifers zu verstehen. Anschließend wird wieder ein Feedback gegeben, um die Sicherheitstools zu verbessern.

Investigation

Ziel dieser Phase ist es, den Pfad von einem angegriffenen Netzwerk oder System über alle Zwischensysteme und Kommunikationspfade zurück zum Ausgangspunkt des Angriffs zu ermitteln. Die erhaltenen Paketaufzeichnungen und Statistiken werden für die Zuordnung des Angriffs verwendet. Diese Phase kann einige zusätzliche Merkmale aus der Analysephase erfordern, daher werden diese beiden Phasen iterativ durchgeführt, um zu einem Ergebnis zu kommen. Die Feststellung der Identität der/des Angreiferin/Angreifers, ist der schwierigste Teil des netzwerkforensischen Prozesses, da sich die/der Angreiferin/Angreifer mittels verschiedener Techniken wie IP-Spoofing oder Stepping-Stone-Angriff verschleiern kann. Diese Phase liefert Daten für die Reaktion auf Vorfälle und die Verfolgung der/des Angreiferin/Angreifers.

Presentation

Die Beobachtungen werden in einer für juristisches Personal verständlichen Sprache dargestellt und gleichzeitig werden die verschiedenen Verfahren erläutert, die zur Erlangung der Schlussfolgerung verwendet wurden. Die systematische Dokumentation ist ebenfalls enthalten, um die rechtlichen Anforderungen zu erfüllen. Die Schlussfolgerungen werden auch mit Hilfe einer Visualisierung dargestellt, so dass sie leicht zu erfassen sind. Dieser Prozess schließt die netzwerkforensische Analyse ab, da die präsentierten Informationen zur Verfolgung der/des Angreiferin/Angreifers führen. Der gesamte Fall wird dokumentiert, um Einfluss auf künftige Untersuchungen zu nehmen und um Feedback für den Einsatz und die Verbesserung von Sicherheitsprodukten zu liefern.

3.2 Andere wichtige Arbeiten

Hunt [52] beschreibt einige Anwendungsfälle der Netzwerk-Forensik und den Unterschied zur digitalen Forensik. Zudem listet er 27 Tools und kategorisiert diese. Diese Arbeit wurde im Jahre 2012 veröffentlicht und bildet somit keine aktuelle Liste von Tools in der Netzwerk-Forensik mehr.

Shrivastava [53] stellt ebenfalls 18 Tools für die netzwerkforensische Untersuchung vor. Diese Arbeit wurde 2016 veröffentlicht, doch auch hier werden Tools aufgelistet, die nicht mehr bestehen. Der Author beschreibt die grundlegende Funktion dieser Tools, kategorisiert diese aber nicht.

Brighi-Michele Ferrazzano [54] stellt verschieden Standards und Leitfäden für digitale Ermittlungen vor und vergleicht sie mit dem Guidelines on Digital Forensic Procedures for OLAF Staff. Dabei werden die Guidelines von Interpol, der RFC 3327 und weiter nicht miteinbezogen. Zudem wird untersucht welche methodischen Anforderungen an die Verarbeitung von digitalen Beweismitteln bestehen.

Nieto u. a. [55] beschreiben in ihrer Arbeit die Rolle der Privatsphäre in digitalen forensischen Ermittlungen und stellen verschiedene datenschutzfreundliche Ansätze vor. Zudem werden verschiedene Standards für die digitale Forensik vorgestellt. Dabei handelt es sich jedoch ausschließlich um ISO/IEC Standards.

Khan u. a.[56] beschreiben die Herausforderungen der Forensik in Software-Defined Networks. Diese Arbeit soll als Motivation und Anleitung zur Entwicklung von innovativen SDN-Forensik Standards und Frameworks dienen.
Spiekermann u. a.[7] beschreiben in ihrer Arbeit ebenfalls Herausforderungen der netzwerkforensischen Ermittlung in virtuellen Netzen und erarbeiten sechs Anforderungen.

Wilkens u. a.[57] beschreiben verschiedene Möglichkeiten zur Entschlüsselung von TLS-Traffic vor. Zudem stellen sie ihren Prototypen vor, der auf dem Prinzip der passiven TLS-Entschlüsselung basiert.

4 Herangehensweise

Dieses Kapitel gliedert sich in fünf Abschnitte. Im Abschnitt 4.1 werden verschiedene Standards und Guidelines die international beziehungsweise europaweit relevant und anerkannt sind zusammengetragen und allgemein beschrieben. Im Anschluss werden die Eigenheiten dieser hervorgehoben. Im Abschnitt 4.2 werden verschiedene Tools die bei der Netzwerk-Forensik zum Einsatz kommen recherchiert, kategorisiert und kurz beschrieben. Im Abschnitt 4.4 werden drei der im vorigen Abschnitt recherchierten Tools ausgewählt und getestet. Im Abschnitt 4.5 wird die Netzwerk-Forensik in virtuellen Netzen betrachtet. Dabei wird die Technologie Software-Defined Networking (SDN) inklusive VXLAN (Virtual eXtensible LAN) als Vertreter für andere virtuelle Netze herangezogen. Im Abschnitt 4.6 wird der Einfluss von TLS 1.3 auf die Netzwerk-Forensik betrachtet.

4.1 Standards und Guidelines

4.1.1 Einleitung

Während eines Gerichtsverfahren wird die Art und Weise, wie digitale Daten gesammelt oder gespeichert wurden, oft in Frage gestellt. Dies ist der Tatsache geschuldet, dass bei diesen Tätigkeiten notwendigerweise mit nicht greifbarem Material umgegangen werden muss. Nachlässigkeit, mangelnde Fachkenntnisse oder ungeeignete Methoden können dazu führen, dass die/der Richterin/Richter auf Daten zurückgreift, die falsch identifiziert oder unsachgemäß gesammelt oder gespeichert wurden, was zu fehlerhaften Gutachten und Berichten führt. Wie bei jeder Art von Beweismitteln, einschließlich digitaler Beweismittel, liegt die Beweislast für die Wahrhaftigkeit und Authentizität des Beweismittels bei der Partei, die es einführt.[54] Um dies zu gewährleisten ist die Verwendung von international anerkannten Standards und Guidelines bei der Beschaffung von digitalem Beweismaterial unabdingbar. Dieses Kapitel soll einen Überblick über verschiedene internationale sowie europaweit relevante und anerkannte Standards und Guidelines bieten. Dabei werden diese kurz beschrieben sowie der Zweck dieser hervorgehoben.

4.1.2 ISO/IEC

Seit 2012 haben die International Organisation for Standardization (ISO) und die International Electrotechnical Commission (IEC) technische Standards herausgegeben, die als nützliche Bezugspunkte für digitale Ermittlungen in allen Bereichen dienen. Diese Standards decken also nicht nur straf-, sondern auch zivilrechtliche Verfahren ab, ebenso wie Ermittlungen, die intern innerhalb von Behörden und privaten Organisationen durchgeführt werden und deren Ergebnisse daher möglicherweise nie in einem Gerichtssaal landen. ISO-Normen sind international, unabhängig vom geltenden Recht jedes einzelnen Landes und auch unabhängig von den Werkzeugen und Technologien, die zu ihrer Einhaltung verwendet werden können.[54]

ISO/IEC 27037:2012

Guidelines for identification, collection, acquisition and preservation of digital evidence[1] - Bietet eine Anleitung zum Identifizieren, Sammeln, Beschaffen, Handhaben und Schützen von digitalem forensischem Beweismaterial. Er enthält Leitlinien für die folgenden Geräte und Umstände: Digitale Speichermedien die in Standardcomputern verwendet werden, wie Festplatten, Disketten, Datengeräte mit ähnlichen Funktionen, Mobiltelefone, Personal Digital Assistants (PDAs), Personal Electronic Devices (PEDs), Speicherkarten, mobile Navigationssysteme, digitale Standbild- und Videokameras, Standardcomputer mit Netzwerkanschlüssen, Netzwerke die auf TCP/IP und anderen digitalen Protokollen basieren, sowie Geräte mit ähnlichen Funktionen. Der Standard richtet sich an Leiterinnen/Leiter von forensischen Laboratorien, DEFR (Digital Evidence First Responders), DES (Digital Evidence Specialists) und Incident-Response Teams.

ISO/IEC 27041:2015

Guidance on assuring suitability and adequacy of incident investigative method[2] - Bietet einen Leitfaden für Mechanismen, die sicherstellen, dass Methoden und Prozesse, die bei der Untersuchung von Informationssicherheitsvorfällen eingesetzt werden, für ihren Einsatzzweck tauglich sind. Er enthält Best Practices zur Definition von Anforderungen, zur Beschreibung von Methoden und zum Nachweis, dass die Implementierung von Methoden die Anforderungen erfüllt.

[1] https://www.iso.org/standard/44381.html [Zugriff am: 16.07.2021]
[2] https://www.iso.org/standard/44405.html [Zugriff am: 16.07.2021]

ISO/IEC 27042:2015

Guidelines for the analysis and interpretation of digital evidence[3] - Dieser Standard bietet einen Leitfaden für den Prozess der Analyse und Interpretation digitaler Beweise, die einen Teil des forensischen Prozesses bilden. Er legt ein generisches Framework fest, der Good Practices in diesem Bereich umfasst.

ISO/IEC 27043:2015

Incident investigation principles and processes[4] - Bietet einen allgemeinen Überblick über alle Prinzipien und Prozesse der Untersuchung von Vorfällen, ohne bestimmte Details innerhalb der einzelnen behandelten Gebiete vorzuschreiben.

ISO/IEC 27050:2019

Electronic discovery[5] - Dieser mehrteilige Standard betrifft Aktivitäten im Bereich der elektronischen Ermittlung, einschließlich, aber nicht beschränkt auf die Identifizierung, Bewahrung, Sammlung, Verarbeitung, Überprüfung, Analyse und Produktion von elektronisch gespeicherten Informationen[6], ein Fachbegriff, der im Wesentlichen forensische Beweismittel in Form digitaler Daten bezeichnet.

ISO/IEC 30121:2015

Governance of digital forensic risk framework[7] - bietet einen Leitfaden für leitende Organe von Organisationen, wie eine Organisation am besten auf digitale Ermittlungen vorbereitet werden kann, bevor diese stattfinden. Dieser Standard ist anwendbar für die Entwicklung von strategischen Prozessen und Entscheidungen in Bezug auf die Aufbewahrung, die Verfügbarkeit, den Zugriff und die Kosteneffizienz von digitalen Beweisen. Diese Internationale Norm gilt für alle Arten und Größen von Organisationen.

[3] https://www.iso.org/standard/44406.html [Zugriff am: 16.07.2021]
[4] https://www.iso.org/standard/44407.html [Zugriff am: 16.07.2021]
[5] https://www.iso.org/standard/78647.html [Zugriff am: 16.07.2021]
[6] Electronically Stored Information (ESI)
[7] https://www.iso.org/standard/53241.html [Zugriff am: 16.07.2021]

4.1.3 Andere

BSI IT-Forensik Leitfaden

Der Leitfaden[8] erläutert den Einsatz von IT-Forensik und eignet sich sowohl als Grundlagenwerk zur tiefergehenden Einarbeitung in die Thematik als auch als Nachschlagewerk für einzelne praxisbezogene Problemstellungen. Der Leitfaden richtet sich in erster Linie an Betreiberinnen/Betreiber von IT-Systemen, Administratorinnen/Administratoren und Sicherheitsverantwortliche. Die im Leitfaden aufgestellten Grundsätze und ein Teil des Inhalts sind jedoch auch für die Strafverfolgung in den Landeskriminalämtern relevant. Darüber hinaus ist der BSI-Leitfaden ein wichtiges Nachschlagewerk für Strafverteidigerinnen/Strafverteidiger zur Anfechtung digitaler Beweismittel. Die beschriebenen Prozesse und Vorgehensweisen werden anhand von verschiedenen Szenarien und Beispielen aus der täglichen Praxis umfassend und anschaulich beschrieben.

NIST

NIST SP 800-72[9] und 800-101 Rev 1[10] - Diese Richtlinien bieten grundlegende Informationen über digitale Geräte und forensische Tools, die für die Bewahrung, Erfassung, Untersuchung, Analyse und Berichterstellung von digitalen Beweisen entwickelt wurden. Sie konzentrieren sich in erster Linie auf mobile Geräte, einschließlich Personal Digital Assistants (PDAs), Smartphones und Tablets. Diese Guidelines richten sich an forensische Prüferinnen/Prüfer, Response-Teams, die einen Computersicherheitsvorfall behandeln und Sicherheitsbeauftragte von Organisationen, die mitarbeiterinnenbezogene/mitarbeiterbezogene Vorfälle untersuchen. Sie setzen Kenntnisse über traditionelle Methoden der digitalen forensischen Methoden voraus.[54]

NIST SP 800-86[11] - Diser Guide soll Organisationen bei der Untersuchung von Computersicherheitsvorfällen und bei der Fehlerbehebung in der IT helfen, indem er eine praktische Anleitung zur Durchführung von Computer- und Netzwerkforensik bietet. Der Leitfaden stellt die Forensik aus der Sicht der IT dar, nicht aus der Sicht der Strafverfolgung. Insbesondere beschreibt die Publikation die Prozesse für die Durchführung effektiver forensischer Aktivitäten und gibt Hinweise zu verschiedenen Datenquellen, einschließlich Dateien, Betriebssystemen (OS), Netzwerkverkehr und Anwendungen. Die Publikation soll nicht als allumfassen-

[8] https://www.bsi.bund.de/DE/Themen/Oeffentliche-Verwaltung/Sicherheitspruefungen/IT-Forensik/forensik_node.html [Zugriff am: 16.07.2021]

[9] https://www.nist.gov/publications/guidelines-pda-forensics-recommendations-national-institute-standards-and-technology [Zugriff am: 17.07.2021]

[10] https://www.nist.gov/publications/guidelines-mobile-device-forensics [Zugriff am: 17.07.2021]

[11] https://csrc.nist.gov/publications/detail/sp/800-86/final [Zugriff am: 17.07.2021]

de Schritt-für-Schritt-Anleitung für die Durchführung einer digitalen forensischen Untersuchung verwendet oder als Rechtsberatung ausgelegt werden. Ihr Zweck ist es, die Leserinnen/Leser über verschiedene Technologien und deren mögliche Verwendung bei der Reaktion auf einen Vorfall oder bei der Fehlersuche zu informieren.

European Network of Forensic Science Institutes

Das European Network of Forensic Science Institutes (ENFSI)[12] wurde 1995 gegründet. Heute umfasst es 37 europäische Länder, darunter die meisten EU-Mitgliedsstaaten. Als Netzwerk von Expertinnen/Experten widmet sich das ENFSI dem Zweck, Wissen und Erfahrungen auszutauschen und gegenseitige Vereinbarungen auf dem Gebiet der forensischen Wissenschaft, einschließlich des Bereichs der digitalen Beweise, zu treffen. Zu diesem Zweck ermutigt das ENFSI alle Laboratorien, die Teil des Netzwerks sind, sich an Best Practice und internationale Standards zu halten, um Qualität und Kompetenz zu gewährleisten. Das Best Practice Manual for the Forensic Examination of Digital Technology[13] bietet einen Rahmen für Verfahren, Qualitätsprozesse und Schulungsprozesse für forensische Untersuchungen in der IT. Es konzentriert sich auf die Bereitstellung von Leitlinien für forensische Labore, die internationale und lokale regulatorische Standards einhalten müssen.[54]

Council of Europe

Electronic Evidence Guide[14] - Der Zweck des Leitfadens ist es, Fachleute der Strafjustiz zu unterstützen und anzuleiten, wie sie elektronische Beweismittel so identifizieren und handhaben können, dass ihre Authentizität für die spätere Zulässigkeit vor Gericht gewährleistet ist. Er bietet unter anderem einen Überblick über die Arten von Problemen, die häufig beim Umgang mit elektronischen Beweismitteln auftreten und gibt Ratschläge, wie damit umzugehen ist. Dieser Leitfaden wurde für Länder erstellt, die gerade dabei sind, ihre eigenen Regeln und Protokolle für den Umgang mit elektronischen Beweismitteln zu entwickeln und zu etablieren. Die meisten bestehenden Leitfäden wurden für die Strafverfolgungsbehörden erstellt, aber dieser Leitfaden richtet sich an ein breiteres Publikum, einschließlich Richterinnen/Richtern, Staatsanwältinnen/-Staatsanwälten und anderen am Justizsystem Beteiligten, die über elektronische Beweismittel informiert werden müssen.

[12] https://enfsi.eu/ [Zugriff am: 17.07.2021]
[13] https://enfsi.eu/wp-content/uploads/2016/09/1._forensic_examination_of_digital_technology_0.pdf [Zugriff am: 17.07.2021]
[14] https://rm.coe.int/0900001680a22757 [Zugriff am: 18.07.2021]

Electronic Evidence in Civil and Administrative Proceedings[15] - Die Richtlinien zielen darauf ab, die Verwendung und Verwaltung von elektronischen Beweismitteln innerhalb von Rechtssystemen und in der Gerichtspraxis zu erleichtern. Sie sollen sicherstellen, dass spezifische Herausforderungen im Zusammenhang mit elektronischen Beweismitteln adressiert werden, wie die potenzielle Beweiskraft von Metadaten und die Leichtigkeit mit der elektronische Beweismittel manipuliert, verfälscht oder gelöscht werden können. Sowie die Einbeziehung einer dritten Partei bei der Erhebung und Beschlagnahme von elektronischen Beweismitteln. Die Richtlinien gelten für die Beilegung von Streitigkeiten sowohl in Zivil- als auch in Verwaltungsverfahren. Die Richtlinien richten sich an die Mitgliedsstaaten des Europarates und sollen bezwecken, dass die Staaten ihre Prozesse im Zusammenhang mit elektronischen Beweisen in Zivil- und Verwaltungsverfahren verbessern und stärken können.

ENISA

Electronic evidence - a basic guide for First Responders[16] - Dieser Leitfaden wurden mit dem Ziel entwickelt, die Zusammenarbeit zwischen Computer Emergency Response Teams (CERTs) und den Strafverfolgungsbehörden zu unterstützen und zu festigen. Sie sollen den CERTs bei ihrer Aufgabe helfen, die Strafverfolgungsbehörden bei der Sammlung von Beweisen zu unterstützen. Zu diesem Zweck integrieren der Guide die Fülle an Material, das zum Thema digitale Forensik existiert, um CERTs in einem für sie oft neuen Bereich eine Anleitung zu geben, wie sie mit potenziellen digitalen Beweisen und dem Prozess der Beweissammlung umgehen können. Der Leitfaden behandelt die verschiedenen Phasen, auf die First Responder bei der Durchführung von digitaler Forensik oder elektronischer Beweissicherung stoßen und beschreibt unter anderem, wie sie sich vor und während des Eintreffens am Tatort verhalten sollten. Mit diesem Guide kann ein CERT First Responder mit der Sammlung elektronischer Beweise auf angemessene Weise umgehen und eine gute Kommunikation mit den Strafverfolgungsbehörden führen.[54]

Introduction to Network Forensics Handbook V1.1[17] - Dieses Dokument bietet einen umfassenden Überblick über die Netzwerk-Forensik. Es dient als Schulungsmaterial und richtet sich in erster Linie an nationale, staatliche und andere Arten von CSIRTs (Computer Security Incident Response Teams), die ihre Fähigkeiten, Effektivität, Servicequalität und Zusammenarbeit mit anderen Teams und Stakeholdern verbessern wollen. Neben den Grundlagen und Good Practices bietet das Dokument drei verschiedene Anwendungs-

[15] https://rm.coe.int/guidelines-on-electronic-evidence-and-explanatory-memorandum/1680968ab5 [Zugriff am: 17.07.2021]
[16] https://www.enisa.europa.eu/publications/electronic-evidence-a-basic-guide-for-first-responders [Zugriff am: 17.07.2021]
[17] https://www.enisa.europa.eu/topics/trainings-for-cybersecurity-specialists/online-training-material/documents/introduction-to-network-forensics-handbook.pdf/view [Zugriff am: 17.07.2021]

fälle wo die Netzwerk-Forensik zum Einsatz kommt.

ACPO – Association of Chief Police Officers

Good Practice Guide for Computer-Based Electronic Evidence[18] - ist in erster Linie für Mitarbeiterinnen/-Mitarbeiter von Strafverfolgungsbehörden geschrieben, die möglicherweise mit digitalen Beweisen umgehen müssen. Dieser Leitfaden ist für die Verwendung bei der Wiederherstellung computergestützter elektronischer Beweismittel gedacht. Er ist kein umfassender Leitfaden für die Untersuchung dieser Beweismittel. Die hier gegebenen Ratschläge und Good Practices wurden formuliert, um das Personal bei der Bearbeitung von mutmaßlicher digitaler Kriminalität zu unterstützen und sicherzustellen, dass sie alle relevanten Beweise rechtzeitig und auf angemessene Weise sammeln.

European Anti-Fraud Office (OLAF)

Guidelines on Digital Forensic Procedures for OLAF Staff[19] - Diese vom Europäischen Amt für Betrugsbekämpfung (OLAF) herausgegebenen Leitlinien sind interne Regeln, die von den OLAF-Mitarbeiterinnen/ Mitarbeitern in Bezug auf die Identifizierung, den Erwerb, die Abbildung, die Sammlung, die Analyse und die Aufbewahrung digitaler Beweismittel zu befolgen sind. Sie orientieren sich an den technischen Normen von ISO und ACPO. Ziel dieser Leitlinien ist es, Regeln für die Durchführung digitaler forensischer Operationen in einer Weise aufzustellen, die die Integrität der Beweise und der Beweiskette sicherstellt, so dass die Beweise in Verwaltungs-, Disziplinar- und Gerichtsverfahren zulässig sind. In dieser Hinsicht unterscheiden sich die OLAF-Leitlinien von rein technischen Normen, da sie die technischen Anforderungen mit den spezifischen EU-Bestimmungen in Beziehung setzen, in denen sie ihre Rechtsgrundlage finden.

Interpol

Guidelines for Digital Forensics First Responders[20] - Dieser Leitfaden soll den digitalen Forensikerinnen/ Forensikern in den Strafverfolgungsbehörden bei der Durchsuchung und Beschlagnahme von elektronischen Beweismitteln Unterstützung und Empfehlungen bieten, um deren Integrität zu gewährleisten. Diese Leitlinien stützen sich auf den Electronic Evidence Guide des Council of Europe, auf das Digital Evidence

[18] https://www.7safe.com/docs/default-source/default-document-library/acpo_guidelines_computer_evidence_v4_web.pdf [Zugriff am: 17.07.2021]

[19] https://ec.europa.eu/anti-fraud/sites/default/files/guidelines_en.pdf [Zugriff am: 17.07.2021]

[20] https://www.interpol.int/content/download/16451/file/Guideline%20for%20First%20Responders%20Leaflet%20to%20be%20published%20on%20public%20INTERPOL.pdf [Zugriff am: 18.07.2021]

Collection Certificate Manual des National Center of Excellence in Cybersecurity in Spain (INCIBE)[21] und auf andere Leitfäden für bewährte Praktiken der Strafverfolgungsbehörden in Bezug auf die Beschlagnahme und den Umgang mit elektronischen Beweismitteln.
Global Guidelines for Digital Forensics Laboratories[22] - Die Leitlinien beschreiben die Verfahren für die Einrichtung und Verwaltung eines Labors für digitale Forensik (DFL) und enthalten technische Leitlinien für die Verwaltung und Verarbeitung elektronischer Beweismittel. Diese Leitlinien sind als Musterdokument zu verstehen, das von Ländern verwendet werden kann, die den Aufbau ihrer Kapazitäten im Bereich der digitalen Forensik in Erwägung ziehen. Die Leitlinien richten sich hauptsächlich an zwei verschiedene Gruppen. Die erste Gruppe sind die Strateginnen/Strategen und Managerinnen/Manager der digitalen Forensik, die Entscheidungen für die DFL treffen. Die zweite Gruppe umfasst das technische Personal, das täglich mit elektronischen Beweismitteln zu tun hat. Darüber hinaus können auch Staatsanwältinnen/Staatsanwälte, Richterinnen/Richter und Rechtsanwältinnen/Rechtsanwälte von diesem Dokument profitieren, um etwa den Prozess der digitalen Forensik besser zu verstehen.

RFC 3227

Im RFC 3227 - Guidelines for Evidence Collection and Archiving[23] werden wichtige Schritte und Schutzmaßnahmen bei einer Beweissicherung erläutert. Dieses Dokument spezifiziert eine Internet Best Current Practices für die Internet-Community und wurde im Jahre 2002 veröffentlicht. Der Zweck dieses Dokuments ist es, Systemadministratorinnen/Systemadministratoren Richtlinien für die Sammlung und Archivierung von Beweisen zu geben, die für einen solchen Sicherheitsvorfall relevant sind. So wird etwa die Reihenfolge definiert, mit der forensische Daten gesichert werden sollen. Dafür wird der Begriff Order of Volatility (Flüchtigkeitsreihenfolge) verwendet, dieser besagt, dass die Daten, die am flüchtigsten sind als erstes gesichert werden sollen.

4.1.4 Conclusio

In diesem Kapitel wurden die wichtigsten internationalen und europaweit gültigen Standards und Guidelines zusammengetragen und beschrieben. Auf Netzwerk-Forensik spezialisierte Standards oder Guidelines konnten nicht identifiziert werden. Als Außnahme gilt hier das Introduction to Network Forensics Handbook welches sich ausschließlich mit der Thematik Netzwerk-Forensik beschäftigt. Dabei gilt zu beachten, dass

[21] Dieser Handbuch konnte nicht ausfindig gemacht werden.
[22] https://www.interpol.int/content/download/13501/file/INTERPOL_DFL_GlobalGuidelinesDigitalForensicsLaboratory.pdf [Zugriff am: 18.07.2021]
[23] https://www.ietf.org/rfc/rfc3227.txt [Zugriff am: 18.07.2021]

sich alle Standards und Guidelines mit dem Thema digitale Beweismittel oder digitale Forensik auseinandersetzen, welche die Netzwerk-Forensik miteinschließt.

Während die ISO/IEC 27041:2015, 27043:2015, 27050:2019 und 30121:2015 alle Phasen des digitalen forensischen Prozesses abdecken, behandelt der ISO/IEC 27037:2012 Standard nur die Phasen Collection und Preservation. Der ISO/IEC 27042:2015 beschäftigt sich mit den Phasen Analysis, Investigation und Presentation. Die Internationalen ISO/IEC Standards sind kein Referenz- oder Normdokument für regulatorische und gesetzliche Sicherheitsanforderungen. Sie unterstreichen zwar die Bedeutung dieser Einflüsse, können sie aber nicht konkret benennen, da sie von dem jeweiligen Land, der Art des Unternehmens und anderen Faktoren abhängig sind. Die NIST und ENISA Guidelines gehen im Gegensatz zu den ISO/IEC Standards nicht nur auf organisatorische oder technische Prozesse ein, sondern auch sehr in technische Details, wie zum Beispiel das Verwenden von unterschiedlicher Tools. Diese Guidelines gehen ebenfalls nicht genauer auf regulatorische oder gesetzliche Anforderungen ein, können diese aber in Einzelfällen kurz anschneiden. Der RFC 3227 bietet lediglich einen kleinen Überblick über ein paar grundlegende Prinzipien, die es bei der Sammlung und Archivierung von digitalen Beweismitteln zu beachten gilt.

Die aktuelle Version des BSI IT-Forensik Leitfadens wurde 2011 aktualisiert. Der Leitfaden stellt neben dem eigens entwickelten forensischen Prozesses auch grundlegende Methoden vor. Dabei wird systematisch aufgezeigt und beschrieben, wie die einzelnen Methoden arbeiten und den forensischen Prozess in den jeweiligen Untersuchungsschritten unterstützen. Darüber hinaus werden für die jeweiligen Methoden exemplarische ausgewählte, konkrete Werkzeuge vorgestellt. Zudem werden in dem Leitfaden verschiedene Basisszenarien und komplexe Fallbeispiele behandelt. Auf regulatorische oder gesetzliche Anforderungen wird nicht näher eingegangen.

Der Electronic Evidence Guide vom Council of Europe bietet neben den Grundlagen (wie die verschiedenen Quellen von Beweismitteln, Prinzipien einer digitalen Untersuchung) eine allgemeine, aber umfassende Beschreibung der einzelnen Phasen des forensischen Prozesses. Es werden keine konkreten Tools oder Szenarien wie beim Leitfaden des BSI vorgestellt. Dafür geht dieser Guide mehr auf den juristischen Aspekt ein. Es werden verschiedene rechtliche Probleme bei der Beschaffung von digitalen Beweisen, wie etwa internationale Strafverfolgung, welche Rollen in einer digitalen Ermittlung berücksichtigt werden müssen, aufgezeigt. Im letzten Kapitel werden verschiedene Straf- und Zivilverfahren vorgestellt. Die Electronic Evidence in Civil and Administrative Proceedings Guidlines behandeln nur die Verwendung und Verwaltung elektronischer Beweismitteln in Rechtssystemen und in der Gerichtspraxis. Dabei bietet sie grundlegende Leitlinien zur Verwendung, Erhebung, Beschlagnahme, Übermittlung, Relevanz, Zuverlässigkeit, Lagerung und Aufbewahrung von elektronischen Beweismitteln.

Das Best Practice Manual for the Forensic Examination of Digital Technology des ENFSI behandelt die Erfassung und Analyse digitaler Systeme, einschließlich aller Aspekte des forensischen Prozesses von der Beschlagnahme bis zur Erstellung und Archivierung des Abschlussberichts. Dabei wird nicht Schritt für Schritt beschrieben, wie bestimmte forensische Prozesse abzulaufen haben, oder konkrete Tools vorgestellt, sondern es geht auf die abstrakten Prozesse, die damit verbundenen möglichen Risiken und Fehlern ein. Es werden Kriterien festgelegt für das Personal (Abteilungsleiterinnen/Abteilungsleiter/, technische Expertinnen/Experten, Analystinnen/Analysten, Assistentinnen/Assistenten), die Ausrüstung, das Referenzmaterial, die Arbeitsumgebung und die Archivierungspraktiken.

Der Good Practice Guide for Computer-Based Electronic Evidence von ACPO bietet einen umfassenden Überblick über digitale Geräte, die an einem Tatort bei forensischen Ermittlungen vorgefunden werden können und wie mit diesen umgegangen werden soll. Es werden auch die einzelnen Rollen (Ermittlerin/Ermittler, externe forensische Beraterinnen/Berater und andere) und deren Aufgaben beschrieben. Auch werden spezielle Themen wie die Handhabung von Bildern mit pedophilen Inhalten, Sicherstellung von Video-Beweismaterial oder die Untersuchung von Mobiltelefonen. Zudem werden Fragen vorgeschlagen, die beim Erstkontakt mit den Opfern die forensische Ermittlung beschleunigen sollen. Dieser Guide bietet Ermittlungsbehörden einen Einblick wie bei einer digitalen Ermittlung umzugehen ist und was dabei zu beachten ist. Dabei kann er als Checkliste gesehen werden, die bei einer forensischen Ermittlung abgearbeitet werden kann. Auf technische Details oder Tools wird nicht eingegangen.

Die Guidelines on Digital Forensic Procedures for OLAF Staff basieren auf den ISO/IEC 27037:2012 und APCO Standards und definieren technische sowie rechtliche Richtlinien. Neben den Anforderungen an das forensische Labor, beschreiben die Guidelines die Abfolge der Vorgänge, die eine/ein forensische/forensischer Ermittlerin/Ermittler durchführen muss, von der Identifizierung potenzieller digitaler Beweismittel über die Beschaffung und den Transport bis hin zu den Analysetätigkeiten. Ziel dieser Guidelines ist es, der/dem Ermittlerin/Ermittler die Grenzen der Technik aufzuzeigen sowie die Ziele und Sicherheitsvorkehrungen zu beschreiben, die das Gesetz von ihrer/seiner Arbeit verlangt. Auf technische Details sowie konkrete Tools wird in diesen Richtlinien nicht eingegangen.

Die Guidelines for Digital Forensics First Responders von Interpol basieren auf den Electronic Evidence Guide des Council of Europe, den Digital Evidence Collection Certificate Manual von INCIBE und anderen nicht erwähnten Richtlinien. Neben generellen Richtlinien was bei einer Durchsuchung und Beschlagnahmung von digitalem Beweismaterial zu beachten ist, werden sehr spezifische Prozeduren beschrieben. Zu den Themen zählen Smartphone, Server, Computer, Speichermedien, IoT Geräte, Spielekonsolen, Dronen, Autos, Schhiffsausrüstung und andere. Dabei wird sehr genau beschrieben was auf den verschiedenen Ge-

räte an Beweismittel gefunden werden kann, was bei der Sammlung der Beweismittel zu beachten ist und welche Tools unterstützung bieten können. Diese Guidelines werden regelmäßig aktualisiert und liegen aktuell in der Version 7-2 vor. Diese wurde im März 2021 veröffentlicht und ist somit das aktuellste Dokument. Die Global Guidelines for Digital Forensics Laboratories ebenfalls von Interpol beschreiben Verfahren für die Einrichtung und Verwaltung eines Labors für digitale Forensik. Dabei werden Anforderungen an die Räumlichkeiten, das Personal und dem Equipment definiert. Zudem werden Prozesse zur Verwaltung von forensischen Fällen definiert. Des Weiteren werden die einzelnen Phasen einer forensischen Ermittlung allgemein beschrieben.

4.2 Tools

4.2.1 Einleitung

Netzwerk-Forensik zielt darauf ab, Ereignisse aus der Netzwerkkommunikation zu verstehen bzw. zu rekonstruieren, was oft Expertenwissen erfordert wie etwa bei der Interpretation der Low-Level-Netzwerkprotokolle, um das große Ganze zu sehen.[16] Um einen Teil der Komplexität zu beseitigen, sind geeignete Tools unerlässlich.[58]

Tools zur Unterstützung in der Netzwerk-Forensik ermöglichen das Überwachen von Netzwerken, alle Informationen über verdächtigen Netzwerkverkehr zu sammeln und bieten Unterstützung bei der Auflösung von Cybercrime-Vorfällen. Sie helfen auch bei dem Schutz des geistigen Eigentums, der Betwertung der Netzwerkperformance, der Analyse von Datendiebstahl und des Missbrauchs von Ressourcen, zudem können sie bei der Vorhersage von möglichen Angriffszielen unterstützen.[59]

Forensik-Tools können den gesamten Netzwerkverkehr erfassen, ermöglichen es der/dem Benutzerin/Benutzer, wichtige Merkmale im Netzwerkverkehr zu entdecken und ihn nach seinen Bedürfnissen zu analysieren.[8] Der Angriffsverkehr kann wiederholt werden und die Bewegungen der Angreiferinnen/Angreifer können auf böswillige Absichten analysiert werden. Zudem ermöglichen sie die langfristige Speicherung von Aufzeichnungen des Netzwerkverkehrs zur schnellen Analyse. Diese Tools werden Network Forensics Analysis Tools (NFATs) genannt.[60]

Neben den NFATs gibt es noch eine Fülle von Network Security und Monitoring (NSM) Tools. Meist erfüllen diese nur einen bestimmten Zweck, wie etwa das Mitschneiden des Netzwerkverkehrs oder das Überwachen von Netzwerken. Viele dieser Tools wurden mit Blick auf Informationssicherheit und nicht auf die Beschaffung und Verarbeitung von Beweisstücken entwickelt. Daher haben solche Tools meist defizite aus forensischer Sicht.[16] Aber sie können bei verschiedenen Aufgaben in der forensischen Analyse helfen

und unterstützen.[8]
Die NSM-Tools wurden aufgrund ihres Aufgabenbereiches in verschiedene Kategorien unterteilt. Nachfolgend werden die verschiedenen Kategorien von Tools vorgestellt und der Einsatzzweck dieser erläutert. Zudem werden zu den jeweiligen Kategorien verschiedene populäre Tools kurz vorgestellt. Dabei wurde ein besonderes Augenmerk auf Open-Source Tools gelegt.

4.3 Auswahl der Tools

Die Auswahl der Tools geschieht anhand folgender Kriterien:

- Aktualität: Es werden nur Tools aufgenommen, die noch aktiv betreut werden. Das bedeutet Tools, die im Jahr 2019 kein Update mehr bekommen haben, werden nicht berücksichtigt.
- Lizenz: Es werden nur Open-Source Tools berücksichtigt.
- Umfang: Es wird darauf Wert gelegt, dass die Liste an Tools einen umfassenden Überblick bietet. Jedoch wird an dieser Stelle darauf hingewiesen, dass diese Liste an Open-Source Tools nicht als komplett zu betrachten ist.
- Wissenschaftliche Erwähnung: Besonders Wert wird auf Tools gelegt, die in wissenschaftlichen Arbeiten erwähnt werden.

4.3.1 Ausnahmen

Da es sich in der Kategorie NFATs um die Königsklasse der Netzwerk-Forensik Tools handelt wird hier ein Beispiel für ein kommerzielles Tool, namentlich NIKSUN NetDetector aufgelistet. Dieses Tool fand in verschiedenen wissenschaftlichen Publikationen[8][53][52] eine Erwähnung und wurde deswegen ausgewählt.

Eine weiter Ausnahme stellt die Kategorie SIEM dar. Hier werden neben den Open-Source alternativen, zwei kommerzielle Lösungen vorgestellt. Diese wurden ausgewählt, da sie sich im Gartner Quadrant[24] in den Top 3 befinden.

Aufgrund der Vielzahl an Open-Source Monitoring Tools, wurden hier nur fünf Tools vorgestellt. Die Tools wurden ausgewählt, da sie sich in der Liste des Gartner Quadrant[25] befinden und eine Open-Source äquivalent bereitstellen.

[24] https://www.gartner.com/reviews/market/security-information-event-management [Zugriff am: 01.04.2021]
[25] https://www.gartner.com/reviews/market/it-infrastructure-monitoring-tools [Zugriff am: 01.04.2021]

4.3.2 NFATs

In diesem Abschnitt werden einige populäre NFATs vorgestellt, die speziell zur Unterstützung von Forensikerinnen/Forensikern entwickelt wurden. Sie zielen darauf ab, die Analyse zu erleichtern, indem sie die Extraktion von Artefakten automatisieren und intuitive Benutzeroberflächen bereitstellen. Normalerweise haben diese Tools einen Top-Down-Ansatz, der die Analyse vereinfacht und Zeit spart.[5] Ein Network Forensics Analysis Tool sollte laut Cohen (Punkt 1-3) und Pluskal u. a. (Punkt 4-8) folgende Features mit sich bringen:

1. Effiziente Verarbeitung von großen Dateien.
2. Extraktion von High-Level-Informationen
3. Validierung der Ergebnisse
4. Verarbeitung von nicht standardisierten oder unvollständigem Datenverkehr
5. Robuste Datenentkapselung
6. Unterstützung für Overlay-Netzwerke
7. Identifikation von Anwendungsprotokollen

Xplico

Xplico[26] ist ein Open-Source Tool zur forensischen Analyse. Xplico ist in der Lage, die Anwendungsdaten eines Protokolls aus aufgezeichneten Paketen zu rekonstruieren. So können von einer PCAP-Datei Emails (POP, IMAP und SMTP) und verschiedene Anwendungsdaten (HTTP, SIP, FTP, TFTP, ...) extrahiert werden. Es kann der Netzwerkverkehr in Echtzeit von einer Netzwerkschnittstelle oder von einer PCAP-Datei decodiert werden. Das Tool verwendet Port Independent Protocol Identification (PIPI) zur Erkennung von Anwendungsprotokollen, unabhängig vom verwendeten Port. Xplico zerlegt die Daten auf Protokollebene und rekonstruiert und normalisiert sie für die Verwendung in Manipulatoren. Die Manipulatoren übersetzen den Code, korrelieren und aggregieren dann die Daten für die Analyse und stellen die Ergebnisse in einer visualisierten Form dar.[2]

Dshell

Dshell[27] ist ein auf Python basierendes Open-Source Network Forensic Analysis Framework, entwickelt von dem U.S. Army Research Laboratory. Dshell wird als flexibles und skalierbares Tool beworben, da es

[26] https://www.xplico.org/ [Zugriff am: 03.04.2021]

[27] https://github.com/USArmyResearchLab/Dshell [Zugriff am: 03.04.2021]

vollständig in Python geschrieben und modular aufgebaut ist. So können Plugins auf eigene Bedürfnisse angepasst, erweitert und neue geschrieben werden. Neben dem Mitschneiden des Netzwerkverkehrs und dem Auswerten von PCAP-Dateien, lassen sich auch forensische Analysen durchführen. Das Framework ermöglicht das Zusammensetzen von IPv4 und IPv6 Netzwerk-Streams, um Anfragen nachverfolgen zu können. Es beinhaltet auch eine Geolokalisierung und ein IP-to-ASN Mapping für jede Verbindung. Es ermöglicht die schnelle und effiziente Dekodierung und Analyse von Netzwerkdaten. Dshell wird mit verschiedenen Dekodern und Plugins bereitgestellt.

NetworkMiner

NetworkMiner[28] ist ein Open-Source Network Forensic Analysis Tool (NFAT) für Windows. Es kann als passiver Netzwerk-Sniffer/Packet-Capturing-Tool verwendet werden, um Betriebssysteme, Sitzungen, Hostnamen, offene Ports usw. zu erkennen, ohne dass dabei Datenverkehr im Netzwerk entsteht. NetworkMiner kann auch PCAP-Dateien für die Offline-Analyse parsen und übertragene Dateien und Zertifikate aus PCAP-Dateien rekonstruieren. Es wurde nicht entwickelt um den Datenverkehr im Netzwerk, sondern Daten über Hosts im Netzwerk zu sammeln. Die Hauptansicht der Benutzeroberfläche ist Host-zentriert (Informationen werden pro Host gruppiert) und nicht paketzentriert wie etwa bei Wireshark.

Netfox Detective

NFX Detective[29] ist ein Open-Source Tool für die forensische Analyse von Netzwerken, das Methoden zur Extraktion von Anwendungsinhalten aus der Kommunikation mit unterstützten Protokollen implementiert. Es wurde für das Betriebssystem Windows entwickelt. Die Funktionalität umfasst:

- Analyse von mehrere PCAPs in einer einzigen Sitzung.
- Unterstützt große PCAP-Dateien bis zu Gigabytes.
- Unterstützt verschiedenen Ansichten mit unterschiedlicher Detailstufen.
- Ermöglicht Filterung und Volltextsuche im erfassten Datenverkehr.
- Beinhaltet eine Sammlung von Parsern und Inhaltsextraktionsmethoden.

NFX Detective ist eine erweiterbare Plattform, die an individuelle Anforderungen angepasst werden kann.

[28] https://www.netresec.com/?page=networkminer [Zugriff am: 03.04.2021]
[29] https://github.com/nesfit/NetfoxDetective [Zugriff am: 03.04.2021]

NIKSUN NetDetector

Die kommerzielle NetDetector[30] Suite überwacht alle Daten, die durch das IP-Netzwerk fließen. Verwendet Deep Packet Inspection-Techniken, um alle Anwendungen, Sitzungen und Inhalte, die das Netzwerk durchlaufen, genau zu erkennen, zu klassifizieren und zu analysieren. Metadaten werden in Echtzeit über alle Inhalte erstellt, einschließlich E-Mail, FTP, HTTP und DNS. Das Tool warnt bei verdächtigem Datenverkehr auf Basis des Metadateninhalts, zeichnet alle Daten auf und indiziert gleichzeitig alle Daten in Echtzeit, um eine schnelle Suche zu ermöglichen. Es ist die einzige Sicherheitsüberwachungs-Appliance, die signaturbasierte IDS-Funktionalität mit statistischer Anomalieerkennung, Analytik und tiefer Forensik mit vollständiger Anwendungsrekonstruktion und Dekodierung auf Paketebene integriert.

4.3.3 Packet Analyzing and Sniffing Tools

Mit spezieller Software können Netzwerkdaten analysiert und der Netzwerkverkehr nach Typen unterteilt werden. Diese Tools werden Packet Analyzer oder auch Packet Sniffer bezeichnet. Sie werden verwendet um Pakete im Netzwerkverkehr mitzuschneiden und diesen anschließend zu analysieren.[62] Diese Tools bestehen aus zwei Hauptkomponenten. Zum einen das Packet Capture, dabei werden passiv alle Data-Link-Layer-Frames empfangen. Zum anderen aus dem Packet Analyzer, der die Rohdaten der erfassten Pakete aufgrund von RFC oder anderen Standards dekodiert und den Inhalt aller Felder innerhalb einer Protokollnachricht anzeigt.[63] Sie können unter anderem auch zur Fehlerbehebung in einem Netzwerk, zur Erkennung von Eindringlingen in das Netzwerk und zur Protokollierung des Netzwerkverkehrs für forensische Zwecke verwendet werden.[64] In diesem Abschnitt werden verschiedene Open-Source Tools vorgestellt.

TCPFlow

tcpflow[31] ist ein Tool, das Daten erfasst, die als Teil von TCP-Verbindungen übertragen werden. Jeder TCP-Flow wird in einer eigenen Datei gespeichert, um so die Protokollanalyse und Fehlersuche zu vereinfachen. tcpflow versteht TCP-Sequenznummern und rekonstruiert Datenströme korrekt, unabhängig von erneuten Übertragungen oder außerplanmäßiger Zustellung. Es gibt zudem die Möglichkeit Daten von TCP-Dumps zu extrahieren.

[30] https://www.niksun.com/product.php?id=112 [Zugriff am: 03.04.2021]
[31] https://github.com/simsong/tcpflow [Zugriff am: 04.04.2021]

NFDump

Nfdump[32] ist ein Toolset zum Sammeln und Verarbeiten von Netflow- und Sflow-Daten, die von Netflow/Sflow-kompatiblen Geräten gesendet werden. Das Toolset unterstützt Netflow v1, v5/v7, v9, IPFIX und SFLOW, sowohl IPv4 als auch IPv6. NfSen ist ein grafisches, webbasiertes Frontend für Nfdump. Mittels diesem Frontend kann zum Beispiel ein Zeitrahmen zusammen mit Filtern wie Quell- und Zieladressen, Ports und Protokollen ausgewählt werden.

TCPdump

TCPdump[33] ist ein bekannter Paket-Sniffer für die Kommandozeile. Er ermöglicht es TCP-Pakete, UDP- und ICMP-Pakete mitzuschneiden. Er bietet die Möglichkeit den mitgeschnittenen Netzwerkverkehr in der Kommandozeile anzuzeigen oder in einer Datei zu speichern. Die Ausgabe enthält den Zeitstempel, das Protokoll des Pakets, den Absender und Empfänger des Pakets mit ihrer jeweiligen Portnummer sowie spezifische Informationen über jedes Paket. Der Paket-Sniffer ist verfügbar für die meisten Unix-Systeme und wird von vielen Herstellern bereits im Grundsystem mitgeliefert.

ngrep

ngrep[34] ist ein Packet-Analyzer-Tool für die Kommandozeile. Das Besondere an diesem Tool ist die Fähigkeit, nach einem regulären Ausdruck in der Payload des Pakets zu suchen und die passenden Pakete auf der Konsole auszugeben. Es gibt die Möglichkeit den Netzwerkverkehr direkt von der Schnittstelle zu erfassen und diesen in eine PCAP-Datei zu speichern oder PCAP-Dateien zu lesen. Es werden unter anderem folgende Protokolle unterstützt: IPv4 und IPv6, TCP, UDP, ICMPv4 und ICMPv6, IGMP, SLIP, PPP, FDDI und Ethernet.

Kismet

Kismet[35] ist ein Wireless-Sniffer zum Entdecken von Funknetzwerken und Geräten. Das Programm arbeitet mit Wi-Fi-Schnittstellen, Bluetooth-Schnittstellen, einiger SDR-Hardware (Software Defined Radio) wie dem RTLSDR und anderer spezialisierter Erfassungshardware. Kismet sammelt passiv Pakete, ohne in den Netzwerkverkehr einzugreifen. Das heißt es ist in der Lage, ohne protokollierbare Pakete zu senden, die

[32] https://github.com/phaag/nfdump [Zugriff am: 04.04.2021]
[33] https://www.tcpdump.org/ [Zugriff am: 04.04.2021]
[34] https://github.com/jpr5/ngrep [Zugriff am: 04.04.2021]
[35] https://www.kismetwireless.net/ [Zugriff am: 04.04.2021]

Anwesenheit von drahtlosen Zugangspunkten als auch von drahtlosen Clients zu erkennen und diese miteinander zu verknüpfen. Weitere Features von Kismet sind, das Erkennen von aktiven Wireless-Sniffern innerhalb des Funknetzes sowie einiger bekannter Angriffe. Außerdem gibt es die Möglichkeit den erfassten Netzwerkverkehr zu protokollieren und zu speichern. Kismet unterstützt zudem Channel Hopping, um so viele Funknetze wie möglich aufzuspüren.

SiLK

SiLK[36] (System for Internet-Level Knowledge) ist eine Sammlung von Tools zur Traffic-Analyse, die vom CERT Network Situational Awareness Team (CERT NetSA) entwickelt wurde, um die Sicherheitsanalyse großer Netzwerke zu erleichtern. Die SiLK-Tool-Suite unterstützt die effiziente Sammlung, Speicherung und Analyse von Netzwerkflussdaten. Eine SiLK-Installation besteht aus zwei Kategorien von Anwendungen: dem Packing-System und der Analyse-Suite. Das Packing-System sammelt IPFIX, NetFlow v9 oder NetFlow v5 und konvertiert die Daten in ein platzsparenderes Format, indem es die gepackten Datensätze in dienstspezifischen binären Flat Files aufzeichnet. Die Analyse-Suite besteht aus Werkzeugen, die diese Dateien lesen und verschiedene Abfrageoperationen durchführen.

passiveDNS

PassiveDNS[37] dient zum passiven Sammeln von DNS-Einträgen. Dabei schneidet es den Datenverkehr von einer Schnittstelle mit oder liest eine PCAP-Datei und gibt die Antworten des DNS-Servers in einer Protokolldatei aus.

Scapy

Scapy[38] ist ein Python-basiertes Programm/Library zur interaktiven Paketmanipulation. Es ist in der Lage, Pakete von einer Vielzahl von Protokollen zu dekodieren oder sie zu manipulieren und wieder in das Netzwerk einzuspeisen. Zudem ist das Programm in der Lage die Netzwerk-Pakete zu erfassen und zu speichern oder von einer PCAP-Datei zu lesen.

[36] https://tools.netsa.cert.org/silk/ [Zugriff am: 05.04.2021]
[37] https://github.com/gamelinux/passivedns [Zugriff am: 04.04.2021]
[38] https://scapy.net/ [Zugriff am: 04.04.2021]

Stenographer

Stenographer[39] ist eine Lösung zum Erfassen von Paketen, die darauf abzielt, alle Pakete von der Netzwerkkarte schnell auf die Festplatte zu schreiben und dann einen einfachen und schnellen Zugriff auf kleine Teilmengen (< 1%) dieser Pakete zu ermöglichen. Zudem verwaltet es das Löschen der erstellten Dateien, wenn die Festplatte voll ist.

Arkime

Arkime[40] dient zum Erfassen, Speichern und Indexieren des Netzwerkverkehrs im PCAP-Format, um so einen schnellen Zugriff zu ermöglichen. Es verfügt über eine eigene Weboberfläche und nutzt Elasticsearch für das Durchsuchen der Datenbank.

PcapDB

PcapDB[41] ist ein Packet Capturing Tool, das entwickelt wurde, um den erfassten Netzwerkverkehr für eine schnelle Suche zu optimieren. Nach dem Mitschneiden des Verkehrs werden die Pakete zur Sortierung in Flows und zur Indexierung weitergegeben. Die sortierten Flows und Indexe werden anschließend auf die Festplatte geschrieben. Das Tool bietet ein Webinterface, die es der/dem Benutzerin/Benutzer ermöglicht, das Erfassungssystem zu verwalten, Suchen durchzuführen und Ergebnisse abzurufen. PcapDB besitzt die Fähigkeit, den Datenverkehr in Knoten über geografisch getrennte Netzwerke hinweg zu erfassen. Die Installation besteht aus einem Search Head und einem oder mehreren Capture Nodes.

Aircrack-ng

Aircrack-ng[42] ist eine komplette Tool-Suite zur Beurteilung der WiFi-Netzwerksicherheit. Zum einen kann der Netzwerkverkehr mitgeschnitten und gespeichert werden und zum anderen können auch Angriffe durchgeführt oder WPA/WEP Schlüssel geknackt werden.

Wireshark

Wireshark[43] ist ein weit verbreiteter Paket- und Protokoll-Analyzer. Das Programm stellt die Daten des aufgezeichneten Netzwerkverkehrs in Form einzelnen Pakete dar. Dafür werden die Daten in der grafischen

[39] https://github.com/google/stenographer [Zugriff am: 04.04.2021]
[40] https://github.com/arkime/arkime [Zugriff am: 04.04.2021]
[41] https://github.com/dirtbags/pcapdb [Zugriff am: 04.04.2021]
[42] https://www.aircrack-ng.org/ [Zugriff am: 04.04.2021]
[43] https://www.wireshark.org/ [Zugriff am: 04.04.2021]

Oberfläche übersichtlich und entsprechend ihren Protokolle aufbereitet. Es gibt auch die Möglichkeit mittels dem Utility TShark, den Datenverkehr in der Kommandozeile zu analysieren. Die Daten können dabei live von einer Netzwerkverbindung erfasst werden oder aus einer Datei mit bereits erfassten Paketen gelesen werden. Es können verschiedene Einstellungen, Timer und Filter gesetzt werden, um die Ausgabe des aufgezeichneten Datenverkehrs zu filtern. Zudem gibt es die Möglichkeiten eigene Plugins zu erstellen, um etwa unbekannte Protokolle zu analysieren.

4.3.4 Network Monitoring Tools

Die Überwachung eines Netzwerks ist für das Netzwerkmanagement von entscheidender Bedeutung und wird für viele kritische Aufgaben eingesetzt. Eine Hauptfunktion der Netzwerküberwachung ist die frühzeitige Erkennung von Trends und Mustern sowohl im Netzwerkverkehr als auch bei den Geräten.[65] Die Netzwerküberwachung ist eine schwierige und anspruchsvolle Aufgabe, die ein wichtiger Teil der Arbeit einer/eines Netzwerkadministratorin/Netzwerkadministrators ist. Netzwerkadministratorinnen/Netzwerkadministratoren sind ständig bestrebt, den reibungslosen Betrieb ihrer Netzwerke aufrechtzuerhalten.[66] Mittels Network Monitoring Tools lassen sich verschiedenste Statistiken über das Netzwerk überwachen, wie etwa die Netzwerkperformance, QoS, Delay, Bandbreite und den Status der verschiedenen Netzwerkdienste. Monitoring-Tools bieten eine Vogelperspektive auf das gesamte oder ein Segment des Netzwerks. Nachfolgend werden einige Open-Source Network Monitoring Tools vorgestellt.

Nagios Core

Nagios Core[44] ist ein in C geschriebenes Monitoring-Tool, das Host-Resourcen, Router/Switches, Netzwerkservices wie SMTP, HTTP, POP3 und Ping überwacht. Es kann Berichte generieren und bei Bedarf das zuständige Personal alarmieren. Das Tool bietet zudem die Möglichkeit eine Netzwerk-Hierarchie der Hosts zu erstellen. Durch Plugins lässt sich Nagios leicht erweitern. So sind beispielsweise mittels Plugins individuelle Uberprüfungungen von Services oder auch Visualisierungen möglich. Dieses Tool gibt es auch in einer kommerziellen Version.

Zabbix

Zabbix[45] ist eine Software, die zahlreiche Parameter eines Netzwerks sowie den Zustand von Anwendungen und Diensten sowie Server/Hosts überwacht. Das Besondere an dieser Software ist die verständliche

[44] https://github.com/NagiosEnterprises/nagioscore [Zugriff am: 05.04.2021]
[45] https://github.com/zabbix/zabbix [Zugriff am: 05.04.2021]

Bedienung über eine moderne Weboberfläche, mit der sämtliche Konfigurationen getätigt werden können. Es bietet Templates und ein Auto-Discovery Feature, das bei großen Netzwerken viel Zeit ersparen kann. Zudem gibt es noch die Möglichkeit, Berichte zu erstellen oder sich bei der Überschreitung von Schwellwerten benachrichtigen zu lassen.

Icinga 2

Icinga 2[46] ist ein Monitoring-System das auf Verfügbarkeitsüberwachung ausgelegt ist. Zu den Kernfunktionen gehören die Überwachen von Netzwerkdiensten (wie SMTP, POP3, HTTP, NNTP oder Ping) und von Host-Ressourcen. Ein einfaches Plugin-Design ermöglicht die leichte Erweiterung von Icinga 2. Es besteht aus mehreren Teilkomponenten, so bildet der Core das Grundgerüst. Er regelt die Abläufe, das heißt wie wird was und wann überwacht. Die Überwachung selbst erledigen die Plugins, diese werden vom Core aufgerufen. Die Weboberfläche nennt sich Incinga Web 2 und dient zur Darstellung der Ergebnisse.

Checkmk Raw Edition

Checkmk[47] ist ein Monitoring Tools, die auf Plugins basiert. Die Plugins (laut Website über 1900) ermöglichen die Erweiterung der Lösung um hersteller- und technologiespezifische Komponenten zu Überwachen. Diese ermöglichen unter anderem auch das Monitoring von Public-Cloud-Anwendungen und Kubernetes. Der Monitoring-Kern der Open-Source Variante bildet Nagios und ist auf bis zu 1000 Hosts skalierbar. Die Grundidee von Checkmk ist es, alle Informationen über eine Zielmaschine auf einmal abzurufen. Für jeden zu überwachenden Host wird Checkmk von Nagios nur einmal pro Zeitperiode aufgerufen. Er kontaktiert einen kleinen Daemon namens „mknagios“ auf der Zielmaschine, der alle relevanten Informationen über den Host ausgibt. Checkmk verarbeitet nun diese Informationen und extrahiert alle Elemente, die für die Überwachung konfiguriert wurden und prüft sie gegen konfigurierte Levels. Anschließend werden die Ergebnisse an Nagios gesendet. Einer der Hauptvorteile dieses Ansatzes ist eine massive Reduzierung der benötigten CPU-Ressourcen - sowohl auf dem Host und auf der Zielmaschine.

4.3.5 Network Intrusion Detection Systems (NIDS)

NIDS werden eingesetzt, um Unternehmen dabei zu helfen, potenzielle Angriffe auf das Netzwerk zu finden und zu verhindern, dass diese ausgeführt werden.[67] Bei NIDS gibt es grundsätzlich zwei verschiedene Ansätze: Signaturbasierte Erkennung und Anomalieerkennung. Der signaturbasierte Ansatz kommt in den

[46] https://github.com/Icinga/icinga2 [Zugriff am: 05.04.2021]
[47] https://github.com/tribe29/checkmk [Zugriff am: 10.08.2021]

meisten IDS zum Einsatz. Dabei lernen diese Systeme aufgrund von Angriffssignaturen, was ein Angriff ist. Sie sind hocheffektiv, bei der Erkennung von Angriffen, auf die sie programmiert sind. Sie können das Netzwerk jedoch nicht gegen neue Angriffe verteidigen, weil sie die Angriffe, die nicht in ihre Signaturenliste vorhanden sind, nicht erkennen können. Die Anomalieerkennung verwendet Instanzen des normalen Datenverkehrs, um Normalbetriebsprofile zu erstellen. Sie erkennt Anomalien als Aktivitäten, die von dieser Baseline abweichen. Solche Methoden können neue Arten von Netzwerkangriffen erkennen, die bisher nicht bekannt waren, erfordern jedoch ein Training zur Erstellung von Profilen. Darüber hinaus ist es nicht einfach, ein genaues und aktuelles Profil für den Normalbetrieb zu erstellen, was zu hohen Fehlalarmraten führen kann.[68] In diesem Abschnitt werden drei verschiedene Open-Source NIDS vorgestellt.

Snort

Snort[48] ist ein signatur-basiertes Network Intrusion Detection System (NIDS). Es analysiert die Pakete im Netzwerkverkehr anhand von Regeln (Signaturen), um Würmer, Schwachstellenausnutzung, Port-Scans und anderes verdächtiges Verhalten zu erkennen. Snort arbeitet hauptsächlich in drei Modi. Im Sniffer-Modus liest es nur die Netzwerkpakete und zeigt sie auf einer Konsole an. Im Logger-Modus protokolliert es die Pakete und speichert sie auf der Festplatte. Im Intrusion Detection-Modus analysiert es den Netzwerkverkehr anhand definierter Regelsätze. Diese Regelsätze definieren was erkannt wird und welche Maßnahmen ergriffen werden. Zudem gibt es noch die Möglichkeit Snort als NIPS (Network Intrusion Prevention System) zu verwenden, dabei wird der Netzwerkverkehr verworfen bzw. umgeschrieben, falls eine bösartige Aktivität festgestellt wird.

Suricata

Suricata[49] ist eine Open-Source-Engine zur Erkennung von Netzwerkbedrohungen, die Funktionen wie Intrusion Detection (IDS), Intrusion Prevention (IPS) und Network Security Monitoring bietet. Suricata inspiziert den Netzwerkverkehr mithilfe einer leistungsstarken und umfangreichen Regel- und Signatursprache und verfügt über eine leistungsfähige Lua-Skriptunterstützung zur Erkennung komplexer Bedrohungen. Viele Funktionen und Features sind vergleichbar mit denen von Snort. Doch Suricata unterscheidet sich unter anderem bei diesen Punkten: Es unterstützt Multi-Threading, Hardware-Beschleunigung, Hashing und File-Extraction. Zudem kann Suricata TLS/SSL-Zertifikate, HTTP-/DNS-Anfragen erfassen und protokollieren.

[48] https://www.snort.org/ [Zugriff am: 06.04.2021]

[49] https://suricata-ids.org/ [Zugriff am: 06.04.2021]

Zeek

Zeek[50] ist ein passives Open-Source-Framework zur Analyse des Netzwerkverkehrs. Es kann wie ein Network Intrusion Detection System (NIDS) eingesetzt werden, jedoch bietet es mehr Features. Einschließlich der Durchführung von Incident-Response, Forensik, Dateiextraktion und Hashing. Zeek unterscheidet sich von den anderen NIDS, da es sowohl ein signatur- als auch ein anomaliebasiertes NIDS ist. Seine Analyse-Engine wandelt den erfassten Datenverkehr in eine Reihe von Ereignissen um. Ein Ereignis könnte zum Beispiel eine Verbindung zu einer Website sein. Zudem wird ein Policy Script Interpreter bereitgestellt, der verwendet wird, um mit Ereignissen zu interagieren und zu verstehen, was diese Ereignisse in Bezug auf die Netzwerksicherheit bedeuten.

4.3.6 Security Information and Event Management (SIEM)

Logs spielen eine sehr wichtige Rolle in der Informationssicherheit. Sie sind sehr nützlich für die Analyse des Netzwerks, so dass eine Organisation vor einem Einbruch geschützt werden kann oder im Falle eines Einbruchs sich dessen bewusst wird und eine effektive Reaktion auf einen Vorfall bieten kann. Die Log-Verwaltung und -Überwachung selbst ist eine sehr anspruchsvolle Aufgabe. Es ist sehr schwierig, jedes einzelne Log von jeder Appliance zu sehen und im Falle eines Vorfalls rechtzeitig auf die Ursache zu schließen. Deshalb gibt es ein SIEM-Lösungen, die diese Aufgaben effektiv und effizient erledigt.[69]

Der Begriff SIEM wurde 2005 von Williams u. a. eingeführt. Das SIEM hat zwei Arten von Systemen ersetzt, die zuvor getrennt waren - Security Information Management (SIM) und Security Event Management (SEM) Systeme. Ersteres diente der langfristigen Speicherung, Analyse und dem Reporting, während letzteres Events in Echtzeit sammelte.[71] Allgemein vereint das SIEM-System die Funktionen von SIM und SEM in einem Sicherheitsmanagementsystem, das relevante Daten aus mehreren Quellen sammelt und korreliert, Berichte ausgibt, Abweichungen identifiziert und entsprechende Maßnahmen ergreift.[72]
SIEM-Systeme sind ein wichtiges Werkzeug in SOCs (Security Operation Centers). Sie sammeln Sicherheitsevents und Logs aus vielen verschiedenen Geräten/Appliances (wie Sicherheitsgeräte, Netzwerkinfrastruktur, Systeme und Anwendungen) des gesamten Netzwerks an einem zentralen Ort. Anschließend normalisieren sie diese Daten in ein gemeinsames Format, speicheren die normalisierten Events für die forensischen Analysen und korrelieren die Ereignisse, um bösartige Aktivitäten in Echtzeit zu identifizieren.[35]

[50] https://zeek.org/ [Zugriff am: 06.04.2021]

Letztendlich bietet eine SIEM-Lösung eine zentrale Sicht auf Aktivitäten innerhalb des Unternehmensnetzwerks, um in Echtzeit auf Bedrohungen reagieren zu können. Eventdaten werden mit kontextbezogenen Informationen über Benutzerinnen/Benutzer, Anlagen, Bedrohungen und Schwachstellen kombiniert, um zusätzliche Erkenntnisse zu gewinnen. Sie konsolidiert und analysiert die Eventdaten aufgrund von definierten Verhaltensregeln auf Abweichungen, um potenzielle Bedrohungen zu erkennen.[51]
Es gibt eine große Auswahl an SIEM-Plattformen, von Freeware-Lösungen bis hin zu Enterprise Security Management-Plattformen. Die meisten dieser Plattformen ermöglichen es Analystinnen/Analysten, eine gefilterte Suche und Log-Korrelations-Prüfungen durchzuführen.[73]

ELK Stack

Der ELK Stack[52] ist ein Paket von Open-Source-Technologien zum Sammeln, Durchsuchen, Analysieren und Visualisieren großer Datenmengen, die aus verschiedenen Quellen stammen. Er besteht aus drei Teilkomponenten: Elasticsearch, Logstash und Kibana. Elasticsearch ist eine Suchmaschine und Analytics Engine. Logstash ist eine serverseitige Datenverarbeitungspipeline, die Daten aus unterschiedlichen Quellen gleichzeitig ingestiert, sie umwandelt und dann an einen Speicherort, zum Beispiel an Elasticsearch, sendet. Kibana ermöglicht die Visualisierung von Daten durch Diagramme und Tabellen in Elasticsearch.

Alien Vault OSSIM

Alien Vault OSSIM[53] ist ein Open-Source SIEM Produkt und ist nur als serverbasierte Software verfügbar. Es hat keine eingebaute Berichtsunterstützung für Compliance-Initiativen. Es bietet drei Berichtsvorlagen, aber nicht speziell für Compliance-Berichte. Zu den Features zählen Asset Discovery, Bewertung von Schwachstellen, Intrusion Detection und Event-Korrelation. Es bietet keine Unterstützung für das Log-Management.

IBM QRadar

IBM QRadar[54] ist eine kommerzielle SIEM-Lösung und biete eine flexible Architektur, welche es ermöglicht die SIEM-Lösung entweder On-Premises oder in einer Cloud-Umgebung zu betreiben. Es erfordert jedoch Red Hat Enterprise Linux (RHEL) Server 6 als darunterliegendes Betriebssystem. Diese Lösung

[51] https://www.splunk.com/de_de/data-insider/what-is-siem.html [Zugriff am: 20.07.2021]
[52] https://www.elastic.co/de/ [Zugriff am: 20.07.2021]
[53] https://cybersecurity.att.com/products/ossim [Zugriff am: 20.07.2021]
[54] https://www.ibm.com/at-de/products/qradar-siem [Zugriff am: 20.07.2021]

bietet neben den klassischen Features eines SIEM zusätzlich noch Threat-Intelligence, eine hochskalierbare Datenbank und automatischen Normalisieren und Parsen von Protokollen.

Splunk

Splunk[55] ist eine sehr bekannte SIEM-Lösung die Daten egal von welchem Format oder Quelle für Benutzerinnen/Benutzer zugänglich und nutzbar macht. Splunk wirbt mit dem Begriff Data-to-Everything Plattform. Damit ist eine Datenanalyseplattform gemeint, die es ermöglichen soll, sämtliche Daten, die in einem Unternhemen anfallen, sichtbar, untersuchbar, überwachbar und analysierbar zu machen. Neben den klassischen Features von SIEM-Lösungen bietet Splunk risikobasierte Warnmeldungen, künstliche Intelligenz und Maschine Learning zur besseren Entscheidungsfindung. Es kann On-Premises oder in der Cloud betrieben werden.

4.4 Paxistest der Tools

4.4.1 Auswahl der Tools

Die NFATs bilden die Königsklasse unter den netzwerkforensischen Tools, deshalb werden aus dieser Kategorie zwei Tools ausgewählt. Um die gleichen Voraussetzungen zu schaffen, sollen beide Tools auf dem gleichen Betriebssystem funktionieren. Zudem müssen die Tools PCAP(NG)-Dateien analysieren können. Daher fällt die Auswahl auf NetworkMiner und Netfox Detective. Beide dieser Tools sind für das Betriebssystem Windows entwickelt worden und sind für die forensische Analyse von PCAP(NG)-Dateien geeignet. Um zu sehen wie sich im Vergleich zu den NFATs ein Packet Analyzer schlägt wird zusätzlich das Tool Arkime ausgewählt. Dieses Tool wird auf einer Ubuntu-Maschine getestet. Für eine kurze Beschreibung der Tools siehe Abschnitt 4.2.

Name	NetworkMiner (free edition)[56]
Version	2.7.1.0
Aktualität	18.06.2021
Lizenz	GNU General Public License version 2 (GPLv2)

Tabelle 4.1: NetworkMiner

[55] https://www.splunk.com/de_de [Zugriff am: 20.07.2021]

Name	Netfox Detective[57]
Version	2.1.0.0
Aktualität	18.10.2019
Lizenz	Apache License, Version 2.0

Tabelle 4.2: Netfox Detective

Name	Arkime[58]
Version	3.0.0
Aktualität	18.08.2021
Lizenz	Apache License, Version 2.0

Tabelle 4.3: Arkime

4.4.2 Vorgehensweise

Testumgebung

Um für die zwei NFATs die gleichen Bedingungen zu schaffen, wird die identische virtuelle Maschine erstellt. Arkime kann unter Windows nicht betrieben werden, daher wird für dieses Tool eine virtuelle Maschine mit dem Betriebssystem Ubuntu erstellt. Die Virtualisierungssoftware die dafür verwendet wird lautet VMware® Workstation 16 Pro Version 16.1.2 build-17966106. Die Spezifikationen der virtuellen Maschinen in der die Tools getestet werden, wird in Tabelle 4.4 für NetworkMiner und NFX Detective und in Tabelle 4.5 für Arkime dargestellt.

Betriebssystem	Windows 10 Enterprise Evaluation, x64-basierter Prozessor Version 21H1 Betriebssystembuild 19043.1055
Prozessor	2 Kerne
Installierter RAM	4 GB

Tabelle 4.4: Virtuelle Maschine Windows

Betriebssystem	Linux Ubuntu Version 20.04 LTS Kernel Release 4.4.0-31-generic
Prozessor	2 Kerne
Installierter RAM	4 GB

Tabelle 4.5: Virtuelle Maschine Ubuntu

Auswahl der Testdaten

Um die Fähigkeiten der Tools zu analysieren, wird eine große PCAP-Datei benötigt. Hierzu wird die case001.pcap[59] von der Forensik-Challenge The Case of the Stolen Szechuan Sauce[60] verwendet.

Name	case001.pcap
Größe	188 MB (MegaByte)
SHA-256	09abf49efea1852e047987d92907704d47f36d75f6c8056e2cafa6cc027791cb
Dateityp	PCAPNG

Tabelle 4.6: PCAPNG-Datei

Arbeitsschritte

Dieser Abschnitt bietet einen Überblick über die Arbeitsschritte, die bei jedem der Tools durchgeführt werden.

- **Installation/Konfiguration:** Dokumentation und Bewertung der Installation/Konfiguration.
- **Erzeugung eines Netzwerkmitschnittes:** Hier soll herausgefunden werden ob dies mit dem Tool möglich ist, wenn ja werden die einzelnen Schritte die nötig sind Dokumentiert.
- **Analyse der PCAP-Datei:** Dabei sollen verschiedene Fragen beantwortet werden siehe Tabelle 4.8.
- **Bewertung:** Die Tools werden anhand verschiedener Kriterien bewertet, siehe Tabelle 4.8.
- **Stärken/Schwächen:** Es werden die verschiedenen Stärken und Schwächen herausgearbeitet.
- **Conclusio:** Es wird das Ergebnis der Bewertung dargestellt.

[59] https://dfirmadness.com/case001/case001-pcap.zip [Zugriff am: 02.09.2021]
[60] https://dfirmadness.com/the-stolen-szechuan-sauce/ [Zugriff am: 02.09.2021]

Bewertungsschema und Kriterien für die Analyse

In der Tabelle 4.7 wird das Bewertungsschema und in der Tabelle 4.8 werden die Kriterien für die Analyse dargestellt.

Zeichen	Bedeutung	Punkte
++	Das Kriterum wird erfüllt und wurde getestet.	2
+	Das Kriterum wird laut Entwickler erfüllt.	2
~	Das Kriterium wird zum Teil erfüllt.	1
-	Das Kriterium wird nicht erfüllt.	0
o	Das Kriterium konnte nicht getestet werden und wird laut Entwickler nicht unterstützt	0

Tabelle 4.7: Bewertungsschema für die Analyse

Nr.	Kriterium	Beschreibung
1	Installation/Konfiguration	Dabei soll beantwortet werden, wie schwierig die Installation ist und ob die Installationsanleitung akkurat ist.
2	Erzeugung Netzwerkmitschnitt	Ist es möglich mit dem Tool einen Netzwerkmitschnitt im PCAP-Format zu erzeugen?
3	Import von PCAP(NG)-Dateien	Ist es möglich eine PCAP(NG)-Datei zu importieren?
4	Analyse der PCAP(NG)-Datei	Welche Protokolle kommen vor? Welche IP-Adressen kommunizieren miteinander? Wer erzeugt den meisten Traffic? Welche IP-Adresse weist die meisten Verbindungen auf? Können Credentials identifiziert werden?
5	Extraktion von Dateien	Können Dateien aus der PCAP-Datei extrahiert werden?
6	IPv6	Wird IPv6 unterstützt?
6	Encapsulation Protokolle	Werden Encapsulation Protokolle wie etwa VLAN oder GSE?
7	Application Protocol Identification	Wird das richtige Protokoll unabhängig vom Port erkannt?
8	Filter	Kann die Ansicht nach IP-Adressen, Protokollen, Domänen gefiltert werden?
9	Export	Können die analysierten Daten in verschiedenen Formaten (PCAP(NG), CSV, XML, JSON,...) exportiert werden?
11	Zeitzone	Kann die Zeitzone selbstständig konfiguriert werden?
12	Farbkodierung	Ist es möglich anhand von verschiedenen Faktoren (IP-Adresse, Protokoll,...) die Daten mit einer Farbe zu markieren?
13	Plugins	Kann das Tool mittels selbst erstellten Plugins erweitert werden?
14	Bedienung	Ist die Bedienung intuitiv und weitestgehend selbsterklärend?
15	GUI	Besitzt das Tool eine grafische Benutzeroberfläche?
16	Aktualität	Wird das Tool regelmäßig gewartet?

Tabelle 4.8: Kriterien für die Analyse

4.4.3 NetworkMiner (free edition)

Installation

Beim NetworkMiner handelt es sich um eine portable Anwendung, was bedeutet, dass keine Installation erforderlich ist. Die ZIP-Datei[61] wird heruntergeladen und anschließend entpackt. Danach kann die NetworkMiner.exe gestartet werden. Die Übersicht von NetworkMiner wird in Abbildung 4.1 dargestellt.

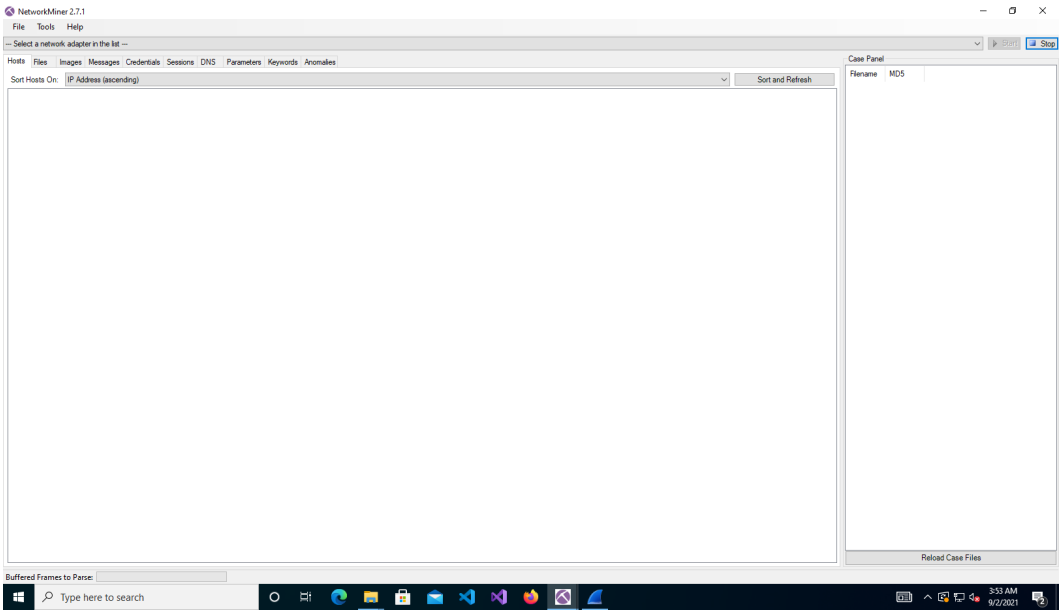

Abbildung 4.1: Übersicht von NetworkMiner

Übersicht der Funktionen

Die Funktionen der Register sind wie folgt:

- Hosts: Es werden alle identifizierten Hosts, mit OS-Fingerprinting und verschiedenen Statistiken, aufgelistet. Es Kann nach IP-Adresse, MAC-Adresse, Hostnamen, gesendete/empfangene Pakete/Bytes, Anzahl der offenen TCP-Ports, Betriebssystem und Router Hops Distanz sortiert werden.
- Files: Hier werden die extrahierten Files aufgelistet, diese werden automatisch im Ordner AssembledFiles abgespeichert. Zudem kann nach Keyword gefiltert werden.
- Images: Es werden die als Bilder identifizierten Dateien angezeigt.
- Messages: Hier werden E-Mails aufgelistet.
- Credentials: Es werden Benutzernamen, Passwörter und Hashes von HTTP, FTP, IMAP, POP3, SMTP,

[61] https://www.netresec.com/?download=NetworkMiner [Zugriff am: 02.09.2021]

Kerberos Hashes, MS SQL, NTLM Hashes und RDP aufgelistet.

- Sessions: Hier werden alle Sessions aufgelistet, mit Client/Server IP, Hostname, Port und Protokoll, sowie die Startzeit und die Framenummer.
- DNS: Hier werden die DNS-Queries aufgelistet.
- Parameters: Es werden alle Arten von Informationen angezeigt, die aus dem Netzverkehr extrahiert werden, wenn eine Kombination aus Name und Wert vorliegt.
- Keywords: Hier kann man den Mitschnitt nach eigenen Keywords durchsuchen.
- Anomalies: Es werden verschiedene Fehler oder Ungereimtheiten, die beim Parsen vorgekommen sind aufgelistet.

Erzeugung Netzwerkmitschnitt

Um einen Netzwerkmitschnitt zu erstellen, muss NetworkMiner als Administrator ausgeführt werden. Danach kann der Netzwerkadapter, bei dem der Netzwerkverkehr mitgeschnitten werden soll, ausgewählt werden. Anschließend wird der Mitschnitt mittels Betätigung des Start-Knopfes gestartet. Mittels Betätigung des Stop-Knopfes, wird das Mitschneiden beendet. Es wird automatisch eine PCAP-Datei in dem Ordner *NetworkMiner_2-7-1/Captures* erstellt. Die Bezeichnung der PCAP-Datei folgt diesem Schema: *NM_YYYY-MM-DDTHH-MM-SS*

Import von PCAP(NG)-Dateien

Um eine PCAP-Datei zu importieren, gibt es zwei Möglichkeiten. Entweder über *File->Open* oder über Drag & Drop. Die freie Version von NetworkMiner unterstützt keine PCAPNG-Dateien, daher ist dieses Kriterium nur zum Teil erfüllt. Das Laden der PCAP-Datei dauerte 3 Minuten und 21 Sekunden (Vergleichswert Wireshark: 10 Sekunden).

Analyse der PCAP(NG)-Datei

Da es sich bei den Testdaten um eine PCAPNG-Datei handelt, wurde diese mittels Wireshark in eine PCAP-Datei umgewandelt, damit die Analysefähigkeiten von NetworkMiner untersucht werden können.

1. Welche Protokolle kommen vor?

Um zu identifizieren welche Protokolle in dem Mitschnitt vorkommen, muss man in das Session-Tab wechseln. Hier werden die verschiedenen Sessions mit der Portnummer abgebildet. Die Zuordnung zum Protokoll funktioniert nur teilweise. Protokolle wie Kerberos und HTTP/S werden nur teilweise zugeordnet und bei

den anderen Portnummern findet überhaupt keine Zuordnung statt, siehe Abbildung 4.2. Zudem gibt es keine Übersicht, welche Protokolle vorkommen, sprich man muss die ganzen Sessions durchscrollen. Man kann zwar nach bestimmten Protokollen suchen aber diese Funktion ist sehr fehlerbehaftet. In der Abbildung 4.3 wurde nach dem Port 3306 (Meterpreter) in der Spalte C.port gesucht, die Abfrage bringt Resultate obwohl es keine geben sollte, da keine Meterpreter-Session in der PCAP-Datei vorkommt.

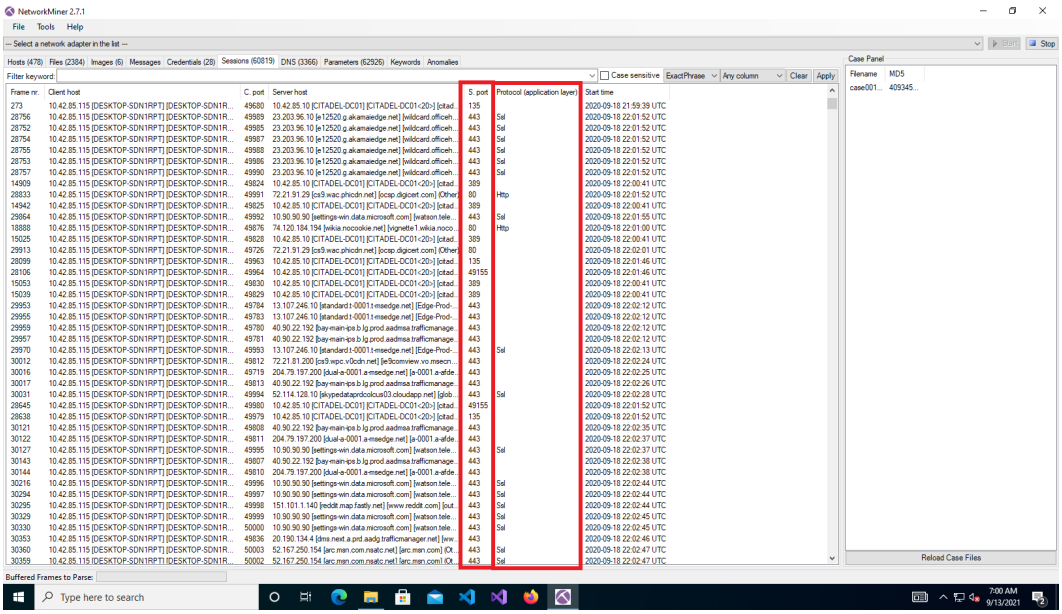

Abbildung 4.2: NetworkMiner Protokolle

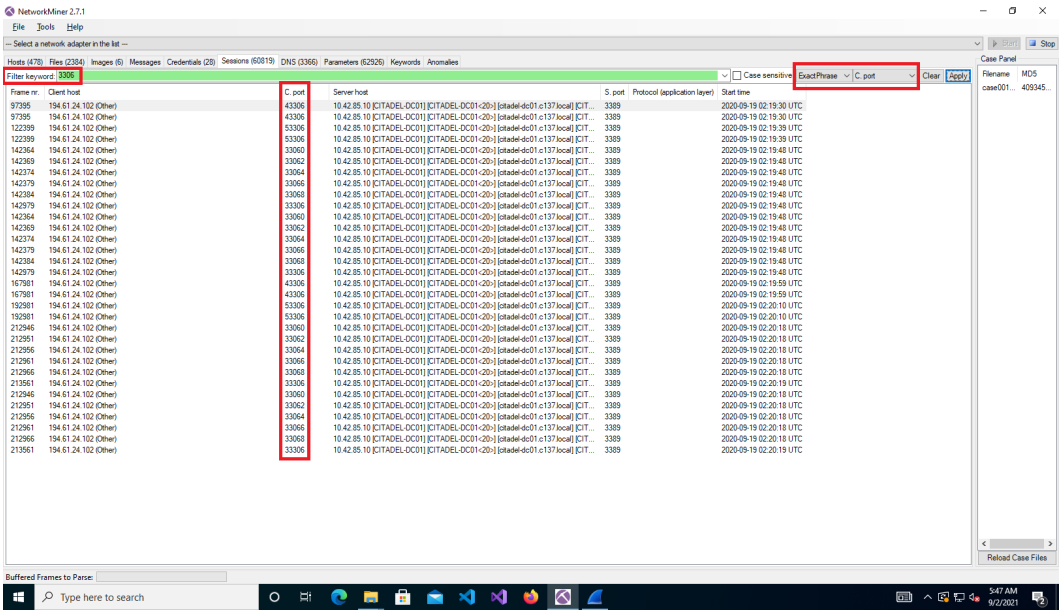

Abbildung 4.3: NetworkMiner Meterpreter

2. Welche IP-Adressen kommunizieren miteinander?

Dies wird ebenfalls in dem Session-Tab angezeigt. Es wird zwar angezeigt wer mit wem kommuniziert, jedoch ist es bei einer großen PCAP-Datei sehr unübersichtlich, hier würde eine grafische Darstellung oder ähnliches, Erleichterung bringen. Es gib die Möglichkeit nach einzelnen IP-Adressen zu filtern, dann werden alle Sessions dieser IP-Adresse angezeigt. Wenn man die Verbindungen einer bestimmten IP-Adresse herausfinden möchte, ist dies mit NetworkMiner sehr gut möglich.

3. Wer erzeugt den meisten Traffic?

Diese Frage ist durch die Sortierung nach gesendeten Bytes im Hosts-Tab sehr leicht zu beantworten, siehe Abbildung 4.4.

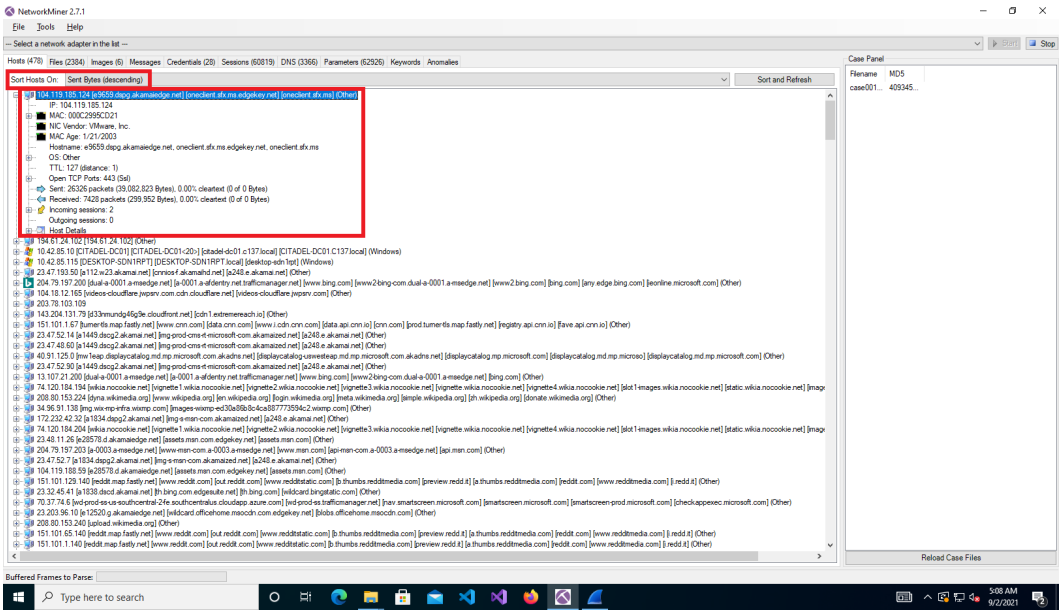

Abbildung 4.4: NetworkMiner Traffic

4. Welche IP-Adresse weist die meisten Verbindungen auf?

Es ist zwar möglich diese Frage mittels dem Session-Tab zu beantworten, jedoch ist dies eine sehr aufwendige Arbeit. Man muss sich zuerst alle IP-Adressen im Hosts-Tab notieren und anschließend im Session-Tab einzeln nach diesen IP-Adressen filtern und sich die Anzahl der Sessions notieren.

5. Können Credentials identifiziert werden?

Es wurden folgende Credentials automatisch identifiziert, siehe Abbildung 4.5.

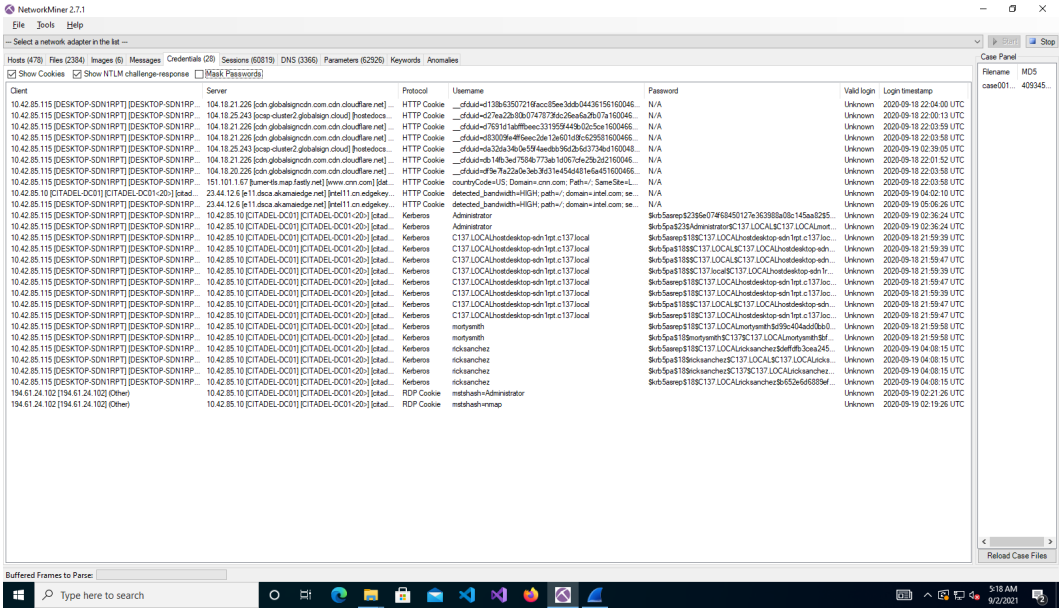

Abbildung 4.5: NetworkMiner Credentials

Extraktion von Dateien

Die Extraktion von Dateien wird von Networkminer automatisch durchgeführt. Dabei kann der Ordner, in dem die Dateien extrahiert werden, nicht geändert werden. Die Dateien werden in den Ordner *NetworkMiner/AssembledFiles* gespeichert. Falls Malware unter diesen Daten ist, birgt es ein Sicherheitsrisiko, dies sollte man beim Verwenden im Hinterkopf behalten. Die extrahierten Dateien werden im Files-Tab aufgelistet, siehe Abbildung 4.6. Zudem werden Bilder im Images-Tab angezeigt, siehe Abbildung 4.7.

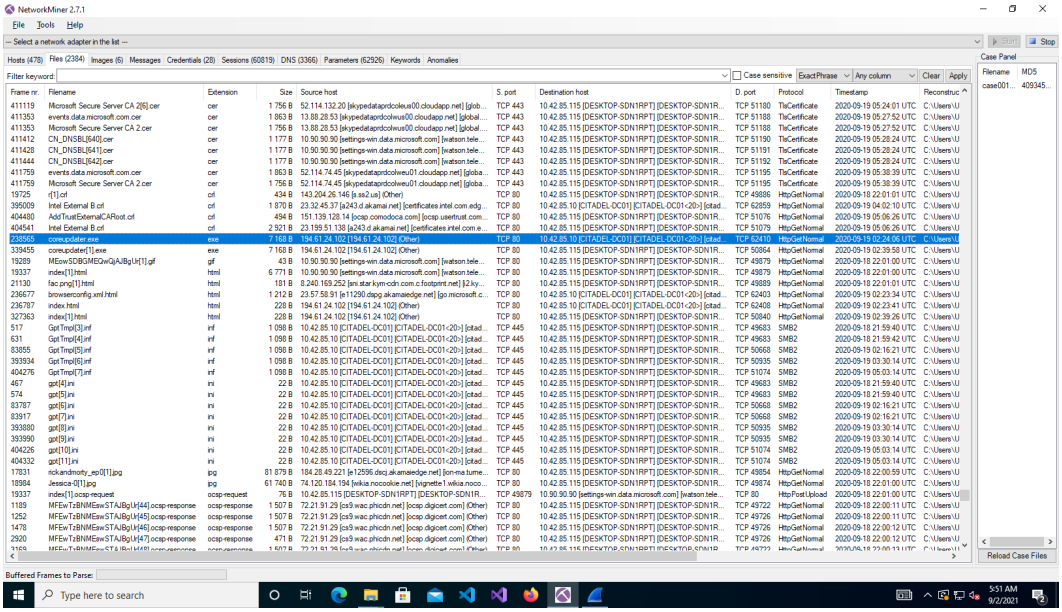

Abbildung 4.6: NetworkMiner Files

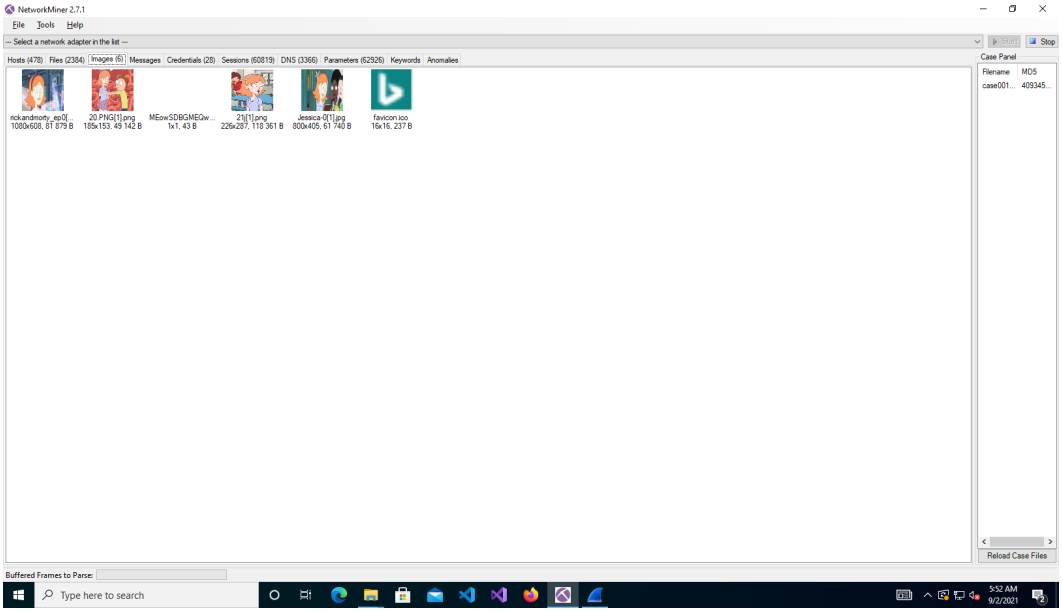

Abbildung 4.7: NetworkMiner Images

IPv6

IPv6 wird von NetworkMiner unterstützt.

Encapsulation Protokolle

Es werden GRE (Generic Routing Encapsulation), 802.1Q, PPPoE (Point-to-Point Protocol over Ethernet), VXLAN, OpenFlow, SOCKS (Internetprotokoll, das Netzwerkpakete zwischen einem Client und einem Server über einen Proxyserver austauscht.), MPLS (Multiprotocol Label Switching) und EoMPLS (Ethernet over MPLS) unterstützt.

Application Protocol Identification

Wird von der kostenlosen Version nicht unterstützt.

Filter

Die Ansicht kann je nach Register aufgrund von Keywords oder vorgegebenen Parameter gefiltert werden.

Export

Die kostenlose Version von NetworkMiner bietet keine Möglichkeit, die Daten zu exportieren. Es können auch keine Einträge gelöscht oder bearbeitet werden. Nur der selbst mitgeschnittene Netzwerkverkehr wird automatisch in eine PCAP-Datei gespeichert.

Zeitzone

Die Zeitzone kann in der kostenlosen Version nicht verändert werden.

Farbkodierung

Die kostenlose Version unterstützt keine Farbmarkierung.

Plugins

NetworkMiner ist nicht darauf ausgelegt, dass man das Tool mittels Plugins erweitert.

Bedienung

Die Bedienung ist selbsterklärend und intuitiv, jedoch bei einer großen Datenmenge unübersichtlich.

GUI

NetworkMiner besitzt eine grafische Benutzeroberfläche.

Aktualität

NetworkMiner liegt zum Zeitpunkt dieser Arbeit in der Version 2.7.1.0 vor. Diese wurde am 18.06.2021 veröffentlicht.

Bewertung

Nr.	Kriterium	Zeichen	Punkte
1	Installation/Konfiguration	++	2
2	Erzeugung Netzwerkmitschnitt	++	2
3	Import von PCAP(NG)-Dateien	∼	1
4	Analyse der PCAP(NG)-Datei	∼	1
5	Extraktion von Dateien	++	2
6	IPv6	++	2
6	Encapsulation Protokolle	+	2
7	Application Protocol Identification	o	0
8	Filter	++	2
9	Export	-	0
11	Zeitzone	-	0
12	Farbkodierung	-	0
13	Plugins	-	0
14	Bedienung	++	2
15	GUI	++	2
16	Aktualität	++	2
17	**Gesamtpunkte**		**20**

Tabelle 4.9: Bewertung der Analyse von NetworkMiner

Stärken/Schwächen

Da es sich bei NetworkMiner um eine portable Anwendung handelt, muss dieser nicht installiert werden und ist somit sofort einsetzbar. Ein weiterer Vorteil ist, dass Dateien aus dem Netzwerkmitschnitt automatisch extrahiert werden. Für größere PCAP-Dateien ist die Oberfläche unübersichtlich und die Analyse gestaltet sich oft als umständlich. Die kostenlose Version unterstützt einige wichtige Kriterien nicht, wie etwa das Einstellen der Zeitzone, das Exportieren der analysierten Daten oder die Unterstützung des PCAPNG-Dateiformates.

4.4.4 Netfox Detective

Installation

Netfox Detective[62] wird auf Github oder SourceForge als MSI-Datei bereitgestellt. Voraussetzung für die einwandfreie Installation ist Microsoft .NET Framework 4.6 oder höher.

Nach der Installation kann NFX Detective gestartet werden. Als erster Schritt, muss ein neuer Workspace erstellt werden. Der Workspace stellt eine Verzeichnisstruktur in einem Dateisystem dar, in der alle Daten, die sich auf den Workspace beziehen, gespeichert sind. Dieser kann entweder über *DETECTIVE->New Workspace...* oder über das Plus-Symbol unter Recent Workspaces erstellt werden, siehe Abbildung 4.8. Danach erscheint ein neues Registerblatt namens Investigation manager. Hier muss eine neue Investigation erstellt werden. Diese wird entweder über *DETECTIVE->New Investigation...* oder über das Plus-Symbol unter Investigations erstellt, siehe Abbildung 4.9.

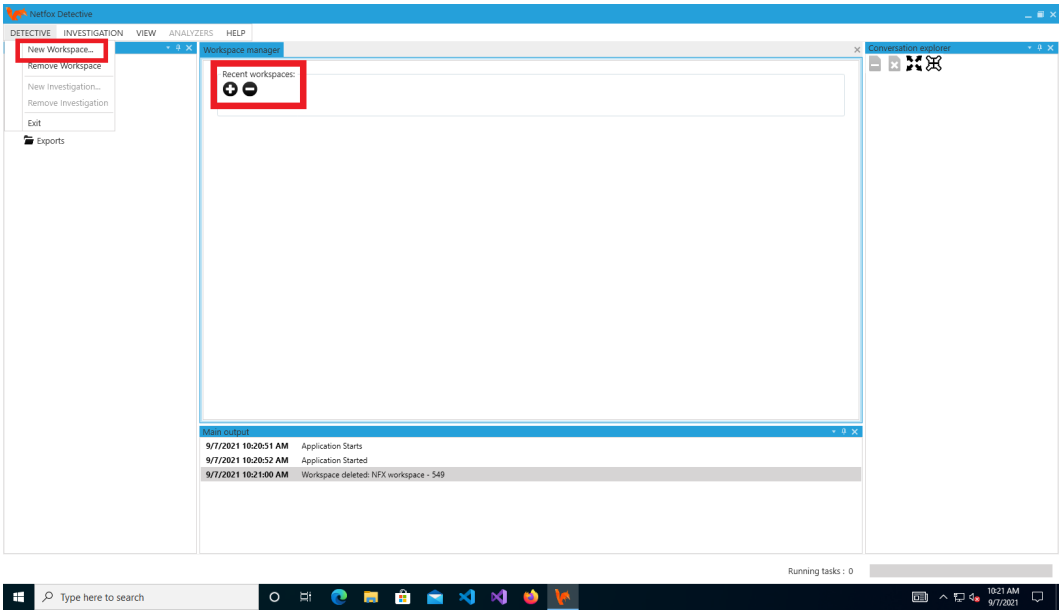

Abbildung 4.8: NFX Detective Workspace

[62] https://github.com/nesfit/NetfoxDetective [Zugriff am: 07.09.2021]

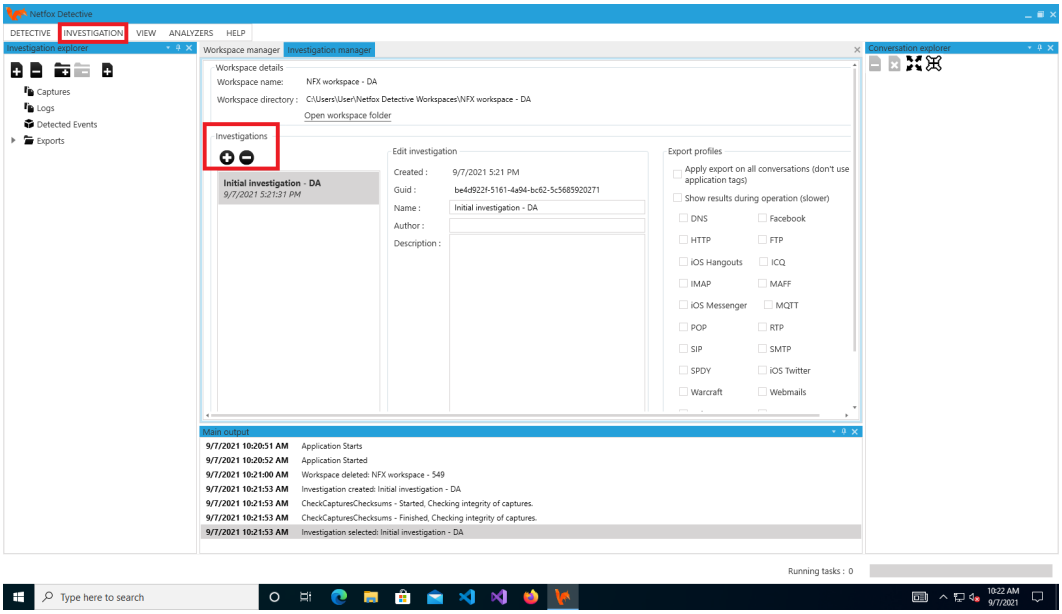

Abbildung 4.9: NFX Detective Investigation

Übersicht der Funktionen

Die grafische Benutzeroberfläche ist wie folgt aufgebaut:

- Investigation Explorer: Er ist das Hauptnavigationsfeld der Anwendung. Er organisiert die PCAP(NG)-Dateien, Logs, erkannten Ereignisse und exportierten Objekte.
- Workspace manager: Hier werden die verschiedenen Workspaces angezeigt, zwischen denen man wechseln kann. Auch können hier neue Workspaces angelegt oder gelöscht werden.
- Investigation manager: Hier können Untersuchungsschritte dokumentiert werden und Einstellungen für das Exportieren von Dateien festgelegt werden.
- Conversations overview: Es werden Informationen über die PCAP-Datei angezeigt, wie etwa der SHA1-Hashwert. Des weiteren ist es möglich einen SSL/TLS-Schlüssel hinzuzufügen. Auch können hier wieder Einstellungen für das Exportieren von Dateien festgelegt werden. Hier werden auch die verschiedenen Conversations (Pakete, die die gleiche Quell- und Zieladresse besitzen) der Layer 3 (gleiche IP-Adressen), 4 (gleiche IP-Adressen und Ports) und 7 (eine einzelne Sitzung zwischen einem Client und einer Server-Anwendung). Sowie die Frame-Ansicht die die einzelnen Frames linear auflistet.
- Conversations detail: Hier werden verschiedene Statistiken des Netzwerkmitschnittes angezeigt. Außerdem werden die vorkommeneden Anwendungs- und Transportprotokolle, sowie die Timeline (wann wurde Traffic gesendet), Hosts Traffic (welcher Host hat wie viel Daten gesendet) und Conversations

(welche Conversation hat wie viel Traffic verursacht und wie lange hat diese gedauert) grafisch mittels Diagramms dargestellt.

Erzeugung Netzwerkmitschnitt

Das Erstellen eines Netzwerkmittschnittes ist mit NFX Detective nicht möglich.

Import von PCAP(NG)-Dateien

NFX Detective unterstützt sowohl PCAP- als auch PCAPNG-Dateien. Im eine PCAP(NG)-Datei zu importieren gibt es zwei Möglichkeiten. Entweder über *Investigation->Add Capture* oder über das Plus-Symbol im Investigation Explorer, siehe Abbildung 4.10. Das Laden der PCAPNG-Datei dauerte 3 Minuten und 8 Sekunden (Vergleichswert Wireshark: 10 Sekunden).

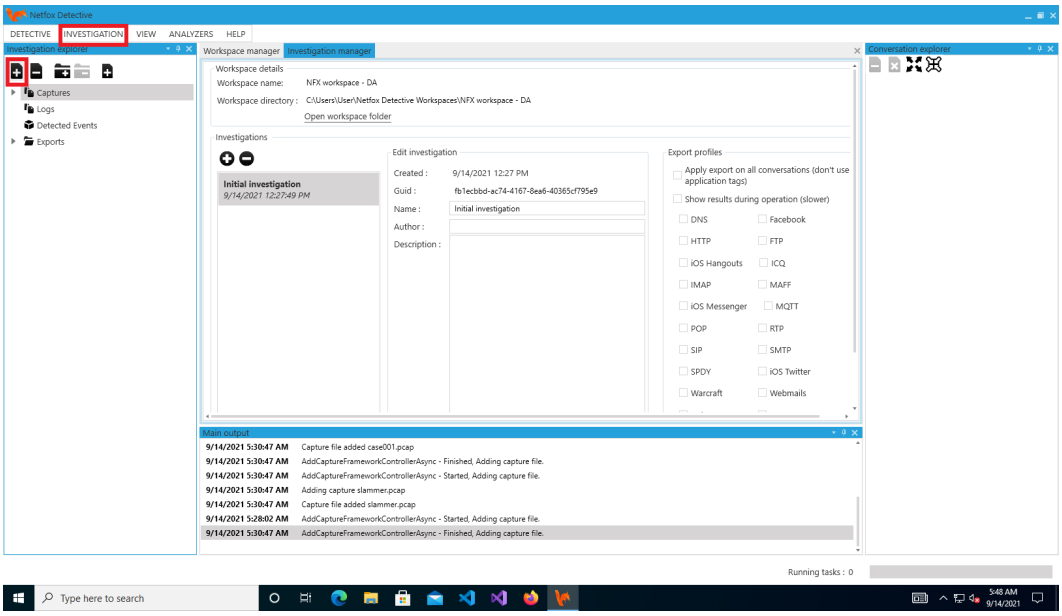

Abbildung 4.10: NFX Detective Capture

Analyse der PCAP(NG)-Datei

1. Welche Protokolle kommen vor?

Um die vorkommenden Protokolle zu identifizieren, muss in das Register Conversations detail und innerhalb dieser auf Application protocols. Hier erhält man einen schönen Überblick der vorgekommen Protokolle, siehe Abbildung 4.11. Ein großer Teil, 70,9%, der Protokolle wird als multiple-protocols gelistet, weitere 3,9% als unknown. Das Problem hierbei ist, dass NFX Detective nur wenige Protokolle unterstützt. Eine Übersicht über die vorkommenden Transport-Protokolle findet man unter Transport protocols, siehe Abbildung 4.12.

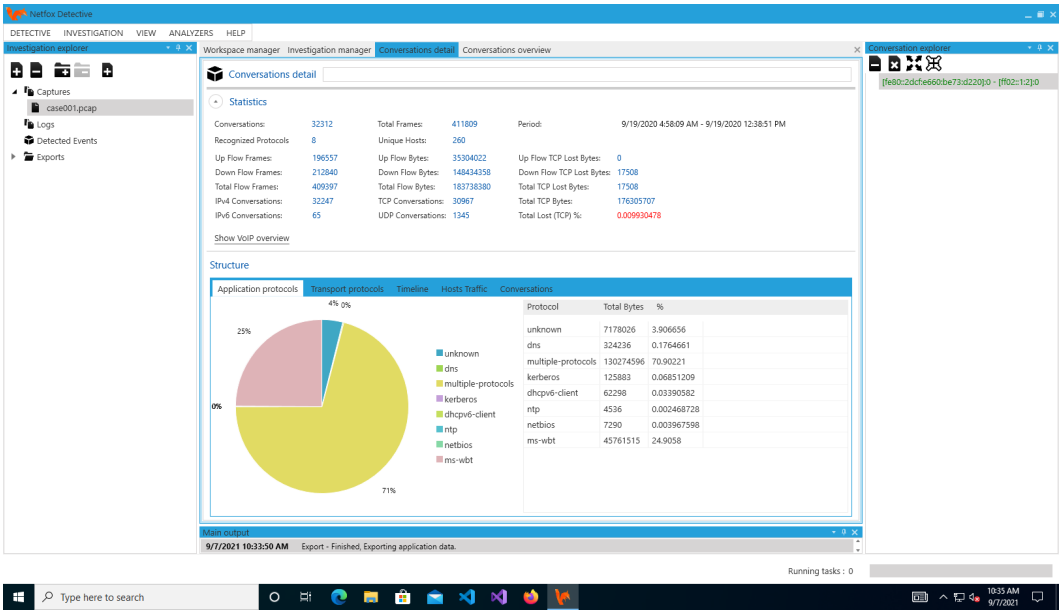

Abbildung 4.11: NFX Detective Protokolle

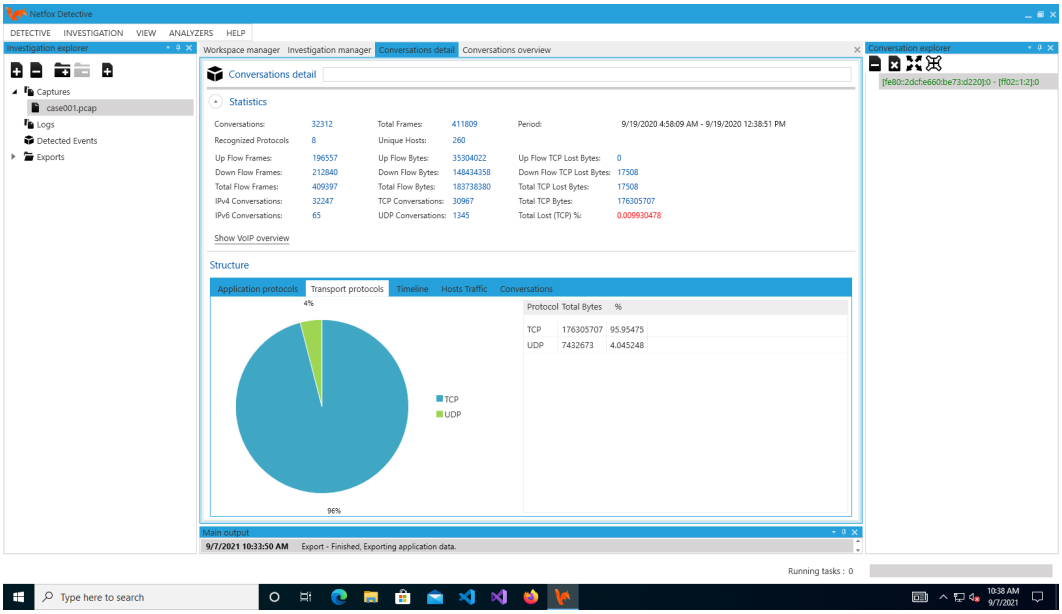

Abbildung 4.12: NFX Detective Transport-Protokolle

2. Welche IP-Adressen kommunizieren miteinander?

NFX Detective bietet im Register Conversations overview verschiedene Ansichten. Zum einen gibt es die Ansichten der verschiedenen Layer (3, 4 und 7) sowie eine Frame-Ansicht, die alle Frames linear auflistet, siehe Abbildung 4.13. Leider kann hier nicht nach Hosts, IP-Adressen oder Protokolle gefiltert werden. Es ist zwar möglich diese Frage zu beantworten, jedoch ohne eine Möglichkeit der Filterung ist dies eine sehr aufwändige Arbeit. Wenn man in einer der Layer-Ansichten auf einen Eintrag doppelklickt, öffnet sich ein neues Registerblatt, mit einer Timeline der zwei Gesprächspartner sowie allen Frames (Doppelklick auf das Frame für Dateilansicht), siehe Abbildung 4.14 und Abbildung 4.15. Dies ist sehr nützlich, um die Kommunikation nachzuvollziehen.

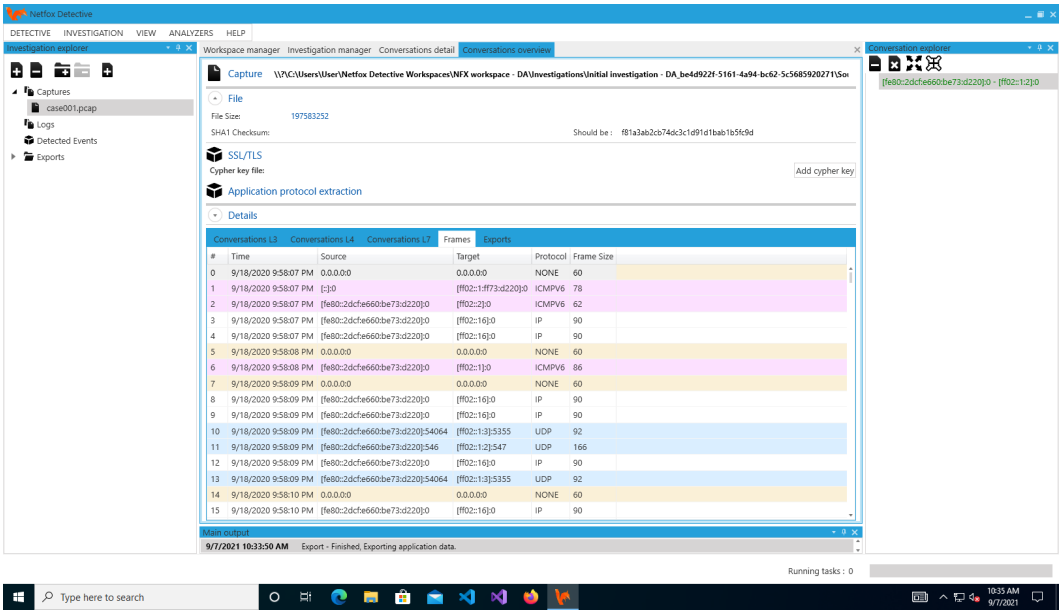

Abbildung 4.13: NFX Detective Frames-Ansicht

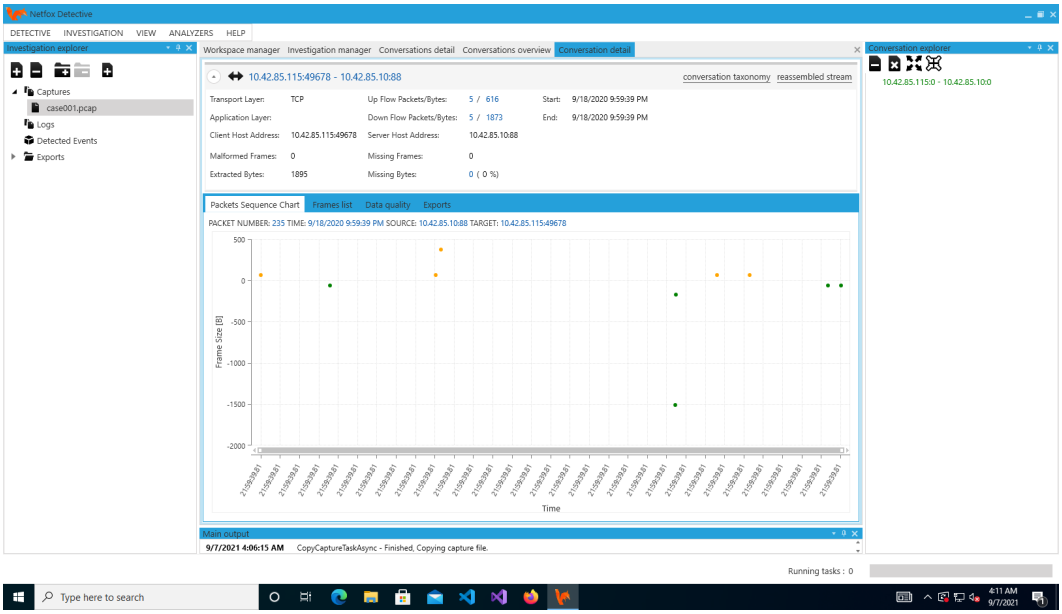

Abbildung 4.14: NFX Detective Conversations Detail

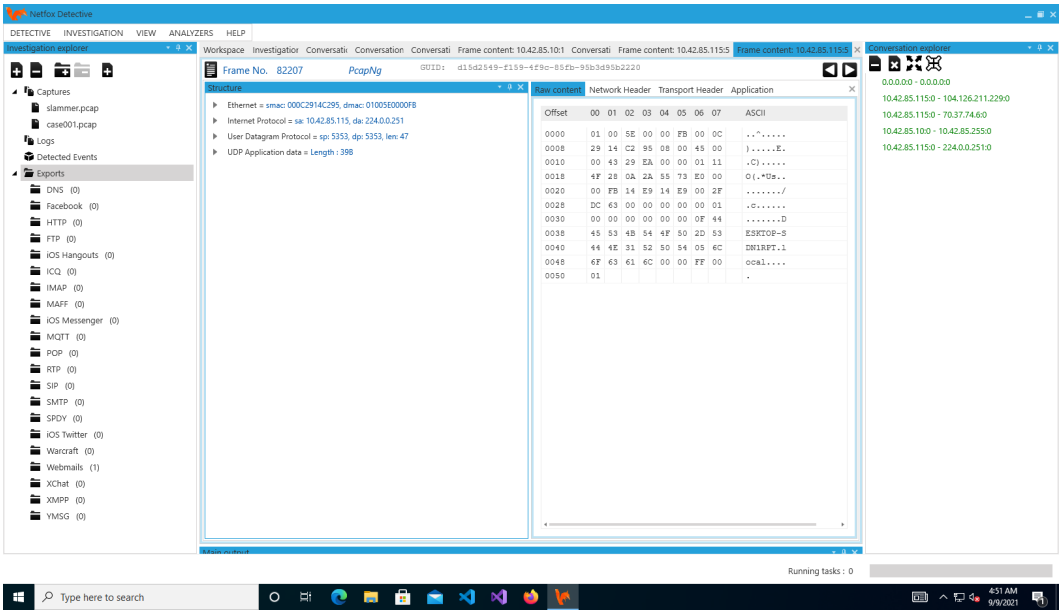

Abbildung 4.15: NFX Detective Frame-Detail

3. Wer erzeugt den meisten Traffic?

Diese Frage ist sehr leicht mit NFX Detective zu beantworten. Hierfür wechselt man in das Register Conversations detail und innerhalb diesen auf Hosts Traffic, siehe Abbildung 4.16

4. Welche IP-Adresse weist die meisten Verbindungen auf?

Diese Frage ist mit NFX Detective nicht zu beantworten, es gibt keine Ansicht, in der die Anzahl der Sessions der Hosts angezeigt wird.

5. Können Credentials identifiziert werden?

Es konnten keine Credentials identifiziert werden.

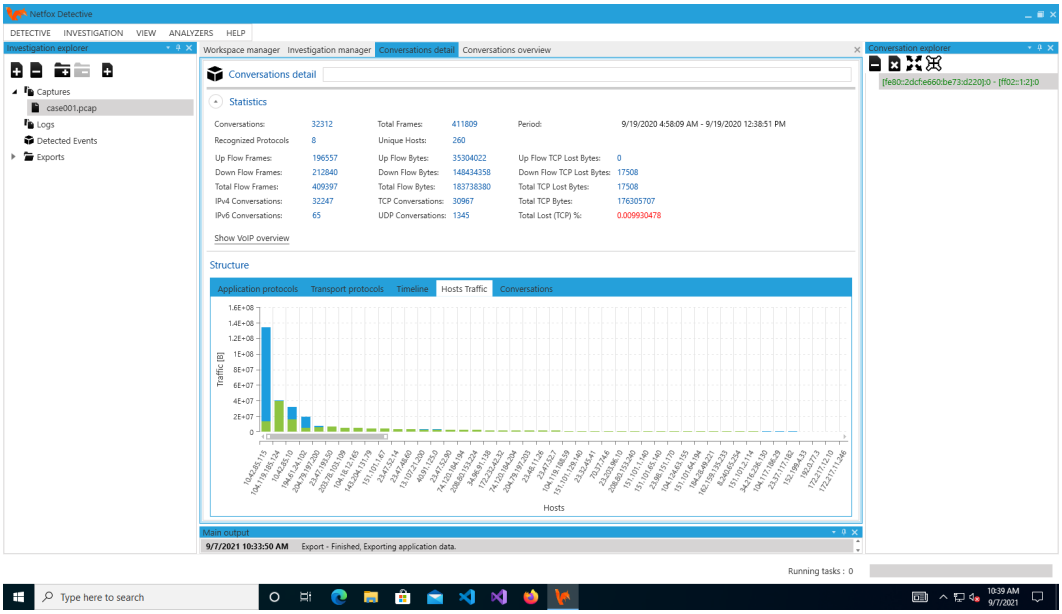

Abbildung 4.16: NFX Detective Hosts Traffic

Extraktion von Dateien

Diese Funktion sollte laut Entwickler für bestimmte Protokolle wie DNS, HTTP, FTP, RTP, POP3, SMTP und weiter zur Verfügung stehen. Doch bei dem Praxistest hat dies nicht funktioniert. Es wurden sämtliche Einstellung des Export profiles durchprobiert, ohne Erfolg. Die Einstellungen für das Export profile findet man im Register Investigation manager, siehe Abbildung 4.17.

IPv6

IPv6 wird unterstützt.

Encapsulation Protokolle

Es werden GRE, 802.1Q und GSE (Generic Stream Encapsulation) unterstützt.

Application Protocol Identification

NFX Detective verwendet SPID (Statistical Protocol IDentification), NBAR (Network-Based Application Recognition), ESPI (Enhanced Statistical Protocol Identification), bayessche Statistik und Random Forests, um das richtige Anwendungsprotokoll herauszufinden.

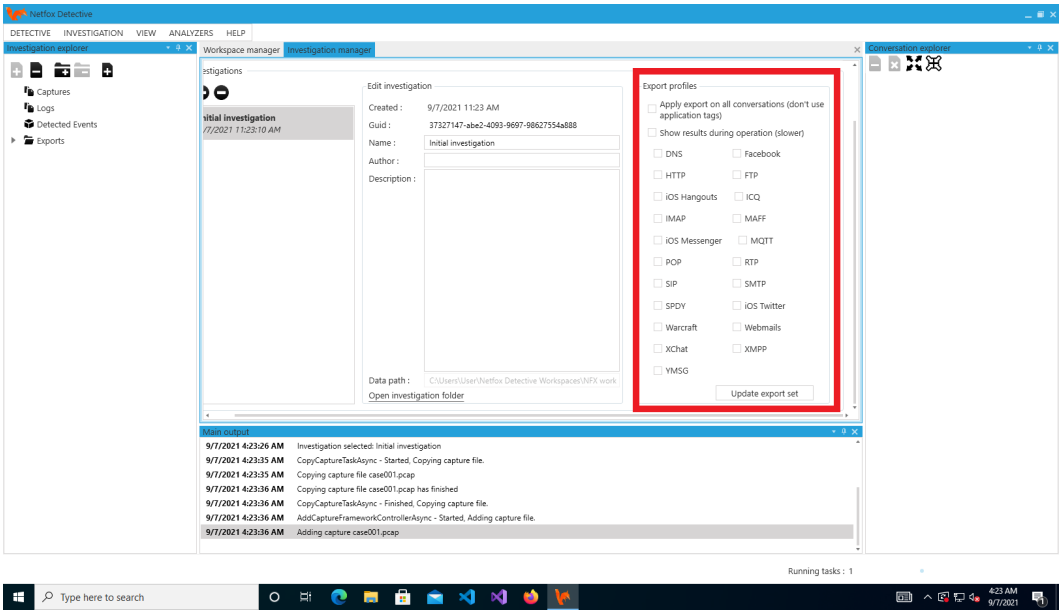

Abbildung 4.17: NFX Detective Export profile

Filter

Standardmäßig unterstützt das Tool keine Abfragen oder Filter.

Export

Die analysierten Daten können nicht in verschiedenen Formaten exportiert werden.

Zeitzone

Die Zeitzone konnte nicht eingestellt werden.

Farbkodierung

NFX Detective unterstützt keine Farbmarkierung.

Plugins

Es gibt die Möglichkeit, neue Extraktionsmodule für andere Anwendungsprotokolle zu erstellen. Auch die Erweiterung des Systems mit benutzerdefinierten Analysemethoden ist möglich. Da NFX Detective ein offenes Datenmodell verwendet, auf das zugegriffen werden kann oder das leicht modifiziert werden kann. Erweitert kann das Tool mittels der Programmiersprache C#.

Bedienung

Man benötigt zwar etwas mehr Zeit wie bei NetworkMiner, damit man die verschiedenen Ansichten und deren Zweck herausgefunden hat, aber die Bedienung ist weitestgehend intuitiv.

GUI

NFX Detective stellt eine grafische Benutzeroberfläche bereit.

Aktualität

Netfox Detective steht zum Zeitpunkt dieser Arbeit in der Version 2.1.0.0 bereit. Diese wurde am 18.10.2019 veröffentlicht und ist damit bereits fast zwei Jahre alt. Deshalb gibt es einen Punkteabzug.

Bewertung

Nr.	Kriterium	Zeichen	Punkte
1	Installation/Konfiguration	++	2
2	Erzeugung Netzwerkmitschnitt	-	0
3	Import von PCAP(NG)-Dateien	++	2
4	Analyse der PCAP(NG)-Datei	$\sim$	1
5	Extraktion von Dateien	-	0
6	IPv6	++	2
6	Encapsulation Protokolle	+	2
7	Application Protocol Identification	+	2
8	Filter	-	0
9	Export	-	0
11	Zeitzone	-	0
12	Farbkodierung	-	0
13	Plugins	+	2
14	Bedienung	++	2
15	GUI	++	2
16	Aktualität	$\sim$	1
17	**Gesamtpunkte**		**18**

Tabelle 4.10: Bewertung der Analyse von NFX Detective

Stärken/Schwächen

Der große Vorteil von NFX Detective sind die übersichtlichen Grafiken, die erstellt werden. Mit diesen hat man schnell einen guten Überblick über die vorhandenen Protokolle, Hosts und Conversations. Bei bestimmten Fragen, die bei der Analyse vorgekommen sind, leistet das Tool hervorragende arbeit, bei anderen wiederum muss man sehr viel manuell durchsuchen, da keine Abfragen oder Filter standardmäßig unterstützt werden. Auch das Extrahieren von Dateien hat leider in dem Test nicht funktioniert. Außerdem kam es bei der Analyse zu mehreren Abstürzen des Tools. Auch können die analysierten Dateien nicht exportiert werden. Der größte Nachteil von NFX Detective ist, dass es nur eine geringe Anzahl an Protokollen unterstützt.

4.4.5 Arkime

Installation

Arkime wird als Debian-Paket (.deb) auf deren Homepage[63] bereitgestellt. Nach dem Herunterladen wechselt man in der Konsole zum Download-Ordner und installiert Arkime mittels:

```
student@ubuntu:~$ sudo apt install ./arkime_3.0.0-1_amd64.deb
```

Danach muss Elasticsearch wie folgt installiert werden:

```
student@ubuntu:~$ sudo su
root@ubuntu:/home/student# wget -qO - https://artifacts.elastic.co/GPG-
KEY-elasticsearch --no-check-certificate | apt-key add -
root@ubuntu:/home/student# echo "deb https://artifacts.elastic.co/packa
ges/7.x/apt stable main" | tee -a /etc/apt/sources.list.d/elastic-7.x.l
ist
root@ubuntu:/home/student# exit

student@ubuntu:~$ sudo apt update
student@ubuntu:~$ sudo apt install elasticsearch
student@ubuntu:~$ sudo systemctl enable --now elasticsearch
```

Durch folgendem Befehl kann die Funktionsfähigkeit von Elasticsearch überprüft werden:

```
student@ubuntu:~$ curl http://localhost:9200
```

Man sollte eine Antwort in der Art wie folgt bekommen:

```
{
"name" : "ubuntu",
"cluster_name" : "elasticsearch",
"cluster_uuid" : "g5xxDwIwQAa2T44yOcIFKA",
"version" : {
    "number" : "7.14.1",
    "build_flavor" : "default",
```

[63] https://s3.amazonaws.com/files.molo.ch/builds/ubuntu-20.04/arkime_3.0.0-1_amd64.deb [Zugriff am: 09.09.2021]

```
    "build_type" : "deb",
    "build_hash" : "66b55ebfa59c92c15db3f69a335d500018b3331e",
    "build_date" : "2021-08-26T09:01:05.390870785Z",
    "build_snapshot" : false,
    "lucene_version" : "8.9.0",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
},
"tagline" : "You Know, for Search"
}
```

Danach wird Arkime wie folgt konfiguriert:

```
student@ubuntu:~$ sudo /opt/arkime/bin/Configure
Found interfaces: lo;ens33
Semicolon ';' seperated list of interfaces to monitor [eth1] ens33
Install Elasticsearch server locally for demo, must have at least 3G of
memory, NOT recommended for production use (yes or no) [no] no
Elasticsearch server URL [http://localhost:9200]
Password to encrypt S2S and other things [no-default] student
Download GEO files? You'll need a MaxMind account https://arkime.com/faq
#maxmind (yes or no) [yes] yes
```

Anschließend muss die Elasticsearch-Arkime-Konfiguration initialisiert werden:

```
student@ubuntu:~$ sudo /opt/arkime/db/db.pl http://localhost:9200 init
```

Jetzt kann ein Adminuser wie folgt erstellt werden:

```
student@ubuntu:~$ sudo /opt/arkime/bin/arkime_add_user.sh admin "Admin
User" password --admin
```

Danach wird der Arkime Capture und Viewer gestartet:

```
student@ubuntu:~$ sudo systemctl enable --now arkimecapture.service
student@ubuntu:~$ sudo systemctl enable --now arkimeviewer.service
```

Mittels folgendem Befehl kann überprüft werden, ob die Services korrekt laufen:

```
student@ubuntu:~$ sudo systemctl status arkimecapture.service
student@ubuntu:~$ sudo systemctl status arkimeviewer.service
```

Wenn alles funktioniert sollte nun der Arkime Capture bereits die Paketerfassung auf der konfigurierten Schnittstelle durchführen. Der Arkime Viewer (GUI) ist unter folgender Adresse im Webbrowser erreichbar:

```
localhost:8005
```

Mittels dem angelegten Admin User kann man sich nun anmelden. Nach erfolgreicher Anmeldung, gelangt man auf die GUI von Arkime, siehe Abbildung 4.18.

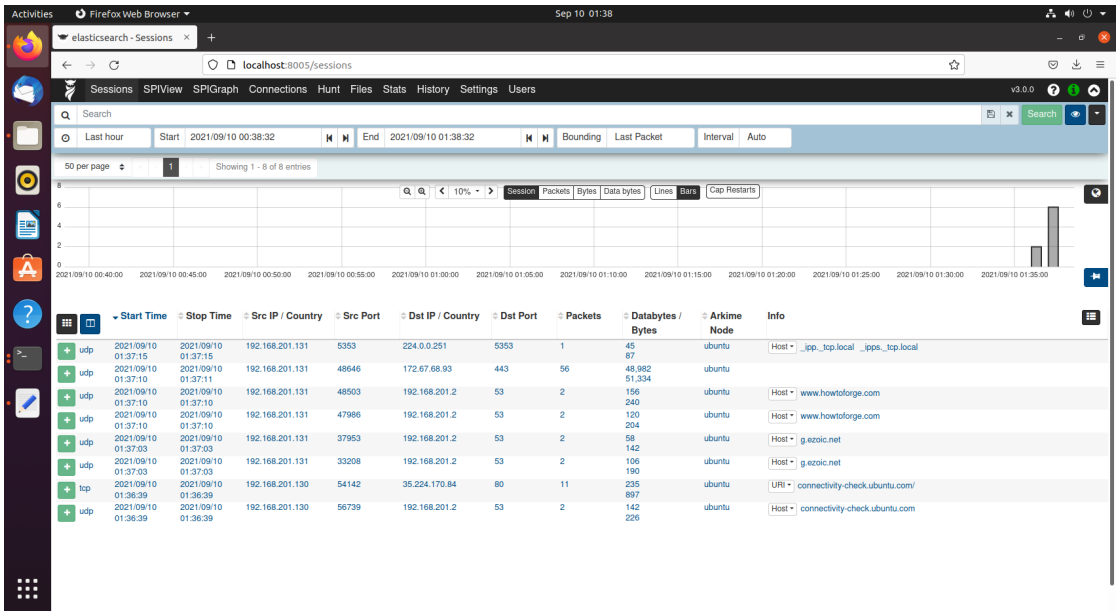

Abbildung 4.18: Arkime GUI

Die Installationsanleitung ist leider nicht auf dem neuesten Stand, von den einzelnen Schritten her ist sie zwar korrekt, jedoch stimmten die Pfade und Namen nicht mit der neuesten Version von Arkime zusammen. Zusätzlich ist die Installation um einiges aufwändiger als die von NetworkMiner und NFX Detective, deshalb bekommt Arkime hier null Punkte.

Übersicht der Funktionen

Arkime setzt sich aus 3 Komponenten zusammen:

- Capture: Eine C-Anwendung, die den Netzwerkverkehr überwacht, PCAP-formatierte Dateien auf die Festplatte schreibt, die erfassten Pakete parst und Metadaten an Elasticsearch sendet.
- Viewer: Eine node.js-Anwendung, die pro Erfassungsmaschine läuft. Sie verwaltet das Webinterface und die Übertragung von PCAP-Dateien.

- Elasticsearch: Die Technologie der Suchdatenbank, die Arkime antreibt.

Die grafische Benutzeroberfläche ist folgendermaßen aufgebaut:

- Sessions: Dies ist die Hauptansicht, die eine Liste der Sessions enthält. Jede Sitzung kann geöffnet werden, um die Metadaten und PCAP-Daten anzuzeigen, siehe Abbildung 4.19. Zudem kann nach verschiedenen Parametern gesucht und gefiltert werden. Auch eine Timeline (wann Daten übertragen wurden).
- SPIView: Diese Ansicht ermöglicht es dem Benutzer, alle eindeutigen Werte für jedes Feld, das Arkime versteht, zu sehen, siehe Abbildung 4.20.
- SPIGraph: Hier können verschiedene Grafiken dargestellt werden. Die Grafiken können verschiedene Kriterien anzeigen und nach diesen auch sortieren. In Abbildung 4.21 wird ein Beispiel für eine Grafik gezeigt.
- Connections: In dieser Ansicht wird ein Verbindungsgraph angezeigt. Mithilfe dessen man auf einen Blick sehen kann welche Parteien miteinander kommunizieren. Die Dicke der Linien gibt Auskunft über die Anzahl der Sessions.
- Hunt: Hier können die Sessions nach Begriffen oder Regex durchsucht werden. Jeder Hunt bekommt dabei eine ID, mithilfe derer die gefundenen Sessions in der Sessions-Registerkarte begutachtet werden können.
- Files: Hier werden die erstellten und hochgeladenen PCAP-Dateien angezeigt.
- Stats: In dieser Ansicht werden verschiedene Statistiken über den Captuer und Elasticsearch angezeigt.
- History: Hier wird die Historie angezeigt.
- Settings: In dieser Registerkarte können verschiedene Einstellungen getroffen werden.
- Users: Hier können die User verwaltet werden.

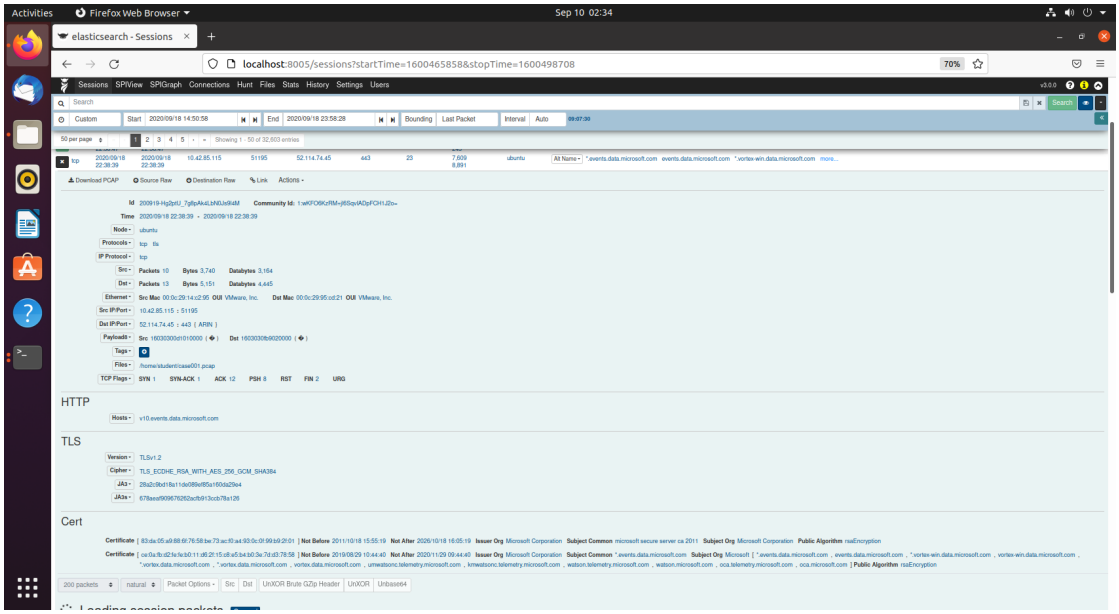

Abbildung 4.19: Arkime Session Detail

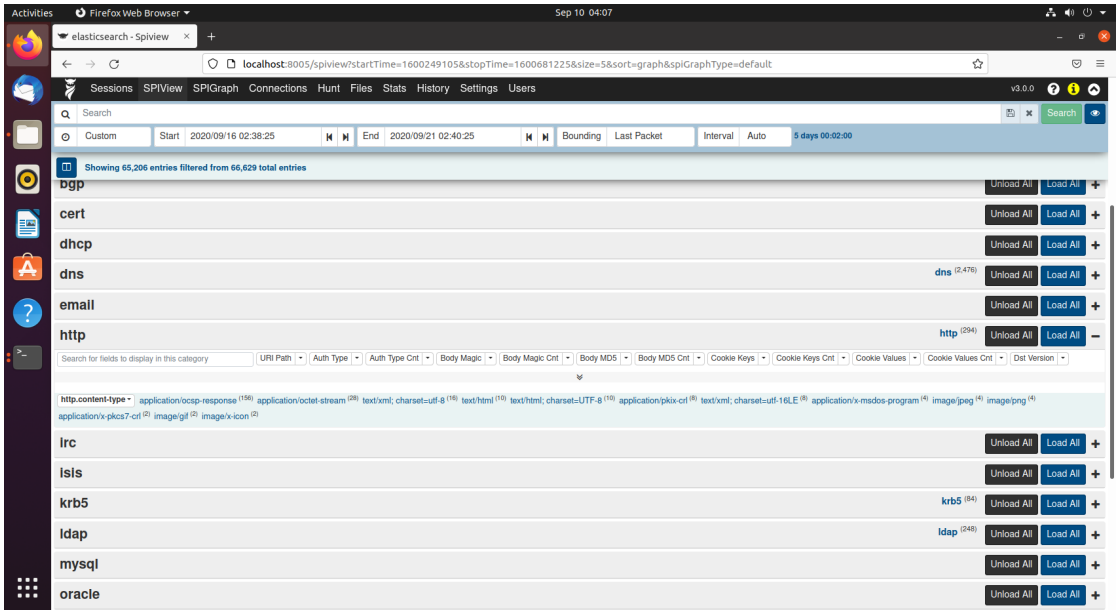

Abbildung 4.20: Arkime SPIView

Abbildung 4.21: Arkime Beispielgrafik

Erzeugung Netzwerkmitschnitt

Durch das Starten des Arkime Capture wird automatisch der Netzwerkverkehr mitgeschnitten auf der konfigurierten Netzwerkschnittstelle. Es ist auch möglich mehrere Netzwerkschnittstellen zu konfigurieren, in dem sie mittels eines Strichpunktes „;;„ trennt. Die erzeugten PCAP-Dateien werden in dem Ordner */opt/arkime/raw* abgelegt.

Import von PCAP(NG)-Dateien

Arkime unterstützt sowohl PCAP- als auch PCAPNG-Dateien. Am besten jedoch mit PCAP-Dateien, bei PCAPNG-Dateien funktionieren ein paar Funktionen nicht, wie die Detail-Ansicht der Session-Pakete (Payload) oder der Hunt. Das Importieren erfolgt mittels folgenden Befehles:

```
sudo /opt/arkime/bin/capture -r /home/student/case001.pcap
```

Das Importieren der Testdaten hat 7,48 Sekunden gedauert (Vergleichswert Wireshark: 3,71 Sekunden).

Analyse der PCAP(NG)-Datei

1. Welche Protokolle kommen vor?

Um herauszufinden welche Protokolle vorkommen, wechselt man in die Registerkarte SPIView. Unter dem Punkt general wählt man das Feld protocols aus. Nun werden alle vorkommenden Protokolle Angezeigt, wie in Abbildung 4.22 ersichtlich.

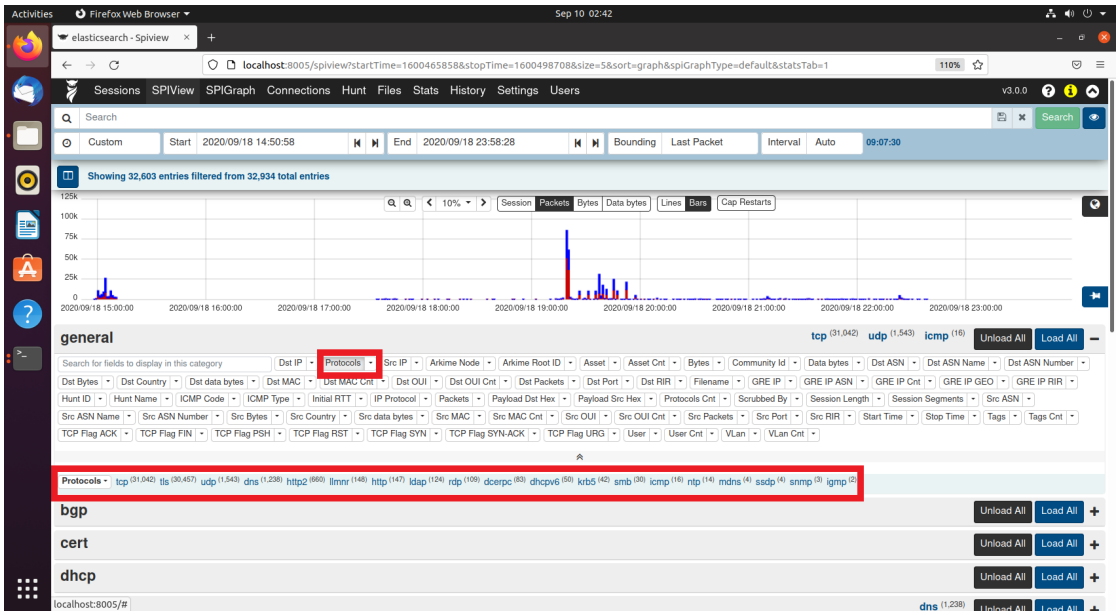

Abbildung 4.22: Arkime Protokolle

2. Welche IP-Adressen kommunizieren miteinander?

Um diese Frage zu beantworten, wechselt man in die Registerkarte Connections. Hier sieht man den Verbindungsgraphen, dieser zeigt an, wer mit wem kommuniziert, siehe Abbildung 4.23

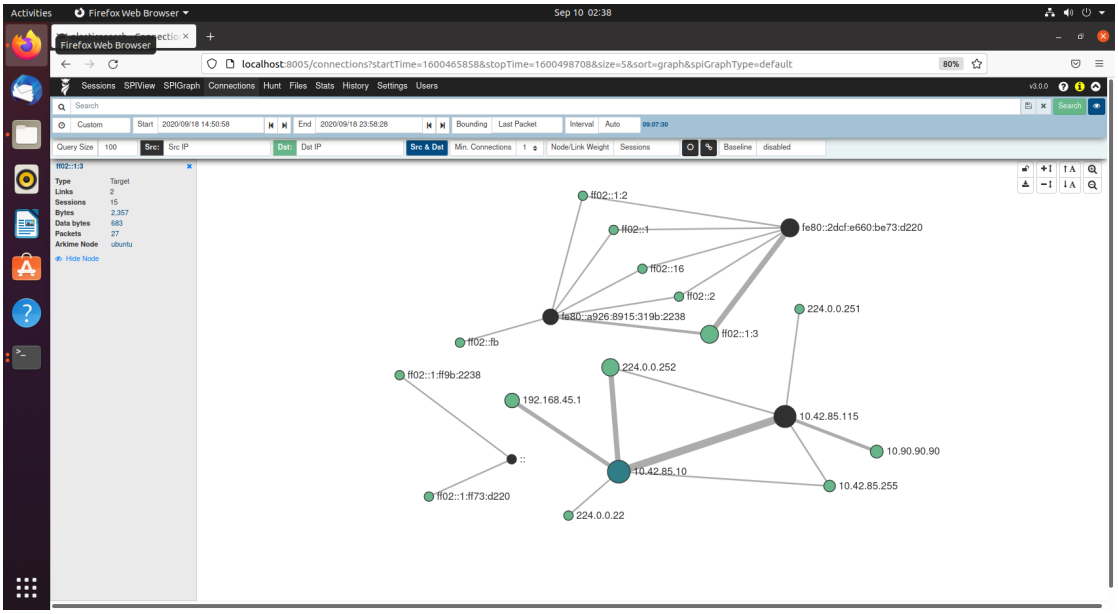

Abbildung 4.23: Arkime Kommunikation

3. Wer erzeugt den meisten Traffic?

In der Registerkarte SPIGraph kann diese Frage beantwortet werden. Hierzu wählt man im Feld SPI Graph das Kriterium Src IP aus und wählt Bytes als Zähler aus. In Abbildung 4.24 wird das Ergebnis präsentiert. Neben der IP-Adresse steht in Klammern die Anzahl der Bytes.

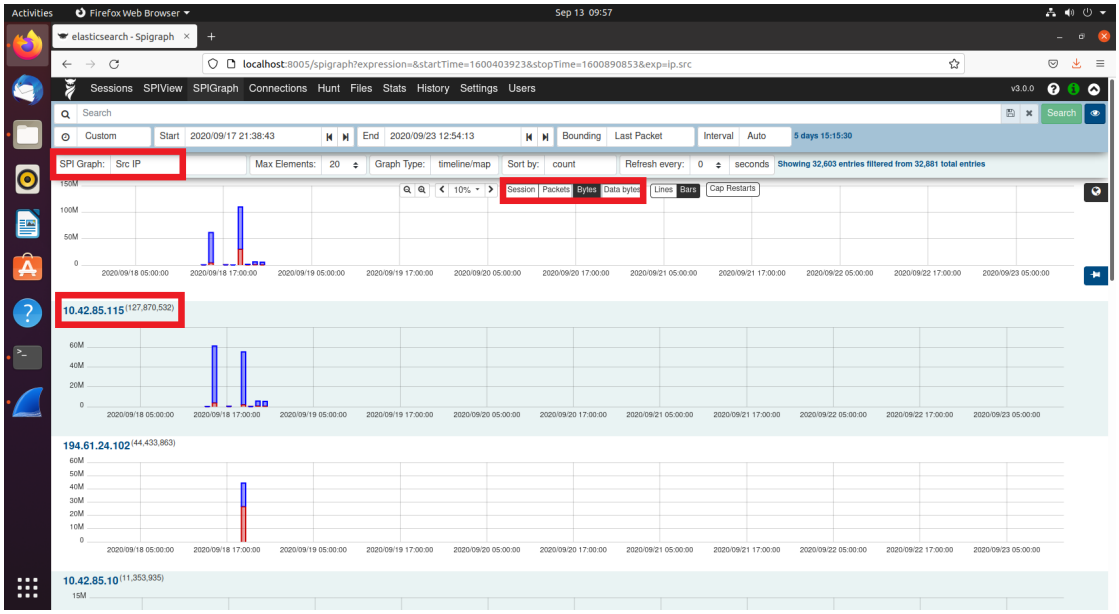

Abbildung 4.24: Arkime Traffic

4. Welche IP-Adresse weist die meisten Verbindungen auf?

Um diese Frage zu beantworten, bleibt man in der Registerkarte SPIGraph. Hier wählt man nun im Feld SPI Graph wieder Src IP aus und als Zähler wird Sessions ausgewählt, siehe Abbildung 4.25.

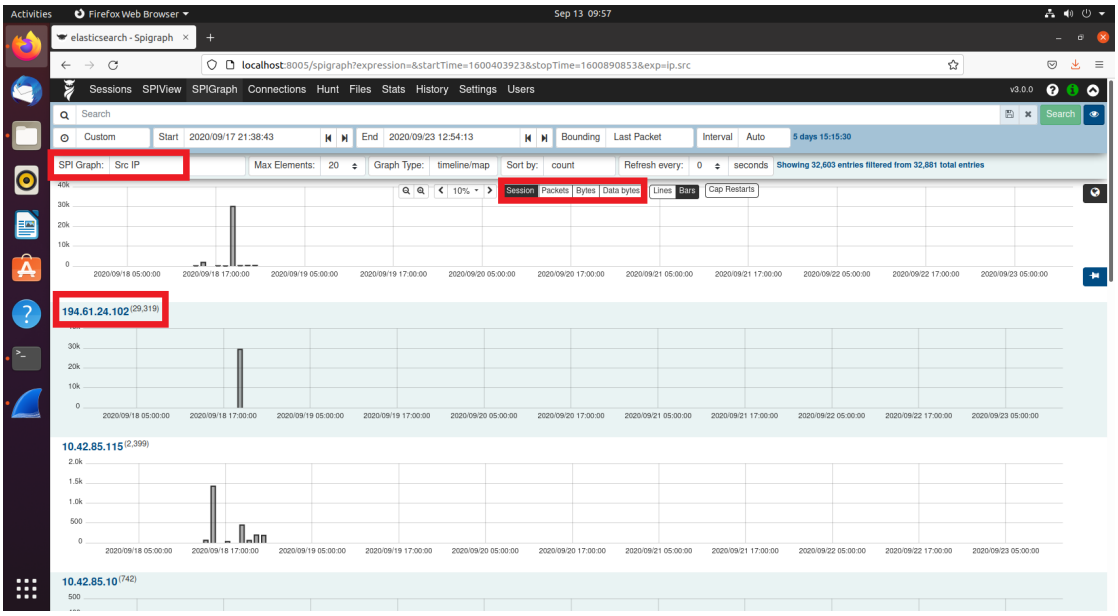

Abbildung 4.25: Arkime Sessions

5. Können Credentials identifiziert werden?

So übersichtlich wie in NetworkMiner werden die Credentials nicht angezeigt. Jedoch gibt es die Möglichkeit mittels Hunt nach zum Beispiel dem User Administrator zu suchen, siehe Abbildung 4.26. Danach wird die Hunt-ID kopiert und in die Session Registerkarte gewechselt. Nur wird nach dieser Hunt-ID gefiltert, siehe Abbildung 4.27. Dies ist etwas umständlich und man muss wissen wonach man suchen möchte. Aber theoretisch funktioniert das Suchen nach Credentials.

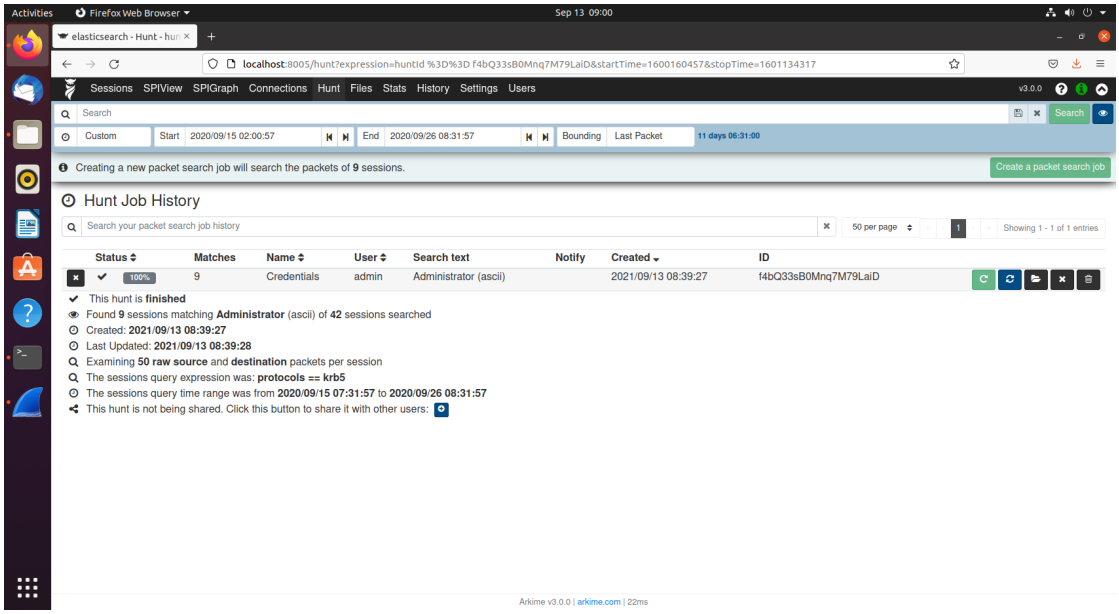

Abbildung 4.26: Arkime Hunt

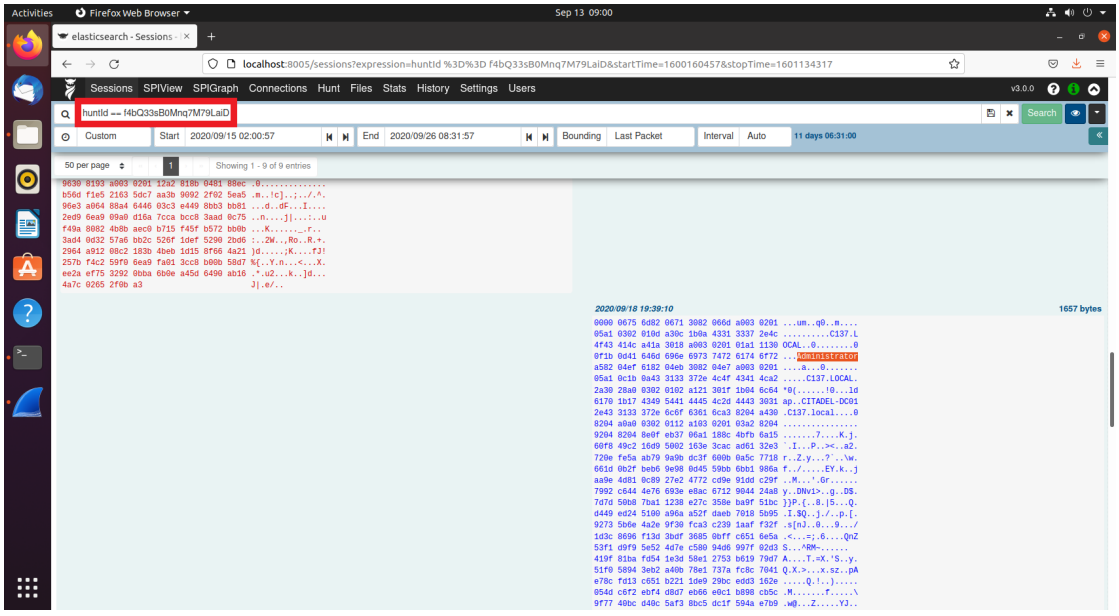

Abbildung 4.27: Arkime Credentials

Extraktion von Dateien

Jegliche Daten können von den Sessions in der Registerkarte Sessions gespeichert werden. Wenn es sich um ein Bild oder eine Grafik handelt, wird diese auf der GUI angezeigt, wie in Abbildung 4.28 dargestellt. Die Abbildung 4.29 zeigt den Download einer EXE-Datei.

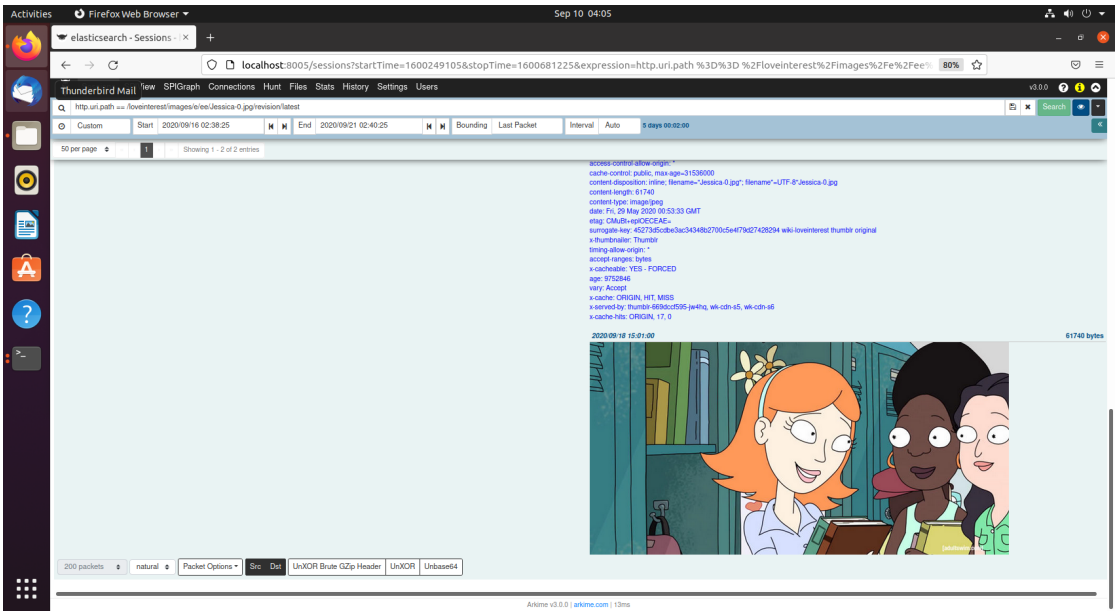

Abbildung 4.28: Arkime Extraktion PNG

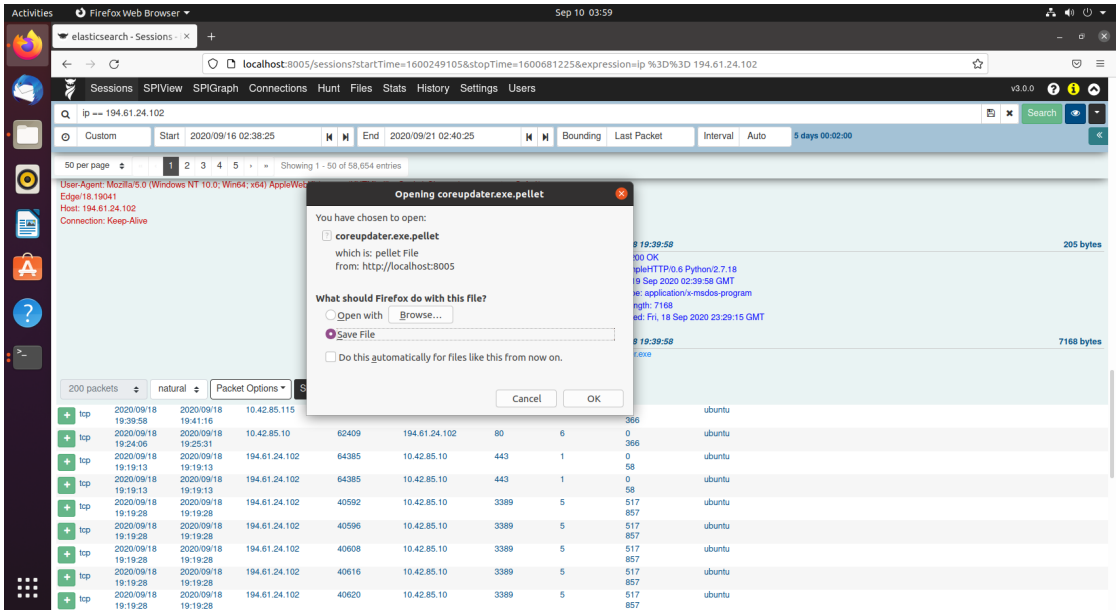

Abbildung 4.29: Arkime Extraktion EXE

IPv6

IPv6 wird von Arkime unterstützt.

Encapsulation Protokolle

Es wird auf der Homepage oder Github leider nicht aufgelistet. Jedoch nach betrachten des Source-Codes müsste Arkime GRE, VXLAN, PPPoE, SOCKS und MPLS unterstützen.

Application Protocol Identification

Ist leider nicht dokumentiert, ob Arkime PIPI oder eine andere Methode zur Identifikation von Protokollen verwendet.

Filter

Es können verschiedene Filter gesetzt werden, sowie verschiedenste Abfragen durchgeführt werden.

Export

Die Analysierten Daten können entweder im CSV- oder PCAP-Format exportiert werden. Dabei können entweder einzelne Sessions, mehrere Sessions oder die ganzen Daten exportiert werden.

Zeitzone

Die Zeitzone kann nicht eingestellt werden, lediglich das Format kann in den Settings zwischen PDT und UTC umgestellt werden.

Farbkodierung

Farbkodierung gibt es nicht, jedoch gibt es die Möglichkeit die Sessions mit einem Tag zu markieren, siehe Abbildung 4.30. Nach diesem Tag kann dann gesucht werden. Da diese Funktion es ermöglicht, Sessions zu markieren und diese ähnlichen Nutzen wie eine Farbkodierung hat, werden hier volle Punkte vergeben.

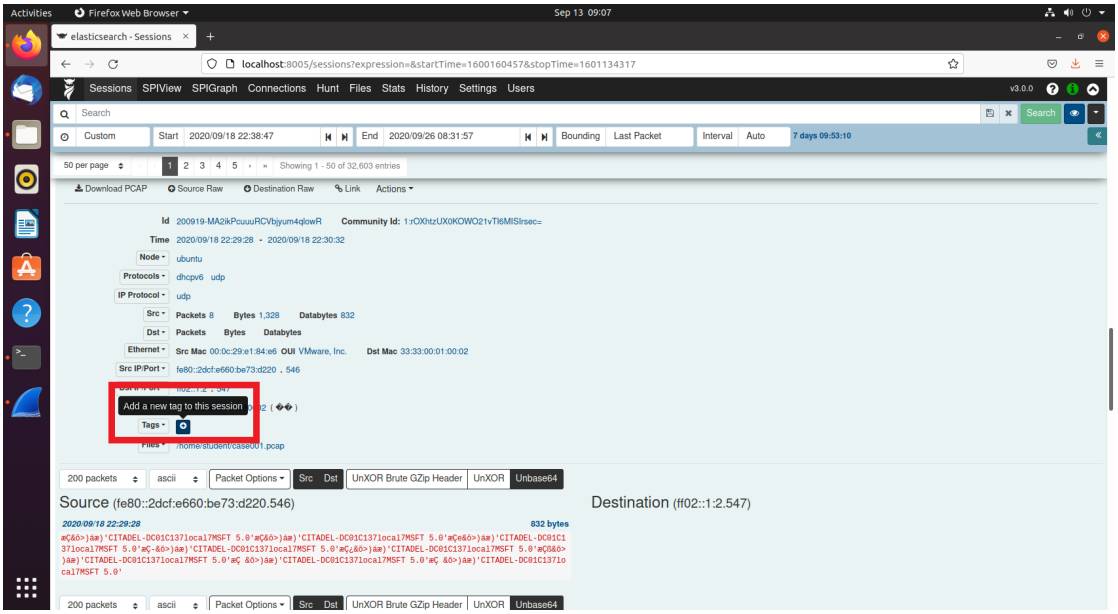

Abbildung 4.30: Arkime Tag

Plugins

Es ist möglich selbst Plugins in der Programmiersprache Lua für Arkime zu entwickeln.

Bedienung

Die Bedienung ist intuitiv und man kommt schnell zurecht.

GUI

Arkime bietet eine grafische Benutzeroberfläche.

Aktualität

Arkime steht zur Zeit dieser Arbeit in der Version 3.0.0 zur Verfügung. Diese wurde am 18.08.2021 veröffentlicht.

Bewertung

Nr.	Kriterium	Zeichen	Punkte
1	Installation/Konfiguration	-	0
2	Erzeugung Netzwerkmitschnitt	++	2
3	Import von PCAP(NG)-Dateien	$\sim$	1
4	Analyse der PCAP(NG)-Datei	++	2
5	Extraktion von Dateien	++	2
6	IPv6	++	2
6	Encapsulation Protokolle	+	2
7	Application Protocol Identification	-	0
8	Filter	++	2
9	Export	++	2
11	Zeitzone	-	0
12	Farbkodierung	++	2
13	Plugins	+	2
14	Bedienung	++	2
15	GUI	++	2
16	Aktualität	++	2
17	**Gesamtpunkte**		**25**

Tabelle 4.11: Bewertung der Analyse von Arkime

Stärken/Schwächen

Die Installation von Arkime ist um einiges aufwändiger als bei den anderen getesteten Tools. Auch nach einem Neustart der VM kam es des Öfteren zu Probleme beim Starten der Instanzen, dabei konnte zum Beispiel nicht auf die Weboberfläche zugegriffen werden, erst nach mehrmaligen starten und stoppen der

Services konnte dies behoben werden. Arkime funktioniert am besten mit PCAP-Dateien, beim Import von PCAPNG-Dateien gehen zwei wichtige Funktionen (Hunt und Session Details) verloren. Doch wenn das Tool erfolgreich in Betrieb genommen wird, bietet es sehr gute Unterstützung bei er Analyse von großen PCAP-Dateien. Dem geschuldet sind unter anderem die übersichtlichen Grafiken, die die wichtigsten Informationen auf einen Blick bereitstellen und die sehr guten Filter/Abfragemöglichkeiten (vergleichbar mit Wireshark). Ein weiterer Vorteil ist das Exportieren der Analysierten Daten, sei es eine einzelne Session, mehrere Sessions von einer Abfrage oder alle Daten. Auch das Markieren der Sessions mittels Tags ist sehr hilfreich bei der Analyse.

4.4.6 Conclusio

NetworkMiners Spezialität ist das Extrahieren von Dateien aus dem Netzwerkmitschnitt. Zu beachten ist jedoch, dass dies automatisch geschieht, der Ausggabeordner nicht verändert werden kann und dass alle Daten, die identifiziert werden extrahiert werden. Ein weiterer Punkt, indem das Tool gute Arbeit leistet, ist das Identifizieren von Hosts inklusive OS-Fingerprinting. Für andere Anwendungsfälle kann NetworkMiner nicht empfohlen werden, da es bei der Analyse oft umständlich bzw. unübersichtlich ist.
Netfox Detective bietet einige übersichtliche Grafiken über Hosts, Protokolle und Conversations. Diese können bei der Analyse sehr hilfreich sein. Das Tool bietet zwar verschiedene Ansichten der Layer, geht es jedoch in die Detail-Analyse erweist sich das Tool als umständlich, da man in verschiedene Ansichten wechseln muss, um genauere Informationen über die einzelnen Pakete zu bekommen.
Abgesehen von der eher aufwändigen Installation, konnte Arkime in diesem Test überzeugen. Das Tool ist sehr übersichtlich und bietet sehr nützliche Grafiken, Filter und Abfragen. Dadurch konnten sämtliche Fragen bei der Analyse beantwortet werden. Es bietet einige nützliche Funktionen, wie etwa das Markieren der Sessions mittels Tags, das Exportieren von Daten im PCAP- oder CSV-Format oder das Durchsuchen der Daten nach Begriffen oder Regex. Zudem konnte das Tool bei der Verarbeitungsgeschwindigkeit der PCAP-Dateien überzeugen.
Dieser Test konnte zeigen, dass Tools die sich als NFAT bezeichnen nicht immer die beste Wahl darstellen. Analystinnen/Analysten und Forensikerinnen/Forensiker sollten sich bei der Auswahl ihrer Tools nicht nur die Kategorie NFAT sondern auch Packet Analyzer ansehen und die Tools anhand ihrer Bedürfnisse sorgfältig auswählen.

4.5 Network Forensics in virtuellen Netzen

4.5.1 Überblick

Mit der Virtualisierung von Computern begannen die Provider, ihre Prozesse und die Infrastruktur in ihren Rechenzentren zu verändern. Die Virtualisierung bietet höhere Flexibilität, verbessert die Automatisierung der Installation, Konfiguration und Bereitstellung von Diensten und bietet gleichzeitig neue Möglichkeiten und Kostenreduzierung. Doch die Implementierung von virtuellen Maschinen (VMs) in eine Rechenzentrumsinfrastruktur wirft neue Probleme für den Provider auf. Aspekte wie Sicherheit, Netzwerkmanagement und Kundenanforderungen waren mit virtuellen Computern allein nur schwer zu erfüllen.[74] Daher benötigt es einen anderen Ansatz, der einfacher und flexibler ist als die traditionellen Lösungen.[75] Die Beschränkungen der derzeitigen Netzwerkinfrastrukturen behindern die Implementierung von hochdynamischen, flexiblen, sicheren und automatisierten Umgebungen, die die Kosten senken und ein anpassbares Netzwerk bieten. Diese Einschränkungen beruhen zum einen auf den Netzwerkprotokolle wie VLAN (Virtual Local Area Network), die die Anzahl der verschiedenen, logisch getrennten Netze begrenzt. Zum anderen bieten die installierten Netzwerkgeräte keine Schnittstellen, die eine automatische Konfiguration der angeschlossenen Geräte auf Grundlage zuvor definierter Regeln bzw. durch die Analyse von Zuständen in Abhängigkeit der aktuellen Netzsituation ermöglichen. Diese Umstände führten zur Entwicklung neuer Netzwerkprotokollen wie Virtual eXtensible LAN (VXLAN), Stateless Transport Tunneling (STT) oder Netzwerkvirtualisierung mit generischer Router Encapsulation (NVGRE), und zur Implementierung neuer Paradigmen wie Software-Defined Networks (SDN) und Network Function Virtualization (NFV).[74] Diese unterschiedlichen Implementierungen von Tunneling- und Verkapselungstechniken ermöglichen die Schaffung separater logischer Netze, die dieselbe zugrunde liegende physische Infrastruktur nutzen. Daher werden diese logischen Netze als Overlay-Netze bezeichnet.[76]
Trotz dieser physischen und logischen Veränderungen in den Netzen von Rechenzentren, besteht weiterhin der Bedarf an netzwerkforensischen Untersuchungen. Diese Arbeit betrachtet die Netzwerk-Forensik in SDN-Netzen mit VXLAN als Overlay-Netz, stellvertretend für andere virtuelle Netze.

4.5.2 SDN

In herkömmlichen Netzwerken gibt es verschiedene Sicherheitslösungen (z. B. Firewall, Unified Threat Management, Deep Packet Inspection, Intrusion Prevention System), um Cyber-Bedrohungen zu verhindern, zu erkennen und zu entschärfen. Viele dieser Lösungen werden als spezialisierte Hardware-Appliances angeboten und ihre Einsatzorte werden sorgfältig festgelegt. Solche Sicherheitslösungen haben die Einschrän-

kung, dass ihre Standorte und Funktionen festgelegt sind und nicht dynamisch geändert werden können. Außerdem sind bestimmte Abwehrmaßnahmen wie die Umleitung des Datenverkehrs, die Einrichtung von Honeypots und die Netzwerktrennung in solchen Netzen kostspielig und instabil.[77] Die Verwaltung und Konfiguration von Netzen ist eine hochkomplexe, anspruchsvolle und zeitaufwändige Aufgabe.[78]

Als aufstrebendes Paradigma vereinfacht und verbessert Software-defined Networking (SDN) die Netzwerkverwaltung, indem es hochflexible Netzwerke bereitstellt, die auf dem Prinzip der Trennung von Kontroll- und Datenebenen basiert.[79] Informationen, die normalerweise über Routing- und Switching-Geräte verteilt sind, werden nun in einem zentralen SDN-Controller verwaltet.

Die SDN-Architektur setzt sich aus drei Hauptkomponenten und zwei schichtübergreifende Kommunikationsschnittstellen zusammen: Anwendungs-, Kontroll- und Datenebene sowie Northbound- und Southbound-Interface. Die Anwendungsebene enthält die Anwendungen, die die Richtlinien, die Paketweiterleitungsschemata und das Netzwerkverhalten definieren.
Die Kontrollebene kann als das Netzwerkbetriebssystem gesehen werden, hier übersetzen SDN-Controller die Richtlinien in Weiterleitungsregeln, die über die gesamte Netzwerkinfrastruktur verteilt werden. Zudem werden hier Low-Level-Netzwerkdetails für Netzwerk- und Verwaltungsanwendungen abstrahiert.
Die Datenebene bezieht sich auf die Weiterleitungsgeräte, die die Netzinfrastruktur bilden und deren Hauptfunktionen darin bestehen, Weiterleitungsaktionen für Datenpakete gemäß den entsprechenden Anweisungen des Controllers zu erzwingen und Netzwerkstatusmaßnahmen zu melden, wenn sie von Netzwerkanwendungen angefordert werden.[80]
Die Verbindung zwischen der Anwendungs- und Kontrollebene wird als Northbound-Interface bezeichnet. Das Southbound-Interface bezeichnet den Kontrollkanal der als Verbindung zwischen Kontroll- und Datenebene dient. Um in einer SDN-Architektur nützlich zu sein, müssen Weiterleitungselemente, hauptsächlich Switches, eine nach Southbound-API unterstützen.[81] OpenFlow ist das am weitesten verbreitete Southbound-Interface Protokoll.[82]

SDN bietet Netzwerkprogrammierbarkeit durch Protokolle wie OpenFlow[83], reduziert den Bedarf an individuell angepassten Netzwerkgeräten und ermöglicht die Implementierung neuer Netzwerkalgorithmen und -protokolle auf der Kontrollebene, ohne dass Netzwerkgeräte verändert werden müssen.[79]
Wie jede neue Technologie hat auch SDN seine eigenen Vor- und Nachteile. Im Hinblick auf die Sicherheit kann die SDN-Technologie beispielsweise dazu genutzt werden, einige Risiken und Schwachstellen, die in

konventionellen Netzwerken häufig ausgenutzt werden, zu verringern oder ganz zu beseitigen. Leider führt die SDN-Technologie neue Schwachstellen und Bedrohungsvektoren ein, die ihrer neuartigen Architektur geschuldet sind.[80] Die Ermittlung der Ursache und der Quelle von Angriffen in SDN ist eine schwierige Aufgabe, denn die in konventionellen Netzwerken verwendeten Methoden zum Sammeln von Beweisen sind für den SDN-Kontext nicht geeignet.[56]

4.5.3 VXLAN

Virtual eXtensible LAN (VXLAN) ist im RFC 7348[64] definiert und ist eine Overlay-Technologie zur Bereitstellung von Layer-2- und Layer-3-Konnektivität über ein herkömmliches IP-Netz. IP-Netze bieten Skalierbarkeit, Load Balancing und vorhersehbare Fehlerbehebung. VXLAN erreicht dies durch das Tunneln von L2-Frames in IP-Pakete. Der VXLAN-Standard definiert die Verkapselung, die einen 8-Byte-Header verwendet, der aus einem 24-Bit-Identifikator (VNID)[65] und mehreren reservierten Bits besteht. Der VXLAN-Header wird zusammen mit dem ursprünglichen Ethernet-Rahmen als UDP-Last platziert. Die 24-Bit-VNID wird verwendet, um L2-Segmente zu identifizieren und die Isolierung zwischen ihnen aufrechtzuerhalten.[75] VXLAN kann als eine Erweiterung von VLAN angesehen werden. VLAN kann nur 4096 Netze identifizieren, was bei weitem nicht ausreicht, um die Anforderungen eines großen Rechenzentrums zu erfüllen. VXLAN kann hingegen 16 Millionen Netzwerke identifizieren.[84]

Die Verwendung von VXLAN erschwert die Erfassung und Aufzeichnung der Netzwerkdaten nicht, da diese unabhängig von den intern verwendeten Protokollen ist. Allerdings beeinflusst die Position des Paketerfassungssystems die Art und Menge der aufgezeichneten Netzwerkpakete. Wenn dieses in der Nähe des verdächtigen Zielsystems läuft, wird die Menge an irrelevantem Netzwerkverkehr reduziert, ist jedoch aber schwieriger zu implementieren. Wenn das Paketerfassungssystem in der Nähe eines Uplinks installiert wird, ist dies einfacher zu implementieren, erhöht aber die Menge an irrelevanten Daten. Zudem müssen bei der Analyse der Netzwerkdaten diese zuerst entkapselt werden müssen.[7]

4.5.4 Herausforderungen

Bewährte Techniken und Methoden scheitern an der der erhöhten Komplexität und Flexibilität der neuen logischen Netzwerke. Alle Phasen des netzwerkforensischen Prozesses sind mit verschiedenen Problemen konfrontiert, die mit den virtuellen Netzwerken zusammenhängen. Die hohe Dynamik der Umgebung er-

[64] https://datatracker.ietf.org/doc/html/rfc7348 [Zugriff am: 12.08.2021]
[65] VNID - VXLAN Network Identifier

schwert die Erfassung und Speicherung der Beweismittel und die Verwendung neuer Netzwerkprotokolle wie VXLAN erschweren die anschließende Analyse der gespeicherten Daten.[7] In der Arbeit von Khan u. a., die als Motivation und Anleitung zur Entwicklung von innovativen SDN-Forensik Standards und Frameworks dienen soll, werden folgende Herausforderungen beschrieben[56]:

Vertrauenswürdige Log-Daten

Logs werden vom Controller, den Northbound- und Southbound-APIs, den Switches und verschiedenen Netzwerkgeräten erstellt. Ein forensischer Ermittler steht vor der Herausforderung der Datenintegrität in den Log-Dateien, die vor jeder unrechtmäßigen oder rechtmäßigen Änderung geschützt werden sollten. Denn eine/ein Angreiferin/Angreifer kann diese ändern, nachdem sie/er seine bösartigen Aktivitäten ausgeführt hat. Im SDN fehlt ein vertrauenswürdiger, aktueller Mechanismus, um die Originalität der in den Log-Dateien enthaltenen Daten zu gewährleisten.

Performance

Ein forensischer Mechanismus benötigt ausreichend Zeit und Rechenleistung, um die aus verschiedenen Quellen eines SDN gesammelten Beweise zu untersuchen. Der Controller ist jedoch für die Verwaltung des gesamten Netzwerks in einem SDN verantwortlich und ist möglicherweise nicht in der Lage, ausreichend Zeit und CPU-Zyklen für die Durchführung einer forensischen Untersuchung bereitzustellen. Dies kann zu Verzögerungen der Ermittlung führen und die/der Angreiferin/Angreifer bekommt dadurch mehr Zeit.

Synchronisation

Die Beweise können von verteilten Standorten in einem SDN gesammelt werden, die von verschiedenen Controllern kontrolliert werden und für verschiedene Domänen zuständig sind. In dieser Situation sollte eine/ein forensische/forensischer Ermittlerin/Ermittler sicherstellen, dass die Beweise synchronisiert werden, um einen genauen forensischen Ermittlungsprozess zu realisieren. Kritisch wird es dann, wenn sich die SDN-Controller an Standorten mit verschiedenen Zeitzonen befinden.

Identifizierung des Quelle

Die Identifizierung der Quelle ist einer der wichtigsten Parameter bei forensischen Untersuchungen, um einen Angriff nach seiner Entdeckung zu entschärfen. Die Identifizierung der Quelle eines Angriffs ist jedoch in einem SDN aufgrund von Spoofing und der häufigen Migration von Instanzen eine schwierige und oft zu komplexe Aufgabe.

Middlebox Seurity

Eine Middlebox ist ein Netzwerkgerät, das eine bestimmte Netzwerkfunktion bereitstellt (Firewall,

LoadBalancer, IDS, Kontentfilter,...). Middleboxen spielen eine wichtige Rolle in der Netzwerk-Forensik, bei der der Netzwerkverkehr untersucht wird, ohne andere Netzwerkabläufe zu beeinträchtigen. Middleboxen müssen ausreichend sicher sein, um genaue Ergebnisse für die Untersuchung des Netzwerkverkehrs zu liefern. Es gibt zahlreiche Möglichkeiten für eine/einen Angreiferin/Angreifer, Middleboxen zu beeinflussen, die dann bei der Untersuchung des Netzwerkverkehrs gefälschte Ergebnisse liefern. Im Vergleich zu traditionellen Netzwerken ist es für ein SDN einfacher, Middleboxen für die forensische Untersuchung zu verwenden, als forensische Tools einzusetzen. Die vollständige Abhängigkeit von Middleboxen ist jedoch riskant. Eine Middlebox, die in einer sicheren Umgebung läuft, gewährleistet die Vertrauenswürdigkeit der Ergebnisse des forensischen Prozesses. Die Sicherheit von SDN bei der ursprünglichen Entwicklung wurde nicht als eines der Hauptmerkmale angesehen, so kann man bei SDN von keiner sicheren Umgebung sprechen.

4.5.5 Forensik Frameworks

Die SDN-Forensik ist noch ein sehr junges Forschungsfeld, deshalb gibt es zum Zeitpunkt dieser Arbeit nur wenige Studien, die sich mit der Untersuchung von Netzwerkangriffen oder Fehlerereignissen in SDN-basierten Netzwerken beschäftigen.

ForCon

Spiekermann u. a. stellen in ihrer Arbeit ForCon[74] vor. Einen forensischen Controller zur Durchführung von Netzwerkuntersuchungen in OpenFlow-gesteuerten Netzwerken mit Open vSwitch. Die Entwicklung des Tools ForCon wurde durch die Notwendigkeit vorangetrieben, einen Datenerfassungsprozess zu implementieren, der in der Lage ist, die virtuelle Umgebung zu überwachen und eine schnelle und valide Reaktion auf Netzwerkänderungen zu ermöglichen. ForCon ist in Python implementiert und benötigt keine zusätzliche Setup-Routine auf einer bestimmten virtuellen oder physischen Maschine im Netzwerk. Es wird über eine Kommandozeilen-Schnittstelle bedient. Durch Eingabe der entsprechenden Parameter steuert ForCon seine verteilten Agenten, um die Umgebung zu überwachen und einen stabilen und korrekten forensischen Prozess zu gewährleisten.

Das System besteht aus dem ForCon-Controller, der die Steuerung der Agents übernimmt und zwei verschiedene Agents. Die SDN-Agents laufen auf den physikalischen Rechenknoten, die die verschiedenen VMs und vSwitches hosten. Sie analysieren die installierten vSwitches und die angeschlossenen Geräte der einzelnen vSwitches. Diese Informationen werden zum ForCon-Controller übermittelt. Dieser wiederum

kann, wenn nötig die Manipulation der Flows in Auftrag geben. Zum beispiel um den Traffic der Ziel-VM zum Mirror-Agent zu duplizieren oder wenn die Ziel-VM auf einen anderen physikalischen Rechenknoten migriert wird. Zusätzlich gibt es noch einen Mirror-Agent, dieser befindet sich auf dem physikalischen Host des Capture-Systems. Dieser baut (wenn der Befehl vom ForCon-Controller kommt) ein VXLAN-Tunnel zum vSwitch, an dem die Ziel-VM verbunden ist, auf. Dadurch kann sich das Capture-System an einem beliebigen Bereich im Netzwerk befinden und die Abschottung zum restlichen Netzwerk wird durch den VXLAN-Tunnel gewährleistet.

Bei ForCon handelt es sich nur um einen Proof-of-Concept und funktioniert nur in Kombination mit OVS (Open vSwitch). Zudem ist dieses System nur auf einen konkreten Fall ausgelegt: Eine/Ein forensische/-forensischer Ermittlerin/Ermittler wird in ein DataCenter gerufen und soll die Paketerfassung eines Ziel-Systems starten (es kann nur ein System überwacht werden). Dadurch werden die Anwendungsgebiete dieser Lösung stark reduziert.

SDNForensics

Zhang u. a. entwickelten das Framework SDNForensics[85]. Es wurde auf der Grundlage von SDN-Schichten und Sicherheitsobjekten entwickelt, das heißt es werden verschiedene Extraktionsmethoden für verschiedene SDN-Ebenen durchgeführt. Die Hauptbestandteile von SDNForensics werden im Folgenden beschrieben:

- Datenerfassung: Verschiedene Arten von Daten werden aus unterschiedlichen Schichten gewonnen. Von der Anwendungsebene werden Laufzeitprotokollinformationen eingeholt. Von der Kontrollebene werden Speicher- und Laufzeitinformationen eingeholt. Von der Datenebene werden Speicher-, Netzwerkpaket- und Laufzeitprotokollinformationen abgerufen. Laufzeit-, Speicher- und Festplatteninformationen werden von den Hosts bezogen.
- Datenextraktion: Startzeit, Endzeit und Fehlermeldungen können aus Laufzeit-Logs extrahiert werden. Aus Speicherabbildern können zahlreiche Echtzeitinformationen extrahiert werden. Durch die Analyse eines Speicherabbilds von einem SDN-Switch können beispielsweise der Typ des verwendeten SDN-Switches und Controllers, die angeschlossenen Hosts und die Flussregeln extrahiert werden. Aus dem Speicherabbild von Hosts können laufende Prozesse, geladene Module, Netzwerkverbindungen, Systemkonfigurationsinformationen und laufende interne virtuelle Maschinen extrahiert werden.
- Datenfusion: Informationen unterschiedlicher Art und aus unterschiedlichen Quellen werden integriert. Dabei dient die Clusteranalyse zur Vorverarbeitung der gesammelten Daten zur Bildung von

Clustern.

- Anomaly Detection: Anomalien können mit Hilfe von Algorithmen des maschinellen Lernens erkannt werden, darunter k-nearest neighbor, support vector machine und auf Fuzzy-Logik basierende Methoden.
- Gemäß der obigen Analyse kann das Sicherheitsalarmmodul auf verschiedene Weise ausgelöst werden, zum Beispiel eine Benachrichtigung an die/den Administratorin/Administrator.
- Beweissicherung: Die verschiedenen Arten von Beweismitteln (gesammelten Daten und Analyseergebnisse) werden in geeigneten Dateiformaten oder Datenbanken gespeichert.

Dieses Framework kann Beweismittel sammeln, um wichtige Hinweise zu extrahieren und zu schlussfolgern was genau im gesamten Netzwerk in Bezug auf Sicherheitsprobleme passiert ist. Es handelt sich jedoch nur um ein theoretisches Modell, das einen Versuch darstellt, ein SDN-Forensik-Framework mit grundlegenden Ermittlungsphasen zu schaffen. Es wird nicht näher darauf eingegangen, wie diese Daten gesammelt werden und ein Netzwerkdiagramm, wo die einzelnen Packet-Capture Systeme implementiert werden wird auch nicht bereitgestellt. Es werden lediglich die Voraussetzungen und die Hauptbestandteile beschrieben, ein Prototyp soll folgen.

ForenGuard

Wang u. a. entwickelten in ihrer Arbeit ForenGuard[86], das Forensik- und Diagnosefunktionen auf Flow-Ebene in SDN-Netzwerken bietet. Dabei überwacht und protokolliert ForenGuard die Laufzeitaktivitäten und ihre kausalen Abhängigkeiten, die sowohl die SDN-Kontroll- als auch die Datenebene einbeziehen. ForenGuard arbeitet auf der SDN-Kontrollebene und stört den normalen Betrieb anderer Controller-Anwendungen nicht. Dieser Prototyp kann frühere Aktivitäten sowohl in der Kontroll- als auch in der Datenebene überwachen und zurückverfolgen, um die Grundursache des Problems zu ermitteln. Der Prototyp wurde auf einen Floodlight[66]-Conroller implementiert.

Das System besteht aus drei Modulen:

- Pre-Prozessor: Er analysiert statisch den Quellcode eines SDN-Controllers zur Extraktion von Aktivitäten, erstellt Datenabhängigkeitsdiagrammen und instrumentiert den Controller.
- Activity Logger: Dieses Modul arbeitet als Controller-Komponente und sammelt dynamisch Aktivitäten von der Kontroll- und der Datenebene und baut die kausalen Abhängigkeitsbeziehungen auf.

[66] http://www.projectfloodlight.org/floodlight/ [Zugriff am: 07.08.2021]

- Diagnosis: Hierbei handelt es sich um ein Kommandozeilen-Tool für die Benutzerinnen/Benutzer zur Abfrage der aufgezeichneten Aktivitäten im SDN-Framework. So kann zum Beispiel nach bestimmten Zeitspannen, Quell/Ziel IP/Port oder Routing-Schleifen gefiltert werden.

Einschränkungen dieses Prototypen sind unter anderem, dass die Implementierung derzeit nur auf Floodlight-Controller funktioniert. Zudem wird davon ausgegangen, dass keine bösartigen SDN-Apps in Betrieb sind. Darüber hinaus kann ForenGuard von ausgenutzten Netzwerkanwendungen überlistet werden, sobald der Controller kompromittiert wurde.

PivotWall

OConnor u. a. stellen in ihrer Arbeit PivotWall[87] vor, dieses nutzt eine neuartige Kombination aus Informationsflussverfolgung und Software-Defined Networking (SDN), um eine breite Palette von Angriffen zu erkennen, die von fortgeschrittenen Angreiferinnen/Angreifern eingesetzt werden, einschließlich solcher, die sowohl Protokolle der Anwendungs- als auch der Netzwerkebene missbrauchen. Darüber hinaus ermöglicht er eine Vielzahl von Angriffsreaktionen, einschließlich der Steuerung des Datenverkehrs, sowie fortschrittliche Mechanismen für die forensische Analyse. Dabei wird die Verfolgung des Informationsflusses auf jedem Host in die Netzwerkverteidigung einbezogen. Die Architektur von PivotWall umfasst drei Komponenten:

- Host Agent: Jeder Host im Netzwerk wird von einem modifizierten SimpleFlow Kernel verwaltet. Dieser kennzeichnet Pakete, die von Prozessen stammen, die möglicherweise vertrauliche Informationen gelesen haben und labelt Prozesse, die ein Paket mit einer vertraulichen Kennzeichnung lesen. Der Host Agent verwaltet die Herkunft der Daten und sendet Kontrollnachrichten an die Kontroller.
- Kontrollebene und SDN-Kontroller: Eine leichtgewichtige OpenFlow-Sicherheitsanwendung basierend auf POX[67] erstellt die erforderlichen Änderungen am Netzwerkfluss, damit die Flüsse mit vertraulichen Paketen diese zur Prüfung an die Kontrollebene weiterleiten können. Auf der Kontrollebene implementiert die Sicherheitsanwendung den Richtlinienspeicher, den NIFC-Graphen (Network Information Flow Control) und den Referenzmonitor. Die Sicherheitsanwendung nutzt diese Komponenten, um Datenflüsse auf Verstöße zu untersuchen, und implementiert acht primitive Aktionen zur Behandlung vertraulicher Datenflüsse. Zu diesen Aktionen gehören eindeutige Methoden zur Umleitung, Drosselung oder Modifizierung vertraulicher Datenflüsse.

[67] https://github.com/noxrepo/pox [Zugriff am: 03.08.2021]

- Datenebene: Die Netzwerkhardware implementiert die OpenFlow-Flussmodifikationen, um gekennzeichnete Flüsse zur Überprüfung an die Steuerebene zu liefern. Die Netzwerkhardware ändert die Flows auf der Grundlage der Anweisungen des SDN-Controllers.

PivotWall verwendet die Taint-Analyse zur Erkennung von Angriffen, dadurch können die Protokolle innerhalb der SDN-Anwendung und des Host Agenten zur Erstellung eines Provenance-Graphen verwendet werden, der den Verlauf des Angriffs beschreibt. Die Erstellung dieses Graphen und die Zusammenführung der Daten über das Netzwerk liefern einem Incident-Response-Team und forensischen Analystinnen/Analysten wertvolle Informationen, um den Angriff einzudämmen und die/den Angreiferin/Angreifer aus dem internen Netzwerk zu entfernen. PivotWall wurden zwar gegen eine breite Palette von Angriffswerkzeugen testeten, jedoch handelt es sich sowohl für den Host Agent als auch für die Kontroller-Anwendung nur um Prototypen.

SDNLog-Foren

Duy u. a. schlagen in ihrer Arbeit einen auf Blockchain basierenden Ansatz vor, um die Sicherheit der Log-Verwaltung in SDN für die Netzwerk-Forensik zu verbessern, genannt SDNLog-Foren[88]. Die zwei Hauptaufgaben dieses Ansatzes sind die Log-Sammlung und die Log-Analyse für die Speicherung. Das Modell besteht grundsätzlich aus folgenden Komponenten:

- SDNLog-Foren Agents: Diese befinden sich in den Netzwerkgeräten im SDN (Controller, Switches, Hosts usw.), um Log-Informationen in Echtzeit von diesen Objekten zu lesen und an das Sammelelement zu senden. Zudem wurde das Modul Forensics Manager für diese Komponente entwickelt, um alle Log-Informationen der im SDN-Netzwerk eingesetzten Agenten zu sammeln.
- Log Collectors und Filter: In einigen Fällen werden nicht alle Log-Informationen analysiert und gespeichert, um zu verhindern, dass die Speicherressourcen durch nicht zugehörige Log-Einträge erschöpft werden. Basierend auf den Anforderungen der/des Administratorin/Administrators oder Ermittlerin/Ermittlers, können einige Filterbedingungen auf die gesammelten Log-Informationen angewendet werden.
- Log Manager: Dies ist die Hauptkomponente des Frameworks, das die Verantwortung für mehrere Aufgaben über API-basierte Interaktionen mit anderen Modulen übernimmt. Zunächst bietet es eine API für Log Collector und Filter, um gefilterte Protokolle zu senden. Dann führt der Log Manager zwei gleichzeitige Aufgaben aus, um die Log-Informationen zu speichern. Einerseits wird das Log

unverändert an die Komponente Log Storage übertragen. Andererseits authentifiziert sich der Log Manager bei der Blockchain, um Anfragen zum Senden von Logs zu stellen.

- Log Storage: Bei diesem Element kann es sich um ein Verzeichnis auf einem Host, in einer Cloud oder um eine Datenbank handeln, um Log- und andere Eingabeinformationen lokal im Framework zu speichern und Blockchain zu verwenden. Diese Aktion ist nützlich für die Unterstützung von Log-Analysatoren (wie Php-syslog-ng, Splunk, EventLog Analyze, ...) die eine lokale Analyse erfordern können.

Die Blockchain-basierte Lösung für die Aufbewahrung von Logs zum Zweck der Netzwerk-Forensik im SDN hat ein großes Potenzial, wesentliche Vorteile für Ermittlungsprozesse zu bringen. Diese Methode kann die Integrität, Authentizität, Manipulationssicherheit, Transparenz, Sicherheit und Revisionsfähigkeit digitaler Beweise und operativer Verfahren, die während der Untersuchung durchgeführt werden, bewahren.

4.5.6 Conclusio

Es konnten fünf verschiedene Frameworks für die SDN-Forensik ausfindig gemacht werden. Zu beachten ist, dass es sich bei allen nur um Prototypen handelt und diese in der Praxis keine Anwendung finden. Bei SDNForensics[85] handelt es sich rein um ein theoretisches Framework. Es wurden lediglich Anforderungen und die Hauptbestandteile des Frameworks beschrieben. Es wird aber an einem Prototyp gearbeitet. Forcon[74] kommt mit sehr viel einschränkungen, so funktioniert es nur mit OVS (Open vSwitch) und es kann immer nur ein System überwacht werden. Zudem kommt, dass dieser Prototyp nicht für Incident-Response anwendbar ist, sondern nur für die/den forensische/forensischen Ermittlerin/Ermittler die/der ein Ziel-System überwachen muss. ForenGuard[86] ist ein weiterer Prototyp, der jedoch einige Einschränkungen mit sich bringt. So kann er nur auf einen Floodlight-Controller implementiert werden. Zudem ist das Threat Model gegen das der Prototyp getestet wurde, sehr eingeschränkt. So wird zum Beispiel davon ausgegangen, dass keine bösartigen SDN-Apps in Betrieb sind. Der große Nachteil von ForenGuard ist, dass wenn der SDN-Controller kompromittiert wird, den gesammelten Beweismittel nicht mehr vertraut werden kann. PivotWall[87] ist ein Verteidigungssystem für die Netzwerksicherheit, bietet jedoch auch Mechanismen für die forensische Analyse. Die Tainting-Analyse wird verwendet, um einen Angriff zu überwachen und eine Warnung zu erzeugen, während die Logs, die von Netzwerkanwendungen und Host-Agenten generiert werden, beschreiben, wie der Angriff durch einen Netzwerk-Informationsfluss-Kontrollgraphen (NIFC) erfolgte. Diese Diagramme enthalten jedoch nur den Fluss von vertraulichen Zugriffen und Daten im gesamten Netzwerk, sei es zwischen oder innerhalb von Hosts und konzentrieren sich auf hostbasierte Informationen.

Wenn eine/ein Angreiferin/Angreifer einen Host komplett kompromittiert, kann er den Host Agent deaktivieren und Daten manipulieren. Des weiteren können keine Host Agents auf IoT-Geräte implementiert werden. Dieser Prototyp konzentriert sich hauptsächlich auf Angriffe, die das Ziel haben, Informationen zu exfiltrieren. SDNLog-Foren[88] ist 2019 publiziert worden und ist somit der aktuellste Prototyp. Hauptaugenmerkt liegt bei der Sicherung sensibler Log-Dateien, die es dem SDN-Netzwerk ermöglichen, Beweise für die digitale Forensik zu sammeln und zu speichern. Durch den Einsatz der Blockchain-Technologie werden die Integrität der Daten und die Fälschungssicherheit gewährleistet. Der Prototyp wurde für Testzwecke erfolgreich auf einem Floodlight-Controller implementiert, die Hosts wurden mittels Docker Container realisiert. Das Ergebnis der Experimente wurden nicht besprochen, so ist nicht klar, ob der Prototyp eine gute Performance geliefert hat. Zudem werden auch keine Limitierungen aufgezeigt. Abschließend kann gesagt werden, dass die SDN-Forensik eine noch sehr junge Wissenschaft ist. Es gibt zwar ein paar Prototypen, jedoch bringt jeder dieser einige Limitierungen. Eine Weiterentwicklung dieser sollte vorangetrieben werden, um ein Framework zu schaffen, das in der Praxis Anwendung finden kann.

4.6 TLS 1.3

Transport Layer Security (TLS) ist das weit verbreitete Standardprotokoll zur Sicherung der Client-Server-Kommunikation über das Internet. TLS ist ein zustandsabhängiges, verbindungsorientiertes Client-Server-Protokoll. Es ist das wohl am weitesten verbreitete Sicherheitsprotokoll für die Kommunikation im Internet und bietet Authentifizierung, Integrität und Vertraulichkeit für zwei Parteien.[89] TLS wird in der Regel zur Verschlüsselung des Inhalts des HTTP-Protokolls verwendet. HTTPS-Protokoll oder HTTP Secure oder Hypertext Transfer Protocol over TLS, ist der Standard für sichere Kommunikation im Internet und wird überwiegend in Computernetzen verwendet.[90] Einem Google-Bericht vom Mai 2021 zufolge, nimmt die Nutzung von HTTPS zu und liegt derzeit bei 79-98 % der geladenen HTTPS-Webseiten im Chrome-Browser, abhängig vom Betriebssystem (Windows, Android, Chrome, Linux und Mac).[68] Mit der zunehmenden Menge an verschlüsseltem Netzwerkverkehr im Internet, haben auch Malware damit begonnen, ihre eigene Kommunikation durch Verschlüsselung zu schützen.[90] Zehn Jahre nach TLS 1.2 wurde eine neue Version des populären Transport Layer Security-Protokolls veröffentlicht.[69] Dieses neue Protokoll mit der Bezeichnung TLS 1.3 bringt mehrere Verbesserungen in Bezug auf Leistung und Sicherheit. Die Änderungen tragen dazu bei, Bedrohungen durch eine Vielzahl von Angriffen zu entschärfen. Die neue Version von TLS ist nicht nur sicherer, sondern verkürzt auch die durchschnittliche Zeit, die für einen TLS-Handshake

[68] https://transparencyreport.google.com/https/overview?hl=de [Zugriff am: 19.08.2021]
[69] https://www.ietf.org/blog/tls13/ [Zugriff am: 23.08.2021]

benötigt wird, was jede Verbindung beschleunigt.[90]

4.6.1 Unterschiede zwischen TLS 1.2 und TLS 1.3

Mit dem Ziel, die allgemeine Sicherheit und den Datenschutz zu verbessern, führt TLS 1.3 mehrere Änderungen gegenüber TLS 1.2 ein. Einige dieser Änderungen wirken sich jedoch negativ auf die netzwerkbasierte Sicherheit aus. Diese Änderungen werden im Internet-Draft draft-camwinget-tls-use-cases-05[70] beschrieben. In diesem Abschnitt werden diese Änderungen kurz beschrieben, zu beachten ist, dass auf generelle Änderungen (wie dem schnelleren Handshake) nicht eingegangen wird.

Entfernung von statischen Cipher-Suites

TLS 1.2 unterstützt statische RSA- und Diffie-Hellman (DH) Cipher-Suites, welche es ermöglichen, dass der private Schlüssel des Servers mit serverseitigen Middleboxes geteilt werden kann. Sprich, wenn man Zugriff auf den privaten Schlüssel des Servers hat, kann man den gesamten TLS-Traffic entschlüsseln. TLS 1.3 hat die Unterstützung für diese Cipher-Suites entfernt und verwendet stattdessen den Diffie-Hellman Ephemeral (DHE) und Elliptic-Curve Diffie-Hellman Ephemeral (ECDHE) als primäre kryptographische Mechanismen zum Schlüsselaustausch. Dabei wird für jede Verbindung ein neuer temporärer privater DH-Schlüssel erzeugt, was Perfect Forward Secrecy (PFS) gewährleistet. Dadurch ist es einem Server nicht mehr möglich, der Middlebox im Vorhinein einen Schlüssel[71] mitzuteilen, was wiederum zur Folge hat, dass die Middlebox keinen Zugriff auf die TLS-Sitzungsdaten erhält. PFS ist bei TLS 1.2 optional, bei TLS 1.3 jedoch verpflichtend. Dies hat unter anderem Auswirkungen auf die Netzwerküberwachung, Fehlerbehebung und Compliance.

Verschlüsseltes Server Zertifikat

Bei TLS wird die ClientHello-Nachricht an die Transportadresse des Servers (IP und Port) gesendet. Die ClientHello-Nachricht kann die Server Name Indication (SNI) enthalten, um den Hostnamen anzugeben, den der Client kontaktieren möchte. Dies ist nützlich, wenn mehrere virtuelle Server an einer bestimmten Transportadresse gehostet werden. Außerdem liefert es Monitoring- und Sicherheitseinrichtungen Informationen über die Domäne, die der Client zu erreichen versucht. SNI ist in TLS 1.2 optional, während es in TLS 1.3 verpflichtend ist. Der Server antwortet mit einer ServerHello-Nachricht, die die ausgewählten Verbindungsparameter enthält, gefolgt von einer Certificate-Nachricht, die das Zertifikat des Servers und damit

[70] https://datatracker.ietf.org/doc/html/draft-camwinget-tls-use-cases-05 [Zugriff am: 23.08.2021]

[71] Privater Schlüssel vom Server, dadurch war es möglich den gesamten TLS 1.2-Traffic zu entschlüsseln.

seine Identität enthält. Hierbei ist zu beachten, dass selbst wenn die SNI vom Client bereitgestellt wird, keine Garantie besteht, dass der tatsächlich antwortende Server derjenige ist, der in der SNI des Clients angegeben ist. Die SNI allein, ohne Vergleich des Serverzertifikats, liefert keine zuverlässigen Informationen über den Server, den der Client zu erreichen versucht. Wenn ein Client durch Malware kompromittiert wurde und eine Verbindung zu einem Command-and-Control-Server herstellt, aber eine harmlose SNI angibt, um Schutzfilter zu umgehen, kann er unter TLS 1.3 nicht erkannt werden. Bei TLS 1.2 werden die Nachrichten ClientHello, ServerHello und Zertifikat im Klartext gesendet, bei TLS 1.3 wird die Zertifikatsnachricht jedoch verschlüsselt, so dass die Identität des Servers vor jedem Dritten verborgen bleibt. Dies hat Auswirkungen auf Middleboxes, die Whitelisting/Blacklisting durchführen. Unter TLS 1.3 muss die Middlebox in diesen Szenarien nun jede Verbindung entschlüsseln und prüfen, um über die gleichen Informationen zu verfügen, die sie für ihre Entscheidungen benötigt. Darüber hinaus ist die Middlebox nicht in der Lage, diese Entscheidungen zu treffen, ohne von Beginn des Handshakes an aktiv an der TLS-1.3-Sitzung teilzunehmen. Zudem kann sie die Verbindung nicht verlassen, sobald sie als gutartig eingestuft wurde, ohne die gesamte Verbindung zu unterbrechen. Bei TLS 1.2 konnten die Middleboxen Entscheidungen auf der Grundlage des Zertifikats treffen, ohne die Verbindung aktiv zu entschlüsseln.

Resumption und Pre-Shared Key

In TLS 1.2 und darunter wird die Wiederaufnahme der Sitzung (Session Resumption) durch Session-IDs und Session Tickets gewährleistet. Wenn der Server ein Ticket nicht anerkennen will, kann er einfach wie üblich einen vollständigen TLS-Handshake mit dem Client initiieren. In TLS 1.3 wird der obige Mechanismus durch Pre-Shared Keys (PSK) ersetzt, die als Teil eines anfänglichen Handshakes ausgehandelt und dann in einem nachfolgenden Handshake verwendet werden können, um die Wiederaufnahme unter Verwendung des PSK durchzuführen. TLS 1.3 besagt, dass der Client eine *key_share*-Erweiterung enthalten sollte, um dem Server die Möglichkeit zu geben, die Wiederaufnahme abzulehnen und zu einem vollständigen Handshake zurückzukehren, dies ist jedoch keine absolute Voraussetzung. Davon betroffen sind beispielsweise Middleboxen, die nicht Teil des ursprünglichen Handshakes waren und daher den PSK nicht kennen. Wenn der Client die *key_share*-Erweiterung nicht enthält, kann die Middlebox keinen Fallback zum vollständigen Handshake erzwingen. Wenn die Middlebox-Richtlinie eine Überprüfung der Sitzung vorschreibt, muss sie stattdessen die Verbindung fehlschlagen lassen.

Version Negotiation und Downgrade Protection

Bei TLS enthält die ClientHello-Nachricht eine Liste der unterstützten Protokollversionen. Der Server wählt die höchste unterstützte Version aus und gibt seine Wahl in der ServerHello-Nachricht an. TLS 1.3 ändert die Art und Weise, in der die Versionsaushandlung durchgeführt wird. In der ClientHello-Nachricht wird die TLS-Version 1.3 in der neuen Erweiterung *supported_versions* angegeben, aus Gründen der Abwärtskompatibilität mit TLS 1.2 wird in der ClientHello-Nachricht jedoch die TLS-Version 1.2 im Feld *legacy_version* angeben. Ein TLS 1.3 Server wird erkennen, dass TLS 1.3 ausgehandelt wird, während ein TLS 1.2 Server einfach ein TLS 1.2 ClientHello sieht und mit der TLS 1.2 Aushandlung fortfährt. In TLS 1.3 enthält der Zufallswert in der ServerHello-Nachricht einen speziellen Wert in den letzten acht Bytes, wenn der Server entweder TLS 1.2 oder TLS 1.1 und darunter aushandelt. Dieser spezielle Wert ermöglicht es einem TLS 1.3-Client, eine/einen aktive/aktiven Angreiferin/Angreifer zu erkennen, die/der einen Downgrade-Angriff startet, wenn der Client tatsächlich einen TLS 1.3-Server erreicht hat. Voraussetzung hierfür ist die Verwendung von ephemerale Cipher. Aus Sicht der Netzsicherheit besteht die wichtigste Auswirkung darin, dass bei TLS 1.3 der TLS-Proxy von Beginn des Handshakes an ein aktiver Man-in-the-Middle sein muss.

SNI Verschlüsselung in TLS

Wie bereits erwähnt, ist bei verschlüsselten Serverzertifikaten die Server Name Indication (SNI) in der ClientHello-Nachricht die einzige Information, die im Klartext verfügbar ist, um den Zielserver des Clients anzugeben. Im RFC 8744[72] wird vorgeschlagen, die SNI in der ClientHello Nachricht zu verschlüsseln. Dadurch soll der Datenschutz und die Zuverlässigkeit von TLS-Verbindungen verbessert werden. Betroffen sind hier Middleboxes, die Whitelisting/Blacklisting durchführen. Ohne die SNI verfügt die Middlebox möglicherweise nicht über die erforderlichen Informationen, um diese Tätigkeit durchzuführen.

4.6.2 Möglichkeiten zur Analyse von TLS 1.3 Traffic

TLS Interception via MitM

Die gängigste Methode zur Entschlüsselung des TLS-Verkehrs ist die Verwendung eines Man-in-the-Middle (MitM)-Proxys, der alle Verbindungen aufbricht und somit Zugriff auf die Klartextdaten hat. Dieser Ansatz wird häufig in Unternehmensnetzwerken eingesetzt und gilt in der Branche als Best Practice.[57] Der MitM-Proxy wird zwischen dem Client und dem Server platziert, wo der verschlüsselte Datenverkehr entschlüsselt, auf unerwünschten Traffic gescannt, erneut verschlüsselt und an die Ziel-IP gesendet wird. Der Proxy ver-

[72] https://datatracker.ietf.org/doc/html/rfc8744 [Zugriff am: 23.08.2021]

wendet ein selbstsigniertes Zertifikat, das von der Proxy-CA (Certificate Authority) ausgestellt wurde, um mit dem Client verschlüsselt zu kommunizieren. Dabei muss der Client der Proxy-CA vertrauen, sprich das Root-Zertifikat muss importiert werden. Dieser Ansatz ermöglicht die Verwendung klassischer Methoden zur Erkennung von unverschlüsseltem Malware-Verkehr und stellt eine erhebliche Vereinfachung des Problems dar. Aufgrund der Popularität von MitM-Proxys wurde das Intercepten von TLS in den letzten Jahren vielfach diskutiert. Dabei sind sich alle bei einem Punkt einig, es werden dabei die Sicherheitseigenschaften der inspizierten Verbindungen erheblich geschwächt.[91]–[96] Im Dezember 2019 veröffentlichte die National Security Agency (NSA) einen Bericht[73], in dem ein potenzielles Risiko durch die unsachgemäße Verwendung der TLS Inspection (TLSI) beschrieben wird. Laut diesem Bericht sollte TLSI nicht als Allheilmittel gesehen werden und dass große Anstrengungen unternommen werden sollten, um TLS Inspection richtig zu implementieren. Andernfalls kann das Netzwerk mit TLSI anfälliger und gefährlicher werden als ohne.[90]

Modifizierung des TLS-Protokolls

TLS wurde so konzipiert, dass es zwischen genau zwei Entitäten eine Ende-zu-Ende-Verschlüsselung unterstützt. Es gibt jedoch Modifikationen am TLS-Protokoll, die es mehreren Entitäten ermöglichen den TLS-Verkehr zu entschlüsseln. Naylor u. a.[97] schlagen in ihrer Arbeit Multi-Context-TLS (mcTLS) vor, das TLS erweitert, um Middleboxen zu unterstützen. Dabei wird der TLS-Header um mehrere sogenannte Kontexte erweitert. Diese Kontexte fungieren als Berechtigungssystem und ermöglichen es dem Client, ausgewählten Middleboxen das Lesen und/oder Schreiben der TLS-Payload zu erlauben. Diese zusätzlichen Kontexte erfordern zusätzliche Berechnungen auf dem Client und führen zu erhöhten Latenzzeiten beim Verbindungsaufbau. Zudem ist der modifizierte TLS-Header mit Kontexten inkompatibel zum standardmäßigen TLS und zwingt daher alle Knoten auf dem Pfad zur Unterstützung von mcTLS.

Naylor u. a.[98] schlagen Middlebox TLS (mbTLS) vor, das den TLS-Handshake erweitert. Damit ermöglicht mbTLS ausgelagerten Middleboxen die Teilnahme an einer TLS-Sitzung. Wenn Endpunkte einen TLS-Handshake miteinander durchführen, öffnet jeder Endpunkt eine sekundäre TLS-Sitzung zu einer oder mehreren Middleboxen. Nach dem primären TLS-Handshake senden die Endpunkte den Sitzungsschlüssel an jede der Middleboxen. Falls ein Endpunkt kein mbTLS unterstützt, kann auf das standardmäßige TLS zurückgegriffen werden. mbTLS bietet jedoch keine Informationen über die Geheimhaltung der einzelnen

[73] https://media.defense.gov/2019/Dec/16/2002225460/-1/-1/0/INFO%20SHEET%20%20MANAGING%20RISK%20FROM%20TRANSPORT%20LAYER%20SECURITY%20INSPECTION.PDF [Zugriff am: 24.08.2021]

Segmente oder Änderungen am TLS Verkehr. Stattdessen verlassen sich die Endpunkte auf die Intel Software Guard Extensions (SGX)[74] Technologie , um die Integrität der Middlebox zu gewährleisten. Dadurch wird die Bereitstellung und Wartung erschwert, da die Middleboxen nun eine Hardware-Enklave benötigen. Außerdem wird durch die sekundären TLS-Sitzungen eine zusätzliche Latenzzeit verursacht.

Das vom Europäischen Institut für Telekommunikationsnormen (ETSI) vorgeschlagene Enterprise Transport Security (ETS)[75] Profil ändert den TLS 1.3-Standard dahingehend, dass die Forward Secrecy optional gemacht wird und die Verwendet längerfristiger, statischer Diffie-Hellman-Schlüssel ermöglicht, die über mehrere Sitzungen hinweg wiederverwendet werden. ETS ist mittlerweile als Schwachstelle mit der Kennzeichnung CVE-2019-9191[76] gelistet. Auch wurde von der IETF ein Internet-Draft[77] veröffentlicht, der eine optionale Konfiguration für TLS-Server beschreibt, die die Verwendung eines statischen, privaten Diffie-Hellman-Schlüssels für alle TLS-Verbindungen zum Server ermöglicht. Dieser ist jedoch 2018 abgelaufen.

Middlebox-aware TLS[99] modifiziert TLS so, dass es den Middleboxen ermöglicht wird, an der TLS-Sitzung auf sichtbare und überprüfbare Weise teilzunehmen. Jede teilnehmende Middlebox teilt in Zusammenarbeit mit den beiden Endpunkten eine Sitzung in zwei Segmente mit eigenen Sicherheitsparametern auf. Diese Sicherheitsparameter werden an den jeweiligen Client weitergeleitet. Das maTLS-Protokoll ist darauf ausgelegt, die Middleboxen zu authentifizieren, um die Sicherheitsparameter der Segmente zu überprüfen und die Schreiboperationen der Middleboxen zu kontrollieren. Jede Middlebox verfügt über ein eigenes Zertifikat, das von einer/einem Dritten überprüft werden kann. Wie mcTLC ist auch dies inkompatibel zu Standard-TLS mit allen Nachteilen. Wie bei mcTLS und mbTLS erfordert vor allem der Verbindungsaufbau zusätzliche Rechenzeit und es kommt zu Verzögerungen durch die zahlreichen Entitäten, die potenziell Berechnungen auf der Verbindung durchführen.

Trusted Execution Environments

Trusted Execution Environments (TEE) stellen eine vertrauenswürdige und sichere Laufzeitumgebung bereit. Im Kontext mit TLS werden sie verwendet, um Funktionalitäten einer Middlebox, wie die Netzwerküberwachung, an Kommunikationsendpunkten sicher einzusetzen.

[74] https://www.intel.de/content/www/de/de/architecture-and-technology/software-guard-extensions.html [Zugriff am: 24.08.2021]
[75] https://www.etsi.org/deliver/etsi_ts/103500_103599/10352303/01.03.01_60/ts_10352303v010301p.pdf [Zugriff am: 24.08.2021]
[76] https://nvd.nist.gov/vuln/detail/CVE-2019-9191 [Zugriff am: 24.08.2021]
[77] https://datatracker.ietf.org/doc/html/draft-green-tls-static-dh-in-tls13-01 [Zugriff am: 24.08.2021]

Goltzsche u. a.[100] stellen in ihrer Arbeit ENDBox vor, ein System, das Middlebox-Funktionen sicher auf Client-Rechnern ausführt. Das Konzept kombiniert ein virtuelles privates Netzwerk (VPN) mit Middlebox-Funktionen, die durch eine TEE hardwaregeschützt sind. Dabei werden die Session-Keys über die OpenVPN-Verwaltungsschnittstelle in das TEE gesendet. Mit diesem Ansatz kann ENDBox den Datenverkehr für den Client transparent entschlüsseln. Der Client muss weder einer benutzerdefinierten Zertifizierungsstelle vertrauen noch sieht er andere Zertifikate als die, die von den aufgerufenen Diensten angeboten werden. Um sicherzustellen, dass der gesamte Netzwerkverkehr von der Middlebox abgefangen wird, kann sich ein Client nur über das VPN mit dem Netzwerk verbinden. Der VPN-Client verarbeitet jedes IP-Paket einzeln in vier Schritten. Nachdem das Paket innerhalb der Enklave kopiert wurde, wird es von einer oder mehreren Middlebox-Funktionen entsprechend der Systemkonfiguration verarbeitet. Je nach der spezifischen Funktion kann der Header oder die Payload geändert oder das gesamte Paket als zu verwerfen markiert werden. Nach der Ausführung der Middlebox-Funktionen wird das Paket entweder angenommen oder abgelehnt. Abschließend wird das Paket signiert und verschlüsselt und dann außerhalb der Enklave kopiert, wo es zur Übertragung über das Netz an den im nicht vertrauenswürdigen Bereich laufenden VPN-Client zurückgegeben wird. Aus Sicht des Datenschutzes besteht der Hauptvorteil darin, dass die Verbindungsdaten zwischen Client und Server unangetastet bleiben. Der Ansatz basiert auf Intel SGX und schränkt damit die auf den Clients einsetzbare Hardware ein. ENDBox implementiert Middlebox-Funktionen unter Verwendung des Click Modular Router[78] was die Zahl der Anwendungen einschränkt. Zudem bestehen hohe Wartungs- und Installationsanforderungen.

Trach u. a.[101] stellen ShieldBox vor, ein sicheres Middlebox-Framework für den Einsatz von Hochleistungs Network Functions (NFs) über nicht vertrauenswürdige Server. ShieldBox verarbeitet verschlüsselten Datenverkehr innerhalb eines sicheren Containers, indem es Intel SGX verwendet, um Vertraulichkeit und Integrität zu gewährleisten. Um bestehende Software nutzen zu können, wurde eine Standard-C-Bibliothek (libc) namens SCONE entwickelt, die die Ausführung in einer Intel SGX-Umgebung unterstützt. Für die NFs wird ebenfalls der Click Modular Router verwendet. Zur sicheren Verarbeitung von Daten mit Leitungsgeschwindigkeit integriert ShieldBox eine leistungsstarke I/O-Verarbeitungsbibliothek namens Intel DPDK[79]. Durch Evaluation von verschiedenen NFs und Fallstudien konnten sie feststellen, dass ShieldBox einen nahezu nativen Durchsatz und Latenzzeit erreicht. Dieser Ansatz hat jedoch auch einige Nachteile. Es

[78] https://github.com/kohler/click [Zugriff am: 24.08.2021]

[79] https://www.intel.de/content/www/de/de/communications/data-plane-development-kit.html [Zugriff am: 24.08.2021]

gibt erhebliche Einschränkungen bei der Speichernutzung und viele Operationen (insbesondere Systemaufrufe) nehmen mehr Zeit in Anspruch. Außerdem bleiben die gleichen hohen Installations- und Wartungsanforderungen wie bei Endbox bestehen.

Passive TLS Decryption

Bei der passiven TLS-Entschlüsselung müssen zunächst die Endpunkte, in dem durch die Middlebox geschützten Netz, das TLS-Schlüsselmaterial erhalten. Das benötigte Schlüsselmaterial besteht bei TLS 1.3 aus dem Pre-Master Secret und dem Client Random (Zufallswert). Der Client Random dient dabei zur eindeutigen Identifizierung der Verbindung. Danach müssen diese an die Middlebox übertragen werden. Die Middlebox führt dann eine Schlüsselableitung durch, um alle Sitzungsschlüssel und IVs, wenn eine AEAD-Cipher-Suite[80] verwendet wird, zu erhalten. Schließlich gleicht die Middlebox das Schlüsselmaterial mit den Verbindungen ab und beginnt mit der passiven Entschlüsselung und weiteren Analyse. Für nachfolgende, wiederaufgenommene Verbindungen (TLS Session Resumption) muss die Middlebox dann auch das Schlüsselmaterial mit dem Sitzungsidentifikator verknüpfen. Der TLS-Traffic vom Client wird dupliziert, durch Port Mirroring oder Netzwerk-TAP, und an die Middlebox weitergeleitet. Dies ist entweder die Session-ID, das Session Ticket oder TLS 1.3 PSK (Pre-Shared Key), je nach TLS-Version und Implementierung.

Wilkens u. a.[57] stellen in ihrer Arbeit einen Prototypen vor, der auf diesem Konzept basiert. Der Prototyp besteht aus einer gepatchten Zeek[81] Version als NMS (Network Monitoring System) und einem einfachen Python-Daemon der auf Endpunkten läuft und Schlüsselmaterial an Zeek weiterleitet. Er unterstützt nur eine Cipher-Suite, nämlich *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384*. Diese kommt häufig in TLS 1.2 zum Einsatz und bietet PFS. Dieser verwendet den ECDHE-Schlüsselaustausch mit RSA zur Authentifizierung und AES im Galois/Counter Mode (GCM) zur symmetrischen Verschlüsselung der Payload. Das Schlüsselmaterial wird mittels der SSL-Keylog-Schnittstelle, die von weit verbreiteten kryptografischen Bibliotheken wie openSSL[82] und NSS[83] unterstützt wird, erlangt. Wenn die Schnittstelle aktiviert ist, schreiben die Bibliotheken TLS Pre-Master Secrets und die entsprechenden Client Randoms in die Datei, die in einer Umgebungsvariablen angegeben ist. Die Datei wird von dem Python-Daemon überwacht, welcher diese an Zeek weitergeleitet. Der Python-Daemon hält eine Verbindung zur Zeek-Instanz

[80] Authenticated Encryption with Associated Data
[81] https://zeek.org/ [Zugriff am: 25.08.2021]
[82] https://www.openssl.org/ [Zugriff am: 25.08.2021]
[83] https://developer.mozilla.org/en-US/docs/Mozilla/Projects/NSS [Zugriff am: 25.08.2021]

über Broker[84], der Standard-Kommunikationsbibliothek von Zeek, und überträgt neues Schlüsselmaterial an Zeek, sobald sie in der Datei erscheinen. Der Daemon führt derzeit keine Filterung und unterstützt daher keine selektive Entschlüsselung. Stattdessen werden alle Client-Randoms und Pre-Master-Secrets an Zeek weitergeleitet, unabhängig von den beteiligten Domänen. Die gepatchte Zeek-Version ist in der Lage, sowohl Pre-Master-Secrets als auch abgeleitete TLS-Sitzungsschlüssel zu empfangen. Im ersten Fall wird die TLS-Schlüsselableitung durchgeführt, sobald das erste Paket der entsprechenden Verbindung eintrifft und die daraus resultierenden Sitzungsschlüssel werden zur späteren Verwendung zwischengespeichert. Sobald die Sitzungsschlüssel verfügbar sind, entschlüsselt Zeek die TLS-Anwendungsdaten und leitet den Klartext an seine interne Protokollerkennungs-Engine und nachfolgende Analysatoren weiter. Sowohl die TLS-Schlüsselableitung als auch die eigentliche Entschlüsselung wird von openSSL durchgeführt, da Zeek bereits auf diese Bibliothek verlinkt. Die TLS Session Resumption (Wiederaufnahme von TLS-Sitzungen) wird derzeit nicht unterstützt. Die Evaluation der Arbeit zeigt, dass die TLS-Entschlüsselung die Laufzeit um den Faktor 2,5 erhöht, wenn man sie mit der Analyse von Klartexten vergleicht. Die Komplexitätsanalyse zeigt, dass dieser Ansatz weniger Rechenressourcen benötigt als MitM-Proxys. Sie kommen zur Erkenntnis, dass in ihrer Testumgebung die Schlüssel etwas zu spät ankommen, wenn der Datenverkehr nicht verzögert an das NMS weitergeleitet wird. Eine kleine Verzögerung von 40 ms reicht laut ihrer Analyse aus um 99,99% aller beobachteten TLS-Verbindungen vollständig zu entschlüsseln.

4.6.3 Conclusio

Beim Intercepten der TLS-Verbindung mittels MitM-Proxys gehen die Grundprinzipien, Privatsphäre und Sicherheit, von TLS 1.3 verloren. Der MitM-Proxy erhält den vollen Zugriff auf alle Payloads in Klartext, wodurch die E2E-Datenintegrität verloren geht. Zudem wird durch das Aufbrechen der Verbindungen die Latenzzeit erhöht und man benötigt eine leistungsstarke Appliance, da die TLS 1.3 Verbindungen vom Proxy aufrechterhalten bleiben müssen, bis die Sitzung beendet wird, da es sonst zum Verbindungsabbruch kommt. Der Vorteil eines MitM-Proxys ist, dass der Wartungsaufwand relativ gering ist, da nur der Proxy selbst gewartet werden muss und die CA auf den Clients bereitgestellt werden muss. Des Weiteren muss das TLS-Protokoll nicht modifiziert werden, um Zugriff auf die Klartextdaten zu bekommen. Die Modifizierung des TLS-Protokolls sollte als kritisch betrachtet werden, da es bei diesen Protokollen zu keinen groß angelegten Sicherheitsanalysen von verschiedenen Experten gekommen ist. Das vorgeschlagene Protokoll der ETSI wird sogar als Schwachstelle gelistet und der Internet-Draft der IETF wurde abgelehnt. Daher wird die Modifizierung des Standardprotokolls in dieser Arbeit als keine Alternative angesehen. Die beiden

[84] https://github.com/zeek/broker [Zugriff am: 25.08.2021]

vorgestellten Frameworks in der Kategorie TEE, ENDBox und ShieldBox, verwenden beide Intel SGX, um die Middlebox-Funktionalität an die Endpunkte zu verlagern. Dadurch muss weder das Protokoll modifiziert werden, noch braucht es zusätzliche Middleboxen. Der Aufwand für die Wartung und Bereitstellung ist jedoch hoch, da die Endpunkte mit Enclave-Hardware ausgestattet werden müssen und auf jedem dieser Endpunkte die Software implementiert, konfiguriert und gewartet werden muss. Zudem kommt, dass wenn man zusätzliche Funktionen benötigt, diese aber nicht vom Click Modular Router bereitgestellt werden, sie selbst programmieren muss. Ein weiterer Nachteil dieses Konzeptes ist, dass es keinen zentralen Blick auf das gesamte Netzwerk ermöglicht, da jede Instanz abgeschottet von den anderen arbeitet. Auch sind mehrere Schwachstellen[102] von Intel SGX bekannt. Zur passiven TSL-Entschlüsselung konnte leider nur eine wissenschaftliche Arbeit ausfindig gemacht werden. Der Prototyp von Wilkens u. a. ist in der Lage, bei Vorhandensein von Pre-Master-Secrets oder den entsprechenden Sitzungsschlüsseln, HTTPS/2-Verkehr so zu analysieren, als wäre er im Klartext aufgezeichnet worden. Die größte Einschränkung ist die Unterstützung von nur einer einzigen Ciper-Suite und die fehlende Unterstützung für die Wiederaufnahme von TLS-Sitzungen. Der große Vorteil der passiven TLS-Entschlüsselung ist, dass die E2E-Datenintegrität erhalten bleibt. Dieser Ansatz findet bereits Anwendung in der Praxis, zum Beispiel bietet ExtraHop[85] ihren Kunden diesen an. Sie bezeichnen das Konzept als Session-Key Forwarding[86]. Zu den Kunden, die diese Lösung implementiert haben zählt zum Beispiel Fiserv[87]. Aus netzwerkforensischer Sicht ist dieser Ansatz am besten geeignet, um digitale Beweismittel zu sichern. Da der gesamte Traffic entschlüsselt werden kann und die E2E-Datenintegrität erhalten bleibt.

[85] https://www.extrahop.com/ [Zugriff am: 26.08.2021]

[86] https://www.extrahop.com/company/blog/2020/what-tls-1-3-means-for-secops-visibility/ [Zugriff am: 26.08.2021]

[87] https://www.fiserv.com/ [Zugriff am: 26.08.2021]

5 Conclusio

Die Netzwerk-Forensik ist ein wichtiger Bestandteil bei der Strafverfolgung von Cyberkriminellen und bei der Incident-Response von Unternehmen. Sie ermöglicht die Untersuchung von Sicherheitsvorfällen, indem sie den Angriff anhand von Netzwerkdaten rekonstruiert und ihn im besten Fall bis zur Quelle zurückverfolgt. Zudem gewährleistet sie die Erstellung von authentischen und gerichtsverwertbaren Beweisen. Zu Beginn dieser Arbeit wurden die folgenden Forschungsfragen gestellt und im Laufe der Forschung versucht, diese zu beantworten:

Welche Standards und Guidelines gibt es? Welche Tools können zur netzwerkforensischen Untersuchung verwendet werden? Wie funktioniert die Netzwerk-Forensik in virtuellen Netzen? Wie wirkt sich TLS 1.3 auf die Netzwerk-Forensik aus?

Am Anfang dieser Arbeit wird ein umfassender Einblick in die Grundlagen der Netzwerk-Forensik gewährt. Dabei wurden die wesentlichen Bestandteile der Netzwerk-Forensik beleuchtet. Anschließend wurden international sowie europaweit relevante und anerkannte Standards und Guidelines recherchiert, beschrieben und gegenübergestellt. Es wurden dabei neben verschiedenen ISO/IEC Standards, NIST Richtlinien auch wichtige Guidelines von Organisationen und Behörden beleuchtet, die im europäischen Raum zur Anwendung kommen.

In dieser Arbeit wurden verschiedene Kategorien von Tools, die bei der netzwerkforensischen Untersuchung unterstützen erarbeitet. Zu den jeweiligen Kategorien wurden anschließend Tools recherchiert und allgemein beschrieben. Das Hauptaugenmerk bei diesen Tools lag auf Aktualität und Open-Source. Danach wurden drei dieser Tools, namentlich NetworkMiner, Netfox Detective und Arkime, ausgewählt und in einem Praxistest auf deren Fähigkeiten analysiert. Dabei wurden Testdaten im PCAP(NG)-Format ausgewählt und verschiedene Kriterien zur Bewertung festgelegt. Als Testsieger stellte sich Arkime heraus. Dieses Tool konnte bei der Analyse der Testdaten und dem Erfüllen wichtiger Kriterien überzeugen.

Auch wurde in dieser Arbeit der Ist-Zustand der Netzwerk-Forensik in virtuellen Netzen festgehalten. Dabei

wurden verschiedene Frameworks und Prototypen aus der Wissenschaft recherchiert, beschrieben und deren Stärken und Schwächen dargestellt. Es zeigte sich, dass es sich bei allen Frameworks um reine Konzepte beziehungsweise Prototypen handelt. Es gibt derzeit kein Framework für die netzwerkforensische Untersuchung in virtuellen Netzen, das sich in der Praxis etabliert hat. Diese Recherche zeigt, dass es sich hierbei um ein junges Teilgebiet der Netzwerk-Forensik handelt und es noch viel Forschungsarbeit benötigt, um die Herausforderungen in virtuellen Netzen zu bewältigen.
Anschließend wurden noch die Auswirkungen von TLS 1.3 auf die Netzwerk-Forensik beleuchtet. Dabei wurden die relevanten Änderungen von TLS 1.3 gegenüber TLS 1.2 beschrieben und deren Effekt auf die Analyse des Netzwerkverkehrs erläutert. Darüber hinaus wurden verschiedene Möglichkeiten zur Entschlüsselung von TLS 1.3-Traffic recherchiert, beschrieben und Beispiele aus der Wissenschaft präsentiert. Passive TLS Decryption beziehungsweise Session-Key Forwarding stellte sich als beste Methode heraus. Obwohl es sich aus dem wissenschaftlichen Bereich nur um einen Prototyp handelt, findet dieser Ansatz bereits in der Praxis Anwendung.

5.1 Weiterführende Arbeiten

Diese Arbeit zeigt, dass die Forschung in der Netzwerk-Forensik im Bereich von virtuellen Netzen noch sehr jung ist und nur wenige Frameworks und Prototypen vorgestellt wurden. Daher bietet es sich an, in diesem Bereich weiter zu forschen und ein neues Framework, das die Nachteile der vorgestellten Frameworks beseitigt. Des weiteren kann im Bereich der passiven Entschlüsselung von TLS 1.3 weitergearbeitet werden. Hier bietet es sich an den von Wilkens u. a.[57] vorgestellten Prototypen weiterzuentwickeln, sodass dieser die neuen TLS 1.3 Cipher-Suites unterstützt.

Abbildungsverzeichnis

Tabellenverzeichnis

Glossar

ACPO	Association of Chief Police Officers
AES	Advanced Encryption Standard
API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
CAM	Content Addressable Memory
CERT	Computer Emergency Response Team
CoC	Chain of Custody
CPU	Central Processing Unit
DES	Data Encryption Standard
DFRW	Digital Forensics Research Workshop
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
E2E	Ende-zu-Ende
ECDHE	Elliptic Curve Diffie-Hellman Ephemeral
ENFSI	European Network of Forensic Science Institutes
ENISA	European Union Agency for Cybersecurity
Enklaven	Enklaven sind isolierte Speicherbereiche mit Code und Daten, die sich im Hauptspeicher (RAM) befinden.
eompls	Ethernet over MPLS
ESI	Electronically Stored Information
ESPI	Enhanced Statistical Protocol Identification

FDDI	Fiber Distributed Data Interface
FTP	File Transfer Protocol
GCM	Galois/Counter Mode
GRE	Generic Routing Encapsulation
GUI	Graphical User Interface
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
I/O	Input/Output
ICMP	Internet Control Message Protocol
ID	Identifikation
IDS	Intrusion Detection Systems
IEC	International Electrotechnical Commission
IETF	Internet-Engineering Task Force
IGMP	Internet Group Management Protocol
Integrität	Integrität bedeutet, dass es nicht möglich sein darf, Daten unerkannt bzw. unbemerkt zu ändern.
Intel SGX	Die Software Guard Extensions (Intel SGX) ist eine Reihe von sicherheitsrelevanten Befehlscodes. Der es ermöglicht, private Speicherbereiche, so genannte Enklaven, zu definieren. Deren Inhalt ist geschützt und kann von keinem Prozess außerhalb der Enklave selbst gelesen oder gespeichert werden, auch nicht von priviligierteren Prozessen.

IoT	Internet of Things
IP	Internet Protocol
IPFIX	IP Flow Information Export
IPS	Intrusion Protection Systems
IRC	Internet Relay Chat
ISO	International Organisation for Standardization
IT	Information Technology
LAN	Local Area Network
Lua	Programmiersprache
MAC	Media-Access-Control
Malware	Bösartige Computersoftware
Middlebox	Middleboxen sind Dienste, die "innerhalb" des Netzes laufen und logisch zwischen den Endpunkten von Kommunikationssitzungen liegen. Eine Middlebox ist ein Netzwerkgerät, das eine bestimmte Netzwerkfunktion bereitstellt (Firewall, LoadBalancer, IDS, Kontentfilter,...).
MPLS	Multiprotocol Label Switching
NBAR	Network-Based Application Recognition
NFAT	Network Forensics Analysis Tools
NFV	Network Function Virtualization
NIC	Network Interface Card - Netzwerkschnittstellenkarte
NIDS	Network Intrusion Detection Systems
NIPS	Network Intrusion Protection Systems
NIST	National Institute of Standards and Technology
NMS	Network Monitoring System
NSM	Network Security und Monitoring
NVGRE	Network Virtualization using Generic Routing Encapsulation

OLAF	European Anti-Fraud Office
OS	Operating System
OSPF	Open Shortest Path First
OSS	Open Source Software
PCAP	Dateiformat der Paketerfassung
PCAPng	PCAP Next Generation - Dateiformat der Paketerfassung
PDA	Personal Digital Assistant
PoC	Proof of Concept
POP3	Post Office Protocol Version 3
PPP	Point-to-Point Protocol
PPPoE	Point-to-Point Protocol over Ethernet
Prolog	logische Programmiersprache
PSK	Pre-Shared Key
QoS	Quality of Service
RFC	Request for Comments
RHEL	Red Hat Enterprise Linux
RSA	Rivest–Shamir–Adleman
SDN	Software-Defined Network
SDR	Software Defined Radio
SIEM	Security Information and Event Management
SMTP	Simple Mail Transfer Protocol
SNMP	Simple Network Management Protocol
SOC	Security Operation Center
SPAN	Switch Port Analyser
SPID	Statistical Protocol IDentification

Spoofing	Spoofing nennt man in der Informationstechnik verschiedene Täuschungsmethoden in Computernetzwerken zur Verschleierung der eigenen Identität.
SSL	Secure Sockets Layer
Stepping-Stone	Einen Computer dazu verwenden um mit diesen auf Services eines anderen Computers zuzugreifen.
STT	Stateless Transport Tunneling
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TTPs	Taktiken, Techniken und Prozeduren
UDP	User Datagram Protocol
VNID	VXLAN Network Identifier
WAN	Wide Area Network
WEP	Wired Equivalent Privacy
WPA	Wi-Fi Protected Access

Literatur

[1] Amala V. Rajan, Rejitha Ravikumar und Mariam Al Shaer, "Uae cybercrime law and cybercrimes — an analysis", in *2017 International Conference on Cyber Security And Protection Of Digital Services (Cyber Security)*, 2017, S. 1–6. DOI: 10.1109/CyberSecPODS.2017.8074858.

[2] Amor Lazzez, "A survey about network forensics tools", *International Journal of Computer and Information Technology*, Jg. 2, S. 74–81, Jan. 2013.

[3] Hal Berghel, "The discipline of internet forensics", *Commun. ACM*, Jg. 46, Nr. 8, S. 15–20, Aug. 2003, ISSN: 0001-0782. DOI: 10.1145/859670.859687. Adresse: https://doi.org/10.1145/859670.859687.

[4] Carole Fennelly, "Analysis: The forensics of internet security", *SunWorld*, 2000.

[5] Jan Pluskal, Frank Breitinger und Ondřej Ryšavý, "Netfox detective: A novel open-source network forensics analysis tool", *Forensic Science International: Digital Investigation*, Jg. 35, S. 301 019, 2020.

[6] John R. Vacca, *Computer and Information Security Handbook*, 3. Aufl. San Francisco, Calif: Morgan Kaufmann, 2017, S. 657–659, ISBN: 978-0-128-03929-8.

[7] Daniel Spiekermann und Tobias Eggendorfer, "Towards digital investigation in virtual networks: A study of challenges and open problems", in *2016 11th International Conference on Availability, Reliability and Security (ARES)*, IEEE, 2016, S. 406–413.

[8] Emmanuel S Pilli, Ramesh C Joshi und Rajdeep Niyogi, "Network forensic frameworks: Survey and research challenges", *digital investigation*, Jg. 7, Nr. 1-2, S. 14–27, 2010.

[9] Ahmad Almulhem und Issa Traore, "Experience with engineering a network forensics system", in *Information Networking. Convergence in Broadband and Mobile Networking*, Cheeha Kim, Hrsg., Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, S. 62–71, ISBN: 978-3-540-30582-8.

[10] Gary Palmers, “A road map for digital forensic research”, *First digital forensic research workshop*, S. 27–30, 2001. Adresse: https://dfrws.org/wp-content/uploads/2019/06/2001_USA_a_road_map_for_digital_forensic_research.pdf.

[11] Sudhakar Parate und S. Nirkhi, “A review of network forensics techniques for the analysis of web based attack”, 2012.

[12] Alec Yasinsac und Yanet Manzano, “Policies to enhance computer and network forensics”, in *Proceedings of the 2001 IEEE workshop on information assurance and security*, Citeseer, 2001, S. 289–295.

[13] Eoghan Casey, *Digital Evidence and Computer Crime - Forensic Science, Computers, and the Internet*. Amsterdam, Boston: Academic Press, 2011, S. 7, ISBN: 978-0-123-74268-1.

[14] Keyun Ruan, Joe Carthy, Tahar Kechadi und Mark Crosbie, “Cloud forensics”, in *IFIP International Conference on Digital Forensics*, Springer, 2011, S. 35–46.

[15] Ben Laurie, “Network forensics: Good detective work means paying attention before, during, and after the attack.”, *Queue*, Jg. 2, Nr. 4, S. 50–56, Juni 2004, ISSN: 1542-7730. DOI: 10.1145/1016978.1016982. Adresse: https://doi.org/10.1145/1016978.1016982.

[16] Eoghan Casey, *Network traffic as a source of evidence: Tool strengths, weaknesses, and future needs*, 2004.

[17] Suleman Khan, Abdullah Gani, Ainuddin Wahid Abdul Wahab, Muhammad Shiraz und Iftikhar Ahmad, “Network forensics: Review, taxonomy, and open challenges”, *Journal of Network and Computer Applications*, Jg. 66, S. 214–235, 2016, ISSN: 1084-8045. DOI: https://doi.org/10.1016/j.jnca.2016.03.005. Adresse: https://www.sciencedirect.com/science/article/pii/S1084804516300121.

[18] Karen Kent, Suzanne Chevalier, Tim Grance und Hung Dang, “Guide to integrating forensic techniques into incident response”, *NIST Special Publication*, Jg. 10, Nr. 14, S. 800–86, 2006.

[19] European Union Agency for Network und Information Security, *Introduction to Network Forensics Version 1.1*. European Union Agency for Network und Information Security (ENISA), 2019. Adresse: https://www.enisa.europa.eu/topics/trainings-for-cybersecurity-specialists/online-training-material/documents/introduction-to-network-forensics-handbook.pdf.

[20] R.C. Joshi und Emmanuel S. Pilli, *Fundamentals of Network Forensics - A Research Perspective*. Berlin, Heidelberg: Springer, 2016, S. 97–103, ISBN: 978-1-447-17299-4.

[21] Ahmad Almulhem, "Network forensics: Notions and challenges", in *2009 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, IEEE, 2009, S. 463–466.

[22] Nicole Lang Beebe und Jan Guynes Clark, "A hierarchical, objectives-based framework for the digital investigations process", *Digital Investigation*, Jg. 2, Nr. 2, S. 147–167, 2005, ISSN: 1742-2876. DOI: https://doi.org/10.1016/j.diin.2005.04.002. Adresse: https://www.sciencedirect.com/science/article/pii/S1742287605000307.

[23] Venansius Baryamureeba und Florence Tushabe, "The enhanced digital investigation process model", *Digital Investigation*, 2004.

[24] Richard G Clegg, Mark S Withall, Andrew W Moore, Iain W Phillips, David J Parish, Miguel Rio, Raul Landa, Hamed Haddadi, Kostas Kyriakopoulos, Jordan Auge u. a., "Challenges in the capture and dissemination of measurements from high-speed networks", *IET communications*, Jg. 3, Nr. 6, S. 957–966, 2009.

[25] Chuang Wang, Taiming Feng, Jinsook Kim, Guiling Wang und Wensheng Zhang, "Catching packet droppers and modifiers in wireless sensor networks", in *2009 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, IEEE, 2009, S. 1–9.

[26] Asou Aminnezhad, Ali Dehghantanha und Mohd Taufik Abdullah, "A survey on privacy issues in digital forensics", *International Journal of Cyber-Security and Digital Forensics*, Jg. 1, Nr. 4, S. 311–324, 2012.

[27] Peter Sommer, "Intrusion detection systems as evidence", *Computer Networks*, Jg. 31, Nr. 23-24, S. 2477–2487, 1999.

[28] Dominique Brezinski und Tom Killalea, "Guidelines for evidence collection and archiving", *RFC3227, February*, 2002.

[29] Alexander Geschonneck, *Computer-Forensik (iX Edition) - Computerstraftaten erkennen, ermitteln, aufklären*. Heidelberg: dpunkt.verlag, 2014, S. 65–67, ISBN: 978-3-864-91489-8.

[30] Jasmin Ćosić, Zoran Ćosić und Miroslav Bača, "An ontological approach to study and manage digital chain of custody of digital evidence", *Journal of Information and Organizational Sciences*, Jg. 35, Nr. 1, S. 1–13, 2011.

[31] Mrunali Chopade, Sana Khan, Uzma Shaikh und Renuka Pawar, “Digital forensics: Maintaining chain of custody using blockchain”, in *2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*, IEEE, 2019, S. 744–747.

[32] Nathan Clarke, Fudong Li und Steven Furnell, “A novel privacy preserving user identification approach for network traffic”, *computers & security*, Jg. 70, S. 335–350, 2017.

[33] Gerard Johansen, *Digital Forensics and Incident Response - Incident response techniques and procedures to respond to modern cyber threats, 2nd Edition*. Birmingham: Packt Publishing Ltd, 2020, S. 90–92, ISBN: 978-1-838-64408-6.

[34] Rabiu Mukhtar, A. Al-Nemrat, Mamoun Alazab, Sitalakshmi Venkatraman und Hamid Jahankhani, “Analysis of firewall log-based detection scenarios for evidence in digital forensics”, *Int. J. Electron. Secur. Digit. Forensic*, Jg. 4, Nr. 4, S. 261–279, Okt. 2012, ISSN: 1751-911X. DOI: 10.1504/IJESDF.2012.049761. Adresse: https://doi.org/10.1504/IJESDF.2012.049761.

[35] Sandeep Bhatt, Pratyusa K. Manadhata und Loai Zomlot, “The operational role of security information and event management systems”, *IEEE Security Privacy*, Jg. 12, Nr. 5, S. 35–41, 2014. DOI: 10.1109/MSP.2014.103.

[36] Simson Garfinkel, “Network forensics: Tapping the internet”,

[37] Matt Bromiley, “Enhance Your Investigations with Network Data”, SANS Institute, 2017.

[38] Scott D. Fether, “PCAP Next Generation: Is Your Snifer Up to Snuff?”, SANS Institute, 2018.

[39] Michael Tüxen, Fulvio Risso, Jasper Bongertz, Gerald Combs, Guy Harris und Michael Richardson, “PCAP Next Generation (pcapng) Capture File Format”, Internet Engineering Task Force, Internet-Draft draft-tuexen-opsawg-pcapng-02, Sep. 2020, Work in Progress, 52 S. Adresse: https://datatracker.ietf.org/doc/html/draft-tuexen-opsawg-pcapng-02.

[40] Brian Trammell und Elisa Boschi, “An introduction to ip flow information export (ipfix)”, *IEEE Communications Magazine*, Jg. 49, Nr. 4, S. 89–95, 2011. DOI: 10.1109/MCOM.2011.5741152.

[41] Benoit Claise, Ganesh Sadasivan, Vamsi Valluri und Martin Djernaes, “Cisco systems netflow services export version 9”, 2004.

[42] Cisco Systems Inc., "Cisco ios netflow version 9 flow-record format", 2011. Adresse: `https://www.cisco.com/en/US/technologies/tk648/tk362/technologies_white_paper09186a00800a3db9.pdf`.

[43] Rick Hofstede, Pavel Čeleda, Brian Trammell, Idilio Drago, Ramin Sadre, Anna Sperotto und Aiko Pras, "Flow monitoring explained: From packet capture to data analysis with netflow and ipfix", *IEEE Communications Surveys & Tutorials*, Jg. 16, Nr. 4, S. 2037–2064, 2014.

[44] Idilio Drago, Rafael RR Barbosa, Ramin Sadre, Aiko Pras und Jürgen Schönwälder, "Report of the second workshop on the usage of netflow/ipfix in network management", *Journal of network and systems management*, Jg. 19, Nr. 2, S. 298–304, 2011.

[45] Enos K Mabuto und Hein S Venter, "State of the art of digital forensic techniques.", in *ISSA*, 2011.

[46] Daniel Spiekermann und Jörg Keller, "Unsupervised packet-based anomaly detection in virtual networks", *Computer Networks*, Jg. 192, S. 108 017, 2021.

[47] Michael Donovan Kohn, Mariki M Eloff und Jan HP Eloff, "Integrated digital forensic process model", *Computers & Security*, Jg. 38, S. 103–115, 2013.

[48] Wei Ren und Hai Jin, "Modeling the network forensics behaviors", in *Workshop of the 1st International Conference on Security and Privacy for Emerging Areas in Communication Networks, 2005.*, 2005, S. 1–8. DOI: `10.1109/SECCMW.2005.1588287`.

[49] Kevin Mandia, Chris Prosise und Matt Pepe, *Incident response & computer forensics*. McGraw-Hill New York, 2003, Bd. 2.

[50] Changwei Liu, Anoop Singhal und Duminda Wijesekera, "A logic-based network forensic model for evidence analysis", in *IFIP International Conference on Digital Forensics*, Springer, 2015, S. 129–145.

[51] Prabhjot Kaur, Anchit Bijalwan, RC Joshi und Amit Awasthi, "Network forensic process model and framework: An alternative scenario", in *Intelligent Communication, Control and Devices*, Springer, 2018, S. 493–502.

[52] Ray Hunt, "New developments in network forensics—tools and techniques", in *2012 18th IEEE International Conference on Networks (ICON)*, IEEE, 2012, S. 376–381.

[53] Gulshan Shrivastava, "Network forensics: Methodical literature review", in *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, IEEE, 2016, S. 2203–2208.

[54] Raffaella Brighi-Michele Ferrazzano, “Digital forensics: Best practices and perspective”, in *Digital forensic evidence: towards common European standards in antifraud administrative and criminal investigations*, Wolters Kluwer, 2021, S. 13–47.

[55] Ana Nieto, Ruben Rios, Javier Lopez, Wei Ren, Lizhe Wang, Kim-Kwang Raymond Choo und Fatos Xhafa, *Privacy-aware digital forensics.* 2019.

[56] Suleman Khan, Abdullah Gani, Ainuddin Wahid Abdul Wahab, Ahmed Abdelaziz, Kwangman Ko, Muhammad Khurram Khan und Mohsen Guizani, “Software-defined network forensics: Motivation, potential locations, requirements, and challenges”, *IEEE Network*, Jg. 30, Nr. 6, S. 6–13, 2016. DOI: 10.1109/MNET.2016.1600051NM.

[57] Florian Wilkens, Steffen Haas, Johanna Amann und Mathias Fischer, “Passive, transparent, and selective tls decryption for network security monitoring”, *arXiv preprint arXiv:2104.09828*, 2021.

[58] Simson L. Garfinkel, “Digital forensics research: The next 10 years”, *Digital Investigation*, Jg. 7, S64–S73, 2010, The Proceedings of the Tenth Annual DFRWS Conference, ISSN: 1742-2876. DOI: https://doi.org/10.1016/j.diin.2010.05.009. Adresse: https://www.sciencedirect.com/science/article/pii/S1742287610000368.

[59] Rommel Sira, “Network forensics analysis tools: An overview of an emerging technology”, *GSEC, version*, Jg. 1, S. 1–10, 2003.

[60] Vicka Corey, Charles Peterman, Sybil Shearin, Michael S Greenberg und James Van Bokkelen, “Network forensics analysis”, *IEEE Internet Computing*, Jg. 6, Nr. 6, S. 60–66, 2002.

[61] MI Cohen, “Pyflag–an advanced network forensic framework”, *Digital investigation*, Jg. 5, S112–S120, 2008.

[62] Leslie F. Sikos, “Packet analysis for network forensics: A comprehensive survey”, *Forensic Science International: Digital Investigation*, Jg. 32, S. 200 892, 2020, ISSN: 2666-2817. DOI: https://doi.org/10.1016/j.fsidi.2019.200892. Adresse: https://www.sciencedirect.com/science/article/pii/S1742287619302002.

[63] Mohammed Abdul Qadeer, Arshad Iqbal, Mohammad Zahid und Misbahur Rahman Siddiqui, “Network traffic analysis and intrusion detection using packet sniffer”, in *2010 Second International Conference on Communication Software and Networks*, 2010, S. 313–317. DOI: 10.1109/ICCSN.2010.104.

[64] Felix Fuentes und Dulal C Kar, "Ethereal vs. tcpdump: A comparative study on packet sniffing tools for educational purpose", *Journal of Computing Sciences in Colleges*, Jg. 20, Nr. 4, S. 169–176, 2005.

[65] Sihyung Lee, Kyriaki Levanti und Hyong S Kim, "Network monitoring: Present and future", *Computer Networks*, Jg. 65, S. 84–98, 2014.

[66] Jakub Svoboda, Ibrahim Ghafir, Vaclav Prenosil u. a., "Network monitoring approaches: An overview", *Int J Adv Comput Netw Secur*, Jg. 5, Nr. 2, S. 88–93, 2015.

[67] Ajay Shah, Sophine Clachar, Manfred Minimair und Davis Cook, "Building multiclass classification baselines for anomaly-based network intrusion detection systems", in *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*, 2020, S. 759–760. DOI: 10.1109/DSAA49011.2020.00102.

[68] Pedro Casas, Johan Mazel und Philippe Owezarski, "Unsupervised network intrusion detection systems: Detecting the unknown without knowledge", *Computer Communications*, Jg. 35, Nr. 7, S. 772–783, 2012.

[69] Muhammad Irfan, Haider Abbas und Waseem Iqbal, "Feasibility analysis for incorporating/deploying siem for forensics evidence collection in cloud environment", in *2015 IEEE/ACIS 14th International Conference on Computer and Information Science (ICIS)*, 2015, S. 15–21. DOI: 10.1109/ICIS.2015.7166563.

[70] Amrit Williams und Mark Nicolett, "Improve it security with vulnerability management", *Gartner ID*, Nr. G00127481, 2005.

[71] Axel Buecker, Jose Amado, David Druker, Carsten Lorenz, Frank Muehlenbrock, Rudy Tan u. a., *IT Security Compliance Management Design Guide with IBM Tivoli Security Information and Event Manager*. IBM Redbooks, 2010, S. 15–18.

[72] Hassan Mokalled, Rosario Catelli, Valentina Casola, Daniele Debertol, Ermete Meda und Rodolfo Zunino, "The applicability of a siem solution: Requirements and evaluation", in *2019 IEEE 28th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*, 2019, S. 132–137. DOI: 10.1109/WETICE.2019.00036.

[73] Gerard Johansen, *Digital Forensics and Incident Response - Incident response techniques and procedures to respond to modern cyber threats, 2nd Edition*. Birmingham: Packt Publishing Ltd, 2020, S. 179–182, ISBN: 978-1-838-64408-6.

[74] Daniel Spiekermann, Jörg Keller und Tobias Eggendorfer, "Network forensic investigation in openflow networks with forcon", *Digital Investigation*, Jg. 20, S66–S74, 2017.

[75] Edison F. Naranjo und Gustavo D. Salazar Ch, "Underlay and overlay networks: The approach to solve addressing and segmentation problems in the new networking era: Vxlan encapsulation with cisco and open source networks", in *2017 IEEE Second Ecuador Technical Chapters Meeting (ETCM)*, 2017, S. 1–6. DOI: 10.1109/ETCM.2017.8247505.

[76] T. Anderson, L. Peterson, S. Shenker und J. Turner, "Overcoming the internet impasse through virtualization", *Computer*, Jg. 38, Nr. 4, S. 34–41, 2005. DOI: 10.1109/MC.2005.136.

[77] Ozgur Yurekten und Mehmet Demirci, "Sdn-based cyber defense: A survey", *Future Generation Computer Systems*, Jg. 115, S. 126–149, 2021.

[78] Rahim Masoudi und Ali Ghaffari, "Software defined networks: A survey", *Journal of Network and computer Applications*, Jg. 67, S. 1–25, 2016.

[79] Diego Kreutz, Fernando MV Ramos, Paulo Esteves Verissimo, Christian Esteve Rothenberg, Siamak Azodolmolky und Steve Uhlig, "Software-defined networking: A comprehensive survey", *Proceedings of the IEEE*, Jg. 103, Nr. 1, S. 14–76, 2014.

[80] Juan Camilo Correa Chica, Jenny Cuatindioy Imbachi und Juan Felipe Botero Vega, "Security in sdn: A comprehensive survey", *Journal of Network and Computer Applications*, Jg. 159, S. 102 595, 2020.

[81] Yosr Jarraya, Taous Madi und Mourad Debbabi, "A survey and a layered taxonomy of software-defined networking", *IEEE Communications Surveys Tutorials*, Jg. 16, Nr. 4, S. 1955–1980, 2014. DOI: 10.1109/COMST.2014.2320094.

[82] Fei Hu, Qi Hao und Ke Bao, "A survey on software-defined network and openflow: From concept to implementation", *IEEE Communications Surveys Tutorials*, Jg. 16, Nr. 4, S. 2181–2206, 2014. DOI: 10.1109/COMST.2014.2326417.

[83] Nick McKeown, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker und Jonathan Turner, "Openflow: Enabling innovation in campus networks", *SIGCOMM Comput. Commun. Rev.*, Jg. 38, Nr. 2, S. 69–74, März 2008, ISSN: 0146-4833. DOI: 10.1145/1355734.1355746. Adresse: https://doi.org/10.1145/1355734.1355746.

[84] Yaohua Yan und Hongbo Wang, "Open vswitch vxlan performance acceleration in cloud computing data center", in *2016 5th International Conference on Computer Science and Network Technology (ICCSNT)*, 2016, S. 567–571. DOI: 10.1109/ICCSNT.2016.8070222.

[85] Shu-hui Zhang, Xiang-xu Meng und Lian-hai Wang, "Sdnforensics: A comprehensive forensics framework for software defined network", *Development*, Jg. 3, Nr. 4, S. 5, 2017.

[86] Haopei Wang, Guangliang Yang, Phakpoom Chinprutthiwong, Lei Xu, Yangyong Zhang und Guofei Gu, "Towards fine-grained network security forensics and diagnosis in the sdn era", in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, Ser. CCS '18, Toronto, Canada: Association for Computing Machinery, 2018, S. 3–16, ISBN: 9781450356930. DOI: 10.1145/3243734.3243749. Adresse: https://doi.org/10.1145/3243734.3243749.

[87] TJ OConnor, William Enck, W Michael Petullo und Akash Verma, "Pivotwall: Sdn-based information flow control", in *Proceedings of the Symposium on SDN Research*, 2018, S. 1–14.

[88] Phan The Duy, Hien Do Hoang, Do Thi Thu Hien, Nguyen Ba Khanh und Van-Hau Pham, "Sdnlogforen: Ensuring the integrity and tamper resistance of log files for sdn forensics using blockchain", in *2019 6th NAFOSTED Conference on Information and Computer Science (NICS)*, 2019, S. 416–421. DOI: 10.1109/NICS48868.2019.9023852.

[89] Sean Turner, "Transport layer security", *IEEE Internet Computing*, Jg. 18, Nr. 6, S. 60–63, 2014. DOI: 10.1109/MIC.2014.126.

[90] Paraskevi Dimou, Jan Fajfer, Nicolas Müller, Eva Papadogiannaki, Evangelos Rekleitis und František Střasák, "ENCRYPTED TRAFFIC ANALYSIS", European Union Agency for Cybersecurity (ENISA), 2019.

[91] X de Carné de Carnavalet und Mohammad Mannan, "Killed by proxy: Analyzing client-end tls interception software", in *Network and Distributed System Security Symposium*, 2016.

[92] Mark O'Neill, Scott Ruoti, Kent Seamons und Daniel Zappala, "Tls inspection: How often and who cares?", *IEEE Internet Computing*, Jg. 21, Nr. 3, S. 22–29, 2017. DOI: 10.1109/MIC.2017.58.

[93] Zakir Durumeric, Zane Ma, Drew Springall, Richard Barnes, Nick Sullivan, Elie Bursztein, Michael Bailey, J Alex Halderman und Vern Paxson, "The security impact of https interception.", in *NDSS*, 2017.

[94] Jeff Jarmoc und DSCT Unit, “Ssl/tls interception proxies and transitive trust”, *Black Hat Europe*, 2012.

[95] Mark O’Neill, Scott Ruoti, Kent Seamons und Daniel Zappala, “Tls proxies: Friend or foe?”, in *Proceedings of the 2016 Internet Measurement Conference*, 2016, S. 551–557.

[96] Louis Waked, Mohammad Mannan und Amr Youssef, “To intercept or not to intercept: Analyzing tls interception in network appliances”, in *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, 2018, S. 399–412.

[97] David Naylor, Kyle Schomp, Matteo Varvello, Ilias Leontiadis, Jeremy Blackburn, Diego R. López, Konstantina Papagiannaki, Pablo Rodriguez Rodriguez und Peter Steenkiste, “Multi-context tls (mctls): Enabling secure in-network functionality in tls”, *SIGCOMM Comput. Commun. Rev.*, Jg. 45, Nr. 4, S. 199–212, Aug. 2015, ISSN: 0146-4833. DOI: 10.1145/2829988.2787482. Adresse: https://doi.org/10.1145/2829988.2787482.

[98] David Naylor, Richard Li, Christos Gkantsidis, Thomas Karagiannis und Peter Steenkiste, “And then there were more: Secure communication for more than two parties”, in *Proceedings of the 13th International Conference on Emerging Networking EXperiments and Technologies*, Ser. CoNEXT ’17, Incheon, Republic of Korea: Association for Computing Machinery, 2017, S. 88–100, ISBN: 9781450354226. DOI: 10.1145/3143361.3143383. Adresse: https://doi.org/10.1145/3143361.3143383.

[99] Hyunwoo Lee, Zach Smith, Junghwan Lim, Gyeongjae Choi, Selin Chun, Taejoong Chung und Ted Taekyoung Kwon, “Matls: How to make tls middlebox-aware?”, in *NDSS*, 2019.

[100] David Goltzsche, Signe Rüsch, Manuel Nieke, Sébastien Vaucher, Nico Weichbrodt, Valerio Schiavoni, Pierre-Louis Aublin, Paolo Cosa, Christof Fetzer, Pascal Felber, Peter Pietzuch und Rüdiger Kapitza, “Endbox: Scalable middlebox functions using client-side trusted execution”, in *2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 2018, S. 386–397. DOI: 10.1109/DSN.2018.00048.

[101] Bohdan Trach, Alfred Krohmer, Franz Gregor, Sergei Arnautov, Pramod Bhatotia und Christof Fetzer, “Shieldbox: Secure middleboxes using shielded execution”, in *Proceedings of the Symposium on SDN Research*, Ser. SOSR ’18, Los Angeles, CA, USA: Association for Computing Machinery, 2018, ISBN: 9781450356640. DOI: 10.1145/3185467.3185469. Adresse: https://doi.org/10.1145/3185467.3185469.

[102] Jaak Randmets, “An overview of vulnerabilities and mitigations of intel sgx applications”, 2021.