Datenschutzfreundliche Erkennung und Abwehr von Insiderbedrohungen

Dissertation

zur Erlangung des akademischen Grades

Dr. rer. nat.

an der Fakultät für Mathematik, Informatik und Naturwissenschaften der Universität Hamburg

eingereicht beim Fach-Promotionsausschuss Informatik von

Dipl.-Inf. Ephraim Zimmer

geb. 1988 in Jena

Juli, 2020

Gutachter:

Prof. Dr. Hannes Federrath

Prof. Dr. Dieter Gollmann

Datum der Disputation:

06. Oktober 2020

Danksagung

Diese Dissertation ist mit der Unterstützung durch zahlreiche Personen entstanden, denen ich an dieser Stelle meinen herzlichen und tief verbundenen Dank aussprechen möchte.

Ausdrücklicher Dank gebührt meinem Doktorvater und Betreuer dieser Arbeit Hannes Federrath. Seine Aussagen bezüglich der Insiderproblematik haben meinen thematischen Schwerpunkt ins Rollen gebracht. Seine vielen hilfreichen Anmerkungen sowie die Diskussionen und behutsame Anleitung bei der Durchführung des Promotionsvorhabens haben die Forschungsbeiträge maßgeblich geschliffen. Bedanken möchte ich mich auch bei Dieter Gollmann für seine Bereitschaft, die Dissertation zu begutachten sowie für seine anekdotischen und äußerst lehrreichen Diskussionen und Denkanstöße, die dazu beigetragen haben, an geeigneten Stellen entgegen der Intuition zu denken und so hilfreiche Erkenntnisse zu erlangen.

Natürlich richtet sich mein Dank auch an meine Kolleginnen und Kollegen im Arbeitsbereich Security and Privacy an der Universität Hamburg, die für ein großartiges Arbeitsklima, nützliche Teeküchenideen sowie kritische Diskussionen gesorgt haben. Insbesondere möchte ich Dominik Herrmann, Jens Lindemann, Christian Burkert und Tom Petersen für die gemeinsame Erarbeitung und Publikation anfänglicher Ideen beziehungsweise der Datenschutzaspekte in dieser Arbeit danken. Christian Burkert hat mich darüber hinaus bei der Ergründung der Insider-Grundlagen immer wieder auf Spur gehalten. Herzlichen Dank auch an Matthias Marx und Steffen Haas, die mir wichtige Impulse bei der Entwicklung der Abwehrtechnik gaben. Nicht zuletzt danke ich Dimitra Pons und Britta Böhm, die mir mit ihrer großartigen Unterstützung im Alltagsgeschäft Zeit und Freiraum für die Forschung verschafften.

Keinesfalls vergessen möchte ich an dieser Stelle den Dank an Kristin Kissmann, die meinem Verständnis psychologischer Aspekte auf die Sprünge geholfen und bei jeder Gelegenheit nach neuesten Erkenntnissen gefragt hat. Bettina Martin danke ich ganz sehr für die Validierung meiner Darstellung rechtlicher Verpflichtungen. Für die unermüdliche Fehlersuche und erhebliche Verbesserung der textuellen Darstellung sowie teilweise auch der inhaltlichen Präzisierung danke ich allen Korrekturleserinnen und Korrekturlesern. Dazu zählen neben bereits und nachfolgend genannten auch Johannes Siegert, Tobias Müller und Monina Schwarz.

Ein großer Dank gilt auch meiner Familie, die mich in allen Lebenslagen tatkräftig und finanziell unterstützt und mein Studium sowie meine Promotion ermöglicht hat. Nicht nur kritische und hilfreiche Anmerkungen zur Dissertation, sondern auch immer offene Arme und Herzen haben mir stets den nötigen Rückhalt und große Motivation gegeben. Einen ganz besonderen Dank möchte ich schließlich an meine Frau Judith und meine Kinder richten. Bei ihnen hatte ich immer ein offenes Ohr für Probleme und Schwierigkeiten, die trotz der Fachfremde mit guten Lösungsvorschlägen adressiert wurden. Die stetigen Rückfragen nach dem aktuellen Stand oder die Frage, wann Papa endlich zum Spielen nach Hause kommt, hatten einen unvergleichlich positiven Einfluss auf meine Produktivität. Darüber hinaus habe ich ein einzigartig entgegengebrachtes Verständnis sowie eine derart selbstlose Unterstützung erfahren dürfen, die mich in tiefer Demut und mit allergrößtem Respekt anerkennen lassen, dass ich ohne sie das Promotionsvorhaben niemals zu einem erfolgreichen Ende hätte bringen können.

Zusammenfassung

Insider spielen für ihre Umgebung eine zentrale Rolle bei der Erfüllung von Aufgaben, Erbringung von Leistungen oder Aufrechterhaltung von Diensten. Spezielle Fähigkeiten, die Insidern eine herausgehobene Stellung gegenüber Outsidern verleihen, können allerdings auch einen gewollten oder versehentlichen negativen Einfluss ausüben. Diese gleichzeitige Notwendigkeit auf der einen und Gefahr auf der anderen Seite, die sich in Insidern vereinen, eröffnen ein komplexes Themenfeld, dass in seiner Problematik und Tragweite zunehmend an Bedeutung gewinnt.

In dieser Dissertation wird aufgezeigt, welche grundlegenden Probleme in der Forschung und Entwicklung von Lösungsansätzen vorliegen, die Verbesserungen der Insiderproblematik im Weg stehen. Dafür werden zunächst neue Erkenntnisse im Grundlagenbereich der Insiderthematik erarbeitet und zu einer Insiderontologie zusammengeführt. Diese beinhaltet die Identifizierung der fünf speziellen Insidercharakteristiken *Credentials*, *Knowledge*, *Privileges*, *Trust* sowie *Uncertainty*, mit denen die erwähnten Fähigkeiten von Insidern explizit benannt werden können. Die Insidercharakteristiken begründen jeweils unabhängige Insidertypen, die miteinander kombiniert eine eindeutige Beschreibung verschiedener Insider erlauben. Aus der Kombination von Insidertypen ergibt sich eine natürliche Insidertaxonomie, die als Grundlage für die Systematisierung von existierenden und zukünftigen Forschungsarbeiten dienen kann. Gleichzeitig wird der Unterschiedlichkeit von Insidern in verschiedenen Umgebungen Rechnung getragen, indem eine dynamische Insidermodellierung definiert wird. Aus den entwickelten Grundlagen, die sich zunächst allein auf Insider im Allgemeinen beziehen, wird anschließend der Aspekt einer Insiderbedrohung im Detail herausgearbeitet, von welchem Insidertyp welche Art von Insiderbedrohung ausgeht. Dafür werden die Insiderbedrohungen zunächst in drei verschiedene Bedrohungsklassen strukturiert. Anschließend erfolgt eine detaillierte Betrachtung des Einflusses jeder Insidercharakteristik auf jede Insiderbedrohungsklasse. Die Ergebnisse erlauben eine Zuordnung der Insiderbedrohungen zu den Insidertypen der entwickelten Insidertaxonomie.

Neben den Grundlagen bezüglich Insidern und Insiderbedrohungen befasst sich die Dissertation auch mit Erkennungs- und Abwehrmaßnahmen von Insiderbedrohungen. Zunächst erfolgt ausgerichtet auf die Insidertaxonomie und die identifizierten Klassen von Insiderbedrohungen eine Zuordnung grundlegender Sicherheitsmechanismen zu den Bedrohungen durch einzelne Insidertypen. Darüber hinaus wird ein neuer Ansatz zur detaillierten Erfassung und Analyse von Insideraktivitäten in IT-Umgebungen entworfen und umgesetzt. Dieser basiert auf Ereignisnachrichten aus der Systemaufrufebene von Betriebssystemen. Die Ereignisnachrichten werden in eine neuartige Graphenstruktur überführt, die eine signaturbasierte Erkennung und eine anschließende Abwehr von bedrohlichen Insideraktivitäten ermöglicht. Anhand von realen Angriffsproben wird zusätzlich ein definiertes Insiderbedrohungsszenario ausgeführt und die neue Erkennungs- und Abwehrtechnik damit evaluiert. Die Ergebnisse zeigen Erkennungsraten von Insiderangriffsaktivitäten im Bereich von 99,97% und 100%, wenn zuvor in einer Trainingsphase ähnlich gelagerte Aktivitäten aufgezeichnet wurden.

Die in der Dissertation betrachteten Bedrohungsaspekte umfassen nicht nur solche, die von Insidern ausgehen, sondern auch jene, die auf Insider einwirken, wenn sie Erkennungs- und Abwehrtechniken ausgesetzt sind. Der dabei im Mittelpunkt stehende Insiderdatenschutz wird

rechtlich eingeordnet und abschließend durch ein neu entwickeltes Pseudonymisierungsverfahren in die umgesetzte Erkennungs- und Abwehrtechnik integriert. Dabei liegt der Fokus sowohl auf der Erhaltung der Funktionalität, welche die Verkettbarkeit verschiedener Ereignisse sowie die Identifizierung von Akteuren anhand der Ereignisse umfasst, als auch auf einer technischen Durchsetzbarkeit der wichtigsten Datenschutz-Grundprinzipien, wie etwa der Datenminimierung, der Speicherbegrenzung sowie der Zweckbindung. Zum Einsatz kommen dabei kryptographische Funktionen, deren Kombination neu definierte Pseudonymisierungs-Schutzziele bereitstellen. Dazu zählen beispielsweise die Vertraulichkeit der personenbeziehbaren Daten, die pseudonymisiert werden sollen, oder die Limitierung der Verkettbarkeit verschiedener Ereignisse anhand der gewählten Pseudonyme. Darüber hinaus werden auch schutzwürdige Aspekte herausgearbeitet, die sich erst aus der praktischen Umsetzung einer Pseudonymisierung ergeben.

Mithilfe der Ergebnisse dieser Dissertation können existierende und zukünftige Forschungs- und Entwicklungsarbeiten auf dem Gebiet der Insiderthematik systematisch aufbereitet und somit besser koordiniert werden. Daran ausgerichtet können weitere grundlegende Erkenntnisse erarbeitet und Synergieeffekte sowie Verbesserungen hervorgehoben werden. Die entwickelten Techniken zeigen darüber hinaus neue Wege für die Erkennung und Abwehr von Insiderbedrohungen auf. Gleichzeitig würdigen sie den Insiderdatenschutz und verdeutlichen, dass beides als mehrseitige Interessen und Pflichten ausbalanciert werden muss.

Abstract

For their environment insiders play a central role regarding the completion of tasks and the performance or the maintenance of services. However, special capabilities, which put insiders into an enhanced position compared to outsiders, can have an intended or accidental negative effect. This combined necessity on the one hand and danger on the other opens a complex topic area regarding its difficulty and scope and that is acquiring greater importance.

This thesis uncovers basic problems in the area of research and development of solution approaches which are barriers to improvements of the insider topic. Therefore, new insights in basic research are acquired and combined to an insider ontology. Those insights contain the identification of five particular insider characteristics *Credentials*, *Knowledge*, *Privileges*, *Trust*, and *Uncertainty*. These characteristics allow an explicit description of the mentioned insiders capabilities. Moreover, each characteristic defines an independent insider type and also a clear and unambiguous allocation of different insider types when combined with each other. This combination leads to a natural insider taxonomy serving as foundation for a systematisation of existing and future research. At the same time specifying a dynamic way to create insider models account for the existence of various different insiders in different environments. With those developed foundations, which at first solely focus on insiders in general, insider threats are subsequently identified and focussed upon. It is described in more detail, which insider threat emanates from which insider type. For this investigation, insider threats are structured into three different threat classes followed by a precise consideration of the influence of every insider characteristic on each class of insider threats. The results allow the allocation of insider threats to the insider types of the developed insider taxonomy.

Alongside the foundations regarding insiders and insider threats, this thesis deals with mechanisms of insider threat detection and prevention. To begin with, an allocation of basic security mechanisms to threats of individual insider types is performed, which is orientated towards the insider taxonomy as well as the identified insider threats. Furthermore, a new and in depth approach for acquiring and analysing insider activities in an IT based environment is designed and developed. This approach is based on event messages at the level of system calls from operating systems. The event messages are conveyed into newly created graph structures, which allow a signature based detection and subsequent prevention of threatening insider activities. Based on real attack samples, a defined insider threat scenario is run in order to evaluate the new detection and prevention mechanism. The results of the evaluation show a detection rate of insider attack activities in the area of 99,97in a training phase beforehand.

The considered threat aspects in this thesis not only comprise those which emanate from insiders, but also those which affect insiders when they are imperilled by detection and prevention mechanisms. Insider privacy is hereinafter focussed upon and legally classified according to the current law in Germany and finally integrated into the implemented detection and prevention mechanism via a newly designed and developed pseudonymisation procedure. In the course of this, the focus is put on two contrary objectives. On the one hand, the functionality of the detection and prevention mechanisms shall be preserved, which includes the linkability of different event messages as well as the identification of actors with the help of those event

messages. On the other hand, important privacy principles shall be technically enforceable. These privacy principles include data minimisation, storage limitation, and purpose limitation. For these objectives, cryptographic techniques are combined and deployed, which achieve the protection of newly defined pseudonymisation protection goals. These include for example confidentiality of personal identifiable or quasi-identifiable data, which shall be pseudonymised, or the limitation of linkability of various event messages. On top of that, more subtle aspects, which are worthy of being protected and which emerge as a result of the deployment of pseudonymisation itself are identified and dealt with.

With the contributions of this thesis, existing and future research and development in the area of the insider topic can be systematically treated and superiorly coordinated. Additional fundamental insights can be acquired and synergy effects as well as improvements can be emphasised. Moreover, the herein designed and developed techniques show new ways to detect and prevent insider threats as well as to protect the privacy of insiders. It becomes clear, that both, functionality and privacy, need to be appreciated and balanced as a multilateral conflict of duties and interests.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Fachspezifisches Abkürzungsverzeichnis

AktG	Aktiengesetz
BAG	Bundesarbeitsgericht
BDSG	Bundesdatenschutzgesetz
BetrVG	Betriebsverfassungsgesetz
BSI	Bundesamt für Sicherheit in der Informationstechnik
BSIG	Gesetz über das Bundesamt für Sicherheit in der Informationstechnik
BVerfG	Bundesverfassungsgericht
C-Insider	Credentials-Insider
C-Outsider	Credentials-Outsider
C_A-Insider	Credentials-Associate
C_M-Insider	Credentials-Masquerader
CC	Common Criteria
CEF	Common Event Format
DCGK	Deutscher Corporate Governance Kodex
DSGVO	Datenschutzgrundverordnung
EC	elliptische Kurven
EU	Europäische Union
EuGH	Europäischer Gerichtshof
FBI	Federal Bureau of Investigation
GenG	Gesetz betreffend die Erwerbs- und Wirtschaftsgenossenschaften
GG	Grundgesetz
GmbHG	Gesetz betreffend die Gesellschaften mit beschränkter Haftung
GRCh	Grundrechtecharta
HMAC	Keyed-Hash Message Authentication Code (MAC)
HmbPersVG	Hamburgisches Personalvertretungsgesetz
IDS	Intrusion Detection System
IT-Sicherheitsgesetz	Gesetz zur Erhöhung der Sicherheit informationstechnischer Systeme

K-Insider	Knowledge-Insider
K-Outsider	Knowledge-Outsider
KWG	Kreditwesengesetz
MAC	Message Authentication Code
MDL	Minimum Descriptive Length
NIST	National Institute of Standards and Technology
OT	Oblivious Transfer
OWiG	Gesetz über Ordnungswidrigkeiten
P-Insider	Privileges-Insider
P-Outsider	Privileges-Outsider
PDF	Adobe Protable Document Format
PEEPLL	Privacy Enhanced Event Pseudonymisation with Limited Linkability
PII	Personal Identifiable Information
PZT	Pseudonymzuordnungstabelle
QID	Quasi-Identifier
SysGraph	Systemaufruf-Graph
T-Insider	Trust-Insider
T-Outsider	Trust-Outsider
TOE	Target of Evaluation
TTP	Trusted Third Party
U-Insider	Uncertainty-Insider
U-Outsider	Uncertainty-Outsider
USB	Universal Serial Bus
VM	virtuelle Maschine
VO	Verordnung
WpHG	Wertpapierhandelsgesetz

1 Einleitung

Insider[1] sind wichtige Funktionsträger unserer Gesellschaftsstrukturen. Sie bilden die Grundlage für Organisationseinheiten, Infrastrukturen oder zwischenmenschlichen Interaktionen. Ohne sie kann keine Zusammenarbeit, Gruppenbildung und Aufgabendelegation funktionieren. So braucht es zum Beispiel in einem Unternehmen Insider, um ein Gebäude instand zu halten und Kommunikationsdienste zu administrieren oder auch zur Anleitung von Organisationssubstrukturen und zur Erbringung von Leistungen.

Mit dieser herausgehobenen Stellung von Insidern geht in der Regel ein Auftrag einher, beispielsweise die Erfüllung einer Aufgabe oder die Lösung eines Problems. Diese herausgehobene Stellung für Insider ist alternativlos. Ohne sie kann dieser Auftrag nicht ausgeführt werden. Dabei muss darauf vertraut werden, dass die Insider ihre Aufgaben zuverlässig erfüllen und mit ihrer Verantwortung gewissenhaft umgehen. Denn mit der Befähigung von Insidern zur Erfüllung ihrer Aufgaben entstehen gleichzeitig immer zwei Risiken. Zum einen können diese Befähigungen auf eine Art und Weise vom Insider eingesetzt werden, die so nicht vorgesehen war oder erlaubt ist und zum anderen können diese Befähigungen in die falschen Hände geraten.

Das Problem der Insiderbedrohung ist bereits aus den Zeiten der frühen Menschheitsgeschichte überliefert. So ist im Buch Exodus Kapitel 1 der Bibel in den Versen 8–11 die Angst des damaligen Pharaos von Ägypten beschrieben, dass sich die Volksgruppe der Israeliten, die sich in seinem Land aufhielten, mit seinen Gegnern verbündet und sich von innen heraus gegen ihn richten könnte. Der Pharao ordnete Maßnahmen der Überwachung und Unterdrückung an, um diese Bedrohung abzuwehren. Maßnahmen der Überwachung werden auch heutzutage eingesetzt, die einerseits abschreckende Wirkungen auf derartiges Fehlverhalten von Insidern haben sollen und die andererseits die Erkennung und Abwehr von Fehlverhalten ermöglichen sollen.

In dieser Dissertation stehen Insider sowie Bedrohungen für und durch Insider im Fokus. Das Ziel ist eine eindeutige Charakterisierung verschiedener Insidertypen und eine Systematisierung von Insiderbedrohungen und Gegenmaßnahmen. Darüber hinaus ist die bereits erwähnte spezifische Gegenmaßnahme der umfassenden Überwachung von Insidern Forschungsschwerpunkt dieser Dissertation. Dabei wird untersucht, wie eine solche Gegenmaßnahme sowohl effektiv und effizient als auch datenschutzfreundlich gestaltet und umgesetzt werden kann.

1.1 Motivation

Mitarbeiter des Federal Bureau of Investigation (FBI) haben sich in den vergangenen Jahren tausendfach unautorisiert Informationen über Personen aus einer Datenbank beschafft, die zur Abfrage in Verdachtsfällen bei der Bekämpfung von Terrorismus und Cyberkriminalität vorgesehen war [Wei19]. Die Agenten suchten allerdings aus persönlichem Interesse nach Freunden,

1. In dieser Dissertation wird durchgehend das generische Maskulinum verwendet. Es schließt damit weibliche wie männliche Akteure gleichermaßen mit ein und soll dabei den Lesefluss sowie die Grammatik nicht stören. Das gilt nicht nur für den Begriff *Insider*, sondern auch für alle anderen Begriffe, die mit ihrem Genus ein biologisches Geschlecht implizieren können.

Verwandten oder Mitarbeitern. Im Hinblick auf die Art und Weise des Fehlverhaltens umfassen diese Insiderangriffe allein die Interaktionen der Insider mit einem speziellen Informationssystem. Im Jahr 2019 wurde der Ex-Apple-Anwalt Gene Levoff, seinerzeit zuständig für die Einhaltung von Börsenregeln und insbesondere für die Konformität mit den Insiderhandel-Richtlinien des Unternehmens, selbst wegen Insiderhandel während seiner Amtszeit in mehreren Fällen angeklagt [LF19]. Er soll Insiderinformationen genutzt haben, um mit Wertpapierhandel Gewinne zu erzielen. Das Fehlverhalten des Insiders wird hier erst durch seine Interaktionen am Börsenmarkt, also einem System, das nicht zum Unternehmen gehört, ersichtlich. Bei einem Angriff auf die Firma Sony im Jahr 2014 verschafften sich die Angreifer mithilfe von Sony-Mitarbeitern unerlaubten physischen Zutritt zum Unternehmen und nutzten gestohlene Zugangsdaten eines Systemadministrators, um Zugriff auf sensible Unternehmensdaten zu erhalten und im Anschluss die Spuren ihres Angriffs zu verwischen [Bor14]. Es handelte sich hierbei sogar um mehrere Insiderangriffe. Einerseits sind Mitarbeiter des Unternehmens als Insiderangreifer involviert und andererseits agieren auch externe Angreifer als Insiderangreifer.

Die Vielschichtigkeit des Problems der Insiderbedrohungen, die bereits anhand dieser Beispiele deutlich wird, wurde in der Erforschung und Entwicklung von Gegenmaßnahmen bisher nicht berücksichtigt und in Grundlagenarbeiten bisher nur unzureichend systematisch aufgearbeitet. Sowohl ein Insider an sich als auch die Bedrohungen durch einen Insider umfassen mehrere unterschiedliche Aspekte, die beim Einsatz von Bedrohungserkennungs- und Abwehrmaßnahmen berücksichtigt werden müssen. Dadurch lassen sich Forschungsarbeiten und -ergebnisse, die sich direkt oder indirekt auf einen speziellen Insidertyp fokussieren, nicht mit den Forschungsarbeiten vergleichen, die sich mit anderen Insidertypen beschäftigen. Aus den jeweiligen Arbeiten selbst wird diese Problematik allerdings nicht beziehungsweise nicht explizit ersichtlich. Stattdessen wird der Begriff *Insider* inflationär und allumfassend eingesetzt und verleitet dazu, unterschiedlich gelagerte Forschungen fälschlicherweise miteinander zu vergleichen und mögliche beziehungsweise notwendige Synergieeffekte zu übersehen.

Ein weiterhin unterrepräsentierter Aspekt der Insiderthematik ist die Bedrohung für die Insider selbst, wenn Bedrohungserkennungs- und Abwehrmaßnahmen beispielsweise in Form von Überwachungen eingesetzt werden. Ein besonders brisantes Beispiel ist die Überwachung von Mitarbeitern beim Discounter Lidl, die im Jahr 2008 ans Licht kam und aufzeigte, wie detaillierte Informationen über die finanzielle Situation, persönliche Präferenzen und das Verhalten am Arbeitsplatz dokumentiert wurden [AD08]. Weitere Überwachungspraktiken von Mitarbeitern finden sich beispielsweise in speziell dafür vorgesehenen Softwareprodukten [Sol17]. Das damit verbundene Eingriffspotenzial in die Privatsphäre und die informationelle Selbstbestimmung von Insidern wird häufig in der Theorie und Praxis ignoriert, wurde aber Ende Oktober 2019 von der Datenethik-Kommission der deutschen Bundesregierung als unvertretbar eingestuft [Kre19].

1.2 Forschungsfragen und Methodik

Aus der dargelegten Motivation lassen sich die folgenden Forschungsfragen ableiten, die in dieser Dissertation adressiert werden. Die Forschungsfragen 1 und 2 gehen dem Problem nach, dass die Insiderthematik sowohl in Theorie als auch in Praxis auf keine einheitlichen Grundlagen zurückgreifen kann. Daher zielen sie darauf ab, grundlegende Systematisierungen der Insiderthematik zu entwickeln. Die Forschungsfragen 3 und 4 befassen sich mit einer konkreten Erkennungs- und Abwehrtechnik von Insiderbedrohungen sowie mit den durch solche Techniken

entstehenden Gefahren für den Datenschutz von betroffenen Insidern. Zum Einsatz kommen dabei die folgenden Forschungsmethoden:

- *Literaturrecherche* zu thematisch relevanten Forschungsarbeiten,
- *Systematisierung* existierender Forschungsarbeiten, Methoden und Bedrohungen,
- *Qualitative Analyse* von Definitionen, Szenarien und Verfahren,
- *Entwicklung* von Grundlagen, Strukturen und Verfahren,
- *Implementierung und Evaluation* der Strukturen und Verfahren.

Forschungsfrage 1: Insiderdefinitionen

Welche Charakteristiken definieren einen Insider und welche Insidertypen lassen sich daraus ableiten? Wie hängen diese Insidertypen miteinander zusammen beziehungsweise wie lassen sie sich voneinander abgrenzen? Mit welchen Mitteln lassen sich Insidertypen einerseits aus Insiderszenarien und andererseits aus Forschungsbeiträgen und Methodenbeschreibungen herleiten, um beide Seiten entsprechend des inhaltlichen Fokus zusammenzuführen?

Die Bearbeitung von Forschungsfrage 1 erfolgt in Kapitel 2 und leitet Grundlagen für Forschungen und Entwicklungen im Bereich der Insiderthematik her. Damit kann zukünftig eindeutig aufgezeigt werden, welcher Insider im Mittelpunkt steht. Weiterhin kann die Vergleichbarkeit und Abgrenzbarkeit unterschiedlicher Definitionen, Methoden und Verfahren erreicht werden. Der Fokus liegt hier zunächst allein auf Insidern in Abgrenzung zu Insiderbedrohungen beziehungsweise Insiderangreifern, die mit Forschungsfrage 2 adressiert werden.

Methodisch kommt dabei zunächst die *Literaturrecherche* thematisch relevanter Forschungsarbeiten zum Einsatz, mit deren Hilfe bestehende Probleme identifiziert werden, die eine einheitliche und allgemein akzeptierte und etablierte Definition eines Insiders bisher verhindert haben. Daraus ergibt sich die anschließende *Entwicklung* von Grundlagen, die einerseits den Kern verschiedener Insidertypen identifizieren und zu einer *Systematisierung* von unabhängigen Basis- und kombinierten Insidertypen führen. Andererseits wird eine Modellierung von Insidern *entwickelt*, die den identifizierten Problemen bisheriger Insiderdefinition begegnet und diese bereinigt. Darauf aufbauend wird mithilfe einer *qualitativen Inhaltsanalyse* von Definitionen und Szenarien aufgezeigt, welche Insidermodelle in existierenden Forschungsarbeiten und Insiderszenarien vorliegen. Diese Insidermodelle werden abschließend anhand der entwickelten Grundlagen *systematisiert*.

Die Ergebnisse der Bearbeitung von Forschungsfrage 1 zeigen auf, dass Insider nur vor dem Hintergrund einer speziellen Domäne und ausschließlich durch Charakteristiken, die von dieser Domäne bereitgestellt wurden, definiert werden können. Dabei konnten fünf definierende Insidercharakteristiken identifiziert werden, die einzeln oder beliebig kombiniert zu insgesamt 31 verschiedenen Insidertypen führen. In der qualitativ-inhaltlich analysierten Literatur zur Insiderforschung konnte ein Fokus auf lediglich drei dieser Insidertypen festgestellt werden.

Forschungsfrage 2: Insiderbedrohungen

Welche Bedrohungen für und durch Insider existieren und von welchem Insidertyp geht welche konkrete Bedrohung aus? Inwieweit adressieren existierende Sicherheitsmechanismen einzelne Aspekte dieser Insiderbedrohungen und lassen sich somit den identifizierten Insidertypen beziehungsweise deren Bedrohungen zuordnen?

Die Forschungsfrage 2 wird in den Kapiteln 3 und 4 untersucht. Dabei stehen Bedrohungen einer Domäne durch einen Insider sowie umgekehrt Bedrohungen eines Insiders durch seine Domäne im Mittelpunkt. In diesem Zusammenhang werden Bezüge zu den in Kapitel 2 entwickelten Insidertypen hergestellt. Darüber hinaus werden Erkennungs- und Abwehrmaßnahmen anhand der Insiderbedrohungen eingeordnet und somit deren verborgener Fokus auf verschiedene Insidertypen offengelegt.

Methodisch werden grundlegende Definitionen und Charakterisierungen bezüglich der Insiderbedrohungen *entwickelt*. Anhand dieser wird ausgearbeitet, welche Arten von Insiderbedrohungen existieren und wie sich diese in Anlehnung an die Bedrohungen der klassischen drei IT-Sicherheitsschutzziele Vertraulichkeit, Integrität und Verfügbarkeit *systematisieren* lassen. Anschließend werden diese identifizierten Insiderbedrohungen in detaillierter Weise und nach Vorgabe der Insidertypen aus Kapitel 2 aufgeschlüsselt. Weiterhin erfolgt auf Grundlage der Insidertypen sowie der konkreten Insiderbedrohungen eine *qualitative Analyse*. Darauf aufbauend wird eine *Systematisierung* existierender Sicherheitsmaßnahmen vorgenommen, die entweder dediziert oder zumindest potenziell die Bedrohungen durch Insider erkennen und abwehren.

Die Ergebnisse der Bearbeitung von Forschungsfrage 2 zeigen auf, dass sich Insiderbedrohungen in die *Eskalation*, die *Weitergabe* und die *Verhinderung* von Aspekten, die den Insiderangreifer oder andere Insider zu Insidern machen, aufschlüsseln lassen. Weiterhin wird erkenntlich, dass Insidercharakteristiken, wie sie in Kapitel 2 identifiziert wurden, Verbesserungen von Bedrohungen in den Dimensionen *Erfolg*, *Verdecktheit* und *Auswirkung* erwirken. Detaillierte Betrachtungen dieser Erkenntnisse lassen darüber hinaus Rückschlüsse auf Voraussetzungen und mögliche Abwehrmaßnahmen dieser Insiderbedrohungen zu.

Forschungsfrage 3: Insiderüberwachung

Wie lassen sich Insideraktivitäten möglichst detailliert erfassen und auf Insiderbedrohungen hin auswerten? Mit welchen Techniken lässt sich die dabei anfallende Datenmenge sowohl sinnvoll reduzieren, ohne dabei wichtige Aktivitätsdetails zu verlieren, als auch automatisiert auswerten, ohne dabei an Genauigkeit zu verlieren?

Die Forschungsfrage 3 wird in Kapitel 5 adressiert und legt den Fokus auf die Erkennung und Abwehr der Bedrohungen eines bestimmten Insidertyps, der nach den entwickelten Insidergrundlagen in Abschnitt 2.4 als sogenannter C-P-U-Insider eingeführt wird. Das Kapitel setzt sich mit einer Technik auseinander, die eine tiefgreifende und umfassende Überwachung von Insideraktivitäten erlaubt. Beim praktischen Einsatz dieser Technik muss mit dem komplexen Problem der Handhabung und akkuraten Auswertung der großen Menge an anfallenden Überwachungsdaten umgegangen werden. Die erarbeiteten Forschungsergebnisse leisten dafür wichtige Beiträge.

Methodisch werden speziell erfasste Überwachungsdaten von Aktivitäten an Rechnern derart aufbereitet, dass sie in einer neu entworfenen und *implementierten* Graphenstruktur ausgewertet werden können. Für diese Auswertung wird eine Methode *entwickelt* und *implementiert*, die die Graphenstruktur automatisiert auswertet und markante Charakteristiken in der Graphenstruktur offenlegt. Die Erkennungs- und Abwehrtechnik wird abschließend anhand von Echtdaten *evaluiert*.

Die Ergebnisse der Bearbeitung von Forschungsfrage 3 zeigen auf, dass die Graphenstruktur sowie die daraus offengelegten markanten Strukturcharakteristiken ein geeignetes Mittel für eine regelbasierte Erkennung von bedrohlichen Insideraktivitäten darstellen. Von zentraler Bedeutung

sind dabei die Häufigkeiten speziell angepasster sogenannter Netzwerkmotive. Diese wurden so konzipiert, dass sie für gleiche beziehungsweise ähnliche Aktivitäten Ähnlichkeiten in den Strukturcharakteristiken und für unterschiedliche Aktivitäten entsprechend große Abweichungen hervorrufen.

Forschungsfrage 4: Insiderdatenschutz

Welche Rechte, Pflichten und Anforderungen haben sowohl Insider als auch die Domäne eines Insiders als Gegenpart in Bezug auf den Datenschutz? Wie lassen sich wichtige Prinzipien des Datenschutzes auch im Kontext der Insiderüberwachung praktisch realisieren, sodass eine Insiderüberwachung nicht für andere Zwecke missbraucht werden kann, ohne dabei wichtige Maßnahmen der Domäne für den Schutz vor Insiderbedrohungen unbrauchbar zu machen? Welche zusätzlichen Maßnahmen können umgesetzt werden, um Schwächen von Datenschutztechniken zum Schutz der Insider vor unrechtmäßiger Überwachung bei der praktischen Realisierung zu beheben?

Mit der Bearbeitung der Forschungsfrage 4 in den Kapiteln 6 und 7 werden die gegenläufigen Interessen, Ziele und Rechte der von Erkennungs- und Abwehrtechniken gegen Insiderbedrohungen betroffenen Insider einerseits und der Domäne, die diese Techniken einsetzt, andererseits explizit benannt und bearbeitet. Dabei wird darauf geachtet, dass die mehrseitigen Interessen soweit es geht gewahrt bleiben, ohne die Interessen beziehungsweise Rechte der jeweiligen Gegenseite zu diskriminieren.

Methodisch wird zunächst anhand einer *Literaturrecherche* zur aktuellen europäischen und deutschen Datenschutzrechtslage im Unternehmens- und Beschäftigtenkontext der rechtliche Rahmen analysiert. Anschließend erfolgt ebenfalls anhand einer *Literaturrecherche* die Erarbeitung von Anforderungen an eine datenschutzfreundliche Überwachungstechnik sowie die Einordnung von De-Identifizierungsmaßnahmen im vorliegenden Kontext. Darauf aufbauend wird die in dieser Dissertation entwickelte Erkennungs- und Abwehrtechnik von Insiderbedrohungen mit datenschutzerhöhenden Techniken ausgestattet. Dafür werden zwei unterschiedliche Systemmodelle zur Pseudonymisierung von Ereignisnachrichten *entwickelt*, *implementiert* und *evaluiert*.

Die Ergebnisse der Bearbeitung von Forschungsfrage 4 zeigen auf, dass die Rechtslage in Deutschland sowohl den Datenschutz von Insidern als auch die Rechte von Domänen würdigt und beides im gegenseitigen Interesse ausgeglichen werden muss. Die Pseudonymisierung von Ereignisnachrichten wurde mithilfe kryptographischer Techniken angereichert, sodass der Datenschutz über das Standardniveau hinaus erhöht werden konnte. Ebenfalls ersichtlich und quantifizierbar wird der gleichzeitig negative Einfluss auf die Performanz durch die Implementierung sowie der Vergleich der neuen Datenschutztechnik in verschiedenen Konfigurationsstufen.

1.3 Forschungsbeiträge der Dissertation

Im Rahmen der Dissertation werden vier Forschungsbeiträge geleistet, die sich an eine datenschutzfreundliche Erkennung und Abwehr von Insiderbedrohungen richten. Die Ergebnisse sind für alle akademischen und praxisnahen Forschungs- und Entwicklungsarbeiten im Kontext von Insiderbedrohungen relevant, da grundlegende Systematisierungen erarbeitet werden, die eine Einordnung zukünftiger Beiträge erlauben (Forschungsfragen 1 und 2, Beiträge B1 und B2).

Darüber hinaus werden Vorschläge zur detaillierten Analyse von Insideraktivitäten (Forschungsfrage 3, Beitrag B3) sowie zur Aufrechterhaltung und Wahrung des Datenschutzes erarbeitet (Forschungsfrage 4, Beitrag B4).

B1: Insidertypisierung, -modellierung und -taxonomie

Es werden Grundlagen entwickelt, die eine eindeutige Benennung und Einordnung von verschiedenen Insidertypen erlauben. Dadurch zeigt sich einerseits, dass die existierenden Definitionen von Insidern in der Forschungsliteratur unterschiedliche Aspekte eines Insiders fokussieren. Andererseits wird deutlich, dass eine fehlende explizite Benennung dieser Aspekte zur aktuell vorliegenden unkoordinierten Erforschung der Insiderthematik beiträgt. Weiterhin wird eine Methodik zur Insidermodellierung vorgeschlagen, mit deren Hilfe die entwickelten Insidertypen verschiedenen Szenarien zugeordnet werden können. Dies ermöglicht die Erarbeitung von Insidermodellen zum Beispiel im Kontext von Bedrohungen und Abwehrmaßnahmen. Die Relationen der unterschiedlichen Insidertypen zueinander werden abschießend anhand einer Insidertaxonomie einsortiert. Dabei wird zunächst explizit auf die Betrachtung von Bedrohungs- beziehungsweise Angriffsaspekten verzichtet, die mit Forschungsbeitrag B2 abgedeckt werden.

B2: Systematisierung von Insiderbedrohungen und Abwehrmaßnahmen

Mit den in dieser Dissertation identifizierten Insidertypen sowie der entwickelten Insidertaxonomie werden Bedrohungen, die von Insidern im Allgemeinen und von den einzelnen Insidertypen im Speziellen ausgehen, herausgearbeitet und systematisiert. Als Vorbereitung darauf wird aufgezeigt, dass Insiderbedrohungen nicht gleichbedeutend sind mit Bedrohungen durch Insider, sondern dass spezielle Fähigkeiten und Umstände eine zentrale Rolle spielen, die durch vorhandene Insidercharakteristiken hervorgerufen werden. Darüber hinaus werden Insiderbedrohungen äquivalent zu Bedrohungen der klassischen Triade von Schutzzielen der IT-Sicherheit in drei Kategorien eingeteilt. Zudem wird aufgezeigt, dass die besagten Fähigkeiten und Umstände in drei unterschiedlichen Dimensionen Einfluss auf die Schwere von Insiderbedrohungen haben. Mit diesen Analysen und herausgearbeiteten Erkenntnissen werden anhand der Insidertaxonomie grundlegende Erkennungs- und Abwehrmechanismen systematisiert, um hervorzuheben, welchen Insidertypen mit welchen Abwehrmaßnahmen begegnet werden kann.

B3: Graphenbasierte Bedrohungserkennung und -abwehr

Es wird eine neue Erkennungs- und Abwehrmaßnahme von Insiderbedrohungen konzipiert und implementiert. Dabei kommen Ereignisnachrichten von überwachten Systemaufrufen auf Betriebssystemebene eines Rechners zum Einsatz, die in eine speziell entwickelte Graphenstruktur überführt werden. Mit der Analyse spezieller Charakteristiken dieser Graphenstruktur werden Rückschlüsse bezüglich aufgezeichneter Aktivitäten ermöglicht. Mithilfe von voreingestellten Signaturen können diese Aktivitäten automatisiert eingeschätzt und bewertet werden.

B4: Datenschutzfreundliche Ereignispseudonymisierung

Es werden Techniken zur Erhöhung des Datenschutzes entworfen, implementiert und evaluiert, wodurch Rechte und Interessen der Insider berücksichtigt werden, die von einer Überwachungstechnik zum Schutz vor Insiderbedrohungen betroffen sind. Dabei werden gleichzeitig die Rechte

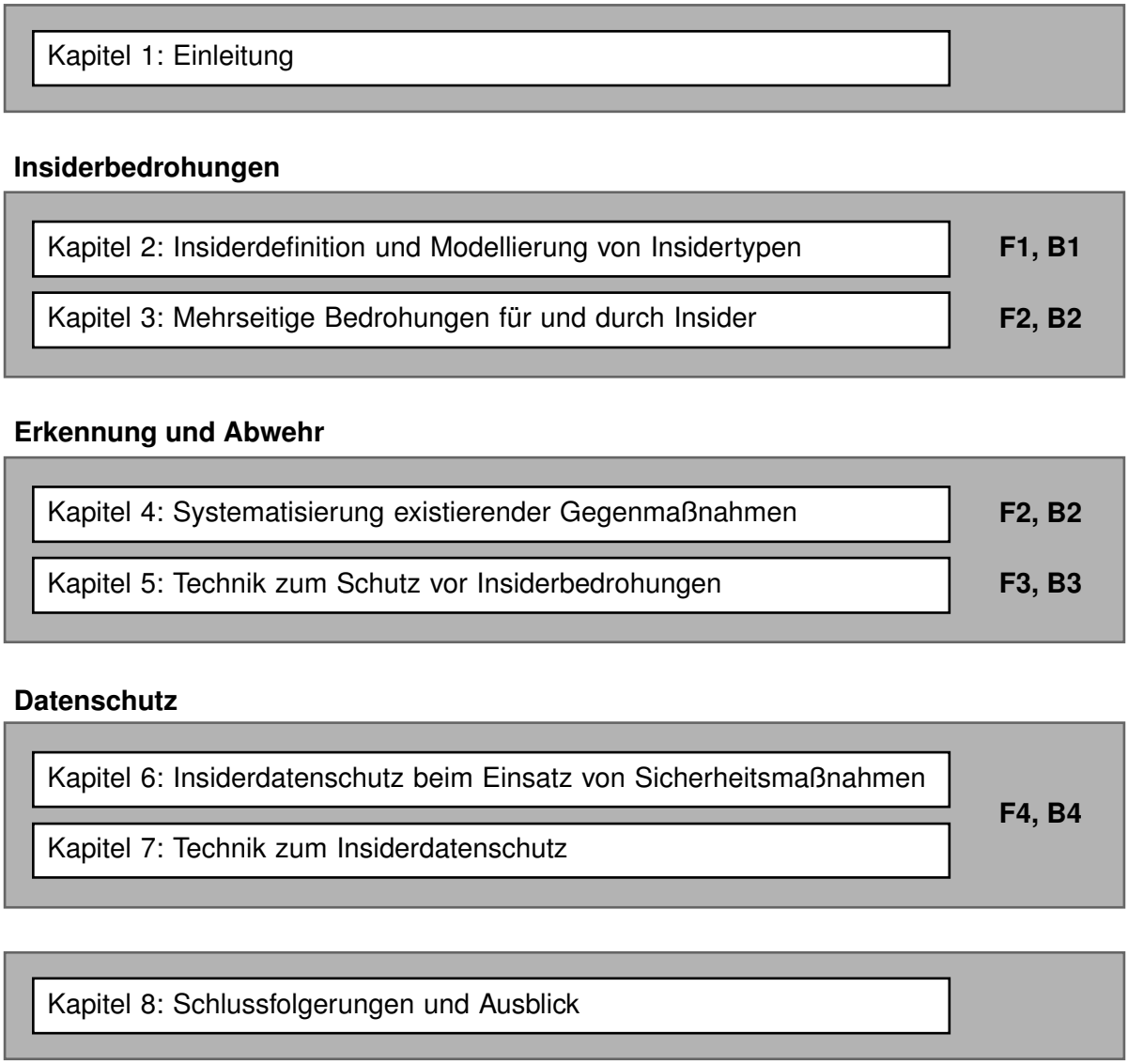


Abbildung 1.1: Aufbau der Dissertation

und Interessen einer Domäne beachtet und bewahrt. Es zeigt sich, dass die gegenläufigen Interessen aller Betroffenen ausbalanciert werden können, dafür aber ein gewisser Performanzverlust in Kauf genommen werden muss.

1.4 Aufbau der Dissertation

Die Dissertation umfasst neben der Einleitung und der Zusammenfassung inhaltlich drei Teile, die in jeweils zwei Kapitel gegliedert sind. Abbildung 1.1 zeigt die Struktur der Dissertation in Verbindung mit den jeweiligen Forschungsfragen sowie den jeweiligen Forschungsbeiträgen.

In Kapitel 1 wurde bereits die Motivation sowie die Forschungsfragen, -methodiken und Beiträge aufgezeigt. Im letzten Abschnitt dieses Kapitels erfolgt eine Einführung in grundlegende Begriffe und Konzepte für diese Dissertation.

Der inhaltlich erste Teil umfasst die systematische Aufarbeitung der Insiderthematik und der Insiderbedrohungen. In der Regel wird dieser Teil als Grundlagenteil aufgefasst, der allerdings in dieser Dissertation neu erarbeitet wird. Aus diesem Grund sollte dieser auch von Lesern gelesen werden, die mit der Insiderthematik bereits vertraut sind. In Kapitel 2 wird die vielschichtige Problematik einer allgemeinen und umfassenden Insiderdefinition aufgezeigt und mit der Einführung von Insidertypen sowie der Insidermodellierung aufgelöst. Weiterhin erfolgt die Erarbeitung einer Insidertaxonomie, mit der alle möglichen Insidertypen relativ zueinander eingeordnet werden (Forschungsfrage 1 und Beitrag B1). Kapitel 3 befasst sich anschließend mit den Bedrohungen, die sowohl von Insidern ausgehen, als auch auf Insider einwirken. Ersteres steht dabei im Fokus und erfolgt detailliert aufgeschlüsselt für die verschiedenen Insidertypen (Forschungsfrage 2 und Beitrag B2).

Im zweiten Teil der Dissertation werden Erkennungs- und Abwehrmaßnahmen von Insiderbedrohungen betrachtet. In Kapitel 4 erfolgt eine Systematisierung von grundlegenden Sicherheitsmechanismen, die beim Schutz vor Insiderbedrohungen eine wichtige Rolle spielen (Forschungsfrage 2 und Beitrag B2). Die Systematisierung richtet sich an der erarbeiteten Insidertaxonomie aus. In Kapitel 5 wird der Fokus auf einen speziellen Sicherheitsmechanismus gerichtet und eine Technik zur signaturbasierten Erkennung und Abwehr von Insiderbedrohungen entwickelt (Forschungsfrage 3 und Beitrag B3).

Im dritten Teil geht es um die Bedrohung von Insidern durch Bedrohungserkennungs- und Abwehrmaßnahmen mittels Überwachung und damit auch durch die in Kapitel 5 entwickelte Technik. Der Fokus liegt daher auf dem Datenschutz. In Kapitel 6 werden sowohl rechtliche als auch funktionale Anforderungen an eine datenschutzfreundliche Erkennungs- und Abwehrtechnik aufgezeigt und gegenläufige Interessen beziehungsweise Pflichten gegenübergestellt. Kapitel 7 befasst sich darauf aufbauend mit datenschutzerhöhenden Maßnahmen, die auf Überwachungstechniken wie die aus Kapitel 5 angewandt werden können.

Abschließend erfolgt in Kapitel 8 eine Zusammenfassung der Inhalte dieser Dissertation sowie ein Ausblick auf weiterführende Aspekte und Arbeiten.

1.5 Grundlegende Begriffe und Konzepte

Die eingehende Betrachtung der Insiderthematik in Bezug auf Definitions- und Bedrohungsaspekte sowie auf Schutzmaßnahmen in dieser Dissertation setzt ein Verständnis der folgenden Begriffe und Konzepte voraus. Weitere Grundlagen bezüglich der inhaltlichen Schwerpunkte dieser Dissertation in der Entwicklung der Erkennungs- und Abwehrmaßnahme sowie der Datenschutzmechanismen sind in Abschnitt 5.1 und Abschnitt 7.1 zu finden.

1.5.1 Der Insider

Eine gründliche Beschreibung beziehungsweise eine feststehende Begriffsdefinition eines Insiders ist an dieser Stelle noch nicht möglich und Versuche existierender wissenschaftlicher Veröffentlichungen, wie sie in Abschnitt 2.1.1 exemplarisch aufgezeigt werden, sind daran bisher immer wieder gescheitert [HP11; Zim+16]. Eine detaillierte Betrachtung der Probleme von Insiderdefinitionen sowie die Erarbeitung von Lösungswegen wird daher in Kapitel 2 vorgenommen.

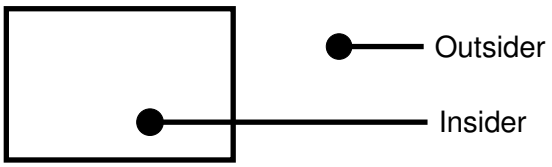


Abbildung 1.2: Stark vereinfachte und inakkurate Auffassung eines Insiders

Allen bisherigen Versuchen von Insiderdefinitionen ist allerdings gemein, dass sie unter einem Insider – im Unterschied zu einem Outsider – im Allgemeinen eine Person verstehen,[2] die sich durch besondere Charakteristiken abhebt und dadurch eine engere Verbindung zu einem Unternehmen oder einem System aufweist. Dieser stark vereinfachten Darstellung, wie Abbildung 1.2 illustriert, fehlt es allerdings an Akkuratesse. So kann ein Insider für bestimmte Teilbereiche eines Unternehmens auch als Outsider angesehen werden oder ein eigentlicher Outsider kann sich unerlaubt in die Lage eines Insiders versetzen und dennoch nicht uneingeschränkt als solcher angesehen werden.

In dieser Dissertation wird zunächst eine explizite Trennung zwischen der Definition eines Insiders (s. Kapitel 2) und der Definition einer Insiderbedrohung (s. Kapitel 3) vorgenommen. Beide dieser Aufgaben weisen ihre eigenen komplexen und unwägbaren Schwierigkeiten auf, etwa die ungünstige Vermischung von Charakteristiken eines Insiders mit Spezifika einer Bedrohung (s. Abschnitt 2.2.7). Die angesprochene engere Verbindung eines Insiders zu einem Unternehmen oder einem System wird im weiteren Verlauf auch als *Insidergrad* (engl. insiderness) bezeichnet.

1.5.2 Die Domäne eines Insiders

Bei der einführenden und sehr groben Beschreibung eines Insiders im vorigen Abschnitt wurde eine enge Verbindung zu einem Unternehmen oder einem System angesprochen. Tatsächlich kann es Insider im Kontext von weiteren, sehr unterschiedlichen Bezugspunkten geben, etwa einem Informationssystem oder Teilen eines Informationssystems, einem Rechnernetz, einem Software- oder Hardwareprodukt, einem Sicherheitsmechanismus oder einem Protokollablauf. Diese unterschiedlichen Bezugspunkte und Kontexte sind von zentraler Bedeutung für die Definition oder Beschreibung eines Insiders, wie auch in Abschnitt 2.2.1 diskutiert wird, denn bei genauerer Betrachtung wird deutlich, dass ein Insider eines speziellen Kontextes nicht zwangsläufig auch ein Insider beziehungsweise derselbe Insider in einem anderen Kontext sein muss. Dieser Bezugspunkt wird nachfolgend als *Domäne eines Insiders* bezeichnet. Wenn nicht anderweitig angegeben, kann ohne Beschränkung der Allgemeinheit von einem Unternehmen als Domäne eines Insiders ausgegangen werden.

Eine Domäne beinhaltet Dienste und interne Ressourcen, etwa Informationssysteme oder Dokumente, mit denen Insider interagieren können. Die Insider einer Domäne sind neben den Diensten und internen Ressourcen natürlich selbst in spezieller Weise Teil der Domäne. Der Einfachheit halber wird deshalb bei der Interaktion eines Insiders mit den Diensten und Ressourcen nachfolgend von einer Interaktion mit der Domäne gesprochen. Ebenso wird auch die Interaktion von Outsidern mit Insidern als Interaktion mit der Domäne bezeichnet.

2. Ein Insider kann sich im Speziellen auch auf ein Gerät oder einen Prozess beziehen. In den meisten Fällen lassen sich allerdings auch Geräte und Prozesse mittelbar oder unmittelbar auf eine Person zurückführen. Daher wird in dieser Dissertation von einem Insider als Person gesprochen.

1.5.3 Angriffe, Angreifer und Bedrohungen

Insbesondere in Kapitel 3 wird auf den Insider als Angreifer sowie dessen Bedrohungen eingegangen. Die nachfolgenden Erläuterungen der zugehörigen Grundbegriffe *Angreifer*, *Angriff* und *Bedrohung*, die dieser Dissertation zugrunde gelegt werden, basieren auf den beiden Glossarien von Shirey [Shi07] und Kissel [Kis13]:

Ein *Angriff* wird beschrieben als eine Methode oder Technik, die von einer Person zur bewussten Verletzung von Sicherheitsmechanismen oder Schutzzielen einer Domäne verwendet wird. Oftmals wird ein Angriff gleichzeitig als bösartig bezeichnet, was allerdings in dieser Dissertation differenziert und in Abschnitt 3.2.1.3 näher erläutert wird.

Ein *Angreifer* kann beschrieben werden als eine Person, die mit bewussten Handlungen oder mit der bewussten Unterlassung von Handlungen Sicherheitsmechanismen einer Domäne oder deren Schutzziele verletzt oder plant, zu verletzen.

Die Bedeutung einer *Bedrohung* umfasst in dieser Dissertation jegliches Ereignis oder jeglichen Umstand, der das Potenzial hat, eine Domäne nachteilig zu beeinflussen. Damit werden Angriffe ebenso eingeschlossen, wie unvorhersehbare Ereignisse und unabsichtliche Handlungen beziehungsweise unabsichtliche Unterlassungen von Handlungen, die alle das Potenzial haben, Sicherheitsmechanismen einer Domäne oder deren Schutzziele zu verletzen. In Anlehnung an Shirey [Shi07] ist ein Angreifer damit einer von mehreren möglichen Bedrohungsagenten und ein Angriff dementsprechend eine von mehreren möglichen Bedrohungsaktionen.

Der Bedrohungsbegriff wird hier bewusst sehr allgemein festgelegt, da aus Sicht einer Domäne jegliche Bedrohung ohne Hintergrundwissen als Angriff aufgefasst und entsprechend abgewehrt werden muss. Nähere Ausführungen dazu finden sich in Abschnitt 3.2.1.3.

1.5.4 Erkennung und Abwehr von Bedrohungen

Mit der Erkennung und Abwehr von Bedrohungen werden Sicherheitsmaßnahmen beschrieben, die die Eintrittswahrscheinlichkeit, die Verdecktheit oder die Auswirkungen von Bedrohungen oder eine beliebige Kombination dieser reduzieren. Nähere Informationen dazu finden sich in der Arbeit von Nowey [Now11, Abschnitt 2.5.8] sowie in Definition 3.3 in Abschnitt 3.2.5 dieser Dissertation. Neben detektiven und reaktiven Maßnahmen sind damit insbesondere auch präventive Maßnahmen eingeschlossen.

Detektive Maßnahmen konzentrieren sich auf die Erkennung von Bedrohungen und reduzieren deren Verdecktheit, sodass bedrohliche oder potenziell bedrohliche Vorgänge nicht unerkannt bleiben, die entweder bereits stattgefunden und somit zu einem Schaden geführt haben oder die gerade stattfinden beziehungsweise erst noch stattfinden werden. Reaktive Maßnahmen bauen auf der Detektion von Bedrohungen auf und leiten angemessene Handlungen ein, die bereits vergangene Bedrohungen behandeln, eventuell entstandene negative Auswirkungen für eine Domäne beseitigen und mit den Erkenntnissen aus Analysen der Vorfälle Verbesserungen im Sinne der bereits genannten Reduzierungen vornehmen. Präventive Maßnahmen nehmen diese Reduzierungen im Vorfeld einer Bedrohung oder spätestens während der Durchführung einer Bedrohungsaktion vor, noch bevor es zu einer konkreten negativen Auswirkung für eine Domäne kommen kann.

1.5.5 Sicherheitsbegriff und Elementare Gefährdungen

Das Bundesamt für Sicherheit in der Informationstechnik (BSI) beschreibt in seiner 2. Edition des IT-Grundschutz-Kompendiums 47 elementare Gefährdungen (G 0.1 – G 0.47), die generelle Aspekte spezifischer Einzelgefährdungen zusammenfassen [Bun19]. Diese elementaren Gefährdungen können in Gefährdungen eingeteilt werden, die aufgrund von *unbeabsichtigten* Ereignissen entstehen, etwa durch Katastrophen, durch äußere Einflüsse und Bedingungen, durch Fahrlässigkeit oder durch Verschleiß, sowie in Gefährdungen, die *beabsichtigt* hervorgerufen werden. Eine Übersicht ist in Tabelle 1.1 zusammengefasst.

Der Schutz vor beziehungsweise die Abwehr von *unbeabsichtigten* Gefährdungen wird im Englischen mit dem Begriff *safety* bezeichnet, während für den Schutz vor beziehungsweise der Abwehr von *beabsichtigten* Gefährdungen der Begriff *security* verwendet wird [FP00]. Im Deutschen gibt es eine solche sprachliche Unterscheidung nicht und wird mit dem Wort *Sicherheit* zusammengefasst. Die in dieser Dissertation betrachteten Gefährdungen, die im weiteren Verlauf als Bedrohungen bezeichnet werden, beschränken sich auf *beabsichtigte* Ereignisse, wobei im Kontext von Insideraktivitäten unbeabsichtigte Bedrohungen, insbesondere in Form von Fahrlässigkeiten, sowie deren Erkennung und Abwehr durchaus sehr relevant sind. Diese Art der Aktivitäten ist allerdings einerseits nicht gleichbedeutend mit den hier gelisteten unbeabsichtigten Gefährdungen und andererseits ohne Kontextwissen von beabsichtigten Bedrohungen zunächst nicht zu unterscheiden. Für eine genauere Diskussion wird auf Abschnitt 3.2.1.3 verwiesen. Dementsprechend wird der Begriff der *Sicherheit* auf die hier gelisteten *beabsichtigten* Gefährdungen bezogen.

1.5.6 Schutzziele der Informationssicherheit

Die Bedrohungen einer Domäne lassen sich in die drei grundsätzlichen Kategorien

- unbefugter Informationsgewinn,
- unbefugte Modifikation von Informationen und
- unbefugte Verhinderung von Ressourcennutzung

einteilen [VK83]. Aus dem Schutz vor diesen Bedrohungen ergeben sich die drei klassischen Schutzziele der Informationssicherheit: *Vertraulichkeit*, *Integrität* und *Verfügbarkeit*. Eine weitere Unterteilung dieser klassischen Schutzziele erfolgt von Pfitzmann [Pfi06] mit den Schutzzielen *Anonymität und Unbeobachtbarkeit*, *Zurechenbarkeit* sowie *Erreichbarkeit und legale Durchsetzbarkeit*. Diese verfolgen zwar ebenso den Schutz vor den drei grundsätzlichen Bedrohungskategorien, legen allerdings ihren Fokus eher auf die System- und Kommunikationsumstände, als auf die Inhalte. Im weiteren Verlauf dieser Dissertation werden die Schutzziele Vertraulichkeit, Integrität, Zurechenbarkeit und Verfügbarkeit für den Kontext von Domänen und Insidern näher betrachtet.

1.5.7 Vertrauensbereich

Der Vertrauensbereich ist jener Bereich eines Systems, dem zur Erreichung eines Schutzziels vollständig vertraut werden muss [FP97]. Er muss frei von Bedrohungen jeglicher Art sein, da zentrale Aufgaben der Informationssicherheit dort durchgeführt werden. Im klassischen Sinn werden dort sicherheitsrelevante Berechnungen durchgeführt sowie Geheimnisse verwaltet und

Tabelle 1.1: Elementare Gefährdungen des IT-Grundschutzkompendiums [Bun19], die aufgrund von unbeabsichtigten beziehungsweise beabsichtigten Ereignissen entstehen

Unbeabsichtigte Ereignisse			
G 0.1	Feuer	G 0.12	Elektromagnetische Störstrahlung
G 0.2	Ungünstige klimatische Bedingungen	G 0.17	Verlust von Geräten, Datenträgern oder Dokumenten
G 0.3	Wasser	G 0.18	Fehlplanung oder fehlende Anpassung
G 0.4	Verschmutzung, Staub, Korrosion	G 0.25	Ausfall von Geräten oder Systemen
G 0.5	Naturkatastrophen	G 0.26	Fehlfunktionen von Geräten oder Systemen
G 0.6	Katastrophen im Umfeld	G 0.28	Software-Schwachstellen oder -Fehler
G 0.7	Großereignisse im Umfeld	G 0.31	Fehlerhafte Nutzung oder Administration von Geräten und Systemen
G 0.8	Ausfall oder Störung der Stromversorgung	G 0.33	Personenausfall
G 0.9	Ausfall oder Störung von Kommunikationsnetzen	G 0.45	Datenverlust
G 0.10	Ausfall oder Störung von Versorgungsnetzen	G 0.46	Integritätsverlust schützenswerter Informationen
G 0.11	Ausfall oder Störung von Dienstleistern	G 0.47	Schädliche Seiteneffekte IT-gestützter Angriffe

Beabsichtigte Ereignisse			
G 0.13	Abfangen kompromittierender Strahlung	G 0.32	Missbrauch von Berechtigungen
G 0.14	Ausspähen von Informationen (Spionage)	G 0.34	Anschlag
G 0.15	Abhören	G 0.35	Nötigung Erpressung oder Korruption
G 0.16	Diebstahl von Geräten Datenträgern oder Dokumenten	G 0.37	Abstreiten von Handlungen
G 0.19	Offenlegung schützenswerter Informationen	G 0.38	Missbrauch personenbezogener Daten[a]
G 0.20	Informationen oder Produkte aus unzuverlässiger Quelle	G 0.39	Schadprogramm
G 0.21	Manipulation von Hard- oder Software	G 0.40	Verhinderung von Diensten (Denial of Service)[b]
G 0.22	Manipulation von Informationen[c]	G 0.41	Sabotage
G 0.23	Unbefugtes Eindringen in IT-Systeme	G 0.42	Social Engineering
G 0.29	Verstoß gegen Gesetze oder Regelungen	G 0.43	Einspielen von Nachrichten
G 0.30	Unberechtigte Nutzung oder Administration von Geräten und Systemen	G 0.44	Unbefugtes Eindringen in Räumlichkeiten

a. Es bestehen große Überschneidungen mit der Gefährdung G 0.36 Identitätsdiebstahl.

b. Es bestehen große Überschneidungen mit den Gefährdungen G 0.24 Zerstörung von Geräten oder Datenträgern, G 0.27 Ressourcenmangel sowie G 0.45 Datenverlust.

c. Es bestehen große Überschneidungen mit der Gefährdung G 0.46 Integritätsverlust schützenswerter Informationen.

gespeichert. Ein solcher Vertrauensbereich ist in der Regel wohldefiniert und abgrenzbar zu den Bereichen eines Systems, denen nicht oder nicht vollumfänglich vertraut werden muss. Anschauliche Beispiele sind Nutzerendgeräte, die den Vertrauensanker bei der sicheren Kommunikation über potenziell kompromittierte Kommunikationsstrecken darstellen.

Nach bisheriger Auffassung wird ein solcher Vertrauensbereich häufig mit der Domäne eines Insiders gleichgesetzt [HP11; MWB15], denn alle Personen, die Teil des Vertrauensbereichs sind beziehungsweise darauf Zugriff haben, können die Domäne von innen heraus und auf eine Art und Weise schädigen, wie es für einen Outsider nicht möglich ist. Daraus ergibt sich, dass der Vertrauensbereich gleichgesetzt wird mit der Abwesenheit von Schutzmaßnahmen und -möglichkeiten innerhalb einer Domäne und somit durch Insider ungehindert bedroht werden kann. Diese vereinfachte Auffassung eines Vertrauensbereiches wird in Abschnitt 2.2 kritisch hinterfragt und dessen Rolle bei der Charakterisierung eines Insiders in Abschnitt 2.3 differenziert betrachtet. Darüber hinaus wird im Laufe dieser Dissertation herausgearbeitet, dass auch innerhalb eines Vertrauensbereichs beziehungsweise der Domäne eines Insiders Schutzmaßnahmen vor diversen Insiderbedrohungen etabliert werden können.

1.5.8 Mehrseitige Sicherheit

Eine Domäne und ihre Insider stellen ein System mit mehreren Akteuren dar, deren Sicherheitsinteressen nicht zwangsläufig deckungsgleich sind. Das ist insbesondere dann der Fall, wenn es sich um ein verteiltes System handelt. Zwar stehen die Sicherheitsinteressen einer Domäne bei der Behandlung der Insiderthematik klar im Fokus, allerdings gilt es auch, die Sicherheitsinteressen aller anderen Akteure einzubeziehen und auszubalancieren. Rannenberg, Pfitzmann und Müller [RPM97] sprechen in diesem Fall von einer *Mehrseitigen Sicherheit*. Dazu gehören auch verschiedene Ebenen der Kooperation, die Federrath und Pfitzmann [FP97] zufolge aufgeteilt werden in die drei Fälle:

- **Unilateral durchsetzbare Sicherheitsinteressen**: Der persönlich erwünschte und erreichbare Schutz eines einzelnen Akteurs ohne die Unterstützung oder die Notwendigkeit anderer Akteure. Dementsprechend können die Sicherheitsinteressen der anderen Akteure nicht existent oder auch gegenläufig sein.
- **Bilateral durchsetzbare Sicherheitsinteressen**: Der durch die Zusammenarbeit zweier Akteure erwünschte und erreichbare Schutz beider Akteure ohne die Unterstützung oder die Notwendigkeit weiterer Akteure. Dementsprechend können die Sicherheitsinteressen der anderen Akteure nicht existent oder auch gegenläufig sein.
- **Multilateral durchsetzbare Sicherheitsinteressen**: Der durch Kooperation mehrerer Akteure erwünschte und erreichbare Schutz. Die Sicherheitsinteressen sind dementsprechend auf ein oder mehrere gemeinsame Interessen ausgerichtet.

Die Autoren identifizieren weiterhin verschiedene Bausteine, mit denen *Mehrseitige Sicherheit* erreicht werden kann. Diese Bausteine umfassen unter anderem Konzelationssysteme zur Erreichung der Vertraulichkeit, das Mix-Konzept zur Erreichung von Anonymität und Unbeobachtbarkeit, Authentifikationssysteme zur Erreichung von Integrität und Zurechenbarkeit sowie Diversität und Redundanz zur Erreichung von Verfügbarkeit. Derartige Bausteine sind allerdings unzulänglich, wenn der Vertrauensbereich eines Systems beziehungsweise von Akteuren verschwimmt und möglicherweise selbst im Fokus von Angriffen und Sicherheitsinteressen steht, so, wie im Kontext der Insiderthematik. Als Konsequenz müssen die Bedrohungen der Sicherheitsinteressen der verschiedenen Akteure in einem Insiderkontext genauer analysiert

werden, um adäquate Bausteine zur Erreichung der *Mehrseitigen Sicherheit* identifizieren zu können (s. Kapitel 3).

1.5.9 Angreifer- beziehungsweise Bedrohungsmodelle

Die Literatur hat einen dynamischen und domänenspezifischen Weg entwickelt, mit dem ein Angreifer oder eine Bedrohung modelliert werden kann [KL14; FP12a]. Ein solches *Angreifermodell* beschreibt den Angreifer mit konkreten Zuweisungen verschiedener Angreifercharakteristiken. Einerseits wird damit die Möglichkeit geschaffen, die maximale Angriffsstärke eines Angreifers festzulegen und zu benennen, gegen die ein System oder Verfahren gerade noch Sicherheit bietet. Andererseits wird damit eine bessere Fokussierung auf die verschiedenen wichtigen Charakteristiken eines Angreifers sowie auf Abwehrmaßnahmen im Kontext verschiedener Organisationen, verschiedener Systemteile, verschiedener Protokolle und verschiedener Sicherheitsmechanismen ermöglicht. Federrath und Pfitzmann [FP12a] erklären, dass ein Angreifermodell die Angreifercharakteristiken *Rolle* (Outsider versus Insider), *Verbreitung* (kontrollierte Stationen und Kanäle), *Verhalten* (passiv versus aktiv) sowie *Ressourcen* (limitiert versus unlimitiert) umfassen und spezifizieren sollte:

- **Rolle(n)**: Welche Rolle oder Kombination von Rollen kann ein Angreifer annehmen? Diese Rollen beschreiben die Relation zwischen dem Angreifer und dem Angriffsziel. Beispiele sind Unbeteiligter, Nutzer, Betreiber, Hersteller, Entwickler, Wartungstechniker oder Designer. Weiterhin können die eher abstrakten Rollen eines Outsiders und eines Insiders herangezogen werden.
- **Verbreitung**: Welche und wie viele Systemteile oder Subsysteme können vom Angreifer mithilfe seiner Rollen genutzt und kontrolliert werden? Die Verbreitung kann als eine Gewichtung der Angreiferrolle(n) angesehen werden.
- **Verhalten**: Es gibt zwei verschiedene Aspekte im Verhalten eines Angreifers. Erstens die Interaktion eines Angreifers mit dem Angriffsziel und zweitens die Protokolltreue des Angreifers mit Bezug auf Grundsätze, Spezifikationen oder Richtlinien. Die Interaktionen sind entweder *passiv*, also praktisch nicht-existent, denn der Angreifer fängt nur die Eingabe- und Ausgabewerte des Angriffsziels ab. Oder die Interaktionen sind *aktiv*, was bedeutet, dass der Angreifer das Angriffsziel kontrollieren beziehungsweise steuern und damit dessen Zustand beeinflussen kann. Die Protokolltreue ist ebenso zweifältig. Entweder ist das Verhalten eines Angreifers *protokolltreu*, was bedeutet, dass sein Verhalten keine erkennbare Abweichung von erwarteten oder erlaubten Interaktionsabläufen mit dem Angriffsziel darstellt. Die Autoren bezeichnen diesen Fall auch als *beobachtend*, was allerdings aufgrund der Begrifflichkeit zu Missverständnissen führen kann. Oder das Verhalten ist *protokolluntreu* beziehungsweise *modifizierend*, was die verifizierbare illegitime Interaktion mit dem Angriffsziel bedeutet.
- **Ressourcen**: Wie viele Ressourcen, häufig in Form von Zeit und Geld ausgedrückt, besitzt der Angreifer? Eine grobe Beschreibung dieser Charakteristik ist, ob der Angreifer *komplexitätstheoretisch* beschränkt ist oder nicht.

Die Modellierung eines Angreifers ist nicht auf die Instanziierung dieser vier Charakteristiken allein beschränkt, sondern ist möglicherweise sehr viel spezifischer. Diese Charakteristiken haben sich allerdings bei der Modellierung von Angreifern als sehr nützlich erwiesen. Ein klassischer Ansatz bei der Modellierung eines Angreifers ist die Unterscheidung zwischen den beiden Rollen eines Outsiders und eines Insiders. Es gibt allerdings Situationen, in denen diese eher breite und

inakkurate Unterscheidung die Umstände eines Angriffsziels nicht befriedigen, zum Beispiel wenn ein Angreifer beide Rollen zur gleichen Zeit in sich vereint.

1.5.10 Ontologien und Taxonomien

Bei einer *Ontologie* handelt es sich Gruber [Gru09] zufolge im Kontext von Informationstechnologien um die Definition einer Menge von repräsentativen Grundlagen und Primitiven, mit denen ein Spezialgebiet oder Diskurs strukturiert und modelliert werden kann. Die repräsentativen Grundlagen umfassen in der Regel Klassen oder Mengen, Attribute oder Eigenschaften sowie Relationen zwischen den Klassen- beziehungsweise Mengenelementen. Die Definition der repräsentativen Grundlagen beinhaltet Informationen über deren Bedeutung und über Bedingungen ihrer logisch konsistenten Anwendung.

Dementsprechend handelt es sich bei einer *Taxonomie* um eine leichte Form oder um einen Teil einer Ontologie [GZ09], mit dem die spezielle *ist-ein* Relation zwischen den Klassen- beziehungsweise Mengenelementen beschrieben wird und somit die eigentliche Klassifizierung vorgenommen wird [IW08].

In der Literatur werden diese beiden Begriffe häufig synonym verwendet. Dieser Dissertation werden jedoch nachfolgend die erläuterten Begriffsbedeutungen zugrunde gelegt.

2 Insiderdefinition und Modellierung von Insidertypen

Der Begriff eines *Insiders* ist im allgemeinen Sprachgebrauch ein feststehender Begriff und wird in der Regel ohne weitere Erläuterungen oder Eingrenzungen verwendet [Sol17; LF19]. Problematisch wird es allerdings, wenn genauer beschrieben werden muss, was ein *Insider* ist, welche Eigenschaften er hat und wie er sich von anderen *Nicht-Insidern* abhebt. So ist es zum Beispiel in Forschungsarbeiten notwendig, den Begriff und den Kontext eines Insiders explizit und detailliert zu definieren, da diese Details oft eine wichtige Rolle in der Einordnung der Forschungsergebnisse spielen. Über die letzten Jahrzehnte hinweg haben sich dabei unterschiedlichste Insiderdefinitionen herausgebildet. Anders als es sich vermuten lässt, haben diese allerdings nicht zu einer klaren Einteilung verschiedener Insiderdefinitionen und einer Zuordnung von Forschungs- und Entwicklungsarbeiten zu feststehenden Insiderbeschreibungen geführt. Stattdessen wird weiterhin erfolglos nach der einen allumfassenden und allgemein akzeptierten Insiderdefinition gesucht. So wird weiterhin mit der Verwendung des Begriffs *Insider* eine homogene Forschungsrichtung suggeriert, obwohl sich die Insiderdefinitionen verschiedener Arbeiten teilweise fundamental unterscheiden. Darüber hinaus werden Forschungsarbeiten zum Thema *Insider* veröffentlicht, die sich mit anderen Forschungsarbeiten vergleichen, dabei allerdings verkennen, dass häufig sehr unterschiedliche Insideraspekte im Fokus stehen und somit eine saubere Vergleichbarkeit eigentlich nicht möglich ist.

Das Ziel dieses Kapitels ist die Erarbeitung und Offenlegung von Erkenntnissen im Bereich der Definition von Insidern, die grundlegenden Charakter haben. Damit wird die Forschungsfrage 1 adressiert (vgl. Abschnitt 1.2). Der Fokus wird dabei zunächst soweit möglich allein auf *Insider* gelegt, bevor in Kapitel 3 die entwickelten Grundlagen auf *Insiderbedrohungen* ausgeweitet und übertragen werden. Diese Grundlagen bieten Systematiken an, die auf dem gesamten Gebiet der Insiderforschung zukünftig angewendet werden können und somit spezifische Abgrenzungen, Zuordnungen und Synergien erlauben.

Wesentliche Inhalte Ausgehend von einer umfassenden Literaturrecherche werden in diesem Kapitel die größten Probleme bei der Definition von Insidern identifiziert und anschließend mit einer grundlegenden Beobachtung sowie einer Reihe von Systematisierungen und einer Modellierungsmethodik aufgelöst (Forschungsbeitrag B1 aus Abschnitt 1.3). Dabei werden Charakteristiken identifiziert, die einzig und allein Insidergrade hervorrufen. Im Gegensatz hierzu stehen Charakteristiken, die den Insideraspekt einer Person nicht berühren, sondern andere durchaus wichtige Eigenschaften explizieren. Mit dieser Erkenntnis werden Typisierungen von Insidern vorgenommen, die jeweils voneinander abgrenzbar, aber auch kombinierbar sind. Weiterhin werden Insidermodelle in Analogie zu Angreifermodellen eingeführt, die es erlauben, alle Insidergrade einer Person methodisch sauber zu identifizieren und nachvollziehbar zu benennen. Am Ende des Kapitels werden die entwickelten Insidertypen miteinander in Beziehung gesetzt und somit eine einfache Insidertaxonomie abgeleitet. Sie erlaubt es in späteren Kapiteln, Insiderbedrohungen sowie Erkennungs- und Abwehrmaßnahmen anhand dieser Insidertaxonomie auszurichten und damit ebenfalls zu systematisieren. Zusammengenommen bilden diese Arbeiten eine grundlegende Insiderontologie (vgl. Abschnitt 1.5.10).

Aufbau des Kapitels Das Kapitel beginnt in Abschnitt 2.1 mit einem Verweis auf Vorarbeiten, in denen Versuche von Insiderdefinitionen sowie von Systematisierungen vorgenommen wurden. Mit der Identifizierung und Beschreibung von Insiderdefinitionsproblemen in Abschnitt 2.2 wird aufgezeigt, welche Lösungen in den nachfolgenden Abschnitten erarbeitet werden. Abschnitt 2.3 richtet den Fokus auf spezielle Charakteristiken, die den Insidergrad einer Person näher beschreiben und anhand derer Insider unterschieden werden können. Damit wird in Abschnitt 2.4 die Typisierung von Insidern vorgenommen und Basis- sowie kombinierte Insidertypen etabliert. Anhand ausgewählter Insiderszenarien wird in Abschnitt 2.5 eine Methodik zur Modellierung von Insidern vorgeschlagen, die sich an den erarbeiteten Insidertypen ausrichtet. Im Anschluss daran wird in Abschnitt 2.6 eine neue Insidertaxonomie vorgestellt, die sich in einer einfachen Art und Weise aus der Typisierung von Insidern ableitet und dadurch einen intuitiven Hintergrund aufweist. Das Kapitel schließt in Abschnitt 2.7 mit einem Fazit ab.

2.1 Existierende Definitions- und Strukturierungsversuche

Bestrebungen der Herausarbeitung von Insiderdefinitionen sowie Versuche, verschiedene Insiderdefinitionen und Insiderszenarien zu systematisieren, sind nicht neu. In der Literatur finden sich zahlreiche Beschreibungen, Charakterisierungen und Definitionen von Insidern, die teilweise bereits mithilfe von Ontologien und Taxonomien versucht wurden einzuordnen und zu strukturieren.

2.1.1 Insiderdefinitionen

Anderson [And80] hat als einer der Ersten die Bedrohungen beschrieben, die von Insiderangreifern ausgehen. Er definiert dabei Insider als Personen, die in der Regel eine Autorisierung zur Nutzung eines technischen Systems oder spezieller Ressourcen auf diesem technischen System haben. Neumann [Neu99] nimmt an, dass Insider relativ zu einem bestimmten IT-System authentifiziert wurden, um innerhalb dieses Systems zu agieren. Die dabei autorisierten Zugriffsrechte und Privilegien werden vom Autor zur Qualifizierung dieser Insider verwendet, was einen sehr frühen Ansatz einer Beschreibung verschiedener Insidergrade darstellt. Wood [Woo00] charakterisiert Insider anhand der Eigenschaften Zugriff, Wissen, Privilegien, Risiken, Taktiken, Motivation und Prozess. Einwechter [Ein02] definiert Insider im Kontext eines Unternehmensnetzwerkes. Für seine Definition richtet er den Fokus auf die drei Charakteristiken Vertrauen, Autorisierung und Wissen. Dem Autor zufolge ist ein Insider eine Person, der einerseits ein autorisierter Zugriff zum Rechnernetz anvertraut wurde, um seine Verpflichtungen dem Unternehmen gegenüber zu erfüllen. Andererseits hat ein Insider oft eine große Menge an Wissen über die Rechnernetzarchitektur inklusive des Wissens darüber, an welchen Stellen anvisierte Dateien oder Systeme verortet sind. Insider werden von Patzakis [Pat03] als bösartige Mitarbeiter und andere vertraute Personen definiert. Brackney und Anderson [BA04] bezeichnen zum einen eine Person mit Zutritt, Zugang beziehungsweise Zugriff auf sensible Informationen und Informationssysteme, der bereits vertraut wird, als Insider. Zum anderen fassen sie auch eine Person mit Zutritt beziehungsweise Zugang, Privilegien oder Wissen von Informationssystemen oder -diensten als einen Insider auf. Butts, Mills und Baldwin [BMB05] beschreiben einen Insider als jede Person, der in einem Informationssystem jegliches Level an Vertrauen erteilt wurde. Pfleeger [Pfl08] stellt Beispiele und Szenarien verschiedener Insider bereit, zum Beispiel

ein Mitglied oder ein ehemaliges Mitglied einer Institution, ein formeller oder informeller Geschäftspartner, jegliche Person, die zur Durchführung bestimmter Aktivitäten autorisiert ist oder eine Person, die durch einen Outsider zur Durchführung von Aktionen getäuscht oder genötigt wird. Hunker und Probst [HP11] liefern einen Überblick verschiedener Insiderdefinitionen und argumentieren, dass das eigentliche Problem den sogenannten „real real insider“ betrifft. Den Autoren zufolge handelt es sich dabei um eine Person, die tief in einer Organisation eingebettet ist, großes Vertrauen genießt und sich in einer Position befindet, in der er große Schäden anrichten kann. Als Beispiele führen sie eine hochrangige Führungskraft oder ein Systemadministrator an. Sie richten darüber hinaus die Aufmerksamkeit auf die Insidereigenschaften Autorisierung, Authentisierung, Vertrauen, Konsequenzen und technische Expertise.

In der Literatur existieren viele weitere Interpretation der Begriffe *Insider*, *Insiderangreifer*, *Insiderangriff* und *Insiderbedrohung*, zum Beispiel in [Com98; BA04; Ein02; Bis05; BMB05; Shi07; Bis+09; Bis+10; Pro+10a; Kap+15]. Viele dieser Definitionen unterscheiden sich in den Insidercharakteristiken, die fokussiert werden, sowie in Annahmen darüber, was einen Insider von anderen Individuen unterscheidet. Diese große Anzahl an Definitionen und insbesondere an vielen Unterschieden untereinander deutet auf die hohe Komplexität hin, die eine grundlegende, sorgfältige und umfassende Insiderdefinition mit sich bringt. Alle genannten Insiderdefinitionen, einschließlich weiterer aus der existierenden Literatur, werden anhand der in den folgenden Abschnitten erarbeiteten Grundlagen in Abschnitt 2.5.3 genau eingeordnet und systematisiert

2.1.2 Insiderontologien und -taxonomien

Im Zuge der Diskussion von Auditierung und Überwachung der Computersicherheit in einem Unternehmen liefert Anderson [And80] eine kleine Taxonomie, indem er verschiedene Insider in die vier Kategorien externer[1] Eindringling, Maskierter, legitimer Nutzer und verborgener Nutzer klassifiziert. Jede Kategorie wird durch den Zugang zu einem Informationssystem sowie durch den Zugriff auf Ressourcen dieses Informationssystems spezifiziert. Neumann [Neu99] versucht, Insider durch deren Abgrenzung zu Outsidern zu klassifizieren und kommt zu dem Schluss, dass Insider nur in Relation zu einem Referenzpunkt eingeordnet werden können. Als Referenzpunkte nennt der Autor Eigenschaften wie Vertrauen, Privilegien oder referenzierte Daten. Ein Outsider beispielsweise, der durch erfolgreiches Eindringen in ein System unberechtigten Zugang erlangt, wird von einem mechanistischen Standpunkt aus gesehen zu einem Insider, der allerdings potenziell weniger Wissen über die Umgebung hat, als ein echter Insider. Der Autor klassifiziert missbräuchliches Verhalten von Insidern in beabsichtigte und unbeabsichtigte Aktivitäten sowie die nachweisbare Natur dieser Missbräuche in erkennbare und verborgene Aktivitäten. Tuglular [Tug00] präsentiert ein Schema von Insider-Computermissbrauchsvorfällen, das die Kategorisierung von Insidervorfällen sowie die Sammlung und Speicherung von Informationen über Insider-Computermissbräuche unterstützen soll. Das Schema beinhaltet die drei Kategorien Vorfall, Antwort und Konsequenzen, die jeweils mehrere Subkategorien enthalten. Wood [Woo00] beschreibt ein Modell von bösartigen Insidern, das zur Simulation und Analyse von Insidervorfällen implementiert werden kann. Magklaras und Furnell [MF01] schlagen eine Architektur für ein Vorhersagewerkzeug von Insiderbedrohungen vor, welches eine Taxonomie für Insidervorfälle beinhaltet. Darin werden die Insider in die drei Basis-Kategorien Systemrolle, Grund des Missbrauchs und Systemkonsequenzen sowie in weitere Unterkategorien einsortiert.

1. Die Terminologie ist an dieser Stelle missverständlich. Der Autor meint damit nicht ausschließlich Outsider beziehungsweise Personen, die nicht Teil des Unternehmens sind.

Auf die gleiche Art und Weise wie Anderson [And80] schlagen Cole und Ring [CR05] vier Kategorien von Insidern vor, die ebenfalls auf dem Grad an Zugang beziehungsweise Zugriff basieren. Diese vier Insiderkategorien werden von den Autoren allerdings (erhöhter) purer Insider, Mitarbeiterinsider, Partnerinsider und Partneroutsider genannt. Pfleeger u. a. [Pfl+10] präsentieren ein Rahmenwerk, das bei der Identifizierung von Ähnlichkeiten und Unterschieden von vorliegenden zu vergangenen Insidervorfällen helfen soll und somit die Erzeugung von Insideraktivitätsclustern für ein Unternehmen unterstützt. Das Rahmenwerk besteht aus zehn Fragen, die die Aspekte Organisation, Individuum, Informationstechnologie und Umgebung abdecken. Eine Antwortmenge dieser Fragen beschreibt die Einordnung eines spezifischen Insiders beziehungsweise einer spezifischen Insiderbedrohung.

Keine der existierenden Bestrebungen hat zu einer weitgehend akzeptierten Beschreibung und Einordnung von Insidern geführt [HP11; MPH13].

Aufgrund dieser Schwierigkeiten und Lücken bei der Einordnung von Insidern wird in den folgenden Abschnitten eine einfache Ontologie von Insidern inklusive einer Insidertaxonomie hergeleitet und erarbeitet, die nicht von vergangenen Insidervorfällen oder spezifischen Unternehmensumgebungen abhängt und keine Implementierung von neuen Rahmenwerken und Richtlinien erfordert.

2.2 Definitionsprobleme

Bei der Analyse existierender Arbeiten zu Insiderbedrohungen und Gegenmaßnahmen wurden die nachfolgend erläuterten Gründe identifiziert, die eine eindeutige, akkurate und vollständige Definition eines Insiders erschweren beziehungsweise bisher unmöglich machten. Lösungen der genannten Probleme werden in den Abschnitten 2.3 bis 2.5 erarbeitet und präsentiert.

2.2.1 Fehlende Benennung einer Insiderdomäne

Wie bereits in Abschnitt 1.5.2 ausgeführt, kann ein Insider nur in Verbindung mit einer bestimmten Domäne beschrieben werden. Eine fehlende Benennung dieser Domäne führt zu uneindeutigen Beschreibungen von Insidern, wie etwa die Insiderdefinition von Pfleeger u. a. [Pfl+10] zeigt:

> „A person with legitimate access to an organization's computers and networks."

Sofern es sich bei der Domäne in dieser Definition um die Organisation handelt, geht es hierbei um einen Insider, der legitimen Zugriff auf Ressourcen und Dienste der Domäne hat. Die Insiderdefinition ändert sich allerdings, wenn es sich bei der Insiderdomäne um die Computer und Rechnernetze der Organisation handelt. In diesem Kontext hat der Insider legitimen Zugang zur Domäne selbst, wobei über Zugriffe auf Ressourcen und Dienste der Domäne keine Aussage gemacht wird.

Damit zeigt sich, dass Definitionen und Eigenschaften von Insidern aus der Sicht verschiedener Domänen unterschiedlich aufgefasst und verstanden werden können. Demnach muss eine Domäne für eine unmissverständliche Beschreibung eines Insiders explizit benannt werden.

2.2.2 Dynamik von Domänen und deren Kontext

Was möglicherweise in der Vergangenheit als eine sehr gut geeignete Definition eines Insiders angesehen wurde, kann heutzutage aufgrund von fortschreitenden Änderungen im Kontext der Domänen als unangemessen und mittlerweile veraltet erscheinen. Werden beispielsweise Unternehmen und deren IT-Umgebungen als Domäne eines Insiders angenommen, dann waren Definitionen eines Insiders als ein Individuum mit privilegiertem, legitimem oder autorisiertem Zugriff zu unternehmensinternen Ressourcen, wie sie etwa in [Com98; Neu99; Pfl08; Pfl+10] angeführt werden, vor Jahrzehnten etablierte und angemessene Insiderdefinitionen. Die IT-Umgebung eines Unternehmens hat sich in den letzten Jahren allerdings grundlegend verändert [HP11]. Ausschlaggebend für diese Veränderungen sind folgende Entwicklungen:

- **Intern und extern vernetzte IT-Systeme**: Durch die Vernetzung vieler und möglicherweise auch aller IT-Systeme untereinander sowie die Verbindung mit externen Rechnernetzen, wie zum Beispiel mit dem Internet, muss ein Insider nun nicht mehr physisch an einem speziellen Ort in einer Domäne anwesend sein. Er kann auch von einem anderen Ort Aktivitäten und Interaktionen mit der Domäne beziehungsweise mit Ressourcen und Diensten der Domäne steuern. Darüber hinaus verändert sich die Identifizierung sowie der Mechanismus zur Prüfung einer behaupteten Identität eines Insiders. Weiterhin werden dadurch interne Systeme und Dienste für externe Personen erreichbar, die zuvor nicht für eine derartige Erreichbarkeit konzipiert wurden. Dadurch werden gewollte aber auch ungewollte Zugriffe von außerhalb einer Domäne ermöglicht. Eine spezielle Entwicklung dieser Vernetzung ist das sogenannte Internet der Dinge (engl. internet of things).
- **Externalisierung von Diensten und Funktionalitäten** (engl. outsourcing): Wenn vormals intern organisierte, durchgeführte und überwachte Aufgaben oder Teilaufgaben an externe Dienstleister ausgelagert werden, erhalten Personen anderer Domänen Einblicke und Zugriffe auf interne Ressourcen und Dienste. Sie werden somit potenziell ebenfalls zu Insidern. Eine besondere Form der Externalisierung von Diensten und Funktionalitäten ist die Nutzung von Cloud-Diensten.
- **Vermischung dienstlicher und privater Geräte**: Eine sehr spezielle Dynamik in Unternehmen ist die Erlaubnis, eigene private Geräte am Arbeitsplatz (engl. bring your own device) oder dienstliche Geräte auch privat zu nutzen. Dadurch kommen diese Geräte domänenübergreifend an verschiedenen Stellen mit unterschiedlichen Sicherheitsanforderungen zum Einsatz und ermöglichen einerseits den Transport von bösartigen und möglicherweise verdeckten Funktionalitäten von außerhalb einer Domäne in die Domäne hinein. Andererseits vereinfachen sie den Abtransport von Informationen und Ressourcen von innerhalb einer Domäne aus der Domäne hinaus.

Zusammengefasst verändern sich dadurch vormals wohldefinierte Nutzergruppen von IT-Systemen hin zu teilweise unbekannten und ungeprüften Nutzern, wie zum Beispiel externen Dienstanbietern und Beratern, Zeit- und Leiharbeitern oder Kunden mit mehr oder weniger privilegiertem Zugriff auf Teile der unternehmensinternen IT. Insiderdefinitionen, deren Fokus allein auf der Charakteristik *Zugriff* liegt, adressieren diese progressiven Veränderungen nicht adäquat.

Derartigen Insiderdefinitionen zufolge wäre sowohl ein externer Cloud-Dienstanbieter, der zwar Zugriff auf Unternehmensdaten hat, aber ansonsten diesem Unternehmen nicht zugehörig ist, als auch ein Kunde dieses Unternehmens, welcher Zugriff auf einen Nutzeraccount hat,

gleichermaßen ein Insider, wie ein lokaler Systemadministrator, der die interne IT-Infrastruktur des Unternehmens wartet und pflegt.

2.2.3 Insider versus Innenseite

Unter anderen geben Wood [Woo00], Shirey [Shi07] und Hunker und Probst [HP11] an, dass die Aktionen eines Insiders direkt assoziiert sind mit der Innenseite einer Domäne, also innerhalb des Perimeters der Domäne stattfinden. Diese explizite oder implizite Annahme ist eine allgemeine Fehleinschätzung bei der Definition von Insidern. Auf der einen Seite kann ein Individuum, das unerlaubt den Zugriff auf die Innenseite einer Domäne erlangt, in diese Domäne von innen heraus agieren und dennoch gleichzeitig als ein Nicht-Insider beziehungsweise Outsider definiert werden. Dabei handelt es sich um einen Fall, der in der Regel als sogenannter *Maskierer* (engl. masquerader) [And80] bezeichnet und kontrovers diskutiert wird, ob er immer noch zur Klasse der Insider gehört oder nicht [Sch+01; Pro+10a]. Auf der anderen Seite kann ein Individuum auf eine Domäne von außen einwirken und dennoch als Insider identifiziert werden. Ein Beispiel wäre ein Ex-Angestellter, der sein ehemaliges Unternehmen durch Zuhilfenahme seines Insiderwissens angreift. Folglich kann ein Insider nicht exklusiv anhand seiner logischen oder physischen Verortung innerhalb des Perimeters einer Domäne definiert werden.

Diese Feststellung sollte nicht verwechselt werden mit der Beobachtung von Neumann [Neu99], der in seinen Ausführungen den Unterschied zwischen logischer und physischer Präsenz innerhalb einer Domäne erläutert. Beide Fälle beschreiben dem Autor zufolge einen Insider. Neumann stellt fest, dass es logische Insider geben kann, die sich physisch außerhalb, und physische Insider, die sich logisch außerhalb der Domäne befinden. Der Autor beschreibt damit Fälle, in denen Insider, die sich außerhalb der Domäne befinden, auf Ressourcen innerhalb der Domäne zugreifen, beispielsweise über einen Fernwartungszugang. Gleichermaßen beschreibt er Fälle, in denen Insider, die sich innerhalb der Domäne befinden, auf Ressourcen außerhalb der Domäne zugreifen. Beide Fälle beleuchten nicht den Unterschied zwischen einem Insider und seiner (logischen oder physischen) Assoziation zur Innenseite oder Außenseite einer Domäne, sondern vielmehr die Tatsache, dass es keinen Unterschied zwischen der physischen und der logischen Verortung eines Insiders relativ zur Domäne gibt. Das bestätigt den Punkt, dass ein Individuum nicht zu einem Outsider wird und damit all seine Eigenschaften als Insider verliert, nur weil er (in diesem Fall logisch) außerhalb der Domäne agiert.

2.2.4 Insider versus Outsider

Jedes Individuum, welches nicht der Definition eines Insiders genügt, muss als ein *Nicht-Insider* aufgefasst werden, der gemeinhin als *Outsider* bezeichnet wird. Eine solche Unterscheidung der beiden Begriffe ist allerdings weder binär noch intuitiv. Es existiert nachweislich eine Abstufung in der Unterscheidung zwischen einem Insider und einem Outsider, was in der Literatur als *Insidergrad* (engl. insiderness) bezeichnet wird [Bis+09]. Individuen erfüllen manchmal weder eindeutig die Definition eines Insiders, noch können sie eindeutig als Outsider bezeichnet werden. Ebenso gibt es Fälle, in denen Individuen die Definition eines Insiders erfüllen, aber auch gleichzeitig von einem gewissen Standpunkt aus als Outsider gesehen werden können. Ein gutes Beispiel ist der Fall des in Abschnitt 2.2.3 bereits erwähnten Maskierers, wie auch das Committee on Information Systems Trustworthiness [Com98] feststellt: „It is equally unclear whether a

traditional spy or saboteur, operating form the inside at the behest of an outside organization, is an *insider*, an *outsider*, or yet a third class of entity."

Bishop u. a. [Bis+10] haben dieses Insiderdefinitionsproblem bereits ansatzweise beobachtet. Allerdings versäumen die Autoren die Betrachtung der Ursachen und der Implikationen sowie die Erarbeitung von Lösungsvorschlägen. Darüber hinaus verfallen die Autoren am Ende doch wieder der binären Ansicht der beiden Begriffe: „Our theme is that the distinction between *insider* and *outsider* is not binary; rather, there are *attackers* [...]. One can call some set of these attackers *insiders*, with the complement being the *outsiders*."

2.2.5 Uneinheitliche Charakterisierungen eines Insiders

Jede bisherige Insiderdefinition fokussiert sich auf eine oder eine ausgewählte kleine Menge von Charakteristiken, die für die jeweilige Definition wichtig erscheint. Beispiele sind Zugriff auf eine Domäne, Wissen über eine Domäne oder Vertrauen, das von Autoritäten der Domäne entgegengebracht wird. In der Realität repräsentiert diese Auswahl die jeweilige Intuition des Autors der Definition oder sie wurde für den Kontext passend ausgewählt, in dem sie benötigt wird. Als ein Resultat dieser Mannigfaltigkeit in der Gewichtung verschiedenster Charakteristiken divergieren Insiderdefinitionen für verschiedene Domänen und für verschiedene Kontexte sehr stark, wie Abschnitt 2.5.3 und Anhang A.2 aufzeigen. Dies macht Definitionen und davon abhängige Forschungsfragen sowie -ergebnisse unvergleichbar.

Abhängig von der Domäne und dessen Kontext sind manche Charakteristiken und Attribute wichtiger als andere, insbesondere wenn es um das übergeordnete Ziel der Einschätzung und Schadensminderung, der Erkennung sowie der Abwehr einer Insiderbedrohung geht. Diese verschiedenen Gewichtungen unterschiedlicher Charakteristiken sind tatsächlich legitim für verschiedene Domänen und deren Kontexte, allerdings führen sie zu einer Vielzahl von konkurrierenden Insiderdefinitionen und vor allem zu inkonsistenten Begrifflichkeiten.

Bishop u. a. [Bis+10] sowie Hunker und Probst [HP11] bearbeiteten diese Problematik bereits und beide Arbeiten kommen zu dem Schluss, dass eine Insiderdefinition immer speziell für eine Evaluationsdomäne gilt und dass bis dato keine Definition universell akzeptiert wurde.

2.2.6 Statische Definitionen eines Insiders

Die Definition eines Insiders ermöglicht die Erarbeitung einer konsistenten und vergleichbaren Auffassung von Insideraspekten für eine Domäne, etwa Nutzen oder Bedrohungsrisiken. Abstrakte und statische Definitionen von Insidern sind allerdings kein passendes Mittel für eine solche Art der Beurteilung, denn spezielle Aspekte oder Charakteristiken eines Insiders haben für die eine Domäne möglicherweise eine größere Bedeutung als für eine andere. Statische Definitionen von Insidern decken dabei nur einen oder sehr wenige Aspekte spezifischer Domänen ab. In anderen Domänen sind sie möglicherweise fehl am Platz.

2.2.7 Insider versus Insiderbedrohung

In der Literatur wird häufig die Definition eines Insiders mit der Definition einer Insiderbedrohung gleichgesetzt [And80; Woo00; MF01; Pat03; MT04; Bis+09]. So definieren Bishop u. a. [Bis+09] beispielsweise einen Insider

> „with regard to two primitive actions: 1. violation of a security policy using legitimate access, and 2. violation of an access control policy by obtaining unauthorized access."

Damit stellt sich allerdings die Frage, was mit Personen ist, die ihren legitimen Zugriff nicht zur Verletzung einer Sicherheitsvorgabe einsetzen oder keine Verletzung einer Zugriffskontrollrichtlinie durch die Erlangung eines unautorisierten Zugriffs hervorrufen.

Viele der dabei vorgeschlagenen Charakteristiken eines Insiders können ausschließlich für die Charakterisierung eines Angreifers oder eines Angriffs verwendet werden. So zum Beispiel die Taktik eines Insiders, die von Wood [Woo00] vorgeschlagen wird, oder die Konsequenzen einer Insiderbedrohung, auf die Magklaras und Furnell [MF01] Bezug nehmen. Dieser Unterschied zur hier forcierten Charakterisierung ist wichtig, denn die Charakterisierung eines Insiders ist zunächst erst einmal unabhängig von der eines Angreifers und kann demzufolge auch von Annahmen über tatsächliche Angriffe getrennt werden.[2] Auf der einen Seite vereinfacht dies die Situation, denn Charakteristiken von Angriffe sind sehr vielfältig und eine detaillierte Charakterisierung von Insiderangriffen beziehungsweise -bedrohungen rufen ganz eigene Schwierigkeiten hervor (s. Abschnitt 3.2.1.2). Auf der anderen Seite spiegelt dies die Tatsache wider, dass ein Insider möglicherweise niemals zu einer konkreten Insiderbedrohung wird.

2.3 Insidercharakteristiken

In der Literatur wurde für die Definition eines Insiders eine Vielzahl von Charakteristiken vorgeschlagen. Allerdings sind nicht alle Charakteristiken für jeden Kontext nützlich (vgl. Abschnitt 2.2.5) oder für die Beschreibung eines Insiders überhaupt geeignet. In diesem Abschnitt werden Aspekte herausgearbeitet, die für die Definition eines Insiders uneingeschränkt wichtig sind. In Abhängigkeit der Domäne, deren Kontext und dem Fokus der definierenden Partei können die folgenden Charakteristiken ausgewählt und unterschiedlich gewichtet werden. Die Liste erhebt keinen Anspruch auf Vollständigkeit, sondern dient vielmehr als Basis für die Beschreibung eines Insiders. Weitere Charakteristiken haben keinen Einfluss auf den *Insidergrad* einer Person, sondern visieren andere Aspekte an, wie zum Beispiel die Stärke eines Insiderangreifers (s. Abschnitt 3.2.1.1). Die Einführung dieser Charakteristiken in einem weiter gefassten Sinn als Variablen mit der Möglichkeit, diese durch spezifische, qualifizierte Werte zu instanziieren, geht über bisherige wissenschaftliche Ansätze hinaus. Sie ermöglicht eine feingranulare Charakterisierung im Gegensatz zu Definitionen, die eine simple und statische Auswahl dieser Charakteristiken darstellen.

Für eine qualifizierte Beschreibung des *Insideraspekts* beziehungsweise des *Insidergrades* (vgl. Abschnitt 2.2.4) eines Individuums in Bezug zu einer spezifischen Domäne können Charakteristiken herangezogen werden, die dem Individuum von dieser Domäne bereitgestellt wurden. Dabei handelt es sich um Charakteristiken, die nur durch die Domäne selbst beziehungsweise

2. Schlussfolgerungen über tatsächliche Angriffe können natürlich dennoch von einem Insidermodell abgeleitet werden (s. Abschnitt 3.2.1).

durch Autoritäten der Domäne absichtlich oder irrtümlich bereitgestellt, gewährt oder kontrolliert werden können.[3]

- **Credentials (C)**, manchmal auch als **Authentication** bezeichnet, definiert eine Zugehörigkeitsrelation, indem die Legitimität für den Besitz eines oder mehrerer Token beschrieben wird, die für die Authentisierung an einer Domäne benutzt werden können. Im Kontext spezieller Domänen, zu der die *Credentials* gehören, sind auch die Bezeichnungen Zutritts- oder Zugangsdaten gebräuchlich. *Credentials* können legitim besessen werden, sofern sie dem Individuum durch die Domäne bewusst gewährt wurden. Sie können allerdings auch unrechtmäßig besessen werden, entweder durch den Erwerb mithilfe eines vorangegangenen Angriffs oder durch die irrtümliche Bereitstellung. Zuletzt kann ein Individuum auch keine *Credentials* besitzen, genau dann wenn es für das Individuum keinen Weg der erfolgreichen Authentisierung gibt.[4] Kurz gesagt kann der konkrete Wert dieser Charakteristik eines der Elemente *no*, *stolen* oder *legit* annehmen.
- **Knowledge (K)** bezieht sich auf die Einblicke, die ein Individuum in eine Domäne hat. Es beschreibt jenes Wissen, das ausschließlich von der Domäne bereitgestellt werden kann. Beispiele sind Wissen über Wirtschaftsgüter, die interne IT-Infrastruktur oder die interne Kommunikation. Der konkrete Grad der Charakteristik *Knowledge*, den ein Individuum besitzt, kann zumeist nicht exakt spezifiziert oder direkt evaluiert werden. Demnach bietet es sich an, eine Abschätzung oder Approximation anhand der diskreten Werte *negligible*, *low*, *middle* oder *high* vorzunehmen. Weiterhin ist die Einschätzung, wobei es sich um viel beziehungsweise wenig Insiderwissen handelt, höchst domänenspezifisch. Beispielsweise kann es in einer bestimmten Domäne einen großen Unterschied zwischen dem Wissen über viele aber unwichtige Dinge auf der einen Seite und dem Wissen über wenige aber sehr wertvolle Dinge auf der anderen Seite geben. Dieses sogenannte Insiderwissen darf nicht verwechselt werden mit spezifischen Ausprägungen der Charakteristik *Credentials*, etwa Passwörter, die teilweise ebenfalls als *Wissen* bezeichnet werden.
- **Privileges (P)**, manchmal auch als **Access** oder **Authorisation** bezeichnet, bezieht sich auf den Grad an durchsetzbaren und verifizierbaren Rechten an erlaubten Interaktionen mit Ressourcen einer Domäne. Diese sind oftmals direkt verknüpft mit der Charakteristik *Credentials*, denn der Prozess der Authentifizierung via Zutritts- oder Zugangsdaten stellt dem authentifizierten Individuum normalerweise auch eine Menge von Zugriffsrechten zur Verfügung. Mit anderen Worten qualifiziert diese Charakteristik die Zugehörigkeitsrelation, die zwar nicht immer, aber in der Regel durch *Credentials* definiert wird. Beispiele von *Privileges* sind Lese-, Schreib-, Lösch- und Ausführungsrechte in einem Dateisystem, das Recht auf Wirtschaftsgüter zuzugreifen, sie zu nutzen oder sie zu transportieren, oder die Wartung einer Domäne über eine dedizierte Schnittstelle zu administrieren. Der konkrete Grad von *Privileges*, der einem Individuum zugeteilt wurde, ist bestimmt durch die Tatsache, dass die konkreten Zugriffsrechte bei mindestens einer Autorität oder einem Mechanismus der Domäne explizit bekannt sein müssen, um sie im Laufe der Autorisierung und auch danach zu bewilligen, durchzusetzen und zu überprüfen. Eine qualifizierende Zuweisung kann anhand der diskreten Werte *negligible*, *low*, *middle* oder *high* erfolgen. Der Grad dieser Charakteristik hängt sehr stark von den Objekten und ihrem Wert für die Domäne ab, mit denen dem Individuum erlaubt wurde zu interagieren.

3. Der Kürze halber und wenn nicht anders angegeben, wird nachfolgend der Begriff *Domäne* als Synonym für *Autoritäten der Domäne* verstanden. Vergleiche dazu auch Abschnitt 1.5.2.
4. Hierzu zählt auch der Besitz von falschen oder ungültigen *Credentials*.

Der wesentliche Unterschied zwischen *Credentials*, die eine Zugehörigkeitsrelation zu einer Domäne definieren, und *Privileges*, die diese Zugehörigkeitsrelation zur Domäne genauer beschreiben beziehungsweise qualifizieren, wird definiert durch den Blickwinkel einer Domäne. Mit einem Unternehmen als Domäne stellt beispielsweise eine gültige Zutrittskarte die legitimen *Credentials* des Insiders dar. Mit dieser Zutrittskarte identifiziert er sich gegenüber der Domäne und erlaubt die Prüfung seiner legitimen Zugehörigkeit. Zusätzlich vorhandene Zugangsdaten, etwa in Form von Schlüsseln zu Räumen oder von Benutzernamen und Passwort für die Anmeldung an einem Rechner, sind dann Teil der *Privileges* des Insiders. Die Verwendung von Zugangsdaten anderer Insider, um sich an einem Rechner anzumelden, ändert die Identität gegenüber der Domäne nicht. Er identifiziert sich weiterhin mit seiner Zutrittskarte. Sofern allerdings der Rechner als Domäne definiert wird, werden der Benutzername und das Passwort zu den *Credentials* des Insiders und dessen *Privileges* werden durch den Zugriffskontrollmechanismus des Rechners festgelegt. Die Verwendung von Zugangsdaten anderer Insider ändert gleichzeitig die Identität des Insiders gegenüber der Domäne. Aus diesem Beispiel wird deutlich, dass *Credentials* und *Privileges* unabhängig voneinander betrachtet werden müssen.

- **Trust (T)**, oder präziser **Individual Trust**, beschreibt den Grad an Treu und Glauben, den eine Domäne in die Integrität und Ehrlichkeit eines Individuums legt. Formeller ausgedrückt kann *Trust* beschrieben werden durch die Wahrscheinlichkeit, mit der ein Mitglied der Domäne bereit ist, bei der Interaktion mit diesem Individuum von formalen Richtlinien und Protokollen abzuweichen. Der konkrete Grad dieser Charakteristik hängt maßgeblich vom persönlichen Verhältnis zwischen dem Individuum, dem sie zugeordnet wird, und den Mitgliedern der Domäne ab. Mögliche diskrete Werte sind *negligible*, *low*, *middle* oder *high*. Bei der Abschätzung müssen ebenfalls Eigenschaften in Betracht gezogen werden, die allein schon aufgrund eines Status, einer Stellung oder der vorzuweisenden Erfahrung einen gewissen Grad an *Trust* hervorrufen, wie zum Beispiel Seniorität.
- **Uncertainty (U)**, oder anders ausgedrückt **Structural Trust**, wird definiert als die absichtliche oder strukturelle Abwesenheit von Überwachungs-, Prüf- oder Kontrollmechanismen. Mit einem hohen Maß an *Uncertainty* kann ein Individuum beispielsweise Aktionen ausführen, ohne dass diese während der Ausführung überprüft oder im Nachhinein nachvollzogen werden können. Allgemein gesagt lässt sich *Uncertainty* als der Grad der Erlaubnis oder Toleranz von Aktionen eines Individuums in Abwesenheit von Sicherheits- und Überprüfungsmechanismen bemessen. Der konkrete Wert dieser Charakteristik ist sehr domänenspezifisch und kann mit *negligible*, *low*, *middle* oder *high* ausgedrückt werden.

2.4 Eindeutig unterscheidbare Typen von Insidern

Verschiedene Instanzen von Insidercharakteristiken können zu spezifischen Insidertypen zusammengefasst und kategorisiert werden. Einige dieser Typen wurden in der Literatur bereits erwähnt (vgl. Abschnitt 2.1.2). Sie wurden jedoch bisher weder von spezifischen Charakteristiken abgeleitet oder als dedizierte Insidertypen eingeführt, noch wurden sie miteinander assoziiert oder gegeneinander abgegrenzt. Der hier vorgestellte neue Ansatz bereitet den Weg einer systematischen Charakterisierung eines Insiders und der Ableitung von Abhängigkeiten zwischen den verschiedenen Insidertypen (s. Abschnitt 2.5 und Abschnitt 2.6). Darüber hinaus können gemeinsame Bedrohungsaspekte identifiziert werden, die einzigartig von den gleichen Insidertypen geteilt werden und die sich zwischen verschiedenen Insidertypen unterscheiden

(s. Kapitel 3). Diese Betrachtungen bilden gemeinsam die Grundlage für die Entwicklung spezifischer Gegenmaßnahmen von Insiderbedrohungen.

2.4.1 Basis-Insider- und Outsidertypen

Die von einer Domäne bereitgestellten Insidercharakteristiken, die in Abschnitt 2.3 aufgezeigt und erläutert wurden, machen einen Unterschied in der Abgrenzung zwischen einem Insider und einem Outsider. Weiterhin beschreiben sie, wie sehr ein Individuum in einer Domäne eingebettet ist, was demnach den *Insidergrad* des Individuums begründet. Die nachfolgend beschriebenen Basis-Insidertypen basieren dementsprechend ausschließlich auf diesen Charakteristiken. Eine wichtige Beobachtung dabei ist, dass jede dieser Charakteristiken jeweils einen Insider und einen Outsider in einer eigenen unabhängigen Dimension definiert. Eine Veranschaulichung dessen gibt die Abbildung 2.1 auf Seite 29. Die Missachtung dieser Tatsache sowie eine beliebige Vermischung der Dimensionen in existierenden Insiderdefinitionen hat zu den Insiderdefinitionsproblemen „Insider versus Innenseite“ (vgl. Abschnitt 2.2.3) und „Insider versus Outsider“ (vgl. Abschnitt 2.2.4) geführt.

Einer der Insideraspekte ist die von einer Domäne bereitgestellte Ausprägung von *Knowledge*. Ein Individuum mit *low*, *middle* oder *high Knowledge* in Bezug auf die Einblicke in eine Domäne kann als ein Insider angesehen werden, wohingegen ein Individuum mit *negligible Knowledge* als ein Outsider angesehen werden kann. Diese Unterscheidung kann unabhängig vom *Insidergrad* gemacht werden, der durch andere Charakteristiken qualifiziert wird, zum Beispiel unabhängig davon, ob ein Individuum mit der Innenseite einer Domäne durch den Besitz gültiger *Credentials* assoziiert werden kann oder nicht. Aus diesem Grund, und um terminologische Verwechslungen zu vermeiden, werden die Basis-Insider- beziehungsweise Outsidertypen **Knowledge-Insider (K-Insider)** und **Knowledge-Outsider (K-Outsider)** eingeführt.

Auf die gleiche Art und Weise, und unabhängig von der Beschreibung eines Individuums als K-Insider oder K-Outsider, kann das Individuum als eine andere Art von Insider aufgefasst werden, wenn es im Besitz von gültigen *Credentials* ist. Ebenso kann das Individuum als eine andere Art von Outsider beschrieben werden, wenn keine gültigen *Credentials* vorhanden sind. Diese Dimension beschreibt ein Individuum im Hinblick auf seine örtliche Relation zu einer Domäne, unabhängig von eventuell vorhandenem Insiderwissen. Diese Individuen werden als **Credentials-Insider (C-Insider)** und **Credentials-Outsider (C-Outsider)** bezeichnet. C-Insider können weiterhin in solche Insider unterteilt werden, die diese *Credentials* legitim besitzen und jene, die diese *Credentials* von einem anderen C-Insider gestohlen oder unerlaubt erhalten haben. Erstere werden **Credentials-Associate (C_A-Insider)** und Letztere werden **Credentials-Masquerader (C_M-Insider)** genannt.

Die gleichen Beobachtungen gelten für die Basis-Insidertypen **Privileges-Insider (P-Insider)**, **Trust-Insider (T-Insider)** und **Uncertainty-Insider (U-Insider)**, jeweils mit den konkreten zugehörigen Insidercharakteristikausprägungen *low*, *middle* oder *high* sowie den zugehörigen Basis-Outsidertypen, jeweils mit der zugehörigen Insidercharakteristikausprägung *negligible*.

2.4.2 Kombinationen von Insidertypen

Anderson [And80] führt in seiner Arbeit vier verschiedene Arten von Insidern ein und nennt sie „External Penetrator“, „Masquerader“, „Legitimate User“ und „Clandestine User“, um

Tabelle 2.1: Präzise Beschreibung unterschiedlicher oder identischer Insidernamen aus der Literatur mithilfe der vorgestellten Insider- und Outsidertypen in Verbindung mit der universellen Namenskonvention. Für nähere Details s. Abschnitt 2.5.3 und Anhang A.2.

Referenz	unpräziser Insidername	präziser Insider-/Outsidertyp
Anderson [And80]	External Penetrator	C-Outsider
	Masquerader	C_M-P-Insider
	Legitimate User	C_A-P-Insider
	Clandestine User	C_A-P-U-Insider
Lundin und Jonsson [LJ00]	Masquerader	C_M-Insider
Bowen u. a. [Bow+10]	Masquerader	C_M-K-Insider
	Traitor	C_A-K-Insider
Hunker und Probst [HP11]	Real Real Insider	C_A-P-T-Insider
Kaplan u. a. [Kap+15]	Errant Insider	K-Outsider

sie vom inflationär benutzten Begriff *Insider* zu unterscheiden. Der Autor gründet die neuen Erklärungen der verschiedenen Arten hauptsächlich auf die beiden Charakteristiken *Credentials* und *Privileges*. Unglücklicherweise definieren auch Hunker und Probst [HP11] einen Insider hauptsächlich anhand dieser Charakteristiken, nennen ihn aber „Real Real Insider“. Darüber hinaus wurde speziell der eingängige Begriff „Masquerader“ von anderen Autoren verwendet und von einigen Charakteristiken entledigt, zum Beispiel von Lundin und Jonsson [LJ00] oder Bowen u. a. [Bow+10]. Andere Begriffe, die in der Literatur einen Insider beschreiben sollen, sind zum Beispiel „Traitor“ [Bow+10] und „Errant Insider“ [Kap+15], was zu einer Vermischung von Bezeichnungen für unterschiedliche oder sogar ähnliche Arten von Insidern führt. Diese Vermischung in Verbindung mit den Beobachtungen, dass ein Individuum immer eine Kombination aus den in Abschnitt 2.4.1 eingeführten Basis-Insider- und Outsidertypen darstellt und all diese Typen aufgrund ihrer Unabhängigkeit in jeder Weise miteinander kombinierbar sind, führt zu der Erkenntnis, dass alle bisherigen Ansätze, Begrifflichkeiten für Arten von Insidern zu etablieren, nicht skalieren. Stattdessen braucht es eine universelle Namenskonvention, die auf den erwähnten Basis-Insider- und Outsidertypen gründet. Um lange Namensketten für spezifische Kombinationen der Typen zu vermeiden, bei denen einfach die jeweiligen Namen hintereinander zusammengesetzt werden, aber um gleichzeitig die Eindeutigkeit einer solchen Insiderbenennung zu erhalten, wird die folgende Namenskonvention vorgeschlagen.

2.4.2.1 Universelle Namenskonvention

Jeder Basis-Insider- und Outsidertyp wurde mit dem ersten Buchstaben der zugrunde liegenden korrespondierenden Insider-Charakteristik abgekürzt. Mit diesen Abkürzungen und der Festlegung darauf, dass zuerst die Liste der kombinierten Insider-Abkürzungen in ihrer alphabetischen Reihenfolge, gefolgt von der Liste der kombinierten Outsider-Abkürzungen in ihrer alphabetischen Reihenfolge aufgeführt werden, erhält jede Kombination der Basis-Insider- und Outsidertypen eine vollständig definierte und eindeutige Bezeichnung. Beispielsweise würde ein Individuum, welches ein *Knowledge-Insider*, ein *Credentials-Outsider* und ein *Trust-Insider* ist,

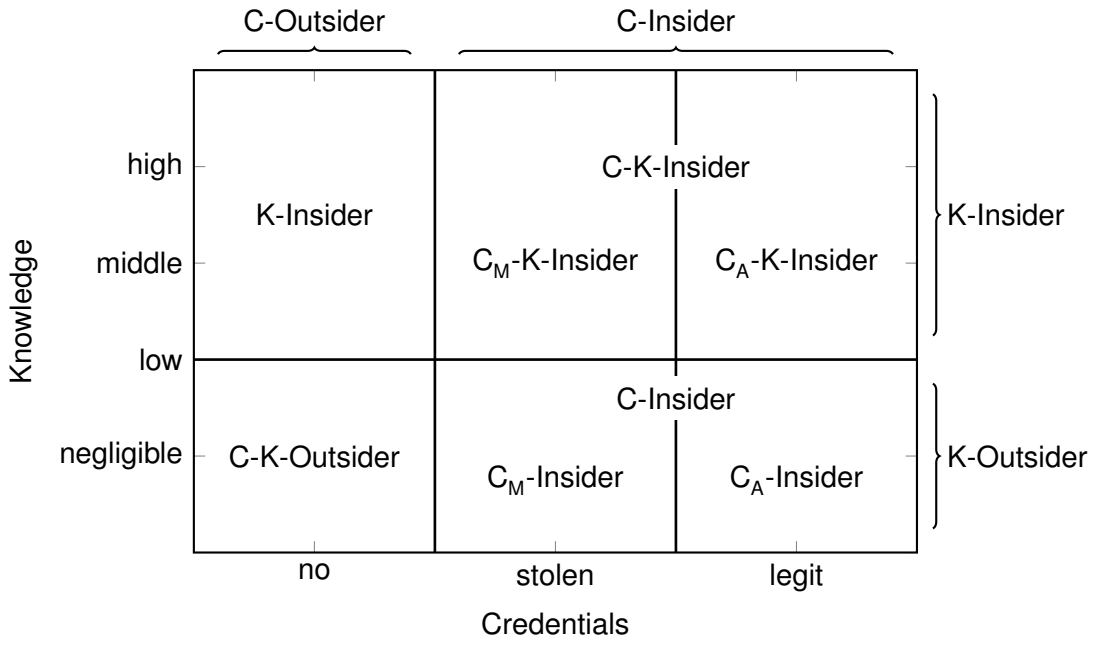


Abbildung 2.1: Die Insidertypen, die durch die Kombination der unterschiedlichen Ausprägungen von *Credentials* und *Knowledge* entstehen

als *K-T-Insider-C-Outsider* benannt werden. Mit der Annahme, dass nicht-aufgeführte Insidertypen den jeweiligen Outsidertypen gleichen, kann die Liste der anhängenden Outsidertypen weggelassen werden, sofern deren explizite Erwähnung nicht relevant ist. Das erlaubt eine kompaktere Bezeichnung von kombinierten Insidertypen. Auf diese Weise kann das Individuum im erwähnten Beispiel als *K-T-Insider* bezeichnet werden.

Mit dieser Methodik lassen sich die Versuche von Benennungen verschiedener Insidertypen in der bestehenden Literatur nun präzise und ausdrucksstark beschreiben. Tabelle 2.1 zeigt die Entsprechungen der neuen Namenskonvention zu den bereits erwähnten Insidernamen. Mehr Details dazu, insbesondere die Definitionstexte und die angenommenen Insiderdomänen der beispielhaft erwähnten Insidernamen, sind in Abschnitt 2.5.3 und Anhang A.2 zu finden.

2.4.2.2 Beispielkombinationen von Knowledge und Credentials Typen

Die Basis-Insidertypen *K-Insider* und *C-Insider*, die von den Charakteristiken *Knowledge* und *Credentials* abgeleitet wurden, sind möglicherweise die intuitivsten und in der Literatur sehr weitverbreiteten Typen, wenn es um Insider geht. Die Kombination dieser beiden Insidertypen, illustriert in Abbildung 2.1, bringt Insidertypen hervor, die Szenarien von hoher praktischer Relevanz einschließen. Sie waren in der Vergangenheit immer wieder schwer zu fassen und wurden kontrovers bisher diskutiert. Aus diesem Grund werden deren Kombination als Beispiele der neuen Namenskonvention für Insidertypen genauer erläutert.

Der *C-K-Outsider* repräsentiert, was intuitiv als Nicht-Insider oder Outsider bezeichnet wird. Es beschreibt ein Individuum, welches keinerlei oder nur sehr wenig Einblicke in eine Domäne und ebenso keine Assoziation zur Innenseite dieser Domäne hat. Es beschreibt also ein Individuum, welches in keinen der Insidertypen passt. Der *C-K-Insider* beinhaltet, was statisch beispielsweise von Einwechter [Ein02] und Mathew u. a. [Mat+10] als Insider beschrieben wird. Es bezeichnet

ein Individuum, welches Insiderwissen über eine Domäne hat und zusätzlich unmittelbar mit der Innenseite dieser Domäne assoziiert ist. Diese beiden Typen decken sich mit der generellen Argumentationslinie und der zugrunde liegenden Intuition, die in der existierenden Literatur zum Thema Insider angenommen wird.

Der *C-Insider-K-Outsider* beschreibt ein Individuum, welches sich erfolgreich gegenüber einer Domäne authentisieren und somit innerhalb des Perimeters dieser Domäne agieren kann, aber darüber hinaus keine weiteren oder nur sehr niedrig eingestufte Insidercharakteristiken hat. Insbesondere hat es keine oder nur sehr wenige Einblicke in die Domäne. Ein Beispiel ist ein *C-K-Outsider*, der erfolgreich die Zutritts- oder Zugangsdaten eines *C-Insider* errät. Das Problem, den *C-Insider-K-Outsider* als Insider oder Outsider zu attribuieren, ist seit Jahrzehnten ungelöst und kann nun durch diese Unterscheidung zwischen Insidertypen gelöst werden. Das Beispiel zeigt, dass jeder Outsider mit gültigen *Credentials*, der dadurch Aktionen auf der Innenseite einer Domäne durchführen kann, nicht mehr als ein Outsider anzusehen ist, sondern nun als ein spezieller Insidertyp zu verstehen ist, der darüber hinaus von einem anderweitigen Insider mit gültigen *Credentials* unterschieden werden kann.

Ebenso repräsentiert der *K-Insider-C-Outsider* diskursive und umstrittene Ausnahmefälle. Dieser Insidertyp illustriert ein Individuum ohne Assoziation zur Innenseite einer Domäne, aber mit einem gewissen Grad an Insiderwissen darüber. Beispiele sind Ex-Angestellte eines Unternehmens, aber auch Angestellte eines Unternehmens, die an Wissen über Interna eines Geschäftsbereichs gelangt sind, ohne keine gültige Zugehörigkeitsrelation zu diesem Geschäftsbereich. Ebenso kann es sich um eine Person handeln, die ursprünglich eine Domäne designt hat, nun aber in keiner Weise mehr mit der Domäne assoziiert ist.

Der *C-Insider*-Typ kann zusätzlich aufgeschlüsselt werden in diejenigen Individuen, die ihre *Credentials* auf legitime Art und Weise erhalten haben und diejenigen, die ihre *Credentials* unerlaubterweise besitzen (vgl. Abbildung 2.1). Die hier als C_A*-Insider-K-Outsider* bezeichneten Fälle sind von einem C_A*-K-Insider* nur durch ihren niedrigeren Grad an Insiderwissen über eine Domäne unterscheidbar. Ein gutes Beispiel für einen solchen C_A*-Insider-K-Outsider* ist ein neuer Mitarbeiter, der noch keine Freigabe für die Kenntnis von wichtigen internen Informationen hat, der sich über interne Konventionen noch nicht bewusst ist oder der noch kein Wissen über Sicherheitsrichtlinien und vorhandene interne Überwachungsmechanismen hat. Der C_M*-Insider-K-Outsider* ist ein *K-Outsider*, der nur an gültige Zutritts- oder Zugangsdaten gekommen ist, die von der Domäne ungewollt an ihn ausgegeben wurden. Sobald dieser auch an Insiderwissen über die Domäne gelangt oder das Insiderwissen schon vorher vorhanden ist, kann man von einem C_M*-K-Insider* sprechen.

2.5 Modellierung eines Insiders

Für die Lösung der in den Abschnitten 2.2.2, 2.2.4 und 2.2.6 beschriebenen Insider-Definitionsproblemen wird der Prozess der *Insidermodellierung* eingeführt. Durch die Probleme wurde deutlich, dass sich Domänen und ihr Kontext im Laufe der Zeit stark verändern können. Weiterhin gibt es keine binäre Unterscheidung zwischen einem Insider und einem Outsider, sondern einen gewissen Insidergrad einer Person. Und schließlich sind statische Definitionen von Insidern kein adäquates Mittel, um die unterschiedlichen Stärken und Schwächen eines Insiders für unterschiedliche Domänen zu erfassen. Die Modellierung von Insidern kann einerseits für die Erzeugung von Insiderdefinitionen im Kontext einer spezifischen Domäne herangezogen werden

und andererseits dazu dienen, ein Individuum mit dem Ziel der Festlegung seines Insidertyps und der Qualifizierung seines *Insidergrades* zu charakterisieren.

Definition 2.1. Ein Insidermodell beschreibt den vorhandenen Insidergrad einer Person in einer speziellen Domäne. Es besteht aus den folgenden Teilen:

1. Eine explizite Benennung der **Domäne**.
2. Eine **Menge von Insidercharakteristiken** (vgl. Abschnitt 2.3), die im Kontext einer Domäne wichtig sind.
3. Eine domänenspezifische **Semantik der Charakteristikausprägungen** für jede der in Punkt 2 identifizierten Insidercharakteristik. Eine solche Semantik kann vernachlässigt werden, sofern sie nicht von der intuitiven Bedeutung abweicht.[5]
4. Eine **Instanziierung** aller Charakteristiken der Menge aus Punkt 2, mit der die konkrete Insidercharakterisierung eines Individuums festgelegt wird.

Die Teile 2 und 3 der Insidermodellierung liefern eine Spezifizierung der Domäne aus Teil 1 und setzen den Kontext, in dem die spezifische Charakterisierung eines Individuums in Teil 4 zu verstehen ist. Teil 3 in Verbindung mit Teil 4 erlaubt die Ableitung spezifischer Insidertypen (vgl. Abschnitt 2.4), die dem entwickelten Insidermodell zugrunde liegen. Nachfolgend werden Beispiele für die Notation von Insidermodellen gegeben. Im Anschluss daran erfolgt die Beschreibung einer formalen Methodik zur Analyse von Insiderszenarien und -definitionen, die abschließend auf eine Menge von Insiderdefinitionen aus der wissenschaftlichen Literatur angewandt wird.

2.5.1 Beispiele von Insidermodellen

Die nachfolgenden Ausführungen vermitteln Beispiele von Insidermodellen anhand verschiedener Szenarien. Der Fokus liegt dabei weiterhin auf den Insidercharakteristiken und nicht auf Bedrohungs- beziehungsweise Angreiferaspekten, auch wenn es sich jeweils um Insiderangriffe handelt.

Beispiel 2.1 (Angestellter)**.** Ein Angestellter gerät in persönliche finanzielle Probleme und versucht, privilegierten Zugriff zu einem Subsystem seiner Organisation zu bekommen, in dem sich hochgradig wertvolle Wirtschaftsgüter befinden, um diese Güter zu stehlen und anschließend zu verkaufen. Für den Erfolg des Vorhabens und die Verdeckung der böswilligen Aktivitäten muss der Angestellte den Grad seiner *Knowledge* sowie *Privileges* erhöhen, um Sicherheits- und Überwachungsmechanismen zu deaktivieren.

Angenommen die gesamte Organisation stellt die Domäne dar und der Angestellte ist das Individuum, das charakterisiert werden soll, dann würde die Menge der wichtigen Charakteristiken eines Insidermodells *Knowledge*, *Credentials*, *Privileges* und *Uncertainty* enthalten. Das Insidermodell dieses Beispiels kann wie folgt zusammengefasst werden:

5. Dazu gehört beispielsweise die Semantik der Charakteristikausprägungen *no*, *stolen* und *legit* der Charakteristik *Credentials*, die offensichtlich und eindeutig ist.

Domäne: Die gesamte Organisation
Insidercharakteristiken: • Credentials: legit → C_A-Insider • Knowledge: low → K-Insider • Privileges: low → P-Insider • Uncertainty: negligible → U-Outsider
Insidertyp: C_A-K-P-Insider

Das Individuum ist ein Angestellter, was bedeutet, dass er berechtigterweise im Besitz von gültigen Zutritts- oder Zugangsdaten ist, um innerhalb des Perimeters der Domäne agieren zu können. Demnach handelt es sich um den Fall eines *C_A-Insiders*. Das Level an *Knowledge* reicht nicht aus, um über vorhandene Sicherheits- und Überwachungsmechanismen Bescheid zu wissen. Andernfalls müsste der Betreffende das Level nicht erhöhen. Als Angestellter hat das Individuum allerdings sehr wahrscheinlich mindestens einen begrenzten Grad an Einsicht in die Domäne, um den Wert und den Standort der Wirtschaftsgüter zu kennen. Es handelt sich demnach um den Fall eines *K-Insiders*. Das Level an *Privileges* ist nicht ausreichend, um existierende Sicherheits- und Überwachungsmechanismen zu manipulieren oder zu umgehen. Andernfalls müsste er das Level nicht erhöhen. Aber als Angestellter mit gültigen Zutritts- oder Zugangsdaten hat das Individuum sehr wahrscheinlich wenigstens einen begrenzen Grad an *Privileges*. Es handelt sich demnach um den Fall eines *P-Insiders*. Der Fakt, dass Sicherheits- und Überwachungsmechanismen vom Individuum manipuliert oder umgangen werden müssen, setzt den Grad an *Uncertainty* auf *negligible*. Damit handelt es sich um den Fall eines *U-Outsiders*.

Mit der Änderung der Domäne dieses Beispiels zu demjenigen Subsystem, welches die wertvollen Wirtschaftsgüter beinhaltet, ändert sich das Insidermodell ein wenig. Die folgende Aufstellung fasst das neue Insidermodell zusammen:

Domäne: Das Subsystem mit den wertvollen Wirtschaftsgütern
Insidercharakteristiken: • Credentials: no → C-Outsider • Knowledge: middle → K-Insider • Privileges: negligible → P-Outsider • Uncertainty: negligible → U-Outsider
Insidertyp: K-Insider

Die Sicherheits- und Überwachungsmechanismen, die für die Entwendung der wertvollen Wirtschaftsgüter manipuliert oder umgangen werden müssen, beinhalten Zugriffskontrollmechanismen der Domäne. Demnach handelt es sich um den Fall eines *C-Outsiders*. Der Fakt, dass das Individuum von diesen Wirtschaftsgütern und deren Wert weiß und ebenso Kenntnisse darüber hat, wie man an diese Güter heran gelangt, setzt den Grad an *Knowledge* auf *middle* oder höher. Es handelt sich demnach um den Fall eines *K-Insiders*. Die durchsetzbaren und verifizierbaren Rechte der Interaktion mit der Domäne sind nicht vorhanden oder höchstens sehr beschränkt. Andernfalls müsste das Individuum nicht erst an einen privilegierten Zugriff auf die Domäne gelangen. Es handelt sich demnach um den Fall eines *P-Outsiders*. Die Sicherheits- und Überwachungsmechanismen scheinen nicht nur die Domäne zu schützen, sondern auch die wertvollen Wirtschaftsgüter innerhalb der Domäne. Demnach ist das Level der *Uncertainty* vernachlässigbar, was zum Fall eines *U-Outsiders* führt.

Beispiel 2.2 (Streifenpolizist)**.** Eine attraktive junge Frau erregt die Aufmerksamkeit eines Streifenpolizisten, als sie ihr korrekt geparktes Auto verlässt. Der Polizist missbraucht seine legalen Befugnisse und macht eine Halterabfrage dieses Autos anhand des Kfz-Kennzeichens, um an die Adresse und Kontaktinformationen der jungen Frau zu gelangen.

Ein sehr ähnlich gelagerter Fall ist im Oktober 2019 über Mitarbeiter des FBI bekannt geworden [Wei19; VT19]. Mit dem Polizeiinformationssystem als Domäne enthält die Menge der wichtigen Charakteristiken eines Insidermodells *Knowledge*, *Credentials* und *Privileges*. Das Insidermodell dieses Beispiels kann wie folgt zusammengefasst werden:

Domäne: Die Fahrzeughalterdatenbank
Insidercharakteristiken: • Credentials: legit → C_A-Insider • Knowledge: high → K-Insider • Privileges: middle → P-Insider • Uncertainty: middle → U-Insider
Insidertyp: C_A-K-P-U-Insider

Der Streifenpolizist besitzt auf legitime Art und Weise gültige *Credentials*, um innerhalb des Perimeters des Polizeiinformationssystems zu agieren. Demnach handelt es sich um den Fall eines C_A*-Insiders*. Der Grad an *Knowledge* muss hoch genug sein, um mit dem Polizeiinformationssystem arbeiten zu können. Es handelt sich somit um den Fall eines *K-Insiders*. Der Streifenpolizist ist ermächtigt, Anfragen an die Fahrzeughalterdatenbank innerhalb des Polizeiinformationssystems zu schicken. Das Level an *Privileges* ist also mindestens so hoch, dass es sich um den Fall eines *P-Insiders* handelt. Die Verletzung der Berechtigung geht aus der situativen Lage hervor, die kein Fehlverhalten der Parkerin beinhaltet, allerdings auf der Ebene der Domäne so nicht erkannt werden kann. Der Streifenpolizist ist in der Hinsicht also ein *U-Insider*.

Beispiel 2.3 (Krankenpfleger)**.** Ein Krankenhauspatient gerät plötzlich in einen kritischen Gesundheitszustand. Ein Krankenpfleger, der in diesem Moment seine Dienstschicht absolviert, erreicht keinen Arzt. Er entscheidet sich deshalb dazu, die ihm bekannten Zugangsdaten eines Arztes zu verwenden, um sich damit in das medizinische Patientendatensystem des Krankenhauses einzuloggen und sich Einblicke in die Patientenakte dieses Patienten zu verschaffen. Er findet dadurch die aktuelle Medikation des Patienten und kann somit eine schnelle und korrekte Entscheidung treffen. Er rettet dadurch das Leben des Patienten.

Eine Falluntersuchung ähnlich gelagerter Fälle von Koppel u. a. [Kop+08] zeigt auf, dass es sich hier nicht nur um ein rein konstruiertes Szenario handelt. Mit dem medizinischen Patientendatensystem des Krankenhauses als Domäne enthält die Menge der wichtigen Charakteristiken eines Insidermodells *Knowledge*, *Credentials* und *Privileges*. Das Insidermodell dieses Beispiels kann wie folgt zusammengefasst werden:

Domäne: Das medizinische Patientendatensystem
Insidercharakteristiken: • Credentials: stolen → C_M-Insider • Knowledge: middle → K-Insider • Privileges: middle → P-Insider • Uncertainty: negligible → U-Outsider
Insidertyp: C_M-K-P-Insider

Der Krankenpfleger kennt die Zugangsdaten eines der Ärzte, was effektiv bedeutet, dass er auf unerlaubte Weise in deren Besitz gekommen ist, selbst wenn sie ihm von diesem Arzt persönlich mitgeteilt wurden. Demnach handelt es sich um den Fall eines C_M*-Insiders*. Der Grad an *Knowledge* muss mindestens *middle* sein, um schnell und korrekt mit dem Patientendatensystem umgehen zu können. Demnach handelt es sich bei dem Krankenpfleger um den Fall eines *K-Insiders*. Mit den Zugangsdaten des Arztes hat der Krankenpfleger die Berechtigungen, um die Patientenakte zu öffnen und zu lesen. Das Level an *Privileges* ist also mindestens *middle*, was zu einem *P-Insider* führt. Ein solcher Vorfall wird in einem Krankenhaus registriert und dokumentiert, sodass spätestens bei Rückkehr des Arztes das Verhalten des Krankenpflegers aufgedeckt werden wird. Die *Uncertainty* ist somit *negligible* und es handelt sich bei dem Krankenpfleger in der gestohlenen Rolle als Arzt um einen *U-Outsider*.

2.5.2 Formale Methodik zur Analyse von Insiderszenarien und -definitionen

Für die Herausarbeitung von Insidermodellen aus Beschreibungen von Insidern, wird in diesem Abschnitt eine Vorgehensweise aufgezeigt, die auf der *Qualitativen Inhaltsanalyse* basiert, wie sie von Mayring und Fenzl [MF14] spezifiziert wird. Damit können einheitliche Modelle aus existierenden Insiderdefinitionen sowie aus Insiderszenarien abgeleitet werden.

Tabelle 2.2: Nummerische Ordnung der Charakteristikausprägungen der in Abschnitt 2.3 aufgeführten Insidercharakteristiken

	-1	0	1	1,5	2	2,5	3
Credentials (C)	no	not stated	–	–	stolen	stolen – legit	legit
Knowledge (K)	negligible	not stated	low	low – middle	middle	low – high	high
Privileges (P)	negligible	not stated	low	low – middle	middle	low – high	high
Trust (T)	negligible	not stated	low	low – middle	middle	low – high	high
Uncertainty (U)	negligible	not stated	low	low – middle	middle	low – high	high

Die *Qualitative Inhaltsanalyse* stellt eine regelgeleitete Analysemethode von Texten dar, die zunächst „qualitativ-interpretativ bleibt und so auch latente Sinngehalte erfassen kann“ [MF14]. Das Vorgehen besteht den Autoren zufolge aus zwei Schritten. Im ersten Schritt werden entweder induktiv an den vorliegenden Texten, oder aber theoriegeleitet-deduktiv, festgestellte Kategorien benannt und beschrieben. Im zweiten Schritt werden Textteile anhand eines Kodierleitfadens diesen Kategorien zugeordnet.

Die vorliegende Anwendung dieser Methodik richtet sich nach der speziellen Technik der *strukturierenden Inhaltsanalyse* [MF14, Abschnitt 38.3.3]. Bei ihr werden ordinal geordnete Kategorien vorab theoriegeleitet entwickelt und die Textteile anschließend anhand der Ordnung in das Kategoriensystem eingeordnet. Schritt 1 wurde in Abschnitt 2.3 bereits deduktiv durchgeführt und umfasst die Kategorie *Insidercharakteristiken* mit den konkret genannten Insidercharakteristiken als Subkategorien:

- ***Credentials* (C)**,
- ***Knowledge* (K)**,
- ***Privileges* (P)**,
- ***Trust* (T)** und
- ***Uncertainty* (U)**.

Die Ordnung innerhalb der Subkategorien ist in Tabelle 2.2 zusammengefasst. Sie ergibt sich aus den konkreten Ausprägungen beziehungsweise diskreten Werten der Insidercharakteristiken, zum Beispiel *negligible*, *low*, *middle* und *high* sowie den Zwischenwerten *not stated*, *low – middle* und *low – high*. Die Definitionen der Subkategorien sind in Abschnitt 2.3 zu finden. Der Kodierleitfaden für Schritt 2, der für die Herausarbeitung eines Insidermodells aus einer Beschreibung oder einem Szenario notwendig ist, ist in Tabelle 2.3 spezifiziert. Der minimale Textbestandteil, der einer Kategorie zugeordnet werden kann, auch als *Kodiereinheit* bezeichnet, bezieht sich auf die vorliegende Insiderbeschreibung. Die *Kontexteinheit*, also das Material, auf das für die jeweilige Kodierung beziehungsweise Interpretation zurückgegriffen werden darf, erstreckt sich auf das gesamte Werk, in dem die Beschreibung eingebettet ist.

Sobald einer Insiderbeschreibung mehr als ein nummerischer Wert für ein und dieselbe Charakteristik zugeordnet werden kann, ergeben sich mehrere Insidertypen für das aus der Kodierung hervorgehende Insidermodell. Damit können mehrere unterschiedliche Outsider- und Insidertypen, auch mehrere kombinierte Insidertypen, aus einer einzigen Insiderdefinition hervorgehen, wie exemplarisch anhand der Insiderdefinition von Butts, Mills und Baldwin [BMB05] in Anhang A.2.17 ersichtlich ist.

Tabelle 2.3: Kodierleitfaden für die Zuordnung von Textteilen verschiedener Insiderbeschreibungen zu den jeweiligen Ordnungen der erarbeiteten Insidercharakteristiken

Ordnung	Subkategorie	Kodierregel	Ankerbeispiel
-1	**C**, **K**, **P**, **T**, **U**	Die Subkategorie wird als nicht-vorhanden beschrieben.	C(-1), T(-1): „Outside affiliates are non-trusted outsiders who use open access to gain access to an organization's resources [and have no legitimate reason to access the building]“ [CR05]; T(-1): „Any individual who has been granted any level of trust in an information system.“ [BMB05].
0	**C**, **K**, **P**, **T**, **U**	Die Subkategorie wird nicht genannt und es lässt sich auch weder aus der Kodiereinheit noch aus der Kontexteinheit explizit darauf schließen.	K(0), T(0), U(0): „An insider can be an employee, student, or other member of a host institution that operates a computer system to which the insider has legitimate access [...]“ [Pfl08].
1	**K**, **P**, **T**, **U**	Die Insiderbeschreibung deutet auf ein eindeutig niedriges Level der Subkategorie hin.	K(1): Der Ex-Mitarbeiter wurde bereits in der Probezeit entlassen und hatte daher nur die Gelegenheit, Einblicke in unwichtige Vorgänge des Unternehmens zu erlangen.
1,5	**K**, **P**, **T**, **U**	Die Beschreibung der Subkategorie deutet auf eine Ordnung zwischen 1 und 2 hin.	K(1,5): „An insider is a database subject who has personal knowledge of information stored in one or more fields marked confidential.“ [Mat+10].
2	**C**	Die Beschreibung der Subkategorie deutet auf eine gültige und akzeptierte Zugehörigkeit zur Domäne hin, die auf einer falschen Identität beruht. In der Regel bedeutet das die Verwendung von gestohlenen beziehungsweise gefälschten Authentisierungsdaten.	C(2): „A masquerader can be defined as a person, either external or internal, who uses an account on the system for which he is not authorized.“ [LJ00].
	K, **P**, **T**, **U**	Die Insiderbeschreibung deutet auf ein eindeutig mittleres Level der Subkategorie hin.	P(2): „Insiders have full control of some [network] nodes.“ [NS05].

2,5	**C**, **K**, **P**, **T**, **U**	Die Beschreibung der Subkategorie deutet auf eine Ordnung zwischen 1 und 3 hin.	T(2,5): „A person with some sort of organizational status that causes members of the organization to view requests or demands as being authorized.“ [Com98].
3	**C**	Die Beschreibung der Subkategorie deutet auf eine legitime Zugehörigkeit zur Domäne hin, was gegenüber der Domäne und gegenüber Dritten auch glaubhaft nachgewiesen werden kann.	C(3): „[P]eople with legitimate access who behave in ways that put our data, our systems, our organizations, and even our businesses' viability at risk.“ [PS09].
	K, **P**, **T**, **U**	Die Insiderbeschreibung deutet auf ein eindeutig hohes Level der Subkategorie hin.	P(3), U(3): „The assumption regarding clandestine users is that the user has or can seize supervisory control of the machine and as such can either operate below the level at which audit trail data is taken or can use privileges or system primitives to evade audit trail data being recorded for him.“ [And80].

2.5.3 Klassifikation existierender Insiderdefinitionen

Der Prozess der Insidermodellierung kann auf existierende Insiderdefinitionen angewendet werden, um sie anhand ihrer zugrunde liegenden Insidertypen zu klassifizieren. Für diese Klassifizierung wurden 83 Insiderdefinitionen aus 47 unterschiedlichen wissenschaftlichen Publikationen extrahiert und analysiert, die mittels Schlüsselwort- sowie Vorwärtsreferenzierungs- und Rückwärtsreferenzierungssuche gefunden wurden. Eine Erkenntnis dieser Untersuchung ist, dass häufig mehr als ein Insidertyp durch eine dieser Insiderdefinitionen beschrieben wird. Darüber hinaus führt die fehlende Spezifizierung einer Domäne zu verschiedenen Insidertypen und manchmal sogar puren Outsidertypen, wenn man verschiedene Domänen annimmt. Butts, Mills und Baldwin [BMB05] definieren beispielsweise einen Insider als „[a]ny individual who has been granted any level of trust in an information system“. Jedes Level an *Trust* schließt *negligible Trust* mit ein, was auf einen *Trust-Outsider (T-Outsider)* hindeutet. Ein anderes Beispiel ist Pfleeger [Pfl08], der einen Insider unter anderem als „an employee, student, or other *member* of a host institution that operates a computer system to which the insider has legitimate access“ beschreibt. Der Insider ist als Angestellter definiert, was mit der Institution als Domäne bedeutet, dass der Insider eigene gültige *Credentials* besitzt. Der legitime Zugriff auf ein Computersystem innerhalb der Domäne bedeutet dann ein befugtes Level an *Privileges*, was zusammen zu einem C_A*-P-Insider* führt. Der Insidertyp verändert sich allerdings mit dem Computersystem als Domäne. Der legitime Zugang zum Computersystem drückt damit eigene *Credentials* des Insiders aus. Weitere Beschreibungen des *Insidergrades* aus Sicht der Domäne gibt es dann allerdings nicht, was zu einem C_A*-Insider* führt.

Die verschiedenen Insidermodelle, die von den 83 analysierten Insiderdefinitionen abgeleitet werden können, sind in Anhang A.2 aufgeschlüsselt und die daraus resultierenden Insidertypen

Tabelle 2.4: Klassifizierung existierender Insiderdefinitionen anhand der zugrunde liegenden Insidertypen. Die Prozentangaben zeigen den Anteil des Insidertyps im Vergleich zu allen in den Publikationen identifizierten Insidertypen.

C-Insider	25.21 %	[And80; LCW92; Neu99; LJ00; Sch02; Pat03; BA04; Jha+04; MT04; Ale+05; CR05; May+05; Shi07; KTB08; Pfl08; PS09; Bow+10; Kan+10; Pfl+10; Kis13; US13; MWB15]
P-Insider	21.85 %	[And80; Com98; Sch02; BA04; CR05; May+05; NS05; Shi07; Pfl08; Bis+09; PS09; Bis+10; Kan+10; Pfl+10; Pro+10a; HP11; US13; MWB15; McG+15; Kha17; SCD17]
C-P-Insider	15.97 %	[And80; Com98; Neu99; Str+00; MT04; Ale+05; CR05; Liu+05; KTB08; Pfl08; FP12b; US13; Kap+15]
C-K-Insider	5.04 %	[Ein02; Chi+05; Bow+10; Mat+10; YP10]
K-Insider	4.20 %	[BA04; May+05; Kan+10; MWB15; McG+15]
P-T-Insider	3.36 %	[BA04; Liu+05; Gre+08; GF10]
C-T-Insider	3.36 %	[BA04; Gre+08; GF10; MWB15]
T-Insider	2.52 %	[Com98; Pat03; BMB05]
K-P-T-Insider	2.52 %	[NRK03; DT09]
K-P-Insider	1.68 %	[Spi03a; YP10]
C-P-U-Insider	1.68 %	[And80; MFB06]
C-K-P-Insider	1.68 %	[Spi03a; MWB15]
C-P-T-Insider	1.68 %	[Bis05; HP11]
K-T-Insider	0.84 %	[GF10]
C-K-P-U-Insider	0.84 %	[Chi+05]
C-K-P-T-Insider	0.84 %	[DT09]
K-P-T-U-Insider	0.84 %	[FP12b]
C-K-P-T-U-Insider	0.84 %	[FP12b]
Outsider	1.68 %	[Pfl08; Kap+15]
C-Outsider	1.68 %	[And80; CR05]
T-Outsider	0.84 %	[BMB05]
K-Outsider	0.84 %	[Kap+15]

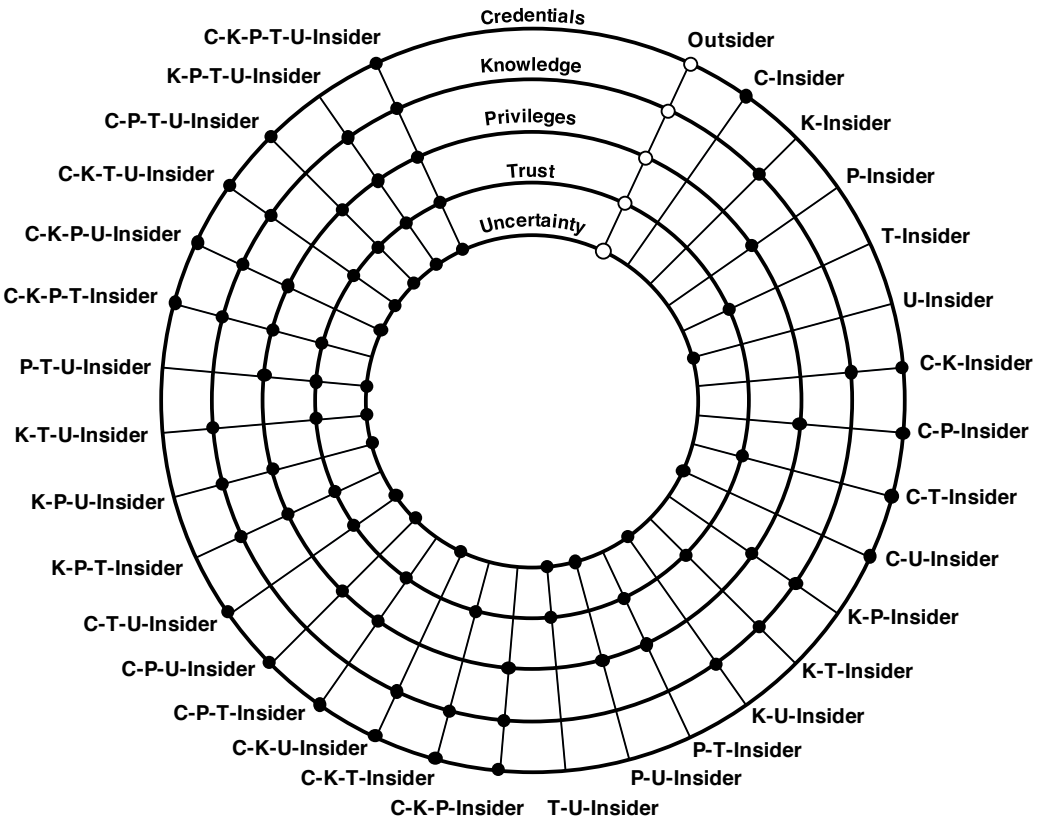


Abbildung 2.2: Die Insidertaxonomie aller Kombinationen von Insidertypen

werden darin aufgeführt. Die Ergebnisse sind in Tabelle 2.4 zusammengefasst, wobei hier der Übersichtlichkeit halber nicht zwischen einem C_A-*Insider* und einem C_M-*Insider* unterschieden wird. Die Tabelle zeigt den Fokus, der in der existierenden Literatur zum Thema Insider vorherrscht. Die mit Abstand meisten Definitionen, die analysiert wurden, konzentrieren sich auf einen *C-Insider*, einen *P-Insider* und einen *C-P-Insider*. Darüber hinaus zeigt die Tabelle die breite Diversität bei der Definition von Insidern. Oftmals fehlen detaillierte Informationen bezüglich der Ausprägung einzelner Insidercharakteristiken, sodass Ambiguitäten bei der Klassifikation einer vorgeschlagenen Insiderdefinition entstehen, die in mehr als einem Insidertyp oder sogar in Outsidertypen resultieren.

2.6 Insidertaxonomie

Für eine Ontologie der Insiderthematik wurden in diesem Kapitel bereits repräsentative Grundlagen und Primitiven (vgl. Abschnitt 1.5.10) geschaffen, die Insidertypen sowie deren Charakteristiken und ihre Anwendungen in Form von Insidermodellen beschreiben. In diesem Abschnitt folgt nun eine Darstellung und Erläuterung einer einfachen Insidertaxonomie, anhand derer die Relationen zwischen den Insidertypen deutlich werden.

Jede von einer Domäne bereitgestellte Charakteristik definiert einen Insidertyp in seiner eigenen unabhängigen Dimension, wobei alle Insidertypen miteinander kombiniert werden können (vgl. Abschnitt 2.4). Mit diesen Erkenntnissen kann eine einfache Insidertaxonomie entwickelt werden, in der die Relationen zwischen den Insidertypen abgebildet sind. Abbildung 2.2 veranschaulicht diese Relationen.

In Bezug auf die Insidercharakteristiken, die eine Domäne einem Individuum bereitstellt, ist jeder Basis-Insidertyp eine echte Verbesserung des jeweiligen Outsidertyps. Beispielsweise gehen die Privilegien eines *P-Insider* über die eines *Privileges-Outsider (P-Outsider)* hinaus. Weiterhin stellt eine Kombination von Basis-Insidertypen eine Verbesserung aller möglichen Sub-Kombinationen dar. Der C_A*-K-P-Insider* des Insidermodells aus Beispiel 2.2 in Abschnitt 2.5 etwa besitzt alle Eigenschaften und Fähigkeiten, die auch ein C_A*-K-Insider*, ein C_A*-P-Insider*, ein *K-P-Insider*, ein C_A*-Insider*, ein *K-Insider* sowie ein *P-Insider* besitzt. Abbildung 2.2 veranschaulicht diese Relationen und zeigt die entwickelte Insidertaxonomie, wobei auch hier für eine bessere Übersichtlichkeit auf die Unterscheidung zwischen einem C_A*-Insider* und einem C_M*-Insider* verzichtet wurde.

Mit detaillierten und adäquaten Insidermodellen (vgl. Abschnitt 2.5) kann eine Domäne folglich den Teil der Insidertaxonomie identifizieren, der für ein spezielles Insidermodell relevant ist. In Verbindung mit der Klassifikation der Insiderdefinitionen aus Abschnitt 2.5.3 können zum Beispiel jene Publikationen identifiziert werden, die zum vorliegenden Insidermodell passen und demnach inhaltlich wichtige Erkenntnisse liefern können. Für den genannten C_A*-K-P-Insider* lassen sich beispielsweise die Arbeiten von Spitzner [Spi03a] sowie Maasberg, Warren und Beebe [MWB15] nennen, die anhand ihrer Insiderdefinitionen ebenfalls einen Fokus auf diesen Insidertyp legen. Spitzner [Spi03a] schlägt zum Beispiel in seiner Arbeit den Einsatz von Honeypots und speziellen Köderdokumenten vor, auch Honeytokens genannt, um das Verhalten von angreifenden Insidern näher zu analysieren. Maasberg, Warren und Beebe [MWB15] wiederum legen den Fokus auf psychologische Indikatoren, um Insiderbedrohungen zu erkennen und zeigen einen Zusammenhang von Insiderbedrohungen mit der sogenannten dunklen Triade von Persönlichkeitsmerkmalen auf. Zusätzlich können übergeordnete Aspekte zum vorliegenden Insidertyp in den Arbeiten gefunden werden, die den C_A*-K-P-Insider* als Sub-Kombination enthalten. In der umgekehrten Richtung können ebenso Teilaspekte zu den Insidertypen gefunden werden, aus denen sich der vorliegende Typ zusammensetzt. All diese Vorschläge setzen allerdings voraus, dass die Insiderdefinitionen beziehungsweise -modelle, die von Autoren in ihren wissenschaftlichen Publikationen verwendet werden, auch zu den inhaltlich vorgeschlagenen Erkenntnissen oder Abwehrmaßnahmen passen. Ein Gegenbeispiel ist die genannte Arbeit von Spitzner [Spi03a], der sich laut seiner Insiderdefinition mit den Bedrohungen eines C_A*-K-P-Insiders* beschäftigt. Die vorgeschlagene Abwehrmaßnahme legt allerdings den Fokus auf die Bedrohungen eines *P-U-Insiders*, wie in Abschnitt 4.7 aufgezeigt wird.

Mit den nachfolgenden Kapiteln 3 und 4 wird der Grundstein für weitere Anwendungsfälle der Insidertaxonomie gelegt. Dadurch können passgenaue Sicherheitsmechanismen und Abwehrmaßnahmen identifiziert und implementiert werden, um sich vor den speziellen Bedrohungen von Insidern zu schützen. Dafür müssen jedoch Bedrohungen und entsprechende Abwehrmaßnahmen den einzelnen Insidertypen zugeordnet und in die Insidertaxonomie eingeordnet werden. Darüber hinaus kann die Wirksamkeit und die Stärke dieser Maßnahmen in Bezug auf den stärksten möglichen Insider abgeschätzt werden, da alle Maßnahmen, die vor den Bedrohungen einer spezifischen Kombination von Basis-Insidertypen schützt, immer auch vor allen Sub-Kombinationen dieser Basis-Insidertypen schützt.

2.7 Fazit

Die Beschreibung und Definition von Insidern und ihren Domänen ist integraler Bestandteil der Insiderforschung. Mangelnde Grundlagen und fehlende Systematiken für diese Insiderdefinition

wurden in diesem Kapitel aufgezeigt und aufgelöst. Zu den Problemen gehören unter anderen die fehlende explizite Benennung einer Insiderdomäne (vgl. Abschnitt 2.2.1), unterschiedliche und oftmals gegensätzliche Charakterisierungen von Insidern (vgl. Abschnitt 2.2.5) sowie statische Insiderdefinitionen (vgl. Abschnitt 2.2.6). Der Kern der Auflösung dieser und weiterer Definitionsprobleme liegt in der Identifizierung von Charakteristiken, die allein für die Qualifizierung von Insidergraden verantwortlich sind. Diese umfassen die Zugehörigkeit zu einer Domäne (*Credentials*), das Wissen über eine Domäne (*Knowledge*), die zulässigen Interaktionen mit den Ressourcen einer Domäne (*Privileges*), das von einer Domäne entgegengebrachte persönliche Vertrauen (*Trust*) sowie das strukturelle Vertrauen (*Uncertainty*) (vgl. Abschnitt 2.3).

Mit diesen von einer Domäne bereitgestellten Charakteristiken lassen sich Basis-Insidertypen etablieren, die beliebig kombinierbar sind und somit jeden möglichen Insidertyp eindeutig beschreiben (vgl. Abschnitt 2.4). Die Einteilung der Insidercharakteristiken in geeignete geordnete Wertigkeiten, zum Beispiel *negligible*, *low*, *middle* und *high*, erlauben eine graduelle Beschreibung der Insidergrade, sodass nicht mehr nur zwischen einem *Outsider* und einem *Insider* unterschieden werden muss, sondern auch Abstufungen verschiedener Insidergrade möglich sind.

Mit diesen Erkenntnissen als Grundlage wurden Insidermodelle sowie eine Methodik zur Erstellung von Insidermodellen eingeführt (vgl. Abschnitt 2.5), die durch die explizite Benennung einer Insiderdomäne sowie durch die Identifizierung und Instanziierung von Insidercharakteristiken aus einer informellen Insiderbeschreibung oder einem Insiderszenario den zugrunde liegenden Insidertyp herausarbeiten. Dadurch erhält die Beschreibung oder das Szenario eine Zuordnung zu einem oder gegebenenfalls mehreren Insidertypen. Anhand von Beispielszenarien (vgl. Abschnitt 2.5.1) sowie von Insiderdefinitionen beziehungsweise -beschreibungen aus existierenden Forschungsarbeiten (vgl. Abschnitt 2.5.3) wurde diese Insidermodellierung präsentiert. Dabei stellte sich heraus, dass der Hauptfokus der untersuchten Forschungsarbeiten auf den C-Insidern, P-Insidern und C-P-Insidern liegt. Mithilfe dieser Modellierungsmethodik lassen sich nun Insiderbeschreibungen und -szenarien Forschungsarbeiten zuordnen, die den oder die gleichen Insidertypen als Fokus haben. Es wird somit ersichtlich, welche Forschungsarbeiten an den gleichen Insidertypen forschen und für welche Szenarien diese Forschung relevant ist.

Weiterhin wurden zur Komplementierung des Forschungsbeitrags B1 (vgl. Abschnitt 1.3) Relationen zwischen allen möglichen Insidertypen in einer Insidertaxonomie erarbeitet und systematisiert (vgl. Abschnitt 2.6), sodass die nachfolgenden Kapitel darauf aufbauen und sowohl Insiderbedrohungen als auch Gegenmaßnahmen zu Insidertypen zuordnen können. Diese entwickelten Grundlagen und Primitiven für den Kontext einer Insiderdefinition ergeben zusammengenommen eine Insiderontologie. Sie ermöglicht in Zukunft schnelle und akkurate Aussagen darüber, welche Erkennungs- und Abwehrmaßnahmen von Insiderbedrohungen miteinander vergleichbar sind und welche miteinander kombiniert werden müssen, um Schutz vor bestimmten Insiderbedrohungsszenarien zu erhalten.

3 Mehrseitige Bedrohungen für und durch Insider

Die Bedrohungen für Domänen durch Insider werden regelmäßig als eine der schwierigsten, teuersten und häufigsten Bedrohungen genannt [Bis+08; Pro+10a; Sil+12]. In einer Studie von Verizon [Ver19] kommen die Autoren bei der Befragung von 390 Unternehmen zu der Erkenntnis, dass die meisten der abgefragten Insiderbedrohungen, wie zum Beispiel unautorisierter Datenzugriff, erst nach Monaten oder gar Jahren entdeckt werden. Weiterhin wird in dieser Studie offengelegt, dass Insiderbedrohungen der häufigste Grund für einen Sicherheitsvorfall und der zweithäufigste Grund für Datenlecks sind. Eine andere Studie von Ponemon Institute LLC u. a. [Pon+19] ordnet den jährlichen Schaden durch Insidervorfälle, der neben demjenigen durch Ransomware die zweitgrößte Wachstumsrate hat, aktuell auf Platz Vier ein, direkt hinter Malware, Web-basierten und Denial-of-Service-Angriffen.

Äquivalent zu den bereits in Kapitel 2 bearbeiteten Problemen von Insiderdefinitionen und darüber hinaus auch in Abhängigkeit davon fehlt es allerdings an grundlegenden und adäquaten Definitionen von Insiderbedrohungen. Dadurch lassen sich derartige Studien, Statistiken und Forschungsarbeiten weder einordnen noch in Beziehung zueinander setzen. In der Regel wird dieser Missstand damit übergangen, dass der Versuch einer Insiderdefinition unternommen wird. Eine Insiderbedrohung ist dann eine klassische Bedrohung, die in der Argumentation derartiger Arbeiten von diesem Insider ausgeht. Eine solche Definition von Insiderbedrohungen ist allerdings weder adäquat noch hilfreich. Denn auch Insiderbedrohungen sind vielfältig und lassen sich in ihren konkreten Ausprägungen nicht auf die Definition eines Insiders reduzieren.

In diesem Kapitel wird daher eine grundlegende Systematisierung von Insiderbedrohungen vorgenommen und somit Forschungsfrage 2 adressiert (vgl. Abschnitt 1.2). Dabei wird äquivalent zum Prinzip der *Mehrseitigen Sicherheit*, wie sie von Rannenberg, Pfitzmann und Müller [RPM97] eingeführt wurde, das Prinzip einer *Mehrseitigen Insiderbedrohung* zugrunde gelegt. Im Fokus stehen dabei die angesprochenen Insiderbedrohungen für Domänen. Behandelt werden außerdem Bedrohungen für Insider durch andere Personen sowie durch die Domäne selbst.

Wesentliche Inhalte Mit einer abstrakten Begriffsdefinition von *Insiderbedrohungen* wird aufgezeigt, dass sich Insiderbedrohungen nicht allein dadurch kennzeichnen, dass sie von Insidern ausgehen, sondern dass konkrete Insidergrade bei Bedrohungsaktionen zum Einsatz kommen und Verbesserungen in drei verschiedenen Dimensionen schaffen. Weiterhin werden Insiderbedrohungen in Analogie zu den klassischen Bedrohungen der IT-Sicherheitsschutzziele *Vertraulichkeit*, *Integrität* und *Verfügbarkeit* charakterisiert und unterteilt. Mit dieser Unterteilung werden Besonderheiten, Details und vor allem Voraussetzungen der Bedrohungen aufgezeigt, die einen bestimmten Insidergrad betreffen. Daraus abgeleitet werden Ansätze zur Erkennung und Abwehr der einzelnen Bedrohungen diskutiert, die sich an den besagten Besonderheiten, Details und Voraussetzungen ausrichten. Die konkret identifizierten Insiderbedrohungen werden weiterhin sowohl in sich als auch anhand der Insidertaxonomie aus Abschnitt 2.6 systematisiert (Forschungsbeitrag B2 aus Abschnitt 1.3). Auf diese Weise erlauben sie Aussagen drüber, welche Insiderbedrohung von welchem Insidertyp ausgehen. Darüber hinaus werden Bedrohungen für Insider durch Outsider sowie durch die Domäne selbst betrachtet. Letzteres dient als Vorbereitung

für die Kapitel 6 und 7. Dabei wird die Gefahr für die Privatsphäre und die informationelle Selbstbestimmung von Insidern aufgezeigt, die von weitreichenden Überwachungsmaßnahmen einer Domäne ausgeht.

Aufbau des Kapitels Das Kapitel beginnt in Abschnitt 3.1 mit der Beschreibung eines Systemmodells von Insiderbedrohungen. Abschnitt 3.2 erarbeitet und diskutiert die Bedrohungen für Domänen durch Insider und stellt damit den Kern dieses Kapitels dar. In Abschnitt 3.3 werden zur weiteren Betrachtung der *Mehrseitigen Insiderbedrohungen* auch die Bedrohungen für Insider durch Outsider hervorgehoben, durch die sich Outsider unerlaubt die Rolle von Insidern aneignen. Anschließend wird in Abschnitt 3.4 die Bedrohung für Insider durch ihre Domäne, etwa in Form von weitreichender Überwachung, adressiert. Die Zuordnung einzelner Insiderbedrohungen zu den Insidertypen und damit die Systematisierung anhand der Insidertaxonomie aus Abschnitt 2.6 erfolgt in Abschnitt 3.5. Abschließend wird in Abschnitt 3.6 eine Erweiterungsmöglichkeit aufgezeigt und in Abschnitt 3.7 ein Fazit gezogen.

3.1 Systemmodell

Da die Szenarien eines Insiders sowie mögliche Bedrohungen sehr stark variieren, kann das Systemmodell nur auf einem sehr abstrakten Niveau festgelegt werden. Drei Beispiele veranschaulichen die Problematik:

Beispiel 3.1. Ein Mitarbeiter eines Unternehmens verschafft sich Zugang zu einem Systemteil, der unternehmensinterne Geschäftsgeheimnisse enthält, sucht nach dortigen wertvollen Informationen und verkauft diese an einen Konkurrenten des Unternehmens.

Abbildung 3.1 illustriert die einzelnen Komponenten aus Beispiel 3.1. Zu beachten ist, dass es sich dabei um eine reine Bedrohung durch einen Insider für die Domäne handelt. Dabei ist es unerheblich, ob der Mitarbeiter das Unternehmensnetzwerk sowie seinen Arbeitsplatzrechner verwendet, um die Geschäftsgeheimnisse zu entwenden, oder ob es um wertvolle Papierdokumente geht, die ohne Zuhilfenahme unternehmenseigener IT physisch entwendet werden.

Beispiel 3.2. Ein Mitarbeiter eines Unternehmens erhält eine E-Mail von einem außenstehenden Angreifer, der sich mithilfe einer gefälschten Absenderadresse als sein Chef ausgibt. In der E-Mail steht die Bitte, die Software im Anhang der E-Mail zu entpacken und auf dem lokalen Rechner auszuführen. Der Mitarbeiter führt die Software auf seinem Rechner aus und bemerkt nicht, dass damit eine Hintertür für den Zugang von außerhalb des Unternehmensnetzwerks eingerichtet wird.

Abbildung 3.2 illustriert die einzelnen Komponenten aus Beispiel 3.2. Im Unterschied zu Beispiel 3.1 handelt es sich hierbei vorwiegend um einen Outsiderangriff, in dem sich der Angreifer die Fähigkeit angeeignet hat, sich glaubhaft als Chef des Unternehmens auszugeben. Sei es durch Übernahme des E-Mail-Kontos des Chefs oder durch Aneignung von genügend Kenntnissen über unternehmensinterne Strukturen, Abläufe und Gepflogenheiten, sodass die E-Mail täuschend echt wirkt. Ausgestattet mit dieser Fähigkeit erfolgen anschließend zwei nachgelagerte Insiderangriffe. Diese bestehen zum einen aus der erfolgreichen Verleitung des Mitarbeiters

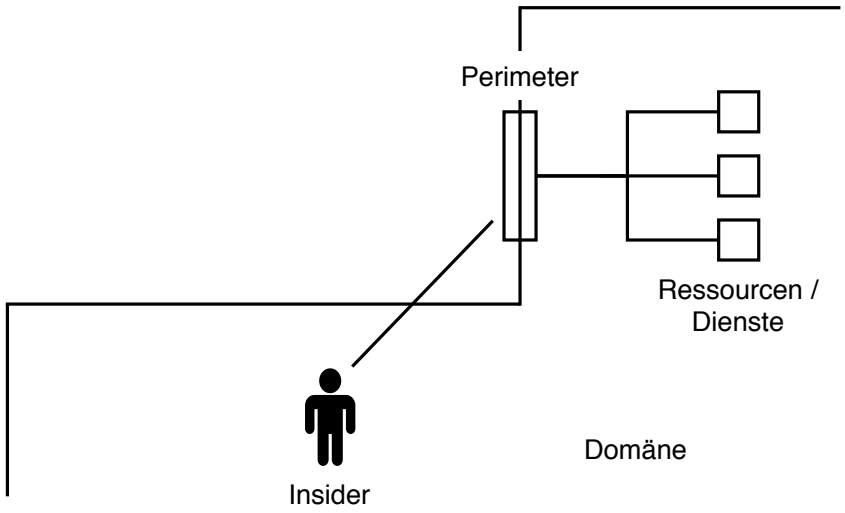


Abbildung 3.1: Komponenten und Kommunikationsverbindungen von Beispiel 3.1

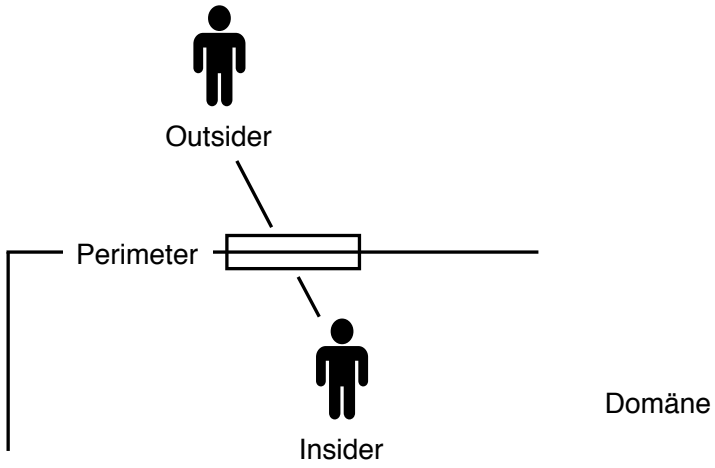


Abbildung 3.2: Komponenten und Kommunikationsverbindungen von Beispiel 3.2

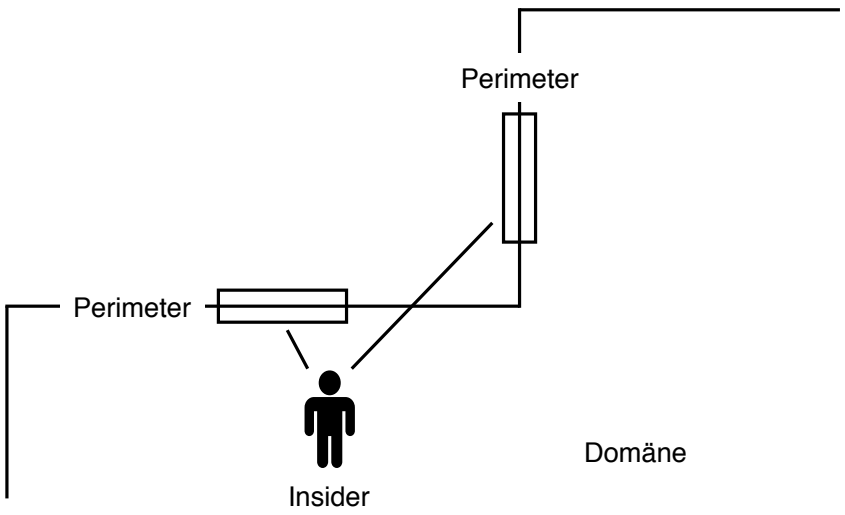


Abbildung 3.3: Komponenten und Kommunikationsverbindungen von Beispiel 3.3

durch den Angreifer, den E-Mail-Anhang auszuführen und zum anderen aus der unbewussten Ausführung des versteckten Schadcodes durch den Mitarbeiter, wodurch unbemerkt die Hintertür im System eingerichtet wird. Der Outsiderangriff ist allerdings die Ausgangslage und steht bei diesem Beispiel im Fokus.

Beispiel 3.3. Ein Unternehmen, dessen Geschäftsmodell auf sensiblen Geschäftsgeheimnissen und geistigem Eigentum beruht, installiert auf allen Arbeitsplatzrechnern sowie im Unternehmensnetzwerk Überwachungsmechanismen, um sowohl Gefahren durch Outsider als auch durch Insider frühzeitig erkennen und verhindern zu können. Dabei fallen umfassende Daten über die Aktivitäten der Mitarbeiter an, die offiziell zur Anomalieerkennung genutzt werden. Inoffiziell werden damit allerdings auch Rückschlüsse auf die Produktivität der einzelnen Mitarbeiter gezogen.

Abbildung 3.3 illustriert die einzelnen Komponenten aus Beispiel 3.3. Bei diesem Beispiel handelt es sich um eine Bedrohung der Insider durch die Domäne, da detaillierte Aktivitätsprofile einzelner Mitarbeiter erstellt werden und für die Analyse der Produktivität missbraucht werden.

Für die präzise Beschreibung der Bedrohungen, die von Insidern ausgehen und diese selbst betreffen, sowie der Schutzziele und Bedrohungsmodelle müssen demnach die folgenden Komponenten betrachtet werden:

1. der Insider,
2. die Domäne und dessen technischer und organisatorischer Perimeter,
3. die domäneninternen Funktionen, Dienste, Ressourcen und Güter sowie
4. (optional) eine dritte Partei (ein Outsider oder ebenfalls ein Insider).

Es gibt Szenarien, in denen eine dritte Partei, wie zum Beispiel ein Outsiderangreifer oder ein Konkurrent der Domäne keine Rolle spielt und demnach auch nicht im Systemmodell berücksichtigt werden muss. Darüber hinaus sind die folgenden Interaktionen zu betrachten:

1. die Interaktion zwischen dem Insider und der Domäne, insbesondere den domäneninternen Funktionen, Diensten, Ressourcen und Gütern,
2. (optional) die Interaktion zwischen dem Insider und der dritten Partei sowie
3. (optional) die Interaktion zwischen der Domäne und der dritten Partei.

Diese Interaktionen können sowohl technisch als auch nicht-technisch erfolgen und beschreiben je nach konkretem Anwendungsfall logische oder tatsächlich physische Kommunikationsverbindungen beziehungsweise Interaktionen.

Der Insider befindet sich nicht automatisch innerhalb des Perimeters der Domäne (vgl. Abschnitt 2.2.3), sondern kann mit den domäneninternen Funktionen, Diensten, Ressourcen und Gütern durch den Perimeter hindurch interagieren, sofern er den nötigen Insidergrad dafür hat (vgl. Abschnitt 2.3). Dieser Perimeter einer Domäne ist definiert durch das zugrunde liegende Insidermodell, insbesondere durch die domänenspezifische Semantik der jeweiligen Charakteristikwerte (vgl. Abschnitt 2.4 und Abschnitt 2.5).

Entsprechend der drei genannten Beispiele werden in den nachfolgenden Abschnitten 3.2 bis 3.4 die unterschiedlichen Bedrohungen für und durch Insider näher betrachtet.

3.2 Bedrohungen für Domänen durch Insider

Mit den drei identifizierten Komponenten sowie den Interaktionen aus Abschnitt 3.1 sind die hier genauer betrachteten Bedrohungen für eine Domäne durch einen Insider eine Richtung, bei der Interaktion zwischen dem Insider und der Domäne stattfindet. Die Rückrichtung, bei der die Domäne auf den Insider einwirkt, wird in Abschnitt 3.4 beleuchtet.

Dieser Abschnitt definiert, charakterisiert und systematisiert dabei zunächst den zentralen Begriff einer *Insiderbedrohung* (s. Abschnitt 3.2.1). Daraus gehen die konkreten und insiderspezifischen Bedrohungen der *unbefugten Weitergabe*, der *unbefugten Eskalation* sowie der *unbefugten Verhinderung von Insidergraden* hervor (s. Abschnitte 3.2.2 bis 3.2.4). Abschließend wird zur Vervollständigung der konkreten Insiderbedrohungen auf Verbesserungen durch Insidergrade eingegangen, die bei jeglichen Insiderbedrohungen eine Rolle spielen (s. Abschnitt 3.2.5).

3.2.1 Insiderbedrohungen neu definiert und charakterisiert

Die in Abschnitt 1.5.3 vorgenommene Festlegung der Grundbegriffe eines Angreifers, eines Angriffs und einer Bedrohung wird hier als Grundlage genommen. Eine Bedrohung umfasst dabei sowohl Angriffe, das heißt bewusst gut- oder bösartige Bedrohungsaktionen, als auch unbewusste Bedrohungsaktionen. Ein Angreifer ist einer von weiteren möglichen Bedrohungsagenten, der bewusst eine gut- oder bösartige Bedrohungsaktion durchführt. Andere Bedrohungsagenten handeln dementsprechend unbewusst.

Mit der Möglichkeit der eindeutigen und umfassenden Definition beziehungsweise Modellierung von Insidern anhand der in den Abschnitten 2.4 und 2.5 erarbeiteten Spezifikationen, können nun angemessene Definitionen der Begriffe *Insiderangreifer*, *Insiderangriff* sowie *Insiderbedrohung* etabliert werden. Entgegen der Intuition, die durch diese Begriffe angestoßen wird, kann ein Insiderangreifer nicht definiert werden als ein Angreifer, der als Insider charakterisierbar ist, oder spezifischer, der als ein oder mehrere Basis-Insidertypen klassifizierbar ist (vgl. Abschnitt 2.4.1). Gleiches gilt für einen Insiderangriff und eine Insiderbedrohung. Der vorhandene Insidergrad eines Angreifers hat möglicherweise keinerlei Relation zu seinem konkreten Angriff. Ein *T-Insider* könnte beispielsweise einen erfolgreichen Bruteforce-Angriff auf das Zugangspasswort eines Mitglieds einer Domäne durchführen. Dieser Angriff ist kein Insiderangriff, denn der Angreifer hätte das Zugangspasswort genauso gut per Bruteforce erfolgreich angreifen können, wenn er kein *T-Insider* gewesen wäre.

Der korrekte Weg, eine Insiderbedrohung zu definieren, führt über die Verknüpfung von Insidercharakteristiken mit der Bedrohung. Mit dem Ansatz der Differenzierung zwischen spezifischen Charakteristiken, die, wie in Abschnitt 2.3 beschrieben, einem Individuum von einer Domäne zur Verfügung gestellt werden, erfolgt eine präzise Definition einer Insiderbedrohung:

Definition 3.1 (Insiderbedrohung)**.** Eine Insiderbedrohung ist eine Bedrohung[1] einer Domäne durch einen Insider, die durch den Einsatz von bereitgestellten Insidercharakteristiken entsteht oder ein verbessertes Potenzial aufweist, die Domäne nachteilig zu beeinflussen.

1. Für eine Einordnung des Begriffs *Bedrohung* wird auf Abschnitt 1.5.3 verwiesen.

Tabelle 3.1: Fallunterscheidung einer Bedrohung durch einen Outsider beziehungsweise einen Insider ohne und mit Einsatz des Insidergrades

	ohne Einsatz des Insidergrades	mit Einsatz des Insidergrades
Outsider	Outsiderbedrohung	✖
Insider	Outsiderbedrohung	Insiderbedrohung

Die Definition bezieht sich sowohl auf Insiderangriffe als auch auf unbewusst durchgeführte Bedrohungsaktionen durch Insider. Ausgenommen sind allerdings unvorhersehbare Ereignisse, wie etwa Katastrophen, äußere Einflüsse oder Verschleiß (vgl. Abschnitt 1.5.5), da diese keinen Bezug zu einem Insidergrad haben. Der Insiderangreifer ist in diesem Fall ein Bedrohungsagent, der von einer Domäne zur Verfügung gestellte Insidercharakteristiken verwendet, um seine Bedrohungsaktion durchzuführen oder zu verbessern.

Tabelle 3.1 veranschaulicht die Relation zwischen einem Insider und dem Einsatz von Insidergraden, die erst in Verbindung miteinander zu einer Insiderbedrohung führen. Der in Definition 3.1 angesprochene Aspekt der Verbesserung wird etwas später in Abschnitt 3.2.5 näher betrachtet.

3.2.1.1 Insiderbedrohungsmodelle

Die klassische Modellierung eines Angreifers wurde in Abschnitt 1.5.9 bereits auf die Angreifercharakteristiken *Rolle(n), Verbreitung, Verhalten* und *Ressourcen* zurückgeführt. Im Kontext von Insiderbedrohungen kann auf diese klassische Angreifermodellierung zurückgegriffen und eine Verfeinerung vorgenommen werden. Sowohl die Rolle(n) als auch die Verbreitung eines Insiderbedrohungsagenten und damit im Speziellen auch eines Insiderangreifers können anhand der in den Abschnitten 2.4 und 2.5 spezifizierten Insidertypen und Insidermodellen ersetzt und damit genauer festgelegt werden. Dadurch wird der Insidergrad des Angreifers eindeutig benannt. Für die Bedrohungsaspekte können nun weitere Charakteristiken herangezogen werden, die nicht den Insidergrad, sondern das Bedrohungspotenzial des Insiders genauer beschreiben. Es handelt sich also um inhärente Charakteristiken eines Bedrohungsagenten, die ohne die Mithilfe oder die Gunst einer Domäne, die in diesem Fall das bedrohte Ziel darstellt, erworben werden können und die nicht von der Domäne steuer- oder kontrollierbar sind. Die Einführung der Charakteristiken als Variablen, die mit geordneten Werten instanziiert werden können, erlaubt eine sehr feingranulare Modellierung eines Insiderbedrohungsagenten.

- **Abilities (A)**, manchmal auch als **Skills** bezeichnet, beschreibt den Grad an praktischen Kompetenzen und Expertise, die ohne domänenspezifische Informationen erworben werden können. Beispiele sind generelle Rechnernetz-Penetrationsfähigkeiten oder Kenntnisse über Software- und Protokollspezifikationen sowie Schwachstellen. Der Grad der *Abilities*, die ein Individuum besitzt, kann mittels Abschätzung oder Approximation auf die diskreten Werte *incompetent*, *proficient*, *skilled* oder *expert* gesetzt werden.
- **Behaviour (B)** umfasst die Interaktion eines Individuums mit der Domäne. Diese kann nicht vorhanden sein, was mit einer Unterlassung von Interaktionen gleichgesetzt wird. Die Interaktion kann aber auch sowohl passiv sein, also eine reine Informationsgewinnung, als auch aktiv, also eine direkte Interaktion und Modifikation des Domänenzustandes. Schließlich kann die Charakteristik *Behaviour* als adaptiv spezifiziert werden. Damit verhält sich ein Individuum auf eine Art und Weise, die den beobachteten Domänenzustand

aufgrund von früheren aktiven Interaktionen einbezieht und damit die zukünftigen aktiven Interaktionen und Modifikationen beeinflusst. Der konkrete Wert dieser Charakteristik kann zusammengefasst eines der Elemente *omitted*, *passive*, *active* oder *adaptive* annehmen.

- **Compliance (C)** betrifft die Konformität und Protokolltreue gegenüber Regeln, Spezifikationen oder Richtlinien. Ein Individuum verhält sich konform, wenn keine Abweichung von erwarteter und/oder definierter Interaktion mit der Domäne vorliegt.[2] Ansonsten verhält sich das Individuum nicht-konform. Die konkrete Ausprägung lässt sich also einteilen in *non-compliant* und *compliant*.
- **Intention (I)**, auch mit **Motivation** oder **Goal** benannt, fasst die intrinsische Natur einer Aktion oder einer Abfolge von Aktionen eines Individuums zusammen. Die *Intention* kann mit einem der drei sich ausschließenden Werte *intentionally good-natured*, *accidental* oder *intentionally malicious* instanziiert werden. Eine Aktion, die ohne *Intention* ausgeführt wird, kann als *accidental* angesehen werden.
- **Resources (R)** beschreibt die allgemeine Stärke eines Individuums bezogen auf Aspekte, die jeweils für die Ausführung einer Aktion, welche Auswirkungen auf eine Domäne hat, zur Verfügung stehen. Solche Aspekte können Rechenleistung, finanzielle Ressourcen, verfügbarer Speicherplatz und Zeit sein. Im Unterschied zur Insidercharakteristik *Privileges* aus Abschnitt 2.3 geht es hierbei explizit nicht um Ressourcen, die von der Domäne selbst zur Verfügung gestellt werden. Die konkrete Zuweisung dieser Charakteristik kann zurückgeführt werden auf die diskreten Werte *blank*, *little*, *limited* oder *unlimited*.

In Ergänzung zu Abschnitt 2.5.2, in dem eine formale Methodik zur Analyse von Insiderszenarien mit dem Fokus auf den reinen Insidergrad einer Person, also dessen Rolle(n) und Verbreitung, vorgestellt wurde, findet sich in Anhang A.3 eine Erweiterung dieser Methodik auf die hier vorgestellten inhärenten Bedrohungscharakteristiken. Damit lassen sich die in Abschnitt 2.5.1 beschriebenen Beispiel-Insidermodelle anhand der zugrunde liegenden Szenariobeschreibungen zu Insiderbedrohungsmodellen erweitern.

Das Insiderbedrohungsmodell entsprechend der Szenariobeschreibung aus Beispiel 2.1 (Angestellter) in Abschnitt 2.5.1 (Seite 31) sieht damit wie folgt aus:

Domäne: Die gesamte Organisation

Insidercharakteristiken:

- Credentials: legit → C_A-Insider
- Knowledge: low → K-Insider
- Privileges: low → P-Insider
- Uncertainty: negligible → U-Outsider

2. In der Definition eines klassischen Angreifermodells von Federrath und Pfitzmann [FP12a] fällt diese Charakteristik unter das sogenannte *Verhalten* eines Angreifers. Die Autoren bezeichnen die hier genannten Aspekte als *beobachtend* und *modifizierend*, was allerdings zu Missverständnissen mit dem ebenfalls von den Autoren einbezogenen *aktivem* und *passivem* Verhalten eines Angreifers führt. Weiterhin konzentrieren sich die Autoren bei den Aspekten *beobachtend* und *modifizierend* darauf, ob eine Protokolluntreue vom Angriffsziel als solche erkannt werden kann. Nur dann wird das Verhalten des Angreifers als modifizierend, andernfalls als beobachtend charakterisiert. Diese Besonderheit muss hier nicht beachtet werden, da das tatsächliche Erkennen oder Nicht-Erkennen der *Compliance* seitens der Domäne unter die Insidercharakteristik *Uncertainty* des Angreifers beziehungsweise Bedrohungsagenten fällt.

Insidertyp: C_A-K-P-Insider
Bedrohungscharakteristiken: • Abilities: skilled • Behaviour: active • Intention: intentionally malicious

Die Bestimmung der Insidercharakteristiken wurde in Abschnitt 2.5.1 bereits detailliert erläutert. Weitere Aussagen können über die inhärenten Charakteristiken *Abilities*, *Intention* sowie *Behaviour* gemacht werden. Für die Manipulation oder Umgehung der Sicherheits- und Überwachungsmechanismen kann wenigstens von einem qualifizierten Level an *Abilities* ausgegangen werden. Die Absichten werden explizit als böswillig beschrieben. Darüber hinaus muss das Individuum aktiv mit der Domäne interagieren, um sein Ziel zu erreichen.

Das zweite Insiderbedrohungsmodell entsprechend der Szenariobeschreibung aus Beispiel 2.2 (Streifenpolizist) in Abschnitt 2.5.1 (Seite 33) kann folgendermaßen zusammengefasst werden:

Domäne: Die Fahrzeughalterdatenbank
Insidercharakteristiken: • Credentials: legit → C_A-Insider • Knowledge: high → K-Insider • Privileges: middle → P-Insider • Uncertainty: middle → U-Insider
Insidertyp: C_A-K-P-U-Insider
Bedrohungscharakteristiken: • Behaviour: active • Compliance: compliant • Intention: intentionally malicious

Der Streifenpolizist muss aktiv mit der Domäne interagieren und die Tatsache, dass er aus Sicht der Domäne prinzipiell die Berechtigung zur Halterabfrage hat und diese in bestimmten Situationen sogar durchführen muss, zeigt, dass er konform und protokolltreu handelt. Die *Intention* ist allein getrieben von persönlichem Vorteil, denn die junge Frau hat kein Verkehrsvergehen begangen. Eine Halterabfrage wäre also nicht notwendig gewesen.

Das letzte Insiderbedrohungsmodell entsprechend der Szenariobeschreibung aus Beispiel 2.3 (Krankenpfleger) in Abschnitt 2.5.1 (Seite 33) kann schließlich wie folgt beschrieben werden:

Domäne: Das medizinische Patientendatensystem
Insidercharakteristiken: • Credentials: stolen → C_M-Insider • Knowledge: middle → K-Insider • Privileges: middle → P-Insider • Uncertainty: low → U-Outsider
Insidertyp: C_M-K-P-Insider
Bedrohungscharakteristiken: • Behaviour: active • Compliance: non-compliant • Intention: intentionally good-natured

Der Krankenpfleger muss aktiv mit der Domäne interagieren und die Tatsache, dass er in seiner Rolle keine Erlaubnis hat, die benötigten Patientendaten einzusehen, zeigt, dass er nicht-konform handelt. Die *Intention* ist getrieben durch die Notwendigkeit und moralische Verpflichtung, den medizinischen Zustand des Patienten in Not wieder zu stabilisieren. Der Krankenpfleger handelt also in guter Absicht.

3.2.1.2 Systematisierung von Insiderbedrohungen

Insiderbedrohungen lassen sich allgemein in die drei klassischen Bedrohungen der Informationssicherheit, also dem unbefugten Informationsgewinn, der unbefugten Modifikation von Informationen und der unbefugten Verhinderung von Ressourcennutzung (vgl. Abschnitt 1.5.6) einordnen. Darüber hinaus erlauben die Spezifika von Insiderbedrohungen auch eine neue Klassifizierung, die besondere Eigenschaften von Insiderbedrohungen hervorheben:

Definition 3.2 (Klassen von Insiderbedrohungen)**.** Bedrohungen einer Domäne, die durch bereitgestellte Insidercharakteristiken entstehen oder verbessert werden, lassen sich einordnen in

a) die unbefugte Weitergabe des Insidergrades (Verletzung der Insidergrad-Vertraulichkeit),

b) die unbefugte Eskalation des Insidergrades (Verletzung der Insidergrad-Integrität) sowie

c) die unbefugte Verhinderung des Insidergrades (Verletzung der Insidergrad-Verfügbarkeit).

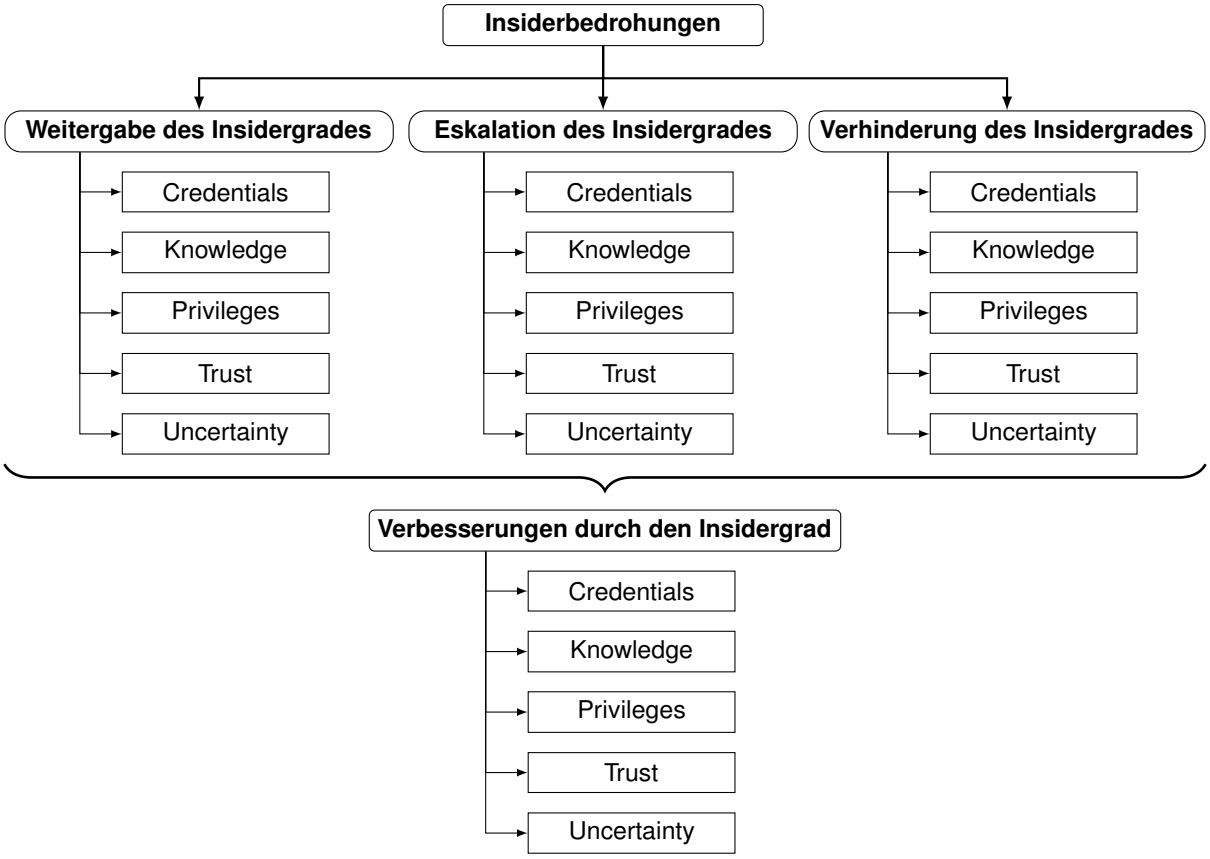


Abbildung 3.4: Systematisierung von Insiderbedrohungen

Die Klasse b) von Insiderbedrohungen umfasst nicht ausschließlich Bedrohungen für Domänen durch Insider, denn die Eskalation des Insidergrades wird gleichermaßen von vielen Outsiderangreifern angestrebt, wie in Abschnitt 3.3 gezeigt wird. Häufig spielt dabei allerdings ein Insider als Gegenpart eine Rolle, dessen Handlungen bewusst oder unbewusst die Weitergabe seines Insidergrades (Klasse a)) zur Folge haben können. Damit zeigt sich die Wichtigkeit, mehrseitige Bedrohungen zu identifizieren und zu untersuchen. Die drei Klassen von Insiderbedrohungen werden in den Abschnitten 3.2.2 bis 3.2.4 detailliert und für jede Insidercharakteristik getrennt betrachtet. Dabei werden sowohl Voraussetzungen als auch mögliche Gegenmaßnahmen aufgezeigt. Der Aspekt der *Verbesserung* ist zentral für die Definition 3.1 (Insiderbedrohung) und trifft auf alle Klassen von Insiderbedrohungen zu. Diese Verbesserungen werden in Abschnitt 3.2.5 diskutiert. Dabei werden ebenfalls mögliche Gegenmaßnahmen speziell ausgerichtet auf die Verbesserungen von Insiderbedrohungen durch die einzelnen Insidercharakteristiken betrachtet. Abbildung 3.4 veranschaulicht die hier aufgezeigte und in den genannten Abschnitten eingehender bearbeitete Systematik von Insiderbedrohungen.

3.2.1.3 Die Intention hinter einer Insiderbedrohung

Anders als die anderen in Abschnitt 3.2.1.1 beschriebenen inhärenten Bedrohungscharakteristiken und entgegen einer weitverbreiteten Auffassung hat die *Intention* keinen Einfluss auf das Bedrohungspotenzial einer Bedrohungsaktion und insbesondere eines Angriffs. Ein Insiderangriff, der zum Beispiel mit einer bösartigen Absicht durchgeführt wurde, muss von der Domäne als ein Angriff angesehen und entsprechend behandelt werden. Sollte die gleiche Bedrohungsaktion

jedoch unabsichtlich ausgeführt werden, kann dies nicht als ein Angriff angesehen werden. Die Auswirkungen dieser Aktion sind aus Sicht der Domäne allerdings identisch. Demnach müssen die Erkennungs- und Abwehrmaßnahmen ebenfalls identisch sein. Einzig die Reaktionen können und sollten sich unterscheiden. So kann zum Beispiel das Bewusstsein für die Auswirkungen einer solchen Aktion sowie Achtsamkeit geschaffen werden. Gegebenenfalls sind Verfahren und Prozesse zu ändern, um derartige Fehler in Zukunft zu vermeiden. Das Gleiche gilt für eine Bedrohungsaktion, die absichtlich und mit gutartiger Motivation ausgeführt wurde. Der Effekt ist aus Sicht der Domäne identisch zu einem Angriff, obwohl die Absicht das genaue Gegenteil ist. Die Gegenmaßnahmen müssen auch hier identisch sein, aber die nachträgliche Behandlung kann und sollte sich unterscheiden, zum Beispiel durch die Änderung von Vorschriften.

Aus Sicht einer Domäne ist es demnach von Vorteil, zwischen den Ausprägungen *intentionally good-natured*, *accidental* und *intentionally malicious* unterscheiden zu können [HP11]. Einerseits kann unabsichtliches Fehlverhalten mit geeigneten Maßnahmen verhindert werden (s. Abschnitt 4.1), insbesondere wenn dadurch häufige oder besonders schwere Fehlerquellen identifiziert und beseitigt werden [Gre+14]. Andererseits kann die nachträgliche Beurteilung eines Fehlverhaltens der *Intention* entsprechend angepasst werden. Die Auswirkungen eines Angriffs sind allerdings immer unabhängig von der zugrunde liegenden *Intention*. In den meisten Fällen kann die *Intention* nicht ohne die Hilfe zusätzlicher Information, wie zum Beispiel der nachträglichen Verknüpfung mehrerer Aktionen, herausgefunden werden, wenn sie überhaupt herausgefunden werden kann.

Die *Intention* sollte also bei der Charakterisierung einer Insiderbedrohung beziehungsweise eines Insiderbedrohungsagenten in Betracht gezogen werden, besonders wenn es um die nachträgliche Untersuchung eines Vorgang geht. Denn die *Intention* kann die Natur und Intensität möglicher Konsequenzen sowohl für den Insider, der die Aktion durchgeführt hat [Pfl+10] als auch für die Domäne selbst beeinflussen.

3.2.1.4 Entzug und Beständigkeit von Insidergraden

Eine naheliegende Abwehrmaßnahme von Insiderbedrohungen sowie eine natürliche Reaktion auf erfolgreiche Insiderbedrohungsaktionen ist der Entzug von Insidercharakteristiken (vgl. Abschnitt 2.3) beim Insiderbedrohungsagenten. Das liegt insofern nahe, da die Insidercharakteristiken einerseits für die Bedrohungsaktionen notwendig sind oder waren und andererseits von der Domäne kontrolliert werden. Letzteres trifft allerdings nur auf die vier Insidercharakteristiken *Credentials*, *Privileges*, *Trust* sowie *Uncertainty* zu. Sie können einem Individuum durch die Domäne aktiv und mit sofortiger Wirkung entzogen werden. Für die Charakteristik *Knowledge* gilt dies nicht ohne weiteres. Eine solche nicht-zurückziehbare Insidercharakteristik hat offensichtlich ein höheres Bedrohungspotenzial als zurückziehbare, denn einmal bereitgestellt oder gewährt, kann sie nicht mehr kontrolliert werden. Ein Insider, der durch diese Charakteristik definiert wurde, bleibt solange ein Insider, bis diese Charakteristik entweder in Vergessenheit gerät oder nicht mehr den tatsächlichen Zustand einer Domäne widerspiegelt. Der aktive Entzug sowie das Vergessen beziehungsweise Nicht-Widerspiegeln des tatsächlichen Domänenzustandes demonstriert eine Eigenschaft der *bereitgestellten Insidercharakteristiken*, die in der existierenden Literatur bisher nicht identifiziert und benannt wurde. Sie wird nachfolgend als **Lastingness (L)** dieser Insidergrade bezeichnet und findet sich als häufige Sicherheitsmaßnahme in den folgenden Abschnitten 3.2.2 bis 3.2.4. *Lastingness* beschreibt die Tatsache, dass jeder Insidergrad zeitlichen Änderungen unterliegen kann. Ein hoher Grad an *Knowledge* eines Individuums beispielsweise

kann sich aufgrund von Änderungen interner Fakten mit fortschreitender Zeit graduell verringern, sofern keine Einblicke mehr in eine Domäne gewährt werden. Die Einführung von *Lastingness* ist der Versuch, die Zeitkomponente als Präzisierung oder Gewichtung der Insidergrade einzusetzen. Der konkrete Wert kann eines der Elemente *negligible*, *low*, *middle* oder *high* sein.

3.2.2 Unbefugte Weitergabe des Insidergrades

Die unbefugte Weitergabe von bereitgestellten Charakteristiken stellt aus Sicht der Domäne immer eine Insiderbedrohung dar. Etwa im Fall eines K-Insiders kann die Weitergabe von Insiderwissen über geistiges Eigentum und Geschäftsgeheimnisse verheerende Folgen für ein Unternehmen haben [Aft15]. Auf Ebene der Insidercharakteristiken entsprechen derartige Bedrohungsaktionen der Verletzung des Schutzziels *Vertraulichkeit* (vgl. Abschnitt 3.2.1.2), wobei die bereitgestellten Insidercharakteristiken Gegenstand des Vertraulichkeitsbereiches sind und der Insider als legitimer Empfänger der Charakteristiken Teil dieses Vertrauensbereichs wird. Die Domäne hat auf die direkte oder indirekte Weitergabe jeglicher Insidergrade von einem Insider zu einer dritten Partei keinen Einfluss. Dadurch kann die Weitergabe an sich nicht verhindert werden. Mögliche Schutzmaßnahmen beziehen sich daher nur auf die Erschwerung vorbereitender Maßnahmen zur Weitergabe, auf die Erkennung der unbefugten Weitergabe post-mortem sowie auf angemessene reaktive Maßnahmen (vgl. Abschnitt 1.5.4). Bezogen auf die empfangende dritte Partei stellt die nachfolgend im Detail aufgeschlüsselte Klasse von Insiderbedrohungen eine unbefugte Erhöhung des Insidergrades dar, die in den Abschnitten 3.2.3 und 3.3.2 behandelt wird.

3.2.2.1 Weitergabe von Credentials

Die Zugehörigkeit zu einer Domäne inklusive des Nachweises der Identität eines Individuums kann durch die folgenden Typen von *Credentials* erbracht werden [GGF17]:

- etwas, das das Individuum weiß (Kenntnis),
- etwas, das das Individuum hat (Besitz),
- etwas, das das Individuum ist (Vorhandensein) oder
- eine Kombination dieser Typen von *Credentials* (Mehrfaktor-Authentisierung; engl. multi-factor authentication).

Alle Arten der *Credentials*, also Kenntnis (zum Beispiel ein Passwort), Besitz (zum Beispiel ein Sicherheitstoken) und Vorhandensein (zum Beispiel ein biometrischer Fingerabdruck), können indirekt weitergegeben werden, indem sie für die Ziele oder auf die Anweisungen einer dritten Partei hin vom Insiderbedrohungsagenten eingesetzt werden. Es handelt sich dabei um eine logische Weitergabe der Domänenzugehörigkeit, da der C-Insider als Handlanger der dritten Partei agiert. Darüber hinaus können Kenntnis und Besitz als Arten von *Credentials* direkt an eine dritte Partei weitergegeben werden. Die dritte Partei wird in allen Fällen zu einem C_M-Insider. Das Ausführen von Schadsoftware, wie etwa ein Trojanisches Pferd, fällt ebenfalls in diese Bedrohungskategorie, da die Schadsoftware mithilfe der *Credentials* des C_A-Insiders unter dessen Identität läuft.[3] Weniger offensichtlich ist die Weitergabe der Domänenzugehörigkeit

3. In diesem Beispiel läuft die Schadsoftware auch mit den *Privileges* des C_A-Insider. Dabei handelt es sich allerdings um den häufigen Nebeneffekt, dass *Privileges* mit den *Credentials* verknüpft sind und eine Weitergabe der *Credentials* auch eine Weitergabe der *Privileges* zur Folge hat.

durch die fahrlässige Verwendung schwacher *Credentials*. Im Fall eines schwachen Passwortes zum Beispiel kommt eine dritte Partei äußerst leicht durch Raten und Ausprobieren in dessen Besitz und kann so als C_M-Insider auftreten.

Sicherheitsmaßnahmen

Da es sich bei *Credentials* um eine zurückziehbare Charakteristik handelt (vgl. Abschnitt 3.2.1.4), kann das nachträgliche Erkennen der unbefugten Weitergabe zum Entzug der *Credentials* durch die Domäne führen. Eine Domäne kann demnach aktiv nach gestohlenen und zur Weitergabe angebotenen *Credentials* suchen und auf solche entsprechend reagieren.

Darüber hinaus sind alle Maßnahmen, welche die Durchführung von schadhaften Interaktionen mit der Domäne unter falscher Identität erkennen, geeignete detektive Sicherheitsmaßnahmen. Sie können einerseits weitere Bedrohungen erkennen und abwehren sowie andererseits ein starkes Indiz für eine bereits erfolgte, bewusste oder unbewusste Weitergabe von *Credentials* liefern und zum Entzug führen. Dazu zählen auch Verfahren zur Erkennung von Anomalien im Verhalten authentifizierter Individuen.

Weiterhin kann die Domäne versuchen, selbst als dritte Partei aufzutreten und an die *Credentials* des eigenen Insiders zu gelangen, während dieser sie gewollt oder ungewollt veräußert.

3.2.2.2 Weitergabe von Knowledge

Insiderwissen über Einblicke in eine Domäne kann unter anderem die folgenden Formen annehmen:

- Betriebsgeheimnisse und geistiges Eigentum,
- Aufbau und Beschaffenheit der (IT-) Infrastruktur,
- im Einsatz befindliche Produkte, etwa Hard- und Software,
- Strukturen, Prozesse und Dienste,
- Konventionen und ungeschriebene Gesetze sowie
- Informationen über Ressourcen und deren Inhalte.

Die unbefugte Weitergabe von *Knowledge* wird häufig auch als *Datenabfluss* (engl. data leackage) bezeichnet [SER12, Kapitel 2]. Die dritte Partei wird dadurch ebenfalls zu einem K-Insider. Da es sich bei *Knowledge* um eine nicht-zurückziehbare Charakteristik handelt (vgl. Abschnitt 3.2.1.4), kann das Erkennen einer bereits vergangenen unbefugten Weitergabe nur bedingt zu reaktiven Maßnahmen führen.

Sicherheitsmaßnahmen

Techniken zur Erkennung und Abwehr von Datenabflüssen (engl. data leackage detection / prevention [SER12, Kapitel 4]) beschäftigen sich mit Mechanismen, die sensible Daten in einem System identifizieren und deren Bewegungen überwachen beziehungsweise stoppen, sobald unerwünschte Bewegungen erkannt werden, beispielsweise über Systemgrenzen hinweg. Sie können die unerlaubte Weitergabe von *Knowledge* letztlich nicht lückenlos erkennen oder verhindern, da nicht-technische Wege der Wissensweitergabe existieren. Sie können aber im Unterschied dazu technische Vorbereitungen beziehungsweise Unterstützungen zur unerlaubten

Weitergabe, etwa das Kopieren von Dokumenten auf ein externes Speichermedium, erkennen und stoppen.

Auf einer nicht-technischen Ebene kann eine Domäne darüber hinaus verschiedene Versionen einer spezifischen Information für verschiedene Individuen bereitstellen und gleichzeitig dokumentieren, welche Version an welchen K-Insider weitergegeben wurde. Sofern diese Information außerhalb der Domäne auftaucht, kann anhand der Version festgestellt werden, welches Individuum die Quelle dieser unbefugten Weitergabe war.

Weiterhin kann die Domäne versuchen, selbst als dritte Partei aufzutreten und an das Insiderwissen des eigenen Insiders zu gelangen, während dieser es gewollt oder ungewollt veräußert.

3.2.2.3 Weitergabe von Privileges

Privileges erlauben dem Besitzer, mit Ressourcen einer Domäne zu interagieren. Beispiele dafür sind sehr domänenspezifisch:

- Lese-, Schreib- und Ausführungsrechte von Dateien im Fall eines Computersystems,
- die Zugriffs, Nutzungs- und Transporterlaubnis von Wirtschaftsgütern im Fall eines Unternehmens oder
- der Zugriff auf Wartungsschnittstellen sowie die Abfrage und Analyse von Ereignissen im Fall eines Rechnernetzes.

Für die unbefugte Weitergabe von *Privileges* durch einen Insider werden weitere Insidergrade benötigt. Sie kann indirekt durch die Weitergabe von *Credentials* geschehen, wobei dies zwei Formen annehmen kann:

1. Durch die physische unbefugte Weitergabe der *Credentials* wird die empfangende dritte Partei befähigt, sich gegenüber der Domäne als Insider zu authentisieren.

2. Der P-Insider agiert bewusst oder unbewusst als Handlanger einer dritten Partei und setzt die Zugangs- und Zugriffsberechtigungen zu beziehungsweise auf Ressourcen einer Domäne für die Ziele oder auf die Anweisungen der dritten Partei hin ein. Es handelt sich dabei um eine logische Weitergabe der Berechtigungen.

Es kann aber auch spezielles Insiderwissen, wie man unbefugt an *Privileges* gelangt, weitergegeben werden. Ein Beispiel dafür ist die Kenntnis über die Art und Weise einer Schwachstelle in der Domäne, dessen Ausnutzung zu den Berechtigungen führt. So werden etwa durch die Weitergabe von sogenannten Jailbreaks (deut. Gefängnisausbruch) Sicherheitsmechanismen mobiler Plattformen ausgehebelt, wodurch unerlaubte Zugriffe auf Hard- und Softwareteile der Plattform ermöglicht werden.

Hier zeigt sich auch, dass die Bedrohung durch die unbefugten Weitergabe von *Privileges* unmittelbar mit der Bedrohung durch die unbefugte Weitergabe von *Credentials* und spezieller *Knowledge* zusammenhängt.

Sicherheitsmaßnahmen

Da es sich bei *Privileges* um eine zurückziehbare Charakteristik handelt (vgl. Abschnitt 3.2.1.4), kann das nachträgliche Erkennen der unbefugten Weitergabe zum Entzug der *Privileges* durch die Domäne führen. Eine Domäne kann demnach aktiv nach unbefugt weitergegebenen *Privileges* suchen und auf solche entsprechend reagieren.

Zusätzlich zu den Sicherheitsmaßnahmen gegen die unbefugte Weitergabe von *Credentials* und *Knowledge* (vgl. Abschnitte 3.2.2.1 und 3.2.2.2) kann die Protokollierung und Prüfung der Zugriffe eines P-Insiders auf die Ressourcen einer Domäne Anomalien in Zugriffsmustern erkennbar machen und somit unbefugt erhaltene *Privileges* aufzeigen.

3.2.2.4 Weitergabe von Trust

Trust erlaubt dem Besitzer, Unterstützung bei der Erreichung seines Ziels durch andere Insider zu erhalten, die durch ihr entgegengebrachtes persönliches Vertrauen eher bereit sind, von formalen Richtlinien und Protokollen abzuweichen. Beispiele für derartiges Vertrauen sind:

- Private freundschaftliche oder langjährige Beziehungen,
- Vertrauen in die Kompetenz einer Person durch dessen Funktionsbeschreibung oder Rolle in einer anderen Domäne oder
- hierarchiebedingter Gehorsam.

Trust kann ausschließlich indirekt unbefugt Weitergegeben werden, da es an die Identität des Besitzers geknüpft ist. Diese Indirektheit kann zwei Formen annehmen:

1. Durch die physische unbefugte Weitergabe der *Credentials* wird die empfangende dritte Partei befähigt, sich gegenüber der Domäne als Insider zu authentisieren.

2. Der T-Insider agiert bewusst oder unbewusst als Handlanger einer dritten Partei und setzt den persönlichen Vertrauensvorschuss einer Domäne für die Ziele oder auf die Anweisungen der dritten Partei hin ein. Es handelt sich dabei um eine logische Weitergabe des Vertrauensvorschusses.

Andere Weitergaben von *Trust* können aus Sicht einer Domäne nicht als unbefugt eingestuft werden. Insbesondere das Aufbauen von persönlichem Vertrauen eines Insiders zu einer unbeteiligten dritten Person kann nicht unter eine unbefugte Weitergabe von *Trust* fallen.

Sicherheitsmaßnahmen

Da es sich bei *Trust* um eine zurückziehbare Charakteristik handelt (vgl. Abschnitt 3.2.1.4), kann das nachträgliche Erkennen der unbefugten Weitergabe zum Entzug von *Trust* durch die Domäne führen. Im 1. Fall der unbefugten Weitergabe von *Trust* greifen alle Sicherheitsmaßnahmen gegen die unbefugte Weitergabe von *Credentials* aus Abschnitt 3.2.2.1. Für den 2. Fall ist insbesondere die Erkennung von Anomalien im Verhalten des Insiders und Auffälligkeiten in dessen Interaktionen mit der Domäne hervorzuheben.

Darüber hinaus kann das Bewusstsein von Insidern für derartige Bedrohungen geschaffen und geschärft werden, sodass die unwissentliche Weitergabe von *Trust* reduziert wird.

3.2.2.5 Weitergabe von Uncertainty

Uncertainty ist der Grad der Erlaubnis oder Toleranz von Aktionen eines Individuums in Abwesenheit von Sicherheits- und Überprüfungsmechanismen. Beispiele für derartiges strukturelles Vertrauen sind:

- fehlende Überwachung von Zutritten beziehungsweise Zugängen zu Systemteilen einer Domäne,
- fehlende Erzeugung und Archivierung von potenziellem Beweismaterial im Fall unbefugter Aktivitäten sowie
- fehlende Kontrolle der Konformität zu Protokollen, Regularien und Gesetzen.

Äquivalent zu *Trust* kann *Uncertainty* ebenfalls indirekt weitergegeben werden, da es an die Identität eines Individuums geknüpft ist und dadurch mit der Weitergabe von *Credentials* zusammenhängt. Diese Indirektheit kann daher ebenso zwei Formen annehmen:

1. Durch die physische unbefugte Weitergabe der *Credentials* wird die empfangende dritte Partei befähigt, sich gegenüber der Domäne als Insider zu authentisieren.

2. Der U-Insider agiert bewusst oder unbewusst als Handlanger einer dritten Partei und setzt den strukturellen Vertrauensvorschuss einer Domäne für die Ziele oder auf die Anweisungen der dritten Partei hin ein. Es handelt sich dabei um eine logische Weitergabe von *Uncertainty*.

Weiterhin kann das Insiderwissen über Umgehungsmöglichkeiten von Kontroll- und Überwachungsmechanismen von einem K-Insider unbefugt weitergegeben werden, wodurch ebenfalls die damit in Verbindung stehende *Uncertainty* indirekt weitergegeben wird. Zusätzlich kann ein P-Insider mit den passenden Zugriffsberechtigungen dafür sorgen, dass Kontroll- und Überwachungsmechanismen derart manipuliert werden, dass die *Uncertainty* für eine andere Person erhöht wird.

Sicherheitsmaßnahmen

Da es sich bei *Uncertainty* um eine zurückziehbare Charakteristik handelt (vgl. Abschnitt 3.2.1.4), kann das nachträgliche Erkennen der unbefugten Weitergabe zum Entzug der *Uncertainty* durch die Domäne führen. Die unterschiedlichen Voraussetzungen für die Weitergabe von *Uncertainty* zeigen Wege auf, wie derartige Bedrohungen erkannt und abgewehrt werden können. Im 1. Fall der genannten indirekten Weitergabe von *Uncertainty* durch *Credentials* greifen alle Sicherheitsmaßnahmen gegen diese Bedrohung aus Abschnitt 3.2.2.1. Für den 2. Fall ist insbesondere die Erkennung von Anomalien im Verhalten des Insiders und Auffälligkeiten in dessen Interaktionen mit der Domäne hervorzuheben.

Eine Erkennung und Abwehr von Anomalien bei Interaktionen mit Kontroll- und Überwachungsmechanismen ist ebenfalls eine passende Sicherheitsmaßnahme gegen die Weitergabe von *Uncertainty* durch den Missbrauch von *Privileges*. Die indirekte Weitergabe durch *Knowledge* kann durch die Sicherheitsmaßnahmen gegen diese Bedrohung aus Abschnitt 3.2.2.2 abgewehrt werden.

Tabelle 3.2: Notwendige (☐ und-verknüpft mit ✔) Insidergrade, die für die Weitergabe von Insidercharakteristiken benötigt werden sowie elementare Bedrohungen (vgl. Abschnitt 1.5.5), die eine Weitergabe der Insidercharakteristiken bedeuten können

Weitergabe von	Notwendiger Insidergrad					Elementare Bedrohungen			
	C-Insider	K-Insider	P-Insider	T-Insider	U-Insider				
Credentials	☑					•	•	•	
Knowledge		☑				•	•		•
Privileges	☐	☐	✔				•	•	
Trust	☐			✔			•		
Uncertainty	☐	☑	☑		✔	•	•	•	
						G 0.19 Offenlegung schützenswerter Informationen			
						G 0.29 Verstoß gegen Gesetze oder Regelungen			
						G 0.32 Missbrauch von Berechtigungen			
						G 0.38 Missbrauch personenbezogener Daten			

3.2.2.6 Qualitative Auswertung der Bedrohungen durch die Weitergabe des Insidergrades

Die herausgearbeiteten Beobachtungen in den Abschnitten 3.2.2.1 bis 3.2.2.5 zeigen auf, welche Voraussetzungen für eine Weitergabe der einzelnen Insidercharakteristiken erfüllt sein müssen. Die gewonnenen Erkenntnisse sind in Tabelle 3.2 zusammengefasst.

Darüber hinaus kann die unbefugte Weitergabe verschiedener Insidercharakteristiken je nach spezieller Ausprägung unmittelbar mit einzelnen elementaren Bedrohungen in Verbindung gebracht werden, die in Abschnitt 1.5.5 eingeführt wurden. So bedeutet die Weitergabe von sensiblen internen Informationen oder Geschäftsgeheimnissen (*Knowledge*) sowie die Weitergabe von Kenntnis oder Besitz als Arten der Zugehörigkeit zu einer Domäne (*Credentials*) eine Offenlegung schützenswerter Informationen (G 0.19). Jegliche Weitergabe eines Insidergrades bedeutet einen Verstoß gegen Gesetze oder Regelungen (G 0.29). Die Weitergabe von Zutrittsberechtigungen (*Credentials*) sowie von Zugangs- und Zugriffsberechtigungen (*Privileges*) ist gleichbedeutend mit einem Missbrauch von Berechtigungen (G 0.32). Und die unerlaubte Weitergabe von persönlichen Informationen über andere Personen (*Knowledge*) bedeutet einen Missbrauch personenbezogener Daten (G 0.38). Tabelle 3.2 fasst auch diese Beobachtungen zusammen.

3.2.3 Unbefugte Eskalation des Insidergrades

Die unbefugte Eskalation von Insidercharakteristiken bedeutet für eine Domäne nicht ausschließlich eine Insiderbedrohung, da genau diese Art der Bedrohung auch von Outsidern ausgehen kann (s. Abschnitte 3.3.1 und 3.3.2). Für die Eskalationen von *Privileges* und *Uncertainty* ist allerdings ein bereits vorhandener Insidergrad zwingende Voraussetzung. Auf diese soll im Folgenden genauer eingegangen werden. Die restlichen Insidercharakteristiken, also *Credentials*,

Knowledge und *Trust*, können sowohl von Insidern als auch von Outsidern eskaliert werden, sodass in diesen Fällen eventuell vorhandene Insidercharakteristiken eine reine Verbesserung einer Eskalation bedeuten und somit unter die Betrachtungen in Abschnitt 3.2.5 fallen.

Auf der Ebene der Insidercharakteristiken bedeutet die unbefugte Eskalation von Insidergraden eine Verletzung der *Integrität*, da die von einer Domäne ausgehende Bereitstellung von Insidercharakteristiken unbefugt modifiziert wird. Bezogen auf die dritte Partei, die unbefugt die Insidercharakteristiken bereitstellt, bedeutet diese Bedrohung eine unbefugte Weitergabe des Insidergrades (vgl. Abschnitt 3.2.2). Diese Weitergabe passiert in manchen Fällen nicht bewusst und auch nicht direkt, sondern unbewusst und/oder indirekt.

3.2.3.1 Eskalation von Privileges

Mögliche Voraussetzungen, die *Privileges* unbefugt zu erhöhen, lassen sich folgendermaßen differenzieren:

1. Durch die vorherige Aneignung oder Überlistung einer fremden Domänenzugehörigkeit (*Credentials*), deren Grad an *Privileges* höher ist als der eigene,
2. durch das Ausnutzen von Wissen über Schwachstellen und Konfigurationsfehler (*Knowledge*), die mehr *Privileges* zur Verfügung stellen oder
3. durch das Hervorrufen von individuellem Fehlverhalten anderer P-Insider, etwa das Abweichen von Vorschriften (*Trust*).

Eine erfolgreiche *Privileges*-Eskalation hat vielfältige negative Auswirkungen auf eine Domäne. Einerseits können Angriffe, die vorher unmöglich waren, dadurch ermöglicht werden. Andererseits können weitere bereitgestellte Insidercharakteristiken eskaliert werden, was insbesondere im Fall von *Uncertainty* dazu führen kann, dass ein Insiderangreifer zusätzliche Möglichkeiten erhält, die eigenen Spuren und diejenigen des Insiderangriffs zu verschleiern oder ganz zu entfernen.

Sicherheitsmaßnahmen

Effektive Sicherheitsmaßnahmen umfassen die Abwehr der genannten Voraussetzungen für die Eskalation von *Privileges*, also der Schutz vor 1. der Eskalation von *Credentials*, 2. potenziell ausnutzbaren Schwachstellen sowie 3. einem Missbrauch von *Trust*.

Weiterhin muss die Bereitstellung von *Privileges* durch eine Domäne derart sorgfältig geprüft und dokumentiert werden, dass eine unbeabsichtigte Bereitstellung ausgeschlossen wird und nachvollziehbar ist, welche erlaubten Interaktionen mit der Domäne für ein Individuum notwendig sind. Dadurch können angemessene und robuste Zugriffskontrollmechanismen etabliert werden. Die fortwährende Überwachung von Zugriffen auf Ressourcen einer Domäne erlaubt zusätzlich den Abgleich mit der dokumentierten Bereitstellung von *Privileges* und damit die Kenntnisnahme von Eskalationen.

Ein weiterer Sicherheitsmechanismus geht aus der Überlegung hervor, dass der Grad der erlaubten Interaktionen mit den Ressourcen einer Domäne nicht unbedingt statisch sein muss. Die erlaubten Interaktionen können damit, an verschiedene Bedingungen geknüpft, unterschiedliche Ausprägungen aufweisen. Etwa die zuvor getätigten Interaktionen mit Ressourcen oder eine

Degradierung in einem zeitlichen Verlauf mit der Möglichkeit der Auffrischung sind Modelle, die bei sorgfältiger Umsetzung einen Sicherheitsgewinn ergeben können. Sie können somit auf die Reduktion der *Lastingness* der *Privileges* abzielen.

3.2.3.2 Eskalation von Uncertainty

Wird die Verfügbarkeit von Überprüfungs-, Überwachungs- und Dokumentationsmechanismen verletzt, lässt sich dadurch eine Eskalation von *Uncertainty* erreichen. Dafür muss allerdings entweder Wissen über deren Existenz, Konfiguration und Schwachstellen sowie deren Ausnutzung (*Knowledge*) oder Zugriffsrechte zur Manipulation der Mechanismen und zum Entfernen von bestimmten Daten (*Privileges*) vorhanden sein. Darüber hinaus können diese Mechanismen mithilfe der Imitation einer fremden Identität (*Credentials*) in die Irre geführt werden.

Die Bedrohung, die von einer Eskalation der *Uncertainty* ausgeht, bezieht sich vor allem auf die Wahrscheinlichkeit, mit der eine nachgelagerte Bedrohung unentdeckt bleibt beziehungsweise mit der ein weiterer Insiderangreifer unerkannt seinen Angriff ausführen kann. Damit hat es direkte Auswirkungen auf alle anderen Insiderbedrohungen.

Sicherheitsmaßnahmen

Mithilfe von Techniken der Ausfallsicherheit und der Resilienz können die Bedrohungen der Verfügbarkeit von Überwachungsmechanismen abgemildert werden. Dazu zählen vor allem die redundante und diversitäre Auslegung von Systemkomponenten sowie die ausreichende Bereitstellung benötigter Ressourcen [FP97].

Mit der Umsetzung des *Vier-Augen-Prinzips* bei administrativen Interaktionen mit Kontroll- und Überwachungsmechanismen werden Manipulationen an den Mechanismen und den zugehörigen Daten erheblich erschwert.

Uncertainty sollte so niedrig wie möglich gehalten werden und besonderer Fokus sollte auf der ungewollten Bereitstellung von *Uncertainty* liegen. Weiterhin kann das sogenannte *Veracity-Prinzip* verfolgt werden [Gol12], bei dem die Manipulation oder Umgehung von Überprüfungs- und Überwachungsmechanismen wiederum Spuren hinterlässt, die auswertbar sind und auf derartige Interaktionen hinweisen.

Da Insiderwissen oder Zugriffsberechtigungen Voraussetzungen für diese Bedrohung sind, könnten *Knowledge*- oder *Privileges*-Eskalationen vorausgehen, die bereits erkannt und abgewehrt werden können.

3.2.3.3 Qualitative Auswertung der Bedrohungen durch die Eskalation des Insidergrades

Die herausgearbeiteten Beobachtungen in den Abschnitten 3.2.3.1 und 3.2.3.2 zeigen auf, welche Voraussetzungen für eine Eskalation der Insidercharakteristiken *Privileges* und *Uncertainty* erfüllt sein müssen. Die gewonnenen Erkenntnisse sind in Tabelle 3.3 zusammengefasst.

Darüber hinaus kann die unbefugte Eskalation der Insidercharakteristiken *Privileges* und *Uncertainty* je nach den speziellen Umständen unmittelbar mit einer Reihe von elementaren Bedrohungen in Verbindung gebracht werden, die in Abschnitt 1.5.5 eingeführt wurden. So kann die

Tabelle 3.3: Notwendige (☑) Insidergrade, die für die Eskalation der Insidercharakteristiken *Privileges* und *Uncertainty* benötigt werden sowie elementare Bedrohungen (vgl. Abschnitt 1.5.5), die eine Eskalation der Insidercharakteristiken bedeuten können

Eskalation von	Notwendiger Insidergrad					Elementare Bedrohungen										
	C-Insider	K-Insider	P-Insider	T-Insider	U-Insider											
Privileges	☑	☑		☑		•		•	•	•	•	•		•		•
Uncertainty	☑	☑	☑			•	•	•	•	•	•	•	•	•	•	

G 0.21 Manipulation von Hard- oder Software
G 0.22 Manipulation von Informationen
G 0.23 Unbefugtes Eindringen in IT-Systeme
G 0.29 Verstoß gegen Gesetze oder Regelungen
G 0.30 Unberechtigte Nutzung o. Administration von Geräten u. Systemen
G 0.32 Missbrauch von Berechtigungen
G 0.36 Identitätsdiebstahl
G 0.37 Abstreiten von Handlungen
G 0.39 Schadprogramm
G 0.40 Verhinderung von Diensten (Denial of Service)
G 0.42 Social Engineering

Manipulation von Informationen (G 0.22) durch einen P-Insider eine Eskalation der *Uncertainty* bedeuten. Jegliche unbefugte Eskalation von Insidercharakteristiken ist gleichbedeutend mit einem Verstoß gegen Gesetzte und Regelungen (G 0.29). Die Eskalation von *Privileges* sowie von *Uncertainty* durch den Missbrauch von fremden *Credentials* oder die Eskalation von *Privileges* durch die Überlistung eines P-Insiders mithilfe von *Trust* ist gleichbedeutend mit einer unberechtigten Nutzung oder Administration von Geräten und Systemen (G 0.30). Diese und weitere Beobachtungen fasst ebenfalls Tabelle 3.3 zusammen.

3.2.4 Unbefugte Verhinderung des Insidergrades

Bei der unbefugten Verhinderung von Insidergraden geht es um die Verringerung von Insidercharakteristiken. Durch die Verletzung der *Verfügbarkeit* auf der Ebene der Insidercharakteristiken können anschließend wichtige Aufgaben nicht mehr erfüllt und wichtige Dienste nicht mehr erbracht werden. Dadurch entsteht der Domäne oder einzelnen Insidern ein Nachteil beziehungsweise ein Schaden, der bei dieser Art von Bedrohungen im Fokus steht.

Zwei Besonderheiten stellen die Insidercharakteristiken *Knowledge* und *Uncertainty* dar. *Knowledge* kann nur dann verhindert werden, sofern sie ausgelagert auf Speichermedien vorliegt und der Zugriff auf diese Speichermedien essenziell für den Insidergrad durch *Knowledge* ist. Dadurch wird die Verhinderung von *Knowledge* nur in Verbindung mit der Verhinderung von *Privileges* möglich. Die Verhinderung von *Uncertainty* stellt aus Sicht der Domäne keine Bedrohung dar.[4] Daher wird die Verhinderung dieser beiden Insidercharakteristiken nachfolgend nicht näher betrachtet.

3.2.4.1 Verhinderung von Credentials

Alle Bedrohungen durch Verhinderung von *Credentials* können nur von einem P-Insider durchgeführt werden, weshalb häufig Bedrohungen durch die Eskalation von *Privileges* vorausgehen.

Das Entwenden oder Zerstören derjenigen *Credentials*, die auf dem Besitz beruhen (zum Beispiel ein Schlüssel oder Sicherheitstoken),[5] verhindert den Insidergrad eines C_A-Insiders, sofern dieser keinen Zugriff mehr darauf hat. Darüber hinaus können die *Credentials* durch eine Manipulation der Zutritts- beziehungsweise Zugangsmechanismen verhindert werden, da so die Domäne keinen Weg mehr hat, einen C_A-Insider zu authentifizieren. Ein Beispiel ist die Änderung eines Nutzerpasswortes, nachdem sich ein Angreifer die Nutzerdaten angeeignet und dadurch die nötigen *Privileges* zum Ändern des Passwortes erhalten hat.

Sicherheitsmaßnahmen

Eine häufige Praxis ist die Rückmeldung einer durch den authentifizierten Nutzer angestoßenen Änderung der *Credentials* über einen hinterlegten Kommunikationsweg, wie etwa eine E-Mail-Adresse. Sofern der legitime Besitzer der *Credentials* diese Änderung nicht veranlasst hat, ist von einer Verhinderung der *Credentials* durch einen Insiderbedrohungsagenten auszugehen.

4. Hier wird davon ausgegangen, dass eine Ausweitung von Kontroll- und Überwachungsmechanismen keine Verletzung von bestehenden gesetzlichen Regelungen bedeutet.
5. Die *Credentials*, die auf der Kenntnis oder dem Vorhandensein beruhen, können nicht entwendet beziehungsweise zerstört werden.

Die Verhinderung wird dadurch zunächst nicht abgewehrt, aber geeignete reaktive Maßnahmen können ergriffen werden.

Entsprechend der benötigten *Privileges* bei der Verhinderung von *Credentials* helfen Mechanismen der Anomalieerkennung und -abwehr im Zugriffsverhalten von P-Insidern auf Ressourcen einer Domäne beim Schutz vor derartigen Bedrohungen.

3.2.4.2 Verhinderung von Privileges

Alle Bedrohungen durch die Verhinderung von *Privileges* können nur von einem P-Insider durchgeführt werden, weshalb häufig Bedrohungen durch die Eskalation von *Privileges* vorausgehen.

Der Entzug von Zugriffsrechten auf Ressourcen einer Domäne beziehungsweise die Zerstörung der Ressourcen selbst bedeutet eine Verhinderung von *Privileges*, sofern die Zugriffsrechte beziehungsweise die Ressourcen nicht rekonstruiert werden können. Diese Bedrohung macht sich beispielsweise sogenannte Ransomware (deut. Erpressungssoftware) zunutze, um im Anschluss an die Verhinderung von *Privileges* ein Lösegeld für gestohlene Daten zu verlangen.

Sicherheitsmaßnahmen

Ein wirksamer Schutz vor der Verhinderung von *Privileges* ist die Möglichkeit der Rekonstruktion wichtiger Ressourcen, die im Fall von kopierbaren Ressourcen etwa in Form von Backups realisiert werden kann.

Entsprechend der benötigten *Privileges* bei der Verhinderung von *Credentials* helfen Mechanismen der Anomalieerkennung und -abwehr im Zugriffsverhalten von P-Insidern auf Ressourcen einer Domäne beim Schutz vor derartigen Bedrohungen.

3.2.4.3 Verhinderung von Trust

Die Möglichkeiten der Verhinderung von *Trust* umfassen das Sähen von Zwietracht und die Verbreitung von Falschinformationen. Dadurch geht eine Domäne davon aus, dass sie sich nicht mehr auf einen T-Insider verlassen kann. Voraussetzung dafür können neben benötigten *Privileges* auch *Knowledge* sein, um Informationen zu manipulieren.

Sicherheitsmaßnahmen

In den letzten Jahren haben sich Forscher mit dem Thema der Erkennung von sogenannten Falschnachrichten (engl. fake news) insbesondere in sozialen Netzwerken befasst [ZZ18]. Entwickelte Ansätze umfassen wissensbasierte Faktenchecks sowie stilbasierte Merkmalextraktion in Verbindung mit Methoden des maschinellen Lernens, um Texte mit falschen Nachrichten automatisiert zu identifizieren.

Um die benötigten Voraussetzungen zur Verhinderung von *Trust* zu erreichen, können eventuell Bedrohungen der *Privileges*- oder *Knowledge*-Eskalation vorausgehen, die mit entsprechenden Abwehrmaßnahmen bereits verhindert werden könnten.

Tabelle 3.4: Notwendige (☑) Insidergrade, die für die Verhinderung von Insidercharakteristiken benötigt werden sowie elementare Bedrohungen (vgl. Abschnitt 1.5.5), die eine Verhinderung der Insidercharakteristiken bedeuten können. *Knowledge* kann von einem P-Insider nur in speziellen Fällen (O) verhindert werden.

Verhinderung von	Notwendiger Insidergrad					Elementare Bedrohungen					
	C-Insider	K-Insider	P-Insider	T-Insider	U-Insider						
Credentials			☑			•		•	•		
Knowledge			O				•	•	•	•	•
Privileges			☑			•		•	•	•	•
Trust		☑	☑				•	•	•		

G 0.21 Manipulation von Hard- oder Software
G 0.22 Manipulation von Informationen
G 0.29 Verstoß gegen Gesetze oder Regelungen
G 0.32 Missbrauch von Berechtigungen
G 0.40 Verhinderung von Diensten (Denial of Service)
G 0.41 Sabotage

3.2.4.4 Qualitative Auswertung der Verhinderung des Insidergrades

Die herausgearbeiteten Beobachtungen in den Abschnitten 3.2.4.1 bis 3.2.4.3 zeigen auf, welche Voraussetzungen für eine Verhinderung der einzelnen Insidercharakteristiken erfüllt sein müssen. Die gewonnenen Erkenntnisse sind in Tabelle 3.4 zusammengefasst.

Darüber hinaus kann die unbefugte Verhinderung verschiedener Insidercharakteristiken je nach spezieller Ausprägung unmittelbar mit einzelnen elementaren Bedrohungen in Verbindung gebracht werden, die in Abschnitt 1.5.5 eingeführt wurden. So kann eine Manipulation von Hard- oder Software (G 0.21) eine Verhinderung von *Credentials* hervorrufen. Dies ist genau dann der Fall, wenn durch die Manipulation der Authentifizierungsmechanismus einer Domäne entsprechend beeinträchtigt wird. Auch die Verhinderung von Zugriffsrechten (*Privileges*) kann mit dieser elementaren Bedrohung in Verbindung gebracht werden. Wohingegen die Manipulation von Informationen (G 0.22) Falschinformationen erzeugen und verbreiten könnte, was die Reputation und Glaubwürdigkeit (*Trust*) anderer Personen negativ beeinflussen kann. In Tabelle 3.4 sind auch diese und weitere Beobachtungen zusammengefasst.

3.2.5 Durch einen Insidergrad verbesserte Bedrohungen

Alle in Definition 3.2 beschriebenen Klassen von Insiderbedrohungen können durch spezielle Insidergrade beziehungsweise durch das Vorhandensein von Insidercharakteristiken nicht nur entstehen, sondern auch verbessert werden. Mögliche Verbesserungen von konkreten Bedrohungsaktionen mithilfe des Insidergrades eines Individuums lassen sich wie folgt einordnen:

Definition 3.3 (Verbesserungen von Insiderbedrohungen)**.** Das Vorhandensein von Insidercharakteristiken verbessert eine Bedrohungsaktion genau dann, wenn

a) die Möglichkeiten der Bedrohungsprävention verringert werden oder, umgekehrt formuliert, die Wahrscheinlichkeit des Erfolgs der Bedrohungsaktion erhöht wird (**Erfolg**),

b) die Erkennbarkeit der Bedrohungsaktion beziehungsweise des Bedrohungsagenten verringert und damit die Verdecktheit erhöht wird (**Verdecktheit**),

c) die Auswirkungen der Bedrohungsaktion auf die Domäne erhöht werden (**Auswirkung**).

Schutzmaßnahmen müssen demnach immer verringernde Einwirkungen auf die Verbesserungen von Insiderbedrohungen haben. Die nachfolgenden Abschnitte führen diese Erkenntnisse mit Blick auf die verschiedenen Insidercharakteristiken weiter aus. Weiterhin wird nachfolgend davon ausgegangen, dass es sich bei einer Bedrohung um einen Insiderangriff handelt und die Verbesserungen durch vorhandene Insidergrade gezielt eingesetzt werden.

3.2.5.1 Verbesserungen durch Credentials

Valide *Credentials* befähigen einen Angreifer, innerhalb des Perimeters einer Domäne zu agieren und sich beziehungsweise eine behauptete Identität gegenüber der Domäne auszuweisen. Bezogen auf die in Definition 3.3 beschriebenen Verbesserungen von Insiderbedrohungen können *Credentials* bei einem Angriff wie folgt aufgeschlüsselt werden:

- **Erfolg**: Ein vormals unmöglicher Angriff kann durch den Einsatz von validen *Credentials* zum Erfolg führen. Das ist zum Beispiel dann der Fall, wenn die *Credentials* das Umgehen von Perimeterschutzmaßnahmen erlauben, die diesen Angriff normalerweise zuverlässig verhindern würden. Auch die Eskalationen von *Trust* und von *Privileges* werden durch den Nachweis einer speziellen Identität begünstigt. Das machen sich beispielsweise *Social-Engineering-Angriffe* zunutze, die häufig subtile Methoden der Authentisierung ausnutzen. Der Urheber einer elektronischen Nachricht wird in der Regel dadurch beim Empfänger authentifiziert, weil er sich zuvor gegenüber dem Kommunikationsdienst authentisiert haben muss, um die Nachricht abzusetzen. Begünstigt wird dies durch den Umstand, dass sich zusätzlich Nachrichten über diesen Kommunikationsweg in der Vergangenheit als authentisch erwiesen haben. Sobald also *Credentials* vorliegen, mit denen dieser Kommunikationsweg benutzt werden kann, können dadurch *Trust* und *Privileges* mit einer höheren Erfolgswahrscheinlichkeit eskaliert werden.
- **Verdecktheit**: Ebenfalls gebunden an *Credentials* ist der Grad an strukturellem Vertrauen (*Uncertainty*), der festlegt, in wie weit die Legitimität der Interaktionen mit einer Domäne mithilfe von Überwachungs- und Prüfmechanismen überprüft werden kann. Die Nutzung von geeigneten *Credentials* kann demnach die Verdecktheit erheblich erhöhen. Darüber hinaus erlaubt die *Credentials*-Eskalation die Vortäuschung einer anderen Identität. Sofern also Spuren eines solchen C_M-Insiderangriffs währen der Ausführung oder im Nachhinein gefunden werden, deuten diese auf einen falschen Urheber hin, wodurch der Angreifer in der Regel ebenfalls eine sehr hohe Verdecktheit erreichen kann.
- **Auswirkung**: Die Auswirkungen einer Bedrohung werden durch den Einsatz von validen *Credentials* nicht beeinflusst.

Eine Besonderheit besteht darin, dass die von einer Domäne bereitgestellten Charakteristiken *Privileges*, *Trust* sowie *Uncertainty* häufig direkt oder indirekt mit der Verwendung von *Credentials* und dem damit verbundenen Identitätsnachweis verknüpft sind. Durch diese Verknüpfungen können weitere Verbesserungen von Insiderbedrohungen erfolgen, die in den Abschnitten 3.2.5.3 bis 3.2.5.5 noch genauer beleuchtet werden.

Sicherheitsmaßnahmen

Credentials sind zurückziehbar (vgl. Abschnitt 3.2.1.4), indem der Authentifizierungsmechanismus einer Domäne derart modifiziert wird, dass diese *Credentials* nicht mehr akzeptiert werden. Somit ist die Prüfung auf den Missbrauch von *Credentials* und deren anschließender Entzug ein effektiver Schutz vor den Verbesserungen von Bedrohungen durch den Einsatz von *Credentials*. Eine solche Prüfung kann beispielsweise in Form einer Anomalieerkennung des Authentisierungsverhaltens von Individuen erfolgen. Damit kann die missbräuchliche Verwendung von gestohlenen *Credentials* erkannt und diese entsprechend entzogen werden. Auch das abweichende Verhalten eines C_A-Insiders kann erkannt und zumindest genauer untersucht werden. Weitere Maßnahmen zur Reduktion der *Uncertainty*, etwa die Aufzeichnung und Korrelation von technisch erfassbaren Authentifizierungsereignissen und die anschließende regelbasierte Erzeugung von Sicherheitsmeldungen, können die Verbesserungen von Insiderbedrohungen eines C-Insiders senken. Die gleichen Sicherheitsmaßnahmen sind auch auf der Ebene der Interaktionen eines Individuums mit einer Domäne sinnvoll. Erkenntnisse auf dieser Ebene der *Privileges*, beispielsweise Anomalien im Interaktionsverhalten eines authentifizierten Individuums mit Ressourcen der Domäne, lassen durch die häufige Verknüpfung von *Privileges* mit *Credentials* Rückschlüsse auf den Missbrauch der *Credentials* zu.

Ein möglicher Schutz vor den nachgelagerten Verbesserungen durch die Verknüpfungen zwischen *Credentials* und den Insidercharakteristiken *Privileges*, *Trust* sowie *Uncertainty* besteht in der Aufhebung oder Abschwächung dieser Verknüpfungen. Das National Institute of Standards and Technology (NIST) schlägt beispielsweise Zusicherungsebenen in Verbindung mit unterschiedlichen Kombinationen von *Credentials* vor [Gra+17]. Die Verwendung dieser Zusicherungsebenen könnte dann unterschiedliche Grade an *Privileges*, *Trust* und *Uncertainty* bedingen. Im Fall von *Uncertainty* könnte die Verknüpfung auch insgesamt aufgehoben werden, indem das gleiche Level an *Uncertainty* für alle Individuen einer Domäne gilt.

Weiterhin ist die Reduktion der *Lastingness* (vgl. Abschnitt 3.2.1.4) von *Credentials* ein möglicher Sicherheitsmechanismus. Insbesondere dann, wenn die *Credentials* eines Individuums mit Terminierung der Domänenzugehörigkeit nicht wieder entzogen werden und demnach unnötigerweise weiter bestehen und unbemerkt verwendet werden können. Die Reduktion der *Lastingness* kann beispielsweise durch periodisches Prüfen der Notwendigkeit geschehen oder durch periodisches Ändern jener *Credentials*, die auf der Kenntnis eines Geheimnisses oder dem Besitz eines Authentisierungsgegenstandes beruhen. Diese periodische Änderung von Geheimnissen und insbesondere von Passwörtern ist unter Experten umstritten [ZMR10; Cra16; Sch16; Hun17], da sie in der Regel dazu führt, dass betroffene Individuen insgesamt schwächere Passwörter wählen und einer allgemein vorhersagbaren Regel folgen, um diese vorgegebene Passwortänderung umzusetzen. Eine derartige Vorgabe ist daher sorgfältig abzuwägen. Die Kombination mehrerer Authentisierungsfaktoren kann dabei eine Lösung bieten, bei der neben dem eigentlichen langlebigen Geheimnis beispielsweise ein weiteres Einmalpasswort zur erfolgreichen Authentisierung benötigt wird. In anderen Bereichen ist die periodische Änderung der *Credentials* hingegen obligatorisch und hat sich als durchaus wichtiges Sicherheitsmerkmal erwiesen. Ein gutes Beispiel sind digitale Schlüsselzertifikate [Woh00].

Ebenfalls zur Reduktion der *Lastingness* von *Credentials* trägt die Überlegung bei, dass der Zustand der Domänenzugehörigkeit nicht unbedingt statisch sein muss [Hun17; Gra+17]. Eine Domänenzugehörigkeit kann damit, an verschiedene Bedingungen geknüpft, unterschiedliche Ausprägungen aufweisen. Etwa der Ort, von dem aus eine Authentisierung stattfindet oder die Intensität, mit der diese Authentisierung stattfindet. Auch eine Degradierung in einem zeitlichen

Verlauf mit der Möglichkeit der Auffrischung sind Modelle, die bei sorgfältiger Umsetzung einen Sicherheitsgewinn erzielen und neuartige Sicherheitsmaßnahmen hervorbringen können.

3.2.5.2 Verbesserungen durch Knowledge

Generell gilt, dass nicht jedes Insiderwissen eine Verbesserung für jede Art von Angriffen ermöglicht, da spezielle Kenntnisse möglicherweise keine Relation zu einem Angriff haben. Ein Beispiel dafür sind Kenntnisse über den Aufbau und die Beschaffenheit der IT-Infrastruktur, während der Angreifer aber Papierdokumente entwenden und veräußern möchte. Dennoch ist Insiderwissen ein mächtiges Werkzeug bei der Verbesserung von Angriffen. Ein Beispiel ist ein *Social-Engineering-Angriff*, bei dem in einer E-Mail mit gefälschtem Absender ein Insider dazu aufgefordert wird, unbefugte Interaktionen mit Ressourcen der Domäne auszuführen. Die Einflechtung von speziellem Insiderwissen über Vorgesetztenstrukturen und interne Hierarchien in diese Nachricht lässt sie für das Opfer sehr viel authentischer wirken und erleichtert somit die Eskalation von *Trust*. Durch den Aufbau von arbeitsplatzbezogenem Druck kann nachgelagert gegebenenfalls die Eskalation von *Privileges* erreicht werden.

Bezogen auf die in Definition 3.3 beschriebenen Verbesserungen von Insiderbedrohungen kann *Knowledge* bei einem Angriff wie folgt aufgeschlüsselt werden:

- **Erfolg**: Für die meisten Arten von Angriffen gibt es spezielles Insiderwissen, welches die Erfolgswahrscheinlichkeit des Angriffs erhöht. Beispielsweise sind bei Angriffen, die Softwaresicherheitslücken ausnutzen, Kenntnisse über die im Einsatz befindlichen Softwareprodukte in einer Domäne hilfreich. Auch Insiderwissen über geistiges Eigentum erhöht die Erfolgsaussichten, eine Domäne etwa durch Erpressung negativ zu beeinflussen. Das Sammeln von Insiderwissen ist teilweise eine Voraussetzung oder zumindest eine übliche Vorbereitungshandlung spezieller Angriffe. Bei einer gezielten Penetration einer Domäne etwa, wird die Informationsgewinnung (engl. reconnaissance) als dedizierte Phase angesehen [HCA11]. Darüber hinaus können Angriffe auf aktive Präventionsmaßnahmen anderen Angriffen vorangestellt werden, sofern Kenntnisse über deren Existenz und deren Funktionsweise vorliegen. Ebenso können derartige Präventionsmaßnahmen mit dem richtigen Insiderwissen umgangen werden. Ein Beispiel ist eine Anomalieerkennungstechnik, die im Fall eines Alarms laufende Interaktionen eines Angreifers mit einer Domäne unterbrechen kann. Mit Kenntnis über dessen Existenz könnte ein Insiderangreifer seinen Angriff derart tarnen, dass die Anomalieerkennung nicht anschlägt. Nicht nur Insiderwissen über die Existenz von Präventionsmaßnahmen kann den Erfolg eines Angriffs erhöhen, sondern beispielsweise auch Insiderwissen über unterschiedliche Möglichkeiten, die einem Angreifer zur Verfügung stehen, um sein Ziel zu erreichen. Weiterhin kann die Eskalation von *Knowledge* erleichtert werden, indem bereits vorhandene *Knowledge* kombiniert und damit gegebenenfalls weiteres Insiderwissen geschöpft wird. Die Eskalation von *Trust* und *Privileges* wird ebenfalls erleichtert, was aus dem bereits angeführten Beispiel von *Social-Engineering-Angriffen* hervorgeht.
- **Verdecktheit**: Vor allem spezielles Insiderwissen über Erkennungs- und Überwachungsmaßnahmen verringert die Erkennbarkeit von Angriffen, denn es ermöglicht die geschickte Umgehung oder Manipulation dieser Maßnahmen. Aber auch andere Kenntnisse können eine derartige Verbesserung von Angriffen herbeiführen. Etwa Kenntnisse über die genaue Verortung von Geschäftsgeheimnissen kann eine anderweitig eventuell notwendige Suche

überflüssig machen und somit dabei helfen, bei unbefugtem Zugriff auf diese Geheimnisse kein Aufsehen zu erregen.
- **Auswirkung**: Mit dem zu einem Angriff passendem Insiderwissen kann ein Angreifer die Auswirkungen seines Angriffs stets maximieren. Ein Denial-of-Service Angriff beispielsweise verursacht den größten Schaden, wenn bekannt ist, welche Schwachstellen in einer Domäne existieren.

Sicherheitsmaßnahmen

Die Verbesserungen von Bedrohungen durch den Einsatz von *Knowledge* können verringert werden, indem unnötig bereitgestelltes Insiderwissen erkannt und dessen Bereitstellung entfernt wird. Diese Bemessung von Aspekten der sogenannten *Kenntnis nur bei Bedarf* (engl. need-to-know; s. Abschnitt 4.3) muss häufig für jedes Individuum oder zumindest für jede Rolle in einer Domäne, Outsider mit eingeschlossen, festgelegt, analysiert und durchgesetzt werden. Zugriffskontrollmechanismen beziehungsweise die Dokumentation über die explizite Bereitstellung von spezifischem Insiderwissen ermöglicht dementsprechend das Nachvollziehen, welches Individuum welche Informationen erhalten hat. Damit können Informationsflüsse modelliert, nachvollzogen und gegebenenfalls gesteuert werden.

3.2.5.3 Verbesserungen durch Privileges

Privileges erlauben einem Angreifer Interaktionen mit Ressourcen der Domäne. Sie werden bei einer Reihe von Insiderangriffen insbesondere zum Erreichen des tatsächlichen Angriffsziels eingesetzt. Dazu zählen: 1. die Eskalation und die Persistierung des eigenen Insidergrades (vgl. Abschnitt 3.2.3), um den Insidergrad zu einem späteren Zeitpunkt für weitere Angriffe zu nutzen, beispielsweise durch die Ausnutzung von Schwachstellen bei Ressourcen, mit denen der Angreifer interagieren kann, 2. die Weitergabe des eigenen Insidergrades an andere Insider (vgl. Abschnitt 3.2.2), beispielsweise durch die Übertragung der eigenen *Privileges* an fremde *Credentials*, sowie 3. die Durchführung von Interaktionen mit Ressourcen der Domäne, um die Verfügbarkeit der Ressourcen zu verletzen (vgl. Abschnitt 3.2.4).

Der Einsatz von *Privileges* bei einem Angriff ist damit sehr vielfältig und von zentraler Bedeutung. Bezogen auf die in Definition 3.3 beschriebenen Verbesserungen von Insiderbedrohungen können *Privileges* bei einem Angriff wie folgt aufgeschlüsselt werden:

- **Erfolg**: Je höher der Grad der *Privileges* ist, desto mehr steigt auch die Wahrscheinlichkeit, dass ein gewünschtes Angriffsziel erreicht werden kann. Beispiele solcher Angriffsziele sind das aktive Zusammentragen von internen Informationen und Insiderwissen zur Eskalation von *Knowledge* oder die Aneignung der in einer Domäne hinterlegten gültigen *Credentials* zur Eskalation der eigenen *Credentials*. Der Zugriffskontrollmechanismus ist in der Regel die letzte Schutzmaßnahme für eine Ressource, die mit den passenden *Privileges* umgangen werden kann. Auch können sonstige Schutzmechanismen deaktiviert oder umgangen werden, die einen Angriff ansonsten verhindern oder abschwächen würden.
- **Verdecktheit**: Mithilfe von *Privileges* können auch Schutzmechanismen deaktiviert oder umgangen werden, die bei der Erkennung eines laufenden Angriffs sowie bei der Aufklärung eines bereits vergangenen Angriffs behilflich sind. Dadurch kann sich ein Insiderangreifer einerseits unter dem Radar bewegen, etwa mit erlaubten Interaktionen, die für sich genommen kein Aufsehen erregen oder sogar als normales Verhalten registriert werden, die

allerdings in ihrer Gesamtheit einen Insiderangriff darstellen. Andererseits kann er aktiven Einfluss auf die eigene *Uncertainty* ausüben und diese eskalieren (vgl. Abschnitt 3.2.3.2), etwa durch die Manipulation von Log- beziehungsweise Aufzeichnungsdaten. Die Verdecktheit des Angreifers kann damit maßgeblich erhöht werden.

- **Auswirkung**: Ein P-Insider kann einen sehr viel stärkeren negativen Effekt mit einem Angriff erzielen, als ein P-Outsider. Das zeigt sich vor allem in der erhöhten Anzahl an Möglichkeiten, die einem solchen Angreifer mit hohen *Privileges* zur Verfügung stehen. Eine höhere Verfügungsgewalt bedeutet häufig auch Zugriff auf wichtige Güter und Infrastrukturen. Ein eingängiges Beispiel ist ein Systemadministrator eines Unternehmens, dessen IT-Infrastruktur von zentraler Bedeutung ist. Der Schaden, den dieser Administrator als Insiderangreifer anrichten kann, ist offensichtlich sehr viel höher, als ein unprivilegierter Nutzer dieser Infrastruktur.

Sicherheitsmaßnahmen

Privileges sind zurückziehbar (vgl. Abschnitt 3.2.1.4), indem der Zugriffskonrollmechanismus einer Domäne derart modifiziert wird, dass diese *Privileges* nicht mehr valide sind. Somit ist die Prüfung auf den Missbrauch von *Privileges* und deren anschließender Entzug ein effektiver Schutz vor den Verbesserungen von Bedrohungen durch den Einsatz von *Privileges*. Eine solche Prüfung kann beispielsweise in Form einer Anomalieerkennung des Interaktionsverhaltens von Individuen erfolgen. Damit kann die missbräuchliche Verwendung sowie das abweichende Verhalten eines P-Insiders erkannt und die *Privileges* entsprechend entzogen oder zumindest genauer untersucht werden. Weitere Maßnahmen zur Reduktion der *Uncertainty* können auch die Verbesserungen von Insiderbedrohungen eines P-Insiders senken. Dazu zählt die Korrelation von technisch erfassbaren Zugriffsereignissen und die anschließende regelbasierte Erzeugung von Sicherheitsmeldungen, die Aufzeichnung und Auswertung von Interaktionen eines Individuums mit den Ressourcen der Domäne oder die Umsetzung von wohlbekannten Sicherheitsprinzipien, wie zum Beispiel das *Vier-Augen-Prinzip*, die *interne Auditierung* oder die *Funktionstrennung* (engl. separation of duties) [PV98; BE01].

Viele Verbesserungen von Bedrohungen durch den Einsatz von *Privileges* entstehen durch einen mangelhaften Schutz der eigenen Ressourcen seitens der Domäne. Ein typisches Beispiel ist die Speicherung von *Credentials*, insbesondere im Kontext von Passwörtern. Ein fehlender Schutz dieser Passwörter erlaubt einem P-Insiderangreifer, sich diese Passwörter anzueignen und sie später für eigene Zwecke zu missbrauchen. Dagegen macht ein korrekter Sicherheitsmechanismus die Aneignung der *Credentials* für einen Angreifer unbrauchbar [Gau12], erhält aber gleichzeitig die Funktionalität der *Credentials*. Derartige Mechanismen könnten auch an anderer Stelle bei anderen Ressourcen etabliert werden.

Besondere Beachtung muss den Möglichkeiten eines P-Insiders zukommen, die Spuren eines Insiderangriffs zu verschleiern und damit die eigene *Uncertainty* zu eskalieren. Zwei Prinzipien sind dabei vielversprechende Sicherheitsmaßnahmen. Zum einen hat Anderson [And80] bereits festgestellt, dass das Nachvollziehen und die Aufklärung von Fehlverhalten anhand der Analyse von Log- und Aufzeichnungsdaten unmöglich ist, sofern diese Daten selbst von einem Insider manipuliert worden sind. Vielmehr braucht es in solchen Fällen Mechanismen, die prüfen und aufzeigen, was nicht oder nicht mehr in den Daten präsent ist. Zum anderen kann ein Netz aus Überwachungsmechanismen dafür sorgen, dass die Manipulation dieser Mechanismen beziehungsweise deren produzierter Daten selbst wieder neue Spuren in anderen Sensordaten

erzeugt. Dieses sogenannte *Veracity-Prinzip* [Gol12] ist immer dann anwendbar, wenn ein P-Insider keine globalen und allumfassenden Zugriffsrechte auf alle diese Mechanismen und deren Daten besitzt.

Weiterhin ist die Reduktion der *Lastingness* von *Privileges* ein möglicher Sicherheitsmechanismus. Er ist angebracht, wenn die *Privileges* eines Individuums bewusst durch die Domäne erhöht wurden, um die Bearbeitung einer speziellen Aufgaben zu ermöglichen. Werden sie im Abschluss der Aufgabe nicht wieder entzogen und bestehen unnötigerweise weiter, können sie unbemerkt erneut verwendet werden. Dieses sogenannte *Least-Privileges-Prinzip* muss häufig für jedes Individuum oder zumindest für jede Rolle in einer Domäne, Outsider mit eingeschlossen, festgelegt, analysiert und durchgesetzt werden. Dadurch können angemessene und robuste Zugriffskontrollmechanismen etabliert werden. Die Reduktion der *Lastingness* kann beispielsweise durch periodisches Prüfen der Notwendigkeit anhand von Aufgabenbeschreibungen oder Rollen geschehen. Die fortwährende Überwachung von Zugriffen auf Ressourcen einer Domäne erlaubt zusätzlich den Abgleich mit der dokumentierten Bereitstellung von *Privileges* und damit die Kenntnisnahme von Eskalationen.

3.2.5.4 Verbesserungen durch Trust

Persönliches Vertrauen beschreibt den Grad an Treu und Glauben, den eine Domäne in die Aufrichtigkeit, Integrität und Ehrlichkeit eines Individuums legt und damit die Bereitschaft einräumt, für das Individuum von Vorschriften und Protokollen abzuweichen. Alle Bedrohungen eines T-Insiders für eine Domäne haben die Eskalation von *Privileges* zum Ziel und zur Folge (vgl. Abschnitt 3.2.3.1) oder verbessern die Eskalationen anderer bereitgestellter Charakteristiken. Dem gegenüber steht also immer ein Insider, der dem Bedrohungsagenten *Trust* bereitstellt und selbst dazu missbraucht wird, seinen Insidergrad weiterzugeben (vgl. Abschnitt 3.2.2). Das bedeutet indirekte Verbesserungen von Insiderbedrohungen durch *Knowledge*, *Credentials*, *Privileges* und *Uncertainty*. Darüber hinaus erlaubt der Einsatz von *Trust* im Speziellen die folgenden Verbesserungen eines Angriffs:

- **Erfolg**: Die Eskalation von bereitgestellten Charakteristiken wird durch den Einsatz von *Trust* signifikant erleichtert. Ein T-Insider kann beispielsweise mithilfe von psychischem Druck eher erreichen, die Insidercharakteristiken eines Opfers zu erhalten, als ein T-Outsider.
- **Verdecktheit**: Das Vorhandensein von *Trust* bei einem Angriff erlaubt eine unterschwellige Art von Verdecktheit. Diese Verdecktheit entsteht durch eine vom persönlichen Vertrauen ausgehende natürliche Unschuldsvermutung. Aufgrund von *Trust* senkt sich in der Regel die Bereitschaft, unbefugte Aktivitäten sowie Verstöße gegen Regeln und Gesetze und damit einen Missbrauch des persönlichen Vertrauens einem T-Insider zuzutrauen. Nachforschungen der Urheberschaft von Aktivitäten in die Richtung des T-Insiders werden dadurch eventuell nicht in Erwägung gezogen.
- **Auswirkung**: Der Einsatz von *Trust* bringt keine zusätzlichen speziellen Verbesserungen von Angriffen in Bezug auf die Auswirkungen dieser Angriffe.

Sicherheitsmaßnahmen

Mögliche Herangehensweisen an Sicherheitsmaßnahmen bezüglich des Vertrauensmissbrauchs durch einen T-Insider können sich auf die damit verbundene Weitergabe von Insidergraden

durch denjenigen Insider konzentrieren, der den persönlichen Vertrauensvorschuss bereitstellt und somit Teil der Bedrohung wird. Dabei können Ansätze aus dem Forschungsgebiet der Psychologie helfen. Levine und Manning [LM14, Abschnitt 10.7] zeigen auf, dass Studien über Gehirnaktivitäten bei vertrauensvollem Verhalten „die Aktivierung in den mit Belohnungen zusammenhängenden Arealen belegen“. Die Verringerung der Bedrohungen durch einen T-Insiders lässt sich demnach durch einen negativen Einfluss auf die Erwartungshaltung für Belohnungen bei der Interaktion mit diesem T-Insider erreichen. Zum Beispiel indem derartige Interaktionen, die vor allem den persönlichen Vertrauensvorschuss für die angebliche Legitimierung einer Handlung in den Vordergrund zu rücken versuchen, einen tiefen moralischen Konflikt mit der Loyalität zur Domäne hervorrufen.

3.2.5.5 Verbesserungen durch Uncertainty

Uncertainty befähigt einen Insider dazu, in Abwesenheit von Überwachungs-, Prüf- und Kontrollmechanismen zu agieren und damit Aktionen auszuführen, die während der Ausführung oder im Nachhinein nicht nachvollzogen werden können. Aus Sicht der Domäne verhält sich der U-Insider dadurch in Bezug auf Regeln, Spezifikationen und Richtlinien konform und protokolltreu. Damit erfolgt eine grundsätzliche Verbesserung für alle möglichen Angriffe auf eine Domäne und *Uncertainty* nimmt eine zentrale Rolle bei allen Insiderbedrohungen ein. *Uncertainty* kann bei einem Angriff die folgenden konkreten Verbesserungen von Insiderbedrohungen verschaffen:

- **Erfolg**: Unter der Annahme, dass alle konkreten Bedrohungsaktionen nur dann verhindert werden können, wenn sie zuvor auch entdeckt wurden, erhöht *Uncertainty* den Erfolg aller möglichen Angriffe durch die erhöhte Verdecktheit. Ein Angriff, der beispielsweise von Abwehrmechanismen nicht registriert wird, kann sein Angriffsziel ungehindert erreichen. Dazu zählen auch Schwachstellen, die von einer Domäne bisher unentdeckt blieben, aber einem Angreifer bekannt sind und von diesem ausgenutzt werden.
- **Verdecktheit**: Der Grad an *Uncertainty* ist mit der Verdecktheit von Aktivitäten eines Insiders äquivalent. Demnach verbessert eine vorhandene *Uncertainty* eines U-Insiders direkt die Verdecktheit von Aktivitäten des Insiders, indem Sicherheits- und Überwachungsmechanismen nicht vorhanden beziehungsweise nicht auf die Aktivitäten des Insiders gerichtet sind oder vom U-Insider umgangen werden können.
- **Auswirkung**: Der Effekt jeglicher Insiderangriffe kann durch das Vorhandensein von *Uncertainty* erhöht werden, da die Angriffe potenziell länger unentdeckt bleibt und somit größeren Schaden anrichten können.

Sicherheitsmaßnahmen

Abgesehen von Kosten- und Performanzaspekten sowie dem potenziell erheblichen Eingriff in die Privatsphäre und die informationelle Selbstbestimmung (s. Abschnitt 3.4) gibt es keinen Grund, warum ein Individuum einen hohen Grad an *Uncertainty* besitzen muss. Aufgrund der großen Gefahr, die von U-Insidern ausgehen, sollte eine Domäne die Ressourcen, Anwendungen und Systemteile hinsichtlich ihrer domänenspezifischen wie auch ihrer technischen Kritikalität identifizieren und entsprechende Überwachungsmechanismen zur Erzeugung, Verknüpfung und Archivierung von Informationen über jegliche Arten von kritischen Vorgängen etablieren. Das dient einerseits der Erkennung von Angriffen während ihrer Durchführung und andererseits der robusten Aufklärung von Vorfällen im Nachhinein (engl. digital forensic readiness [Row04]). Ein konsequentes derartiges Vorgehen, bei dem die *Uncertainty* in einer Domäne gegen Null

Abbildung 3.5: Die Abfolge einzelner Angriffsschritte als Teil einer von Hutchins, Cloppert und Amin [HCA11] identifizierten Angriffskette

läuft, erzeugt allerdings ein System der Totalüberwachung einer Domäne mit allen negativen Folgen in Bezug auf das Arbeitsklima sowie auf die Privatsphäre und die informationelle Selbstbestimmung der Insider. Um dieses Problem zu mildern, werden in Kapitel 6 und Kapitel 7 mögliche Lösungsansätze erarbeitet.

3.3 Bedrohungen für Domänen durch Outsider

Die Bedrohungen für Domänen durch Outsider stehen nicht im Fokus dieser Dissertation und werden demnach auch nicht im Detail behandelt. Dennoch sind die meisten Outsiderangriffe darauf ausgelegt, den Outsider in eine bessere Position gegenüber der Domäne zu bringen. Dadurch kann er einen weiteren Angriff auf die Domäne auszuführen, der ansonsten erfolglos oder zumindest weniger erfolgreich wäre. Diese bessere Position gegenüber der Domäne lässt sich direkt mit einer *Eskalation des Insidergrades* in Verbindung bringen, wie nachfolgend gezeigt wird. Der weitere Angriff stellt dann genau genommen einen Insiderangriff dar und lässt sich somit anhand der Ausführungen in Abschnitt 3.2 einordnen. Diese explizite Aufteilung der Bedrohungen von Domänen durch Outsider in die tatsächliche Outsiderbedrohung, die das Ziel hat, zu einem Insider zu werden, und die anschließende Insiderbedrohung, wird bei der Auswertung von Angriffsszenarien häufig unterschlagen. Sie wird aber offensichtlich, wenn man dabei die unterschiedlichen Insidertypen (vgl. Abschnitt 2.4) berücksichtigt.

3.3.1 Die Angriffskette eines Outsiderangreifers

Zur Erreichung eines Angriffsziels haben Hutchins, Cloppert und Amin [HCA11] sieben Schritte identifiziert, die hintereinander ausgeführt eine Angriffskette (engl. kill chain) bilden. Abbildung 3.5 zeigt die einzelnen Schritte in ihrer Abfolge. Diese Angriffskette muss von einem Angreifer durchlaufen werden, um in ein Informations- oder Kommunikationssystem einzudringen, es in seiner Funktionalität zu stören oder Informationen herauszuschleusen. Die Autoren definieren die Schritte wie folgt:

1. Reconnaissance – Die Recherche, Identifikation und Auswahl von Angriffszielen sowie die Beschaffung von Informationen über die Angriffsziele. Dieser Schritt beinhaltet oft die Durchforstung von Internetquellen nach IP- und E-Mail-Adressen, Domainnamen, sozialen Verbindungen oder Informationen über spezielle Technologien.
2. Weaponization – Die Kopplung eines Fernzugrifftrojaners mit einem Exploit zu auslieferbaren Nutzdaten, typischerweise mithilfe eines automatisierten Software-Werkzeugs. Immer mehr Dateien aus dem Bereich der Nutzeranwendungen, wie zum Beispiel das Adobe Protable Document Format (PDF) oder Microsoft Office Dokumente dienen als derartig präparierte Auslieferungsgegenstände.

3. Delivery – Die Übertragung des präparierten Ausliefergegenstandes zum Angriffsziel. Sehr häufige Auslieferungswege sind E-Mail-Anhänge, Webseiten und Universal Serial Bus (USB) Medien.

4. Exploitation – Die Ausführung des Angriffscodes, der mit dem Auslieferungsgegenstand übertragen wurde. Sehr häufig zielt dieser Schritt auf die Ausnutzung einer Anwendungs- oder Betriebssystemschwachstelle ab. Es könnten aber auch einfach die Nutzer des Informations- oder Kommunikationssystems selbst ausgenutzt oder die Funktion eines Betriebssystems wirksam eingesetzt werden, die automatisiert Code ausführt.

5. Installation – Die Installation eines Fernzugrifftrojaners oder einer Hintertür auf dem System des Angriffsopfers erlaubt dem Angreifer die Einrichtung und Bewahrung von Persistenz innerhalb der Systemumgebung.

6. Command and Control – Typischerweise müssen kompromittierte Rechner aus dem Informations- oder Kommunikationssystem heraus eine ausgehende Verbindung zu einem Kontrollserver aufbauen, um einen Kontrollkanal zu öffnen. Sobald dieser Kontrollkanal offen ist, haben Angreifer direkten Zugriff auf die Innenseite der Zielumgebung.

7. Actions on Objectives – Erst zu diesem Zeitpunkt, nachdem die ersten sechs Schritte durchlaufen wurden, können Angreifer Aktionen durchführen, um ihre ursprünglichen Ziele zu erreichen. Dazu gehören in der Regel das Herausschleusen von Informationen, was die Kollektion, Verschlüsselung und Extraktion von Daten aus der Zielumgebung beinhaltet. Weiterhin gehören die Verletzung der Datenintegrität und der Verfügbarkeit als potenzielle ursprüngliche Ziele dazu. Alternativ könnte der Angreifer nur den Zugriff auf die initiale Opfermaschine wünschen, um sie dann als Sprungpunkt für das Kompromittieren weiterer Systeme und für Bewegungen innerhalb der Zielumgebung zu verwenden.

Die Angriffskette ist umstritten, da sie sich sehr auf den Einsatz von Schadsoftware konzentriert sowie auf Insiderangriffe nicht anwendbar ist [Mal16]. Unabhängig von aller Kritik stellt dieser Vorschlag allerdings eine grundlegende Systematisierung von Outsiderangriffen dar, auf die sich weitere Angriffswege übertragen lassen. Wichtiger jedoch ist die Möglichkeit, anhand der Angriffskette aufzuzeigen, an welchen Punkten auf dem Weg zum ursprünglichen Angriffsziel ein Outsiderangreifer seine Position gegenüber der Domäne zu verbessern versucht, um daraufhin einen oder mehrere weitere und vor allem verbesserte Insiderangriffe durchzuführen.

In Schritt 1 versucht der Outsiderangreifer Einblicke in die Domäne zu erlangen und an Insiderwissen heranzukommen. Damit versucht er unbefugt seine *Knowledge* zu erhöhen. Je besser das dem Angreifer gelingt, um so zielgerichteter und erfolgreicher kann er mit den nächsten Schritten fortfahren. Eine besonders effektive *Reconnaissance* versetzt den Outsiderangreifer somit in die Lage eines K-Insiders.

Die sorgfältige Auswahl und das Vertrautmachen mit geeigneten Software-Werkzeugen sowie die Verknüpfung von Schadsoftware mit einem Auslieferungsgegenstand in Schritt 2 beeinflusst die Charakteristik *Abilities* beziehungsweise wird durch diese Charakteristik beeinflusst. Diese inhärente Bedrohungscharakteristik hat keinen Einfluss auf den Insidergrad des Angreifers (vgl. Abschnitt 3.2.1.1) und versetzt den Angreifer somit auch nicht in die Lage eines Insiders.

In Schritt 3 steht die Erhöhung von *Uncertainty* sowie von *Trust* im Fokus des Angreifers. Die Auswahl des richtigen Übertragungsweges hat zum Ziel, möglichst gängige Datenformate oder Abläufe auf Seite der Domäne zu finden und zu nutzen, sodass keine Auffälligkeit entsteht und

darüber hinaus keine Mechanismen die Übertragung sowie die anschließende Öffnung oder Ausführung des Auslieferungsgegenstandes überwachen oder sogar verhindern. Demnach wird der Angreifer zu einem U-Insider. Ebenfalls kann durch den aus Angreifersicht intelligent gewählten Übertragungsweg indirekt das persönliche Vertrauen, also *Trust*, erhöht werden, indem zum Beispiel der Übertragungsweg einen für das Opfer vertrauten beziehungsweise vertrauenswürdigen Absender vortäuscht und damit zunächst den Angreifer in die Lage eines C_M-Insiders versetzt und damit dann den Grad an *Trust* des vorgetäuschten Absenders übernimmt. Auch das Anregen psychologischer Aspekte zur Begünstigung von Umständen, etwa Freundlichkeit und Hilfsbereitschaft, können *Trust* erhöhen. Der Angreifer wird dadurch zu einem T-Insider.

Die Ausführung des Angriffscodes in Schritt 4 hat zunächst das Ziel, zu einem C_M-Insider zu werden, um dann in Verbindung mit der persistenten Festsetzung in der Systemumgebung in Schritt 5 sowie dem Kontrollkanal in Schritt 6 die *Privileges* zu erhöhen und zu erhalten und dadurch zu einem P-Insider zu werden. Das Opfer öffnet den Auslieferungsgegenstand oder führt diesen aus und gibt damit indirekt seine *Credentials* wissentlich oder unwissentlich an den Angreifer, sodass dessen Aktivitäten unter der Identität des Opfers durchgeführt werden können. Weiterhin wird durch den Schadcode des Angreifers versucht, Dienste oder Ressourcen der Domäne derart zu verändern, dass der Angreifer indirekt höhere Zugriffsrechte erhält. Die Installation in Schritt 5 hat dann das Ziel, diese Zugriffsrechte dauerhaft zu erhalten und gegebenenfalls durch Schritt 6 direkt für den Angreifer zugänglich zu machen.

Im letzten Schritt ist der Angreifer dann in der Lage, den eigentlichen Angriff als C_M-Insider sowie als P-Insider durchzuführen oder gegebenenfalls weitere Angriffe zur besseren Positionierung durchzuführen.

Die Teilangriffe in den Schritten 1 und 3–6 auf dem Weg der Angriffskette lassen sich, wie bereits erwähnt, mit Blick auf den Angreifer in die Bedrohungskategorie *Eskalation des Insidergrades* zusammenfassen, auf die im nächsten Abschnitt kurz eingegangen wird. Bei einem Erfolg der Eskalation kann es sich bei anschließenden Angriffsschritten schon nicht mehr nur um Outsiderbedrohungen, sondern bereits um Insiderbedrohungen handeln (vgl. Abschnitt 3.2.1). Sofern bei den Teilangriffen auf dem Weg der Angriffskette ein Insider als Opfer ausgenutzt wird, sind dessen Handlungen als Insiderbedrohungen aufzufassen und fallen unter die Bedrohungskategorie *Weitergabe des Insidergrades* (vgl. Abschnitt 3.2.2). Der letzte Schritt 7 beinhaltet abschließend entweder weitere Insidergrad-Eskalationen oder hat die *Verhinderung von Insidergraden* als Ziel (vgl. Abschnitte 3.2.3 und 3.2.4).

3.3.2 Eskalation des Insidergrades durch einen Outsider

Eine *Eskalation des Insidergrades* durch einen Outsider liegt genau dann vor, wenn sich ein Individuum unbefugt Insidercharakteristiken aneignet, ohne dabei eventuell eigene vorhandene Insidercharakteristiken auszunutzen. Allerdings sind diese Outsiderbedrohungen keine unmittelbare, sondern nur eine mittelbare Gefahr für Domänen. Die eigentlich nachteiligen Auswirkungen für eine Domäne entstehen erst, wenn ein so unbefugt erlangter Insidergrad verwendet wird, um ihn weiterzugeben oder um damit diesen oder andere Insidergrade zu verhindern (vgl. Definition 3.2 in Abschnitt 3.2.1.2).

Die nachfolgenden Betrachtungen fokussieren sich hauptsächlich auf die Fälle, in denen Outsider ihren Insidergrad eskalieren. Die entsprechenden Insiderbedrohungen, in denen Insider diese Insidergrade eskalieren, lassen sich als jeweilige Verbesserungen der hiesigen Bedrohungen

durch einen vorhandenen Insidergrad einordnen und wurden in Abschnitt 3.2.5 eingehender betrachtet. Sowohl die hier aufgeführten Bedrohungen als auch die möglichen Schutzmaßnahmen sind in der Theorie und in der Praxis nicht neu. Sie beschreiben häufig, was allgemein als Perimeterschutz bezeichnet wird [CBR03, Abschnitt 1.4]. Eine explizite Benennung und Einordnung der bekannten Bedrohungen und Schutzmaßnahmen kann allerdings bei der Einschätzung des Sicherheitsniveaus einer Domäne sowie bei der Auswahl geeigneter Kombinationen von Sicherheitsmaßnahmen helfen.

Bezogen auf den Insider, der dem Outsider unbefugt die Insidercharakteristiken bereitstellt, bedeutet diese Bedrohung eine unbefugte Weitergabe des Insidergrades (vgl. Abschnitt 3.2.2). Diese Weitergabe passiert in manchen Fällen nicht direkt und nicht bewusst, sondern unbewusst und/oder indirekt.

3.3.2.1 Eskalation von Credentials

Die Eskalation von *Credentials* ist beispielsweise durch Erraten oder einen Bruteforce-Angriff auf diejenigen eines C_A-Insiders möglich. Schwach gewählte und mangelhaft geschützte *Credentials* helfen beim Erfolg derartiger Angriffe. Auch wird versucht, anhand geschickter elektronischer Nachrichten oder gefälschter Webseiten die *Credentials* von möglichst vielen oder manchmal auch gezielt von speziellen Personen in Erfahrung zu bringen. Dahinter steckt die einfache Behauptung einer Domänenzugehörigkeit, verbunden mit unterstützenden Maßnahmen wie etwa einer offiziellen Kleidung, einem authentisch wirkenden Dokument oder psychologisch gestützter Appelle an die Freundlichkeit und Hilfsbereitschaft.

In jedem Fall wird der Outsider bei der *Credentials*-Eskalation zu einem C_M-Insider. Mithilfe der *Credentials*-Eskalation können Aktivitäten eines Individuums verschleiert und auf die Identität einer anderen Person projiziert werden. Darüber hinaus eignet sich der ursprüngliche Outsiderangreifer dadurch alle bereitgestellten Charakteristiken an, die an die Identität des bestohlenen Insiders geknüpft sind. Das kann aber muss nicht zwangsläufig auch die zusätzliche Eskalation der *Privileges*, *Trust* und *Uncertainty* implizieren.

Sicherheitsmaßnahmen

Aktuelle bewährte Praktiken (engl. best practices) können dabei helfen, die Bedrohung der Aneignung einer fremden Domänenzugehörigkeit abzuschwächen. Dies kann einerseits durch die Auswahl sicherer *Credentials* und andererseits durch die domänenseitig sichere Handhabung und Aufbewahrung der *Credentials* geschehen, wie etwa im Bereich der Passwortsicherheit [Gau12] und der Mehrfaktor-Authentisierung [Gra+17, Abschnitt 5.1.8]. Einen wichtigen Schutz bietet außerdem das Schaffen und Trainieren des Bewusstseins von Insidern für derartige Bedrohungen,insbesondere jene, die durch gefälschte Behauptungen entstehen und dadurch einen C_A-Insider zur Weitergabe seiner *Credentials* verleiten.

3.3.2.2 Eskalation von Knowledge

Die Eskalation von *Knowledge* durch einen Outsider ist ausschließlich durch Informationen möglich, die von der Domäne wissentlich oder unwissentlich der Öffentlichkeit bereitgestellt werden. Darüber hinaus ist es schwierig bis unmöglich abzuschätzen, welches Insiderwissen

durch die Kombination von vermeintlich harmlosen oder zusammenhangslosen Informationen über oder durch die Domäne entstehen kann. Beispiele über derartiges Erzeugen von Wissen durch anderweitig verfügbares Hintergrundwissen werden in Abschnitt 6.2.2.3 im Kontext der De-Identifizierung zur Erhöhung des Datenschutzes betrachtet.

Sicherheitsmaßnahmen

Die Vertraulichkeit wichtiger interner Informationen muss speziell durch die Domäne abgesichert werden. Dadurch kann die unbeabsichtigte Bereitstellung von *Knowledge* verhindert werden. Etabliert haben sich dabei bereits Verschlüsselungstechniken, das Need-to-Know-Prinzip sowie beauftragte Sicherheitsüberprüfungen, die keine speziellen internen Informationen über ein Unternehmen erhalten, sogenannte Black-Box-Tests. Darüber hinaus müssen allerdings weitere und vor allem subtilere Informationsquellen identifiziert und abgesichert werden. Eine besondere Bedrohung durch die Eskalation von *Knowledge* liegt darin, dass sie nicht zurückziehbar ist (vgl. Abschnitt 3.2.1.4), was bedeutet, dass einmal von einer Domäne bereitgestelltes Insiderwissen nicht mehr von der Domäne kontrolliert werden kann. Die einzige Möglichkeit, die eine Domäne hat, um Insiderwissen einem Individuum wieder zu entziehen, also die *Lastingness* von *Knowledge* zu reduzieren, ist dafür zu sorgen, dass das Insiderwissen den tatsächlichen Domänenzustand nicht mehr widerspiegelt. Dafür lassen sich zum Beispiel Techniken der sogenannten *Verteidigung durch bewegliche Ziele* (engl. moving target defence) [CF14; ZDO14] einsetzen.

3.3.2.3 Eskalation von Trust

Trust kann für einen Outsider zum einen auf konventionellem Weg erlangt werden, wobei unterschiedliche Handlungskontexte, Beziehungskonstellationen und Artefakte eine spezielle Rolle spielen und bedacht werden müssen [CW00].

Zum anderen kommt es im Kontext der Charakteristik *Trust* allerdings auch zu potenziellen Überschneidungen von Domänen. Denn *Trust* ist nicht nur an eine Domäne sondern vor allem an dessen Insider gebunden, die derartiges persönliches Vertrauen aus anderen Domänen zu anderen Personen in die vorliegende Domäne mitbringen. Dadurch kann eine Person aus einer Domäne A, die aus Sicht einer anderen Domäne B einen Outsider darstellt, das persönliche Vertrauen zu einem Insider beider Domänen erhöhen und wird dadurch auch für Domäne B zu einem T-Insider.

Sicherheitsmaßnahmen

Der Schutz vor der unbefugten Erhöhung von *Trust* ist ein komplexes Themenfeld, da persönliches Vertrauen ein wertvolles Gut der zwischenmenschlichen Beziehungen ist und sich der Grad an *Trust* im Zuge von Interaktionen zwischen Individuen zwangsläufig fortwährend ändert und damit immer in Bewegung ist. Ob die Erhöhung von *Trust* unbefugt ist, lässt sich vermutlich nur anhand der intrinsischen Motivation eines Individuums erschließen. Diese *Intention* wiederum lässt sich, wenn überhaupt, häufig nur im Nachhinein eines Vorfalls und durch die Verknüpfung mehrerer Handlungen feststellen (vgl. Abschnitt 3.2.1.3). Demnach beschränken sich Sicherheitsmaßnahmen vermutlich auf die explizite Definition und Abgrenzung verschiedener Vertrauenskontexte, um die Überschneidung von Domänen zu verhindern oder zumindest zu minimieren. Beispielsweise könnte privates persönliches Vertrauen von persönlichem Vertrauen im Kontext einer Geschäftsbeziehung getrennt etabliert werden.

3.4 Bedrohungen für Insider durch Domänen

Eine Domäne hat in zweierlei Hinsicht eine besondere Stellung. Einerseits ist sie in der Position, Insidergrade bereitzustellen und in begrenztem Maß auch zu entziehen, und andererseits hat sie auch die Machtposition in einer Abhängigkeitsbeziehung zwischen der Domäne und dem Insider. Aufgrund dessen ist eine Domäne in der Lage, diese Macht gegenüber ihren Insidern auszunutzen und zu missbrauchen. Eine dieser missbräuchlichen Machtausübungen ist der illegale Eingriff in die Privatsphäre der eigenen Insider.

Eine Domäne hat das berechtigte Interesse und teilweise auch die gesetzlich vorgegebene Pflicht (s. Abschnitt 6.1.2), Schäden an der eigenen Domäne und an den eigenen Insidern sowie Schäden durch die eigenen Insider zu erkennen und abzuwehren. Für diese Aufgabe stehen Sicherheitsmaßnahmen zur Verfügung, die unter anderem auch Daten über die Aktivitäten von Insidern erheben, speichern und weiterverarbeiten. Ein Missbrauch dieser Daten durch die Domäne führt zu einem erheblichen Eingriff in die Privatsphäre und das Recht auf informationelle Selbstbestimmung der von den Maßnahmen betroffenen Insider.[6] Ein entsprechendes Szenario wurde bereits in Beispiel 3.3 aus Abschnitt 3.1 gezeichnet, indem die Daten eines Anomalieerkennungsmechanismus zum Zweck der Produktivitätsbewertung einzelner Mitarbeiter in einem Unternehmen missbraucht wurden. Weitere Beispiele für einen illegalen Umgang mit personenbezogenen Daten von Domänen sind:

- die unerlaubte Erstellung und Aktualisierung von Bewegungs- oder Aktivitätsprofilen (engl. tracking),
- die unerlaubte Erstellung und Aktualisierung von Persönlichkeits- oder Präferenzprofilen (engl. profiling),
- das Nachstellen einzelner Insider (engl. stalking) sowie
- die Stärkung der Machtposition gegenüber den Insidern.

Ein derartiger Machtmissbrauch durch eine Domäne zielt in der asymmetrischen Machtposition der Domäne gegenüber ihren Insidern damit laut Pfitzmann [Pfi06] auf die Bedrohung des Schutzziels der legalen Durchsetzbarkeit ab. Sie ist dem BSI zufolge als die elementare Bedrohung G 0.38 *Missbrauch personenbezogener Daten* einzuordnen und stellt gleichzeitig eine konkrete Ausprägung der elementaren Bedrohung G 0.29 *Verstoß gegen Gesetze oder Regelungen* dar.

Sicherheitsmaßnahmen Der Schutz gegen diese Art der Bedrohung von Insidern durch Domänen erweist sich als sehr komplex, da die Machtasymmetrie als Basis der Bedrohung sowohl ein strukturelles als auch ein kulturelles Problem darstellt. Aktuell etablierte Sicherheitsmaßnahmen beschränken sich hauptsächlich auf die Einrichtung von Meldestellen bei übergeordneten Institutionen, die entweder eine Kontrollfunktion gegenüber der Domäne einnehmen oder die Rechte der Insider vertreten (vgl. Art. 33 DSGVO). Ein weiterer Ansatz, der das Prinzip der *Mehrseitigen Sicherheit* verfolgt (vgl. [RPM97]) und sich technischer Unterstützung bedient, wird in den Kapiteln 6 und 7 erarbeitet und vorgestellt.

6. Sofern es sich um Insider handelt, die beispielsweise zum Betrieb derartiger Sicherheitsmaßnahmen eingesetzt werden und die ihren Zugriff auf diese Daten unbefugt, das heißt nicht von der Domäne autorisiert, missbrauchen, ist dieser Missbrauch als Insiderangriff zu bewerten und entsprechend den Ausführungen in Abschnitt 3.2 einzuordnen. Sofern dieser Missbrauch allerdings von der Domäne autorisiert wird, handelt es sich um eine Bedrohung der betroffenen Insider durch die Domäne.

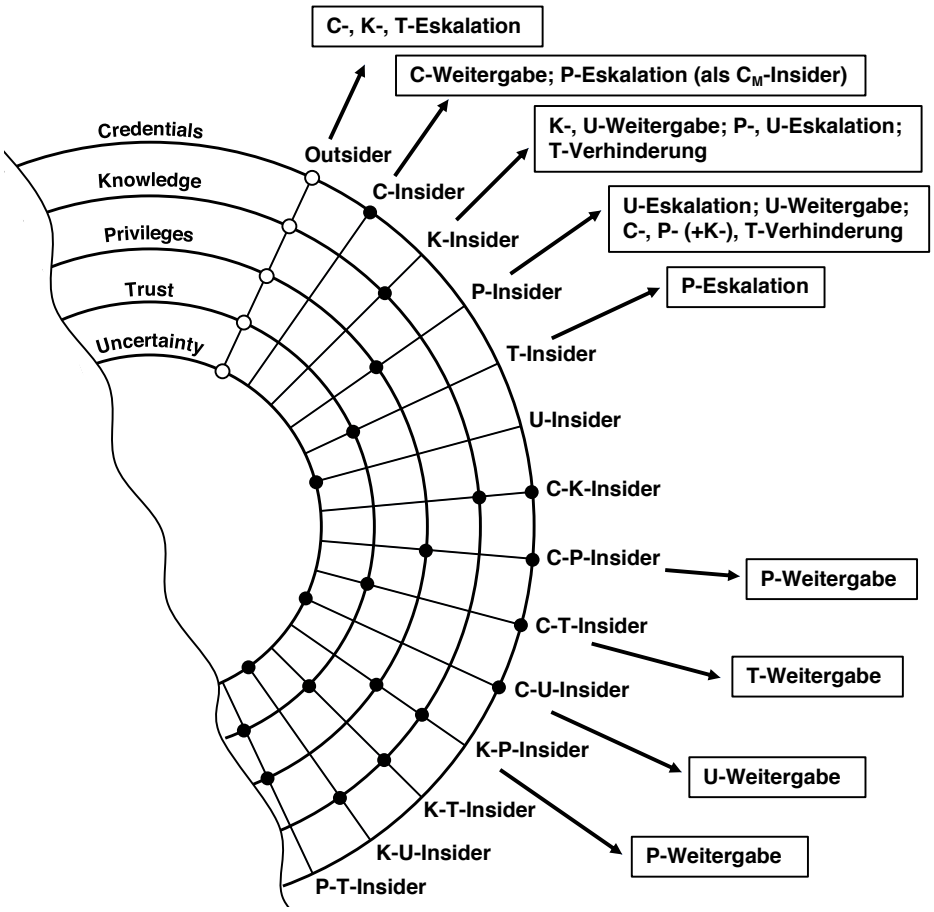


Abbildung 3.6: Einordnung der identifizierten Insiderbedrohungen durch die unbefugte Weitergabe, Eskalation sowie Verhinderung von Insidergraden in die Insidertaxonomie

3.5 Einordnung der Bedrohungen anhand der Insidertaxonomie

Die in Abschnitt 2.6 entwickelte Insidertaxonomie zeigt die Zusammenhänge zwischen allen möglichen Kombinationen der Insidertypen auf. Dadurch wird eine Ordnung deutlich, anhand der spezielle Eigenschaften einzelnen Insidertypen zugeordnet werden können und aus der ersichtlich wird, welche Insidertypen dadurch ebenfalls diese Eigenschaften besitzen.

Die in Abschnitt 3.2 und Abschnitt 3.3 identifizierten Bedrohungen für eine Domäne durch Insider und teilweise auch durch Outsider stellen eben solche Eigenschaften dar und können anhand der ausgeführten Analysen in den genannten Abschnitten einzelnen Insidertypen zugeordnet werden. Die Insidertaxonomie zeigt damit auf, von welchen der insgesamt 31 Insidertypen diese Bedrohungen ausgehen und welche kombinierten Insidertypen demnach auch kombinierte Insiderbedrohungen darstellen. So zeigt beispielsweise Abschnitt 3.2.2.4, dass die Bedrohung durch die Weitergabe der Insidercharakteristik *Trust* von einem C-T-Insider ausgeht. Dieser Insidertyp umfasst allerdings auch die Bedrohungen, die von einem T-Insider, von einem C-Insider sowie von einem Outsider ausgehen. Somit erweitert sich die Liste der Bedrohungen von diesem Insidertyp um die Bedrohungen durch *Credentials*-Weitergabe und *Credentials*-, *Knowledge*-, *Privileges*- sowie *Trust*-Eskalation. Abbildung 3.6 zeigt diese und weitere Zuordnungen der Bedrohungen zu den Insidertypen der Taxonomie.

In Verbindung mit den Anweisungen zur Insidermodellierung aus Abschnitt 2.5 sowie den Erkenntnissen über die Klassifizierung existierender Insiderdefinitionen aus Abschnitt 2.5.3 wird ebenfalls ersichtlich, welche Insidermodelle und welche der analysierten Publikationen, in denen verschiedene Insidertypen definiert werden, zu welcher Insiderbedrohung passen.

3.6 Erweiterungsmöglichkeiten

Die entwickelte Systematisierung der identifizierten Bedrohungen in Abschnitt 3.5 enthält bisher nur die Bedrohungen durch die unbefugte Weitergabe von Insidergraden (vgl. Abschnitt 3.2.2), die unbefugte Eskalation von Insidergraden (vgl. Abschnitte 3.2.3 und 3.3.2) sowie die unbefugte Verhinderung von Insidergraden (vgl. Abschnitt 3.2.4). In Abschnitt 3.2.5 wurden allerdings verschiedene Dimensionen aufgezeigt, in denen einzelne Insidercharakteristiken Verbesserungen von Bedrohungen hervorrufen können. Diese Verbesserungen verändern die Qualität existierender Bedrohungen, sodass sie nach eingehender Analyse in die Insidertaxonomie mit aufgenommen werden können und sollten. Dadurch kann die Frage beantwortet werden, welche Bedrohungen durch die einzelnen Insidertypen verbessert werden und wie genau diese Verbesserung im Detail aussieht. Aus diesen Erkenntnissen können passende Sicherheitsmaßnahmen anhand dieser Verbesserungen von Insiderbedrohungen abgeleitet werden.

3.7 Fazit

In den Fällen, in denen wissenschaftliche Arbeiten im Bereich der Insiderforschung eine Insiderbedrohung als eine Bedrohung durch einen Insider definieren, werden drei grundlegende Tatsachen missachtet. Zum einen gibt es unterschiedliche Typen von Insidern, die diesen Bedrohungen unterschiedliche Qualitäten verleihen. Zum zweiten sind Bedrohungen durch einen Insider, bei denen der Insidergrad keine Rolle spielt, streng genommen Outsiderbedrohungen. Und zum dritten wird damit nicht spezifiziert, welche Insideraspekte bei der Bedrohung eine Rolle spielen und somit den Fokus der Forschung darstellen. Eine vergleichende Betrachtung von Forschungsarbeiten sowie von entwickelten Erkennungs- und Abwehrtechniken von Insiderbedrohungen wird dadurch erschwert beziehungsweise unmöglich. In diesem Kapitel wurden Lösungen für diese Probleme erarbeitet.

Zunächst erfolgte anhand von drei Beispielszenarien die Erarbeitung eines Systemmodells von Insiderbedrohungen, mit dem alle beteiligten Komponenten sowie Interaktionen abstrakt identifiziert werden konnten. Aus dem Systemmodell ging hervor, dass es sich bei Insiderbedrohungen um mehrseitige Bedrohungen handelt, die in den weiteren Abschnitten dieses Kapitels differenziert betrachtet wurden.

Für die Bedrohungen einer Domäne durch einen Insider erfolgte in Abschnitt 3.2 eine neue, detaillierte und konsequente Charakterisierung von Insiderbedrohungen (vgl. Abschnitt 3.2.1), die eine abstrakte Begriffsdefinition, die Modellierung von Insiderbedrohungen anhand von Bedrohungscharakteristiken sowie eine systematische Einteilung von Insiderbedrohungen in die drei Klassen der unbefugten *Eskalation*, *Weitergabe* und *Verhinderung von Insidergraden* beinhaltet. Für alle drei Klassen von Insiderbedrohungen wurden anschließend in den Abschnitten 3.2.2 bis 3.2.4 notwendige Insidergrad-Voraussetzungen herausgearbeitet, wodurch zusätzlich erste Erkenntnisse über mögliche und passgenaue Sicherheitsmechanismen gewonnen und erläutert

werden konnten. Eine jeweils am Ende der Abschnitte durchgeführte qualitative Auswertung der Erkenntnisse fasste die Insidergrad-Voraussetzungen zusammen und erlaubte eine Verknüpfung der Insiderbedrohungen mit den in Abschnitt 1.5.5 genannten elementaren Bedrohungen. Zusätzlich erfolgte in Abschnitt 3.2.5 eine detaillierte Betrachtung der durch einen Insidergrad verbesserten Bedrohungen, die auf jegliche Arten von Insider- und Outsiderbedrohungen zutreffen. Die Verbesserungen lassen sich mit den drei Aspekten *Erfolg*, *Verdecktheit* und *Auswirkung* zusammenfassen. Jede Insidercharakteristik wurde auf diese drei Aspekte hin untersucht. Die gewonnenen Erkenntnisse führten auch dabei zur Herausarbeitung von möglichen und passgenauen Sicherheitsmechanismen.

Mit der Beschreibung von Bedrohungen für Domänen durch Outsider konnte in Abschnitt 3.3 aufgezeigt werden, dass die meisten Outsiderbedrohungen zunächst zum Gegenstand haben, Outsider zu einem Insider werden zu lassen. Anhand der von Hutchins, Cloppert und Amin [HCA11] publizierten Angriffskette wurden zudem die Punkte in der Angriffskette identifiziert, an denen nicht mehr von einer Outsider- sondern von einer Insiderbedrohung ausgegangen werden kann. Zudem wurde anhand der in Abschnitt 2.6 vorgestellten Insidertaxonomie deutlich, dass alle Outsiderbedrohungen auch von Insidern ausgehen können, die ihren Insidergrad für die Verbesserung der Outsiderbedrohungen einsetzen können. Vor diesem Hintergrund wurden spezifische Outsiderbedrohungen genauer betrachtet.

Die Diskussion der Bedrohungen von Insidern durch ihre Domäne in Abschnitt 3.4 rundete die mehrseitigen Bedrohungen für und durch Insider ab. Sie zeigt die Probleme und Gefahren auf, die aufgrund von Sicherheitsmechanismen in der Domäne zur Erkennung und Abwehr von Insiderbedrohungen für die Privatsphäre und informationelle Selbstbestimmung der betroffenen Insider entstehen und gibt einen Ausblick auf die Arbeiten in den Kapiteln 6 und 7.

Mit der Klassifizierung von Insiderbedrohungen sowie der detaillierten Betrachtung aller Insidergrad-Voraussetzungen bei den einzelnen Insiderbedrohungen konnten abschließend und als Liefergegenstand des Forschungsbeitrags B2 (vgl. Abschnitt 1.3) Bezüge zu den Insidertypen der Insidertaxonomie aus Abschnitt 2.6 hergestellt werden. Diese wurden in Abschnitt 3.5 zusammengefasst und illustriert.

4 Systematisierung existierender Gegenmaßnahmen

Mit Kapitel 2 wurden bereits Grundlagen erarbeitet, die es erlauben, die Insider in einer Domäne zu modellieren und daraus eindeutige Insidertypen abzuleiten. Im Idealfall lassen sich damit Forschungs- und Entwicklungsarbeiten identifizieren, die sich mit den Bedrohungsaspekten und zugeschnittene Gegenmaßnahmen dieser Insidertypen beschäftigen. Das setzt allerdings voraus, dass die publizierten Gegenmaßnahmen die Bedrohungsaspekte der zugrunde gelegten Insiderdefinitionen adäquat anvisieren. Um diese Voraussetzung zu ermöglichen, wurden in Kapitel 3 die Bedrohungsaspekte von Insidertypen herausgearbeitet und entsprechende Sicherheitsmechanismen als Schutz gegen die neu charakterisierten und im Detail erläuterten Insiderbedrohungen aufgeführt. In diesem Kapitel werden Sicherheitsmechanismen nun genauer betrachtet, die sich in der Literatur als Gegenmaßnahmen gegen Insiderangriffe etabliert haben. Darüber hinaus erfolgt eine genaue Abgrenzung dieser Sicherheitsmechanismen voneinander anhand der in Abschnitt 2.6 entwickelten Insidertaxonomie. Auf diese Weise tritt hervor, welche Maßnahmen einen wirksamen Schutz gegen Bedrohungen von welchem Insidertyp darstellen. Damit wird der zweite Teil von Forschungsfrage 2 adressiert und bearbeitet (vgl. Abschnitt 1.2).

Wesentliche Inhalte Mit einer genauen Betrachtung grundlegender Erkennungs- und Abwehrmaßnahmen werden Verknüpfungen zu konkreten Insiderbedrohungen aus Abschnitt 3.2 sowie Insidertypen der Insidertaxonomie aus Abschnitt 2.6 ermöglicht. Dafür wird herausgearbeitet, welche Insidercharakteristiken und damit welche Insidertypen von den grundlegenden Gegenmaßnahmen konkret anvisiert werden. In Verbindung mit den Ergebnissen aus Abschnitt 3.5 lässt sich daraus ableiten, welche Insiderbedrohungen damit verknüpft sind. Umgekehrt wird erläutert, welche konkreten Insiderbedrohungen von den grundlegenden Gegenmaßnahmen anvisiert werden, womit Rückschlüsse auf die dafür benötigten Insidercharakteristiken ermöglicht werden. Schließlich erlauben diese Betrachtungen und Verknüpfungen eine Systematisierung der Erkennungs- und Abwehrmaßnahmen anhand der Insidertaxonomie (Forschungsbeitrag B2 aus Abschnitt 1.3). Nicht betrachtet werden dabei allerdings (Insider-) Bedrohungen für die aufgeführten Gegenmaßnahmen selbst.

Aufbau des Kapitels In den Abschnitten 4.1 bis 4.9 werden jeweils grundlegende Erkennungs- und Abwehrmaßnahmen kurz erläutert, einzelnen Forschungsarbeiten im Bereich der Insiderbedrohungen zugeordnet und mit den Insiderbedrohungen aus Kapitel 3 verknüpft. In Abschnitt 4.10 werden diese Verknüpfungen zusammengefasst und die Erkennungs- und Abwehrmaßnahmen aufgeschlüsselt in Voraussetzungen, anvisierte Insidergrade beziehungsweise Insidertypen und anvisierte Insiderbedrohungen. Abschließend wird ein kurzes Fazit in Abschnitt 4.11 gezogen.

4.1 Bewusstseinsbildung und Verhaltenstraining

Das Sensibilisieren für Sicherheitsbedrohungen und das Trainieren von korrektem Verhalten in Bedrohungssituationen ist mittlerweile fester Bestandteil von Sicherheitskonzepten in Unterneh-

men. Es findet sich in den einschlägigen Standards zum Informationssicherheitsmanagement aus der ISO/IEC 27000 Familie und dem IT-Grundschutzkompendium wieder. Im internationalen Standard ISO/IEC 27002 [ISO13] beschreibt Abschnitt 7.2.2 Vorgaben zu *Information Security Awareness, Education and Training*. Das nationale IT-Grundschutzkompendium [Bun19] beinhaltet den Baustein *ORP.3 Sensibilisierung und Schulung*, in dem verschiedene Maßnahmen für verschiedene Schutzbedarfe festgeschrieben sind. Beispielsweise muss das Personal in den sicheren Umgang mit IT-Komponenten eingewiesen werden (M3). Auch sollte der Lernerfolg gemessen und ausgewertet werden (M8).

Das unmittelbare Ziel dieses Sicherheitsmechanismus ist einerseits die Verringerung von unbewusstem Fehlverhalten und versehentlichem Hervorrufen von Sicherheitsbedrohungen. Andererseits soll Unwissenheit und Naivität abgebaut werden, die aktiv von Angreifern auf verschiedenen Wegen und durch geschickte Techniken ausgenutzt werden können. So kann zum Beispiel ein Angreifer die Neugierde von Mitarbeitern eines Unternehmens ausnutzen und manipulierte USB-Sticks auf dem Parkplatz des Unternehmens verteilen. Sofern die Mitarbeiter nicht wissen, dass allein das Anschließen eines manipulierten USB-Sticks ein System kompromittieren kann, hat ein solcher Angriff sehr hohe Erfolgswahrscheinlichkeiten, wie Tischer u. a. [Tis+16] in einem Experiment zeigen konnten. Voraussetzung für die Wirksamkeit dieses Sicherheitsmechanismus ist eine legitime Zugehörigkeit zum Unternehmen, das die Sensibilisierung und das Verhaltenstraining veranlasst. Nur dann ist sichergestellt, dass der Mechanismus die korrekte Zielgruppe erreicht.[1]

Unter Einbezug dieser Beobachtungen geht es um die Abwehr von Bedrohungen, die durch C_A-Insider hervorgerufen werden, die Handlungen mit der *Intention* in der Ausprägung *accidental* durchführen (vgl. Abschnitt 3.2.1.1). Die Bewusstseinsbildung und das Verhaltenstraining bereiten Insider darauf vor, dass sie Angriffsziele sowohl für Outsider- als auch für Insiderangreifer sind, die versuchen, eine Weitergabe des Insidergrades zu erreichen und damit ihren eigenen Insidergrad zu eskalieren.[2] Weiterhin kann Unachtsamkeit und Fehlverhalten sowohl Schwachstellen für Angreifer eröffnen als auch die Verfügbarkeit von Informationen, Ressourcen und Diensten negativ beeinträchtigen. Im Fokus stehen dabei also insbesondere Bedrohungen für eine Domäne, die durch die unbewusste und unbefugte Weitergabe sowie Verhinderung von Insidergraden entstehen (vgl. Abschnitte 3.2.2 und 3.2.4), wobei die unbewusste und unbefugte Eskalation von Insidergraden (vgl. Abschnitte 3.2.3 und 3.3.2) durch die Zielgruppe nicht ausgeschlossen ist.

Es wird deutlich, dass die Abwehr von Insiderbedrohungen der Kern dieses Sicherheitsmechanismus ist. Folgerichtig setzen Greitzer u. a. [Gre+08] ihn explizit in diesen Kontext und zeigen einen interaktiven und Fallbeispiel-basierten Workshop-Ansatz auf, der dabei unterstützen soll. Auch Silowash u. a. [Sil+12] zeigen in Abschnitt *Practice 3: Incorporate insider threat awareness into periodic security training for all employees* schützende Maßnahmen auf, die die Inhalte der genannten Sicherheitsmanagement-Standards auf Bedrohungen für und durch Insider konzentrieren.

Im Hinblick auf die Insidertaxonomie richtet sich diese Sicherheitsmaßnahme also auf den Schutz vor Bedrohungen der unbefugten Weitergabe, Eskalation und Verhinderung von Insidergraden durch C-Insider. Mit den Erkenntnissen über die Zuordnung der genannten Bedrohungen

1. Ob und wie sich die Zielgruppe anschließend an die Vorgaben und Richtlinien der Domäne hält, wird hier nicht näher betrachtet, ist aber durchaus kritisch zu hinterfragen [SS10].
2. Die unbefugte Weitergabe von Insidergraden bedeutet für die empfangende Partei eine unbefugte Eskalation des Insidergrades. Im Fokus steht hier allerdings die unbefugte Weitergabe, da die weitergebende Partei mit dieser Sicherheitsmaßnahme anvisiert wird.

zu den Insidertypen der Taxonomie aus Abschnitt 3.5 ergibt sich der kombinierte Insidertyp C-K-P-T-Insider als stärkster Insiderbedrohungsagent, der theoretisch durch diese Sicherheitsmaßnahme abgewehrt werden kann. Praktisch hängt es davon ab, welche der Insiderbedrohungen durch die Bewusstseinsbildung und das Verhaltenstraining abgedeckt werden. Hinzu kommt, dass trotz der breiten Abdeckung von Insiderbedrohungen durch diese Sicherheitsmaßnahme nicht vergessen werden darf, dass die Umstände, welche die anvisierten Insiderbedrohungen ermöglichen, dadurch nicht ausgeräumt werden, sondern nur das unbewusste Hervorrufen dieser Bedrohungen reduziert wird. Sofern ein Insider allerdings bewusst Insidergrade weitergeben, eskalieren oder verhindern möchte, bleiben dafür alle vorhandenen Gelegenheiten offen.

4.2 Funktionstrennung und minimale Rechtevergabe

Mit der Trennung von Zuständigkeiten für verschiedene Aufgaben in einem Unternehmen (engl. separation/segregation of duties) können zugeschnittene Zugriffsberechtigungen für spezielle Aufgaben und nur an spezielle mit diesen Aufgaben betraute Personen oder Personengruppen vergeben und auch wieder entzogen werden. Das IT-Grundschutzkompendium [Bun19] schreibt dieses Prinzip als Maßnahme M4 für den Baustein *ORP.1 Organisation* vor und auch in ISO/IEC 27002 [ISO13] findet sich die Vorgabe *6.1.2 Segregation of Duties*. Das häufig damit in Verbindung stehende Prinzip der minimalen Rechtevergabe (engl. least privileges principle) sorgt ebenfalls dafür, dass Personen nur diejenigen Zugriffsrechte besitzen, die sie für die Erfüllung ihrer Aufgabe tatsächlich benötigen. Im IT-Grundschutzkompendium ist dieses Prinzip im Baustein *ORP.1.M5 Vergabe von Berechtigungen* festgeschrieben [Bun19].

In der Literatur zu Insiderbedrohungsabwehrmaßnahmen findet sich das Prinzip ebenfalls als wichtiger Bestandteil. Silowash u. a. [Sil+12] beschreiben in Abschnitt *Practice 8 Enforce separation of duties and least privilege* die Vorteile, die die Funktionstrennung sowie die minimale Rechtevergabe für ein Unternehmen mit sich bringen. So limitiert die Aufteilung von Zuständigkeiten auf mehrere Personen sowie die Reduzierung der Zugriffsrechte die Möglichkeiten, die einzelne Mitarbeiter haben, um Angriffe durchzuführen. Viele Angriffe sind dann nur noch mithilfe einer Kollaboration unter mehreren Mitarbeitern möglich, wofür den Autoren zufolge eine natürliche Hemmschwelle besteht. Kandias, Virvilis und Gritzalis [KVG13] beschreiben die Funktionstrennung als eine der wichtigsten Maßnahmen zur Limitierung von potenziellen Insiderbedrohungen im Kontext von Cloud-Computing. Fuchs und Pernul [FP12b] stellen ein Identitätsmanagementsystem vor, mit dem sie Insiderbedrohungen abwehren wollen. Mit diesem System kann die Vergabe von Zugriffsrechten nach dem Prinzip der Funktionstrennung an weitreichende Bedingungen geknüpft werden, die mit herkömmlichen Zugriffskontrollmechanismen nicht durchsetzbar sind. Eine etwas unorthodoxe Meinung zur minimalen Rechtevergabe wird von Sinclair und Smith [SS10] vertreten, die aus eigener Erfahrung aufzeigen, dass eine solche zu verpassten Marktchance von Unternehmen oder verstorbenen Patienten im Gesundheitswesen führen kann.

Das allgemeine Ziel bei der Etablierung dieser beiden Prinzipien ist die Reduktion von unnötigen Berechtigungen und die Trennung verschiedener benötigter Berechtigungen, um deren Nutzung besser eingrenzen und kontrollieren zu können. Beispielsweise muss nur die Personalabteilung eines Unternehmens Zugriff auf die Personalakten der aktuell angestellten Mitarbeiter haben. Ebenso braucht das Reinigungsteam nur während der Reinigung einen Generalschlüssel für alle Räume und kann diesen im Anschluss an die ausgeführte Aufgabe wieder abgeben. Potenziell können damit also all jene Bedrohungen abgewehrt werden, für die *Privileges* benötigt werden.

Das sind im Detail die *Uncertainty*-Eskalation, die *Privileges*- und *Uncertainty*-Weitergabe sowie die *Credentials*-, *Knowledge*-, *Privileges*- und *Trust*-Verhinderung.

4.3 Kenntnis nur bei Bedarf

Die legitimierte Weitergabe von Wissen beziehungsweise der Zugriff auf spezielle Informationen kann auf jenes Maß beschränkt werden, das für eine Aufgabenerfüllung oder Funktionsbeschreibung höchstens benötigt wird (engl. need to know). Aleman-Meza u. a. [Ale+05] verwenden dieses Prinzip, um den Kontext einer Aufgabe zu definieren, der in Verbindung mit Ähnlichkeitsmetriken die Relevanz von Dokumenten, auf die zugegriffen wird, für diesen Kontext bestimmt. Damit versuchen die Autoren das Problem zu lösen, ob ein Dokumentenzugriff für einen Insider beziehungsweise dessen Aufgabenerfüllung notwendig und damit erlaubt ist oder nicht. Maloof und Stephens [MS07] setzten ebenfalls diesen Sicherheitsmechanismus im Kontext der Abwehr von Insiderbedrohungen ein und versuchen bösartige Aktivitäten von Insidern zu erkennen, die zwar ihre gegebenen Zugriffsrechte respektieren und in dieser Hinsicht unauffällig agieren, die aber ihre Zugriffsrechte verwenden, um an Informationen zu gelangen, die sie eigentlich für ihre Aufgabenerfüllung nicht benötigen. Die Autoren stellen ein System vor, welches mit Kontextinformationen über einzelne Mitarbeiter eines Unternehmens sowie mit einem Vergleich des Verhaltens zwischen Mitarbeitern in ähnlichen oder gleichen Mitarbeiterrollen Hinweise über Verletzungen des Need-to-Know-Prinzips sammelt.

Voraussetzung für diese Sicherheitsmaßnahme ist nicht zwingend eine Zugehörigkeit zur Domäne, da auch explizit die Rolle von Outsidern in diese Festlegung mit einbezogen werden kann. Einerseits kann damit die unbefugte *Knowledge*-Eskalation abgewehrt werden, indem diese *Knowledge* in Fällen, wo sie nicht benötigt wird, auch nicht zur Verfügung gestellt wird. Andererseits können durch die Beschränkung von herausgegebener *Knowledge*, also der befugten Verhinderung von *Knowledge*, Bedrohungen durch K-Insider gar nicht erst entstehen.

Die Insiderbedrohung durch *Knowledge*-Eskalation geht sowohl von Outsidern als auch von allen Insidertypen aus, da zum einen kein Insidergrad für eine unbefugte *Knowledge*-Eskalation zwingend erforderlich ist und zum anderen ein bereits vorhandener Insidergrad eine verbesserte Ausgangslage bei der *Knowledge*-Eskalation bieten kann. Im Allgemeinen kann demnach mit dieser Sicherheitsmaßnahme die Bedrohung der *Knowledge*-Eskalation durch jeden beliebigen Insidertypen erkannt und abgewehrt werden. Im speziellen werden allerdings häufig bei der Erkennung der Verletzung des Need-to-Know-Prinzips zusätzlich weitere Insidercharakteristiken vorausgesetzt, wie etwa in den genannten Fällen von Aleman-Meza u. a. [Ale+05] und Maloof und Stephens [MS07]. Dort fokussiert sich der jeweilige Sicherheitsmechanismus auf die *Knowledge*-Eskalation speziell durch erlaubte Zugriffe auf Dokumente. Somit werden vorhandene *Privileges* vorausgesetzt und die Erkennungs- und Abwehrmaßnahme richtet sich damit im Hinblick auf die Insidertaxonomie auf den Schutz vor der *Knowledge*-Eskalation durch P-Insider.

4.4 Schwachstellenanalyse und -behebung

Die aktive Suche nach Sicherheitsschwachstellen in der IT-Infrastruktur sowie der Hard- und Software eines Unternehmens (engl. security assessment / penetration testing) sowie deren Analyse dient in erster Linie dazu, Schwachstellen jeglicher Art zu finden und zu schließen, bevor

sie von Angreifern gefunden und für bösartige Zwecke ausgenutzt werden können [SM15]. Dabei wird häufig die Rolle eines gutartigen Angreifers (engl. white hat hacker) eingenommen und Teile der Angriffskette durchlaufen (vgl. [HCA11] und Abschnitt 3.3.1). Die Erkenntnisse werden in einem Report detailliert aufgeführt, sodass die gefundenen Schwachstellen im Anschluss behoben werden können.

Der Fokus der Schwachstellenanalyse und -behebung ist die Prävention von unbefugten Eskalationen und Verhinderungen der Insidergrade, wie sie in Abschnitt 3.3.1 als Teilschritte der Angriffskette von Hutchins, Cloppert und Amin [HCA11] bereits schrittweise identifiziert und diskutiert wurden. Somit können theoretisch alle Kombinationen von Insidertypen anvisiert werden. Tatsächliche Insiderbedrohungen werden damit allerdings nicht erkannt und die Prävention kommt erst dadurch zustande, dass nach der Schwachstellenanalyse andere Sicherheitsmaßnahmen eingeleitet werden, um diese Schwachstellen zu beheben. Beispielsweise kann mit dem Ergebnis einer *Trust*-Eskalation aus einer Schwachstellenanalyse der entsprechende Mitarbeiter, der als Gegenpart *Trust* unbefugt weitergegeben hat, für sein unbewusstes Fehlverhalten sensibilisiert und für ein angemessenes Verhalten in derartigen Situationen geschult werden (vgl. Abschnitt 4.1). Daraus wird ersichtlich, dass einerseits die Schwachstellenanalyse je nach Intensität und Ausrichtung nur das Potenzial für bestimmte Bedrohungen identifizieren kann, aber an und für sich keine Sicherheitsmaßnahme darstellt. Erst die Schwachstellenbehebung leitet die eigentlichen Sicherheitsmaßnahmen ein und richtet somit den Fokus auf Insidertypen. Diese Schwachstellenbehebung kann andererseits eine Vielzahl an Teilmaßnahmen beinhalten und jede Teilmaßnahme deckt verschiedene Bedrohungsaspekte ab. Diese Aspekte können in der Praxis die anvisierten Kombinationen von Insidertypen einschränken.

Die Autoren Chinchani u. a. [Chi+05] schlagen eine Art der Schwachstellenanalyse vor, um alle möglichen Angriffswege in einem Unternehmen zur Erreichung eines Angriffsziels für einen Insiderangreifer zu identifizieren und mithilfe von Kostenabschätzungen zu bewerten. Die Annahme der Autoren ist, dass ein Insiderangreifer oftmals verschiedene Angriffe ausführen muss, um zusätzliche Informationen oder Berechtigungen zu bekommen und dadurch seine Angriffsstellung zu verbessern und gegebenenfalls auch von einer besseren örtlichen Position den Angriff fortzuführen. Sie modellieren potenzielle Angriffswege als gewichtete Kanten in einem gerichteten Graphen und versuchen damit, den günstigsten Angriffsweg für einen Insiderangreifer zu identifizieren. Damit bleibt es allerdings bei der Schwachstellenanalyse und eine eigentliche Abwehr von Insiderbedrohungen bleibt dadurch offen.

4.5 Plausibilität mehrerer Datenquellen

Das Prinzip der Plausibilität mehrerer Datenquellen (engl. veracity) wird von Gollmann [Gol12] diskutiert und setzt voraus, dass mehrere Sensoren zur Aufzeichnung von Ereignissen in einer Domäne existieren, die nicht alle gleichzeitig unter der Kontrolle eines Angreifers sind. Die Manipulation von Sensoren oder von Sensordaten durch einen Angreifer kann dadurch in Inkonsistenzen mit anderen Sensordaten stehen. Diese Inkonsistenzen lassen sich durch weitere geeignete Maßnahmen erkennen und helfen bei der Analyse der Vorgänge, die durch die Manipulationen verschleiert werden sollten.

Dieses Prinzip konzentriert sich somit nicht vornehmlich auf die Erkennung von Anomalien in Sensordaten, sondern versucht sicherzustellen, dass die Erkennung und Abwehr von Bedrohungen anhand von Sensordaten auch dann zuverlässig funktioniert, wenn Bedrohungen durch Insider

deren Funktionalitäten negativ beeinflussen. Eine derartige negative Beeinflussung lässt sich als *Uncertainty*-Eskalation identifizieren. Sie kann sowohl durch K-Insider ausgelöst werden, indem das vorhandene Insiderwissen genutzt wird, um die Sensoren zu umgehen. Aber auch P-Insider sind dazu in der Lage, mit ihren Zugriffsrechten die Sensoren oder die Sensordaten zu verändern oder zu zerstören.

Maybury u. a. [May+05] setzen unter anderem diesen Sicherheitsmechanismus in einer integrierten Architektur zur Erkennung von Insiderbedrohungen ein und verwenden eine Vielzahl von Datenquellen und Sensoren, die Indikatoren für bösartiges Insiderverhalten liefern. Dazu gehören etwa die Unterdrückung oder Manipulation von Informationen. Auch Bowen u. a. [Bow+10] verwenden das Veracity-Prinzip, um durch eine Kombination von multiplen Sichtweisen auf ein und dasselbe Ereignis eine bessere Fehlerrate ihrer entwickelten Anomalieerkennungstechnik zu erreichen. Dadurch werden beide Systeme mit einem Sicherheitsmechanismus gegen die Insiderbedrohung der *Uncertainty*-Eskalation durch K- oder P-Insider ausgestattet.

4.6 Verräterrückverfolgung

Die Rückverfolgung von Informationen, die vertraulich nur für bestimmte Personen oder Personenkreise zur Verfügung gestellt wurden (engl. traitor tracing), ist eine spezielle Form der Erkennung von Datenabflüssen (s. Abschnitt 4.8). Sie setzt allerdings nicht auf die Erfassung von unerwünschtem Zugriffsverhalten der Bedrohungsagenten, sondern arbeitet mit verschiedenen Informationsversionen, die mit den Identitäten der involvierten Personen verknüpft wird [PG11]. Sobald die Information von einer Person an unautorisierte Dritte weitergegeben wird, lässt sich anhand der Version dieser Information feststellen oder zumindest eingrenzen, wer diese Information weitergegeben haben muss beziehungsweise an einer Weitergabe beteiligt war.

Große Anstrengungen bei der Erforschung und Umsetzung dieses Sicherheitsmechanismus gehen auf die Film- und Musikindustrie zurück, die das unerlaubte Kopieren und Weitergeben von lizenziertem Material erkennen und entsprechend bestrafen möchte. Für diesen Kontext haben etwa Jin, Lotspiech und Nusser [JLN04] eine Technik der Einbettung von Wasserzeichen entwickelt und optimiert, die mit der Veröffentlichung von illegal kopiertem Film- oder Musikmaterial Rückschlüsse auf die Person zulässt, durch die Kopien angefertigt wurden. Yu u. a. [Yu+14] wenden ebenfalls eine Technik zur Einbettung von Wasserzeichen an und setzen diese im Kontext der Insiderbedrohungen für Gesundheitsdaten in Cloud-Umgebungen ein. In den letzten Jahren wurden zudem wiederholt Enthüller von geheimen Dokumenten der National Security Agency durch diese Art der Rückverfolgung überführt, indem die unbefugt angefertigten und verdeckt veröffentlichten Papierausdrucke mit einem Wasserzeichen versehen wurden, das für das bloße Auge nicht sichtbar war [Hol17]. Papadimitriou und Garcia-Molina [PG11] gehen einen anderen Weg bei der Umsetzung dieses Sicherheitsmechanismus und analysieren optimale Strategien, mit denen eine Domäne einzelne Teile von Informationen an beteiligte Personen derart verteilen und dokumentieren kann, dass die Kenntnisnahme von offengelegten Informationsteilen und deren Konstellation Aufschlüsse darüber gibt, von welcher Person diese Konstellation von Informationsteilen weitergegeben wurde.

Die Rückverfolgung von Informationsoffenlegungen eröffnet, unabhängig von der konkreten Ausprägung, die Möglichkeit der Erkennung von Bedrohungen durch die Weitergabe von Informationen und Dokumenten aller Art. Damit wird deutlich, dass es sich um die Insiderbedrohungen der unbefugten *Credentials*- und *Knowledge*-Weitergabe handelt (vgl. Abschnitte 3.2.2.1

und 3.2.2.2), wobei ersteres nur für Authentisierungsarten gilt, die auf Besitz oder Wissen basieren. Die anvisierten Insidertypen lassen sich somit als C- sowie K-Insider identifizieren.

Eine Besonderheit besteht bei diesem Mechanismus darin, dass die unbefugte Weitergabe nur dann erkannt werden kann, wenn die empfangende Partei, die dadurch ihre *Credentials* oder *Knowledge* eskaliert hat, diese Informationen oder Daten offenlegt. Also ist eine Erkennung der unbefugten *Credentials*- oder *Knowledge*-Eskalation auf Seiten der empfangenden Partei Voraussetzung, bevor der Insider zurückverfolgt werden kann, der die Informationen oder Daten weitergegeben hat.

4.7 Köderdokumente

Köderdokumente (engl. decoy documents / honeytokens) werden innerhalb einer Domäne eingesetzt, um Angreifern eine Falle zu stellen, die bereits eine Art von Insidergrad besitzen [Spi03b]. Diese Köderdokumente sollen den Anschein von echten Dokumenten erwecken, die scheinbar wichtige und wertvolle Informationen enthalten. Sie werden innerhalb der Domäne ausgelegt und haben keinen Bezug zu Aufgaben, die von Insidern erledigt werden müssen. Daher gibt es auch keinen Anlass, der eine Interaktion mit diesen Köderdokumenten legitimiert. Das Öffnen beziehungsweise Interagieren stellt somit unerwünschtes Verhalten dar, das mit speziellen Vorkehrungen erkannt werden kann.

Spitzner [Spi03a] schlägt diesen Sicherheitsmechanismus vor, um Insiderbedrohungsagenten zu einem internen Ködersystem (engl. honeypot) zu locken und deren Aktivitäten eingehender zu überwachen und zu untersuchen. Die Köderdokumente beinhalten dabei die Adresse eines als Aktivsystem getarnten Honeypots sowie speziell generierte Zugangsdaten zu diesem Honeypot. Eine alleinige Interaktion mit dem Honeypot stellt laut Autor zunächst noch keinen besonderen Hinweis auf einen Insiderbedrohungsagenten dar. Die Verwendung der generierten Zugangsdaten ist jedoch eindeutig unerwünschtes Verhalten, das auch darauf rückschließen lässt, an welcher Stelle diese Köderinformationen erlangt wurden. Das kann bei der Aufklärung helfen, welcher Insider sich genau hinter dem Bedrohungsagenten verbirgt. Dieses Konzept wird auch von Bowen u. a. [Bow+10] verwendet, die Köderdokumente sowohl mit generierten Zugangsdaten, deren Nutzung überwacht werden, als auch mit Signalfunktionen, die beim Öffnen der Dokumente ein verdecktes aber registrierbares Ereignis auslösen, in einer Domäne auslegen. Ein Patent von Shulman, Cherny und Dulce [SCD17] aus dem Jahr 2017 macht sich diesen Sicherheitsmechanismus ebenfalls zunutze und platziert Köderdokumente auf den Nutzerendgeräten in einer Domäne. Die Nutzung der Informationen in den Köderdokumenten kann überwacht werden und erlaubt eine direkte Verknüpfung mit dem Nutzerendgerät, auf dem das Köderdokument platziert wurde.

Die Erkennung von Ereignissen, die mit Köderdokumenten in Zusammenhang stehen, also die Interaktion mit derartigen Dokumenten oder die Verwendung von darin enthaltenen speziell vorbereiteten Informationen, deutet sehr stark darauf hin, dass ein P-Insider mit den ihm zur Verfügung gestellten Zugriffsrechten unbefugt versucht, seine *Knowledge* zu eskalieren. Sofern die Köderdokumente augenscheinlich wichtige *Credentials* für die Domäne enthalten und der P-Insider diese anschließend auch verwendet, eskaliert er seine *Credentials*. Das besondere an Köderdokumenten ist, dass es sich nicht unbedingt um korrektes Insiderwissen oder um tatsächlich echte *Credentials* handeln muss. Dadurch, dass allerdings mehrere Bedrohungsaktionen damit speziell überwacht und evaluiert werden können, lassen sich Rückschlüsse auf die Absicht

eines Insiders machen, unter der Annahme, dass Köderdokumente sehr wahrscheinlich nur von Insidern gefunden und genutzt werden, die eine bösartige Absicht haben. Dem Einsatz von Köderdokumenten liegt ein gewisser Überwachungsaspekt zugrunde, der auf das Verhalten von Nutzern ausgerichtet ist. Dadurch werden Möglichkeiten geschaffen, einzelne Aktivitäten oder Interaktionsabläufe nachvollziehbar und analysierbar zu machen. Das wiederum reduziert die *Uncertainty* der betroffenen Nutzer und wirkt damit Verbesserungen in der Verdecktheit von Insiderbedrohungen (vgl. Definition 3.3 in Abschnitt 3.2.5) entgegen.

4.8 Erkennung und Abwehr von Datenabflüssen

Die Erkennung und Abwehr von Datenabflüssen (engl. data leakage detection / prevention) befasst sich mit der unautorisierten Verbreitung von Daten und Informationen durch Personen, die erlaubten Zugriff auf diese Daten und Informationen haben [SER12, Kapitel 2]. Dabei können zum Beispiel Verfahren zum Einsatz kommen, welche die Inhalte von Dokumenten an speziellen Datenflusspunkten tiefgreifend analysieren und über die Weiterleitung oder die Unterbindung des Datenflusses anhand von Signaturen oder anhand von Anomalien entscheiden [Mog07]. Andere Techniken setzen etwa auf den Kontext von Datenflüssen, der basierend auf Metadaten der Dokumente oder des Datenflusses diese Entscheidung ermöglicht.

Liu u. a. [Liu+09] stellen ein Framework vor, welches unter anderem Signaturen von den Inhalten spezieller Dokumente erstellt und Datenflussinhalte auf diese Signaturen überprüft. Dabei schlagen die Autoren auch Möglichkeiten vor, wie verdeckte Kommunikationsinhalte auf die Signaturen überprüft und verdeckte Kommunikationen aufgedeckt werden können. Shu u. a. [Shu+16] konzentrieren sich auf Methoden, die definierte Muster in Datenflüssen trotz komplexer Transformationen, wie etwa Einfügungen oder Löschungen, erkennen können.

Dieser Sicherheitsmechanismus nimmt ausschließlich Insiderbedrohungen in den Fokus, da bereits vorhandene *Knowledge* vorausgesetzt wird. Das Ziel ist dann die Erkennung und Abwehr von Bedrohungen durch die Weitergabe dieser *Knowledge*, für deren Vorbereitung oder Durchführung vorhandene *Privileges* eingesetzt werden, zum Beispiel die Zugriffsrechte auf Dokumente mit den anvisierten Informationen. Genau wie beim Einsatz von Köderdokumenten in Abschnitt 4.7 liegt der Erkennung und Abwehr von Datenabflüssen ein gewisser Überwachungsaspekt zugrunde, der auf das Verhalten von Nutzern ausgerichtet ist. Dadurch wird die *Uncertainty* der betroffenen Nutzer reduziert.

4.9 Anomalieerkennung und -abwehr

Sicherheitsmechanismen im Bereich der Anomalieerkennung und -abwehr haben alle das Ziel, Handlungen zu erkennen, die ein System von einem Normal- beziehungsweise gewünschten Zustand in einen davon abweichenden Zustand überführen [CBK09]. Die Überwachung möglichst vieler und im Idealfall aller Aktivitäten innerhalb einer Domäne ist Ausgangspunkt für diese Erkennung und Abwehr von unerwünschten Abweichungen. In der Literatur werden Anomalien per se als diejenigen Abweichungen definiert, die unerwünscht oder bösartig sind [Kan+10; Mat+10; McG+15; GSB17]. Dabei kann die Distanz zu vorgefertigten gutartigen Regeln oder Profilen (engl. white list) oder die Ähnlichkeit zu vorgefertigten bösartigen Regeln oder Profilen (engl. black list) zugrunde gelegt werden. Die Autoren Chandola, Banerjee und Kumar [CBK09]

identifizieren darüber hinaus unterschiedliche Ansätze und Ausrichtungen der Anomalieerkennung. Dazu gehört etwa der Abgleich mit einer Liste von erlaubten beziehungsweise verbotenen Regeln oder Signaturen, die bereits bekannte Merkmale haben. Der Fokus liegt hierbei nicht auf Datenabflüssen, wie in Abschnitt 4.8, sondern allgemeiner auf Aktivitäten oder Zuständen, die einen negativen Einfluss auf eine Domäne haben können. So kann etwa mit einer Liste von Schadsoftwaresignaturen sowohl in einem Rechnernetz als auch im Dateisystem einzelner Nutzerendgeräte überprüft werden, ob diese Signaturen vorhanden sind und somit auf Aktivitäten von Schadsoftware innerhalb der Domäne hinweisen. Ein weiterer von Chandola, Banerjee und Kumar [CBK09] betrachteter Ansatz ist die Aufzeichnung von detaillierten Aktivitäten in einem System und die davon ausgehende Erstellung von Aktivitäts- und Zustandsprofilen, die einen Normalzustand darstellen und die eine Erkennung und Abwehr von unerwünschten Abweichungen zu diesen Profilen ermöglichen. Dabei können Verfahren des maschinellen Lernens und der künstlichen Intelligenz zum Einsatz kommen, die in einer Trainingsphase erlernen, wie normale Aktivitäten aussehen und in einer anschließenden Testphase evaluieren, ob und wie stark neue Aktivitäten von den erlernten normalen Aktivitäten abweichen.

Die Autoren Liu u. a. [Liu+09] konzentrieren sich in ihrer Arbeit explizit auf Insiderbedrohungen. Sie stellen, wie in Abschnitt 4.8 bereits ausgeführt, ein Framework zur Erkennung und Abwehr von Datenabflüssen vor, das auf Signaturen von sensiblen Daten und der Überwachung von Datenflüssen im Hinblick auf diese Signaturen beruht. Für eine bessere Einschätzung, wie gewisse Datenflüsse dekodiert werden müssen, damit die Inhalte sinnvoll interpretiert werden können, greifen die Autoren darüber hinaus auf Regeln zurück, die anhand von Merkmalen der Datenflüsse passende Anwendungen identifizieren. Auch Sibai und Menascé [SM11] verwenden diesen Sicherheitsmechanismus explizit im Kontext von Insiderbedrohungen und stellen ein entwickeltes Framework vor, das sowohl auf eine spezielle IT-Infrastruktur angepasste Regeln zur Erkennung von schadhaften Eingriffen in die Infrastruktur umsetzt als auch weitreichende und feingranulare Regeln für erlaubtes Zugriffsverhalten von Insidern auf Dienste und Ressourcen der Domäne berücksichtigt. Letztere Regeln schränken beispielsweise Zugriffe in Abhängigkeit von Tageszeiten und den Orten, von denen die Zugriffe getätigt werden, ein. Kandias u. a. [Kan+10] verwenden die Anomalieerkennung, um anhand des Nutzerverhaltens in Verbindung mit psychologischen Profilen für jeden Nutzer in einer Domäne eine Bewertung für das Risiko einer Insiderbedrohung abgeben zu können. Mathew u. a. [Mat+10] stellen ein Verfahren vor, dass Abweichungen von Datenbankzugriffen erkennen kann. Dabei konzentrieren sich die Autoren nicht auf die Datenbankanfragen der Nutzer, sondern auf die daraus resultierenden zurückgelieferten Daten der Datenbank. McGough u. a. [McG+15] stellen ein System namens *Ben-ware* vor, welches nicht nur das Verhalten einzelner Nutzer mit hinterlegten Einzelprofilen vergleicht, sondern zudem auch mit einer aggregierten Norm des gesamten Unternehmens. Gamachchi, Sun und Boztas [GSB17] verbinden diesen Sicherheitsmechanismus mit einer vorgeschalteten Aufbereitung von Nutzerinteraktionen untereinander und mit Ressourcen einer Domäne als Graphen. Aus diesen Graphen berechnen die Autoren verschiedene Graphenparameter für einzelne Nutzer und verwenden die resultierenden Ergebnisse als Grundlage für die Erstellung von Profilen beziehungsweise die Erkennung von Abweichungen. Einen nicht-technischen Ansatz liefern Silowash u. a. [Sil+12], die in den Vorgaben bezüglich Bewusstseinsbildung und Verhaltenstraining (vgl. Abschnitt 4.1) Elemente hervorheben, die dafür sorgen sollen, dass sich Insider mit geschärftem Bewusstsein für Fehlverhalten gegenseitig kontrollieren und überwachen. In diesem Fall werden keine technischen Maßnahmen eingesetzt, sondern Personen überwachen sich gegenseitig und melden unerwünschtes Verhalten.

Die Erkenntnisse über aufgetretene Abweichungen von Aktivitäts- und Zustandsprofilen im Verhalten eines Insiders sind nur schwer zu bewerten, denn an und für sich ist eine solche Anomalie häufig nicht gleichbedeutend mit einer Insiderbedrohungsaktion. In diesen Fällen wird auch von sogenannten *Falsch-Positiven* beziehungsweise von der Fehlerrate der Anomalieerkennung gesprochen, denn eine derart erkannte vermeintliche Anomalie bedeutet eine Fehleinschätzung der Sicherheitsmaßnahme. In allen anderen Fällen können die folgenden Insiderbedrohungen erkannt und abgewehrt werden:

- Von normalem Verhalten abweichende Aktivitäten deuten auf unerwünschte Interaktionen mit den Ressourcen einer Domäne hin. Das bedeutet, dass korrekt erkannte unerwünschte Abweichungen von Normalverhalten oder -zuständen alle Bedrohungen eines P-Insiders anvisieren und stoppen können. Als solche konnten in Abschnitt 3.2 konkret die Eskalation und Weitergabe von *Uncertainty* sowie die Verhinderung von *Credentials*, *Privileges* (und damit möglicherweise von *Knowledge*) und *Trust* identifiziert werden.
- Jegliche Aktivitäten und Interaktionen eines C_M-P-Insiders mit den Ressourcen einer Domäne können als Abweichungen von normalem Verhalten des entsprechenden C_A-Insiders erkannt werden, dessen *Credentials* gestohlen wurden. Dadurch lassen sich zunächst die unerwünschten Interaktionen stoppen und darüber hinaus auf eine bereits vorausgegangene unbefugte *Credentials*-Eskalation schließen, die durch das Zurückziehen der gestohlenen *Credentials* im Nachhinein abgewehrt werden kann.
- Wie in Abschnitt 3.2.5.3 ausgeführt, können viele Insiderbedrohungen durch den Einsatz von *Privileges* verbessert werden. Diese Verbesserungen können ebenfalls durch den Einsatz dieses Sicherheitsmechanismus reduziert werden. Dazu gehören zum Beispiel die Eskalationen von *Credentials*, *Knowledge* und *Trust*, die von jedem Outsider unbefugt durchgeführt werden können und demnach auch von P-Insidern verbessert werden können. Im Detail wurden diese und weitere Verbesserungen allerdings nicht eingehend untersucht. Eine eindeutige Identifizierung ist allerdings die Voraussetzung, um an dieser Stelle den Einfluss des beschriebenen Sicherheitsmechanismus auf die verbesserten Insiderbedrohungen benennen zu können.

Genau wie beim Einsatz von Köderdokumenten in Abschnitt 4.7 sowie bei der Erkennung und Abwehr von Datenflüssen in Abschnitt 4.8 liegt der Anomalieerkennung und -abwehr ein gewisser Überwachungsaspekt zugrunde, der auf das Verhalten von Nutzern ausgerichtet ist. Dadurch wird die *Uncertainty* der betroffenen Nutzer reduziert.

4.10 Einordnung der Sicherheitsmaßnahmen anhand der Insidertaxonomie

In den vorigen Abschnitten dieses Kapitels wurde die Einordnung existierender Sicherheitsmaßnahmen anhand der in Abschnitt 2.6 erarbeiteten Insidertaxonomie vorbereitet. Diese Einordnung kann zusammengefasst anhand der Tabelle 4.1 abgelesen werden. Dabei wurden die folgenden Erkenntnisse gewonnen, die sich als wichtige Unterscheidungsmerkmale bei der Einordnung herausstellen.

1. Die erste Erkenntnis betrifft die Unterscheidung, gegen welches Ziel eine Insiderbedrohungsaktion gerichtet ist. Zum einen lassen sich Sicherheitsmaßnahmen anhand der Insiderbedrohungen beziehungsweise -typen einordnen, vor denen sie Schutz bieten beziehungsweise deren Bedrohungsaktionen auf eine Domäne sie erkennen und gegebenenfalls

Tabelle 4.1: Systematisierung von Sicherheitsmaßnahmen anhand der in Abschnitt 2.6 entwickelten Insidertaxonomie sowie der in Kapitel 3 identifizierten Insiderbedrohungen

Sicherheitsmaßnahme	Voraussetzung	Potenziell (○) und definitiv (●) anvisierter Insidergrad					Erkennung und Abwehr	
		Credentials	Knowledge	Privileges	Trust	Uncertainty	Insidertyp	Insiderbedrohung
Bewusstseinsbildung und Verhaltenstraining	C_A-Insider	○	○	○	○		C-K-P-T-Insider	Alle Weitergaben, Eskalationen und Verhinderungen
Funktionstrennung und minimale Rechtevergabe	–			●			P-Insider	U-Eskalation, P- und U-Weitergabe und alle Verhinderungen
Kenntnis nur bei Bedarf	–						Outsider	K-Eskalation
Schwachstellenanalyse und -behebung	–						–	–
Plausibilität mehrerer Datenquellen	keine high Privileges		●	●			K-P-Insider	U-Eskalation
Verräterrückverfolgung	C_A-Insider, K-Insider	●	●				C-K-Insider	C- und K-Weitergabe
Köderdokumente	P-Insider			●		○	P-U-Insider	C- und K-Eskalation
Erkennung und Abwehr von Datenabflüssen	K-P-Insider		●	○		○	K-P-U-Insider	K-Weitergabe
Anomalieerkennung und -abwehr	P-Insider	○		●		○	C-P-U-Insider	C- und U-Eskalation, U-Weitergabe und alle Verhinderungen

verhindern können, sofern sie selbst nicht das Ziel sind. Zum anderen lassen sich Insidertypen identifizieren, deren Insidergrad eine Bedrohung für die Sicherheitsmaßnahmen selbst darstellen. Ein Beispiel ist der Sicherheitsmechanismus der Anomalieerkennung und -abwehr, der in Abschnitt 4.9 behandelt wurde. Dieser legt den Fokus auf Bedrohungen durch P-Insider. Allerdings lässt sich der Mechanismus durch spezielle, vorhandene *Privileges* sowie durch einen entsprechenden Grad an *Knowledge* umgehen. Die *Privileges* ermöglichen potenziell die Deaktivierung oder Manipulation und die *Knowledge* erlauben möglicherweise die geschickte Umgehung des Mechanismus. Die hier erarbeiteten Klassifizierungen der Sicherheitsmechanismen lassen die Bedrohungen der Mechanismen selbst durch spezielle Insidertypen vorerst außer acht. Eine solche Evaluation sollte in ihrer Wichtigkeit allerdings nicht unterschätzt werden. Nur wenn auch die Sicherheitsmechanismen, die gegen spezielle Insiderbedrohungen in einer Domäne eingesetzt werden, selbst vor Insiderbedrohungen geschützt sind, können sie effektiv ihre Aufgaben erfüllen.

2. Die zweite Erkenntnis zeigt auf, dass Sicherheitsmaßnahmen teilweise gewisse Insidergrade voraussetzen, diese allerdings nicht zwangsläufig auch derjenige Insidergrad ist, der von der Maßnahme anvisiert wird. Das wird zum Beispiel anhand der Arbeiten von Aleman-Meza u. a. [Ale+05] sowie von Maloof und Stephens [MS07] deutlich, welche die Maßnahme *Kenntnis nur bei Bedarf* umsetzen (vgl. Abschnitt 4.3). Die vorgestellten Verfahren setzen vorhandene *Privileges* voraus, sodass P-Insider anvisiert werden. Die umgesetzte Sicherheitsmaßnahme an sich zielt allerdings nicht auf die speziellen Bedrohungen von P-Insidern ab, sondern auf die Erkennung und Abwehr von *Knowledge*-Eskalationen, die nicht primär nur von P-Insidern ausgehen, sondern von Outsidern allgemein und durch vorhandene *Privileges* verbessert werden. Diese Erkenntnis findet in Tabelle 4.1 Beachtung und wird in der Spalte *Voraussetzungen* ersichtlich.

3. Die dritte Erkenntnis macht die Mehrschichtigkeit der Abwehr von Bedrohungen deutlich. Denn eine Abwehr von Bedrohungen, die eine unbefugte Eskalation von Insidergraden zum Ziel haben, verhindert darüber hinaus die Insiderbedrohungen der jeweiligen Insidertypen, die aus der unbefugten Eskalation hervorgegangen wären. So verhindert etwa die Sicherheitsmaßnahme *Plausibilität mehrerer Datenquellen* primär die *Uncertainty*-Eskalation durch K-P-Insider. Damit werden nachgelagert ebenfalls diejenigen Bedrohungen verhindert, die von U-Insidern ausgehen. Diese nachgelagerte Abwehr von Insiderbedrohungen, die aus unbefugten Eskalationen von Insidergraden hervorgehen, lässt sich aus Tabelle 4.1 in Verbindung mit den Bedrohungen der einzelnen Basis-Insidertypen erfassen, die in Abbildung 3.6 in Abschnitt 3.5 aufgeführt sind.

4. Die Unterscheidung in potenziell (○) und definitiv (●) anvisierte Insidergrade weist darauf hin, dass manche Insidercharakteristiken nur dann von Sicherheitsmechanismen anvisiert werden, wenn darauf ein spezieller Fokus gelegt wird. Der Mechanismus *Bewusstseinsbildung und Verhaltenstraining* beispielsweise zielt nicht zwangsläufig auf Bedrohungen durch die Weitergabe von *Trust* ab, wenn diese nicht explizit in die Bewusstseinsbildung und das Verhaltenstraining aufgenommen werden.

4.11 Fazit

Die in den Abschnitten 4.1 bis 4.9 behandelten grundlegenden Maßnahmen und Techniken zur Erkennung und Abwehr von Insiderbedrohungen wurden auf vorausgesetzte Insidergrade sowie

anvisierte Insidertypen und -bedrohungen untersucht. Dabei zeigte sich, dass man von den vorausgesetzten Insidergraden nicht unbedingt auf die anvisierten Insiderbedrohungen schließen kann. So ist etwa die in Abschnitt 4.1 behandelte Bewusstseinsbildung eine Gegenmaßnahme gegen alle unbefugten Weitergaben von Insidergraden (vgl. Abschnitt 3.2.2) und damit auch potenziell gegen alle nachgelagerten unbefugten Eskalationen und Verhinderungen von Insidergraden. Die Voraussetzung ist dabei allerdings nur das Vorhandensein legitimer *Credentials*.

Die qualitative Analyse der grundlegenden Erkennungs- und Abwehrmaßnahmen sowie die Herausarbeitung der Zusammenhänge mit den Insiderbedrohungen aus Kapitel 3 erlaubte eine Systematisierung der Gegenmaßnahmen in Anlehnung an die Insidertaxonomie aus Abschnitt 2.6 und damit die Komplementierung des Forschungsbeitrags B2 (vgl. Abschnitt 1.3). Die Ergebnisse lassen nunmehr fundierte Schlüsse darüber zu, welche Gegenmaßnahmen miteinander kombiniert werden können, um eine höhere Schutzabdeckung gegen Insiderbedrohungen zu erhalten. Weiterhin zeigen sie auf, welche Gegenmaßnahmen miteinander verglichen werden können, da sie möglicherweise identische Insideraspekte anvisieren.

5 Technik zum Schutz vor Insiderbedrohungen

Insiderbedrohungen sind vielfältig und schwer zu unterbinden, wie die Ausführungen in den Kapiteln 2 bis 4 aufzeigen. Die Ursachen lassen sich mit der Tatsache in Verbindung bringen, dass Insider gewisse Berechtigungen, Freiheiten oder Kenntnisse besitzen, um ihre Aufgaben bearbeiten zu können. Unberechtigte Anwendungen oder Übertragungen dieser Insidergrade lassen sich aber eventuell nicht klar definieren oder gar technisch überprüfen beziehungsweise verhindern. Jedoch wurde in Abschnitt 2.3 eine spezielle Art von Insidergrad identifiziert, die einerseits ein großes Verbesserungspotenzial bezüglich aller Bedrohungen für eine Domäne hat, die jedoch auch andererseits mit technischen Hilfsmitteln sehr gut kontrolliert werden kann (vgl. Abschnitt 3.2.5.5). Es handelt sich dabei um die Insidercharakteristik der *Uncertainty*. Eine konsequente Reduzierung von *Uncertainty* erlaubt die Überprüfung von Insideraktivitäten in Echtzeit oder im Nachgang einer Aktivität, ohne dabei einen signifikanten Einfluss auf die Erfüllung von Aufgaben seitens der Insider zu nehmen. Dafür muss man an dieser Stelle allerdings zunächst von negativen (datenschutz-)rechtlichen und ethischen Implikationen absehen. Für eine Betrachtung des rechtlichen Rahmens und des Datenschutzes wird auf die Kapitel 6 und 7 verwiesen.

Ausgehend von diesen Erkenntnissen bezüglich *Uncertainty* wird in diesem Kapitel eine Erkennungs- und Abwehrtechnik von Insiderbedrohungen entwickelt und dessen Arbeitsweise und Feinheiten erläutert. Sie ist primär darauf ausgelegt, die Aktivitäten von Insidern an den Rechnern eines Unternehmens mit einem Linux-Betriebssystem möglichst tiefgründig, lückenlos und detailliert aufzuzeichnen und aus der großen Menge und Dichte an Daten hilfreiche Erkenntnisse zu gewinnen. Dabei spielt die weitgehende Automatisierung ebenso eine Rolle, wie die praktische Umsetzbarkeit und effiziente Arbeitsweise. Dieses Kapitel adressiert damit Forschungsfrage 3 aus Abschnitt 1.2 und legt den Fokus auf die technisch unterstützte Analyse von Insideraktivitäten.

Wesentliche Inhalte Für die Analyse von Insideraktivitäten werden umfassende Daten benötigt, die von den Auditierungssystemen moderner Betriebssysteme bei entsprechend feingranularer Konfiguration bereits zur Verfügung gestellt werden. Auf Basis des Linux-Auditierungssystem wird in diesem Kapitel eine Software entwickelt, die Ereignisnachrichten verschiedener Systemaufrufe konsolidiert und für eine Weiterverarbeitung aufbereitet. Aus diesen detaillierten Informationen über getätigte Systemaufrufe an einem Rechner werden anschließend Graphen erzeugt, die Aktivitäten eines Insiders anhand der Beziehungen zwischen den Ressourcen eines Rechners abbilden. In einem weiteren Schritt werden aus diesen Graphen Teilgraphen isoliert, die einzelne Insideraktivitäten sowie Informationsflüsse repräsentieren und somit gesondert analysiert werden können. Die Entwicklung der Analysemethode von Insideraktivitäten in diesem Kapitel umfasst den Einsatz und die Anpassung einer Graphenstrukturcharakteristik namens Netzwerkmotive, die für die Besonderheiten der neuen Graphen weiterentwickelt wurde. Im Zuge dessen wurden Algorithmen entworfen, implementiert sowie optimiert, die es erlauben, anhand der Netzwerkmotive aussagekräftige Signaturen zu erzeugen. Es stellt sich heraus, dass diese Signaturen für spezielle gleichartige Aktivitäten sehr stabil bleiben und für unterschiedliche

Aktivitäten große Unterschiede aufweisen. Mit der Hinzunahme von geeigneten Ähnlichkeits- und Distanzmaßen wird schließlich eine regelbasierte Erkennung von Insiderbedrohungen vorgestellt, die darüber hinaus auch auf Echtzeitanwendungen erweitert wird und somit die aktive Abwehr von Insiderbedrohungen ermöglicht (Forschungsbeitrag B3 aus Abschnitt 1.3). Diese Erkennungs- und Abwehrtechnik wird anhand von realen Angriffsaktivitäten evaluiert und dessen Eignung bestätigt.

Aufbau des Kapitels Zu Anfang werden die für dieses Kapitel benötigten Grundlagen in Abschnitt 5.1 erläutert. Diese umfassen tiefliegende Funktionen von Betriebssystemen in Form von Systemaufrufen, Begriffe und Notationen aus der Graphentheorie sowie charakteristische Eigenschaften von Graphen, die als Netzwerkmotive bezeichnet werden. In Abschnitt 5.2 wird ein Überblick über existierende Arbeiten gegeben, die sich mit der Erkennung und Abwehr von Insiderbedrohungen anhand von Graphen sowie anhand von Systemaufrufen beschäftigen. Mit Abschnitt 5.3 beginnt der konstruktive Teil dieses Kapitels. Darin werden der vorhandene Auditierungsmechanismus des Linux-Betriebssystems sowie benötigte Anwendungsfälle beleuchtet. Weiterhin wird die Entwicklung eines eigenen Plugins zur Erfassung und Aufbereitung von Ereignisnachrichten in Echtzeit beschrieben. Abschnitt 5.4 umfasst die Einführung und Konstruktion von Systemaufruf-Graphen (SysGraphen), die Insideraktivitäten anhand der Beziehungen zwischen den Ressourcen eines Rechners abbilden. Darüber hinaus wird die Extraktion von Teilgraphen beschrieben, die einzelne Insideraktivitäten und Informationsflüsse isolieren. In Abschnitt 5.5 werden Signaturen von Systemaufruf-Graphen (SysGraphen) basierend auf Netzwerkmotiven entwickelt und alle Besonderheiten im Kontext dieser neuartigen Graphen erläutert. Der Einsatz von SysGraphen sowie die Ähnlichkeit und Distanz deren Signaturen werden in Abschnitt 5.6 auf den Kontext der Erkennung und Abwehr von Insiderbedrohungen übertragen und anhand von drei ausgewählten Bedrohungsszenarien veranschaulicht. In Abschnitt 5.7 wird die bis dahin umfassend erläuterte neu entwickelte Erkennungs- und Abwehrtechnik anhand von realen Angriffsdaten evaluiert und dessen Weiterentwicklungspotenzial in Abschnitt 5.8 aufgezeigt. Abschnitt 5.9 schließt das Kapitel mit einem Fazit ab.

5.1 Grundlagen der Technik zur Erkennung und Abwehr von Insiderbedrohungen

Die folgenden Abschnitte behandeln die Grundlagen der in dieser Dissertation entwickelten Erkennungs- und Abwehrtechnik von Insiderbedrohungen. Diese umfassen *Systemaufrufe*, mit denen Programme grundlegende Funktionen des Betriebssystemkerns nutzen können, *Grundlagen der Graphentheorie*, mit deren Hilfe eine besondere Repräsentation von Systemaktivitäten erzeugt wird, sowie *Netzwerkmotive*, die speziell konstruierte Signaturen dieser Aktivitäten erlauben.

5.1.1 Systemaufrufe

Grundlegende Funktionen, die zur Ausführung von Programmen benötigt werden, wie etwa das Öffnen einer Datei, das Starten weiterer Prozesse oder das Aufbauen einer Rechnernetzverbindung, werden vom Betriebssystemkern über sogenannte *Systemaufrufe* (engl. system calls) bereitgestellt [SGG18, Abschnitt 2.3]. Diese Systemaufrufe unterscheiden sich im Detail je

nach Betriebssystem und nach zugrunde liegender Hardwarearchitektur, weisen aber prinzipiell ähnliche oder gleiche Funktionalitäten auf. Sie werden in der Regel von Programmen nicht direkt, sondern über Wrapper-Funktionen aufgerufen. Diese Wrapper-Funktionen werden als Teil einer hardwarespezifischen Standardbibliothek bereitgestellt und haben meist den gleichen Namen sowie die gleichen Aufrufparameter wie die eigentlichen Systemaufrufe. In Unix-ähnlichen (dazu gehören Linux und MacOS) Betriebssystemen lautet eine solche Standardbibliothek beispielsweise `libc` für C-Programme.

Der Vorteil dieser Vorgehensweise ist die Abstraktion von hardwarespezifischen Details für den Programmierer, der in seinem Programm nur die Wrapper-Funktionen verwenden muss. Damit kann das Programm auf unterschiedlichen Hardwarearchitekturen laufen, für die eine Implementierung der verwendeten Standardbibliothek vorhanden ist. Weiterhin können von den Wrapper-Funktionen spezielle Vorgaben für die Aufrufparameter sowie Fehlerfälle und Funktionsweisen der Systemaufrufe behandelt und umgesetzt beziehungsweise abgefangen und gegebenenfalls gelöst werden. Die Wrapper-Funktionen rufen die eigentlichen Systemaufrufe über eine Systemaufrufschnittstelle des Betriebssystems mithilfe festgelegter architekturspezifischer Nummern auf. Von diesen Vorgängen sowie vom Kontextwechsel in den Betriebssystemkern bekommt das aufrufende Programm nichts mit.

Ein Beispiel für einen Systemaufruf in Unix-ähnlichen System ist `openat()`, mit dessen Hilfe ein Dateideskriptor zu einem angegebenen Dateisystemobjekt erstellt werden kann. Über diesen Dateideskriptor können anschließend, je nach speziellen Berechtigungen, die von der Funktion `openat()` angefordert und bei Erfolg erteilt werden, Daten von einer Datei gelesen oder in eine Datei geschrieben werden, oder beides. Die genauen Details dieses Systemaufrufs finden sich in der Hilfe- und Dokumentationsseite (engl. manual page) von `openat()`:[1]

```
   int   openat(int fd, const char *path, int oflag, ...);
Rückgabewert     a0          a1            a2
```

Im Falle einer fehlerfreien Ausführung stellt bei diesem Systemaufruf der Rückgabewert den neu erzeugten Dateideskriptor dar. Die Werte `a0`, `a1` und `a2` sind die Funktionsargumente. Ein für besondere Situationen benötigtes weiteres Funktionsargument, hier mit „. . . “ gekennzeichnet, kann nachfolgend ignoriert werden. In `path` ist der Pfad zum Dateisystemobjekt enthalten, das geöffnet werden soll. Der Wert `fd` wird nur in den Fällen beachtet, in denen der angegebene Pfad ein relativer Pfad ist. Dann zeigt `fd` auf denjenigen Ordner im Dateisystem, von dem aus der Pfad relativ bestimmt werden soll. Der Wert `oflag` enthält die angesprochenen speziellen Berechtigungen. Diese Zahl ergibt sich aus der `ODER`-Verknüpfung verschiedener Werte, die jeweils in aussagekräftigen Variablen gespeichert sind und unterschiedliche Bedeutungen haben. Dazu gehören unter anderem die folgenden sogenannten Dateistatusindikatoren (engl. file status flags):

- `O_RDONLY` – Öffne ausschließlich zum Lesen,
- `O_WRONLY` – Öffne ausschließlich zum Schreiben,
- `O_RDWR` – Öffne zum Lesen und zum Schreiben,
- `O_APPEND` – Füge die Daten bei jedem Schreiben an das Ende an.

Diese Dateistatusindikatoren werden später noch benötigt. Der `openat()` Systemaufruf und einige der wichtigsten Systemaufrufe, die für den in der vorliegenden Arbeit präsentierten

1. Aufrufbar auf Unix-ähnlichen Systemen mit dem Kommandozeilenbefehl `man 2 openat`.

Tabelle 5.1: Relevante Systemaufrufe Unix-ähnlicher Betriebssysteme und ihre Funktionalitäten

Systemaufrufe	Beschreibung
`openat()`	Erstellt einen Dateideskriptor zu einem angegebenen Pfad
`close()`	Schließt einen Dateideskriptor
`read()`	Liest Daten von einer Ressource
`write()`	Schreibt Daten in eine Ressource
`sendfile()`	Liest Daten von einem Dateideskriptor und schreibt diese in eine Ressource
`unlink()`	Löscht eine Datei an einem angegebenen Pfad
`rename()`	Ändert den Namen oder den Speicherort einer Datei
`exit_group()`	Beendet den aufrufenden Prozess sowie seine Threads
`kill()`	Beendet den angegebenen Prozess sowie dessen Threads[2]
`accept4()`	Empfängt eine Verbindung an einem angegebenen Socketdeskriptor
`connect()`	Initiiert eine Verbindung zu einer angegebene Adresse über einen angegebenen Socketdeskriptor
`sendto()`	Sendet Daten an eine angegebene Adresse
`recvfrom()`	Empfängt Daten von einer angegebenen Adresse

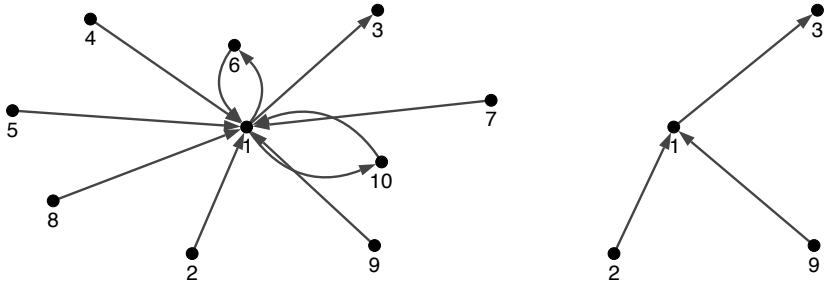
Schutzmechanismus beziehungsweise für dessen aktuelle Implementierung relevant sind, finden sich in Tabelle 5.1.

5.1.2 Grundlagen der Graphentheorie

Ein *Graph* $\mathcal{G} = (V, E)$ ist definiert durch eine Menge von Knoten V (engl. vertices, auch Ecken oder Punkte genannt) und einer Menge von Kanten E (engl. edges, auch Linien genannt). Die Kanten stellen jeweils eine Verbindung zweier Knoten dar. Im vorliegenden Kontext sind nur *gerichtete Graphen* von Interesse, bei denen eine Kante jeweils einen Anfangsknoten und einen Endknoten hat. Die Menge der Kanten E ist also eine Teilmenge aller geordneten Knotenpaare $(a, b) : V \times V$ mit a als Anfangs- und b als Endknoten. Eine geordnete Knotenfolge $(a_0, \ldots, a_n)$ mit allen $a_i \in V$ und $n > 0$ wird Pfad der Länge n zwischen dem Knoten a_0 und dem Knoten a_n genannt, wenn zwischen allen direkt aufeinanderfolgenden Knotenpaaren eine Kante $e_i = (a_i, a_{i+1}) \in E$ für alle $i \in \{0, \ldots, n-1\}$ existiert. Eine Kante $e \in E$ ist demnach der kürzest mögliche Pfad der Länge 1. Die Menge der Knoten wird graphisch üblicherweise als Punkte und die Menge der gerichteten Kanten als Pfeile zwischen diesen Punkten dargestellt. Eine solche Darstellung zweier kleiner Beispielgraphen ist in Abbildung 5.1 illustriert.

Ein Graph $\mathcal{G}' = (V', E')$ wird *Teilgraph* oder *Subgraph* von $\mathcal{G}$ genannt, sofern $V' \subseteq V$ eine Teilmenge von V und $E' \subseteq E$ eine Teilmenge von E ist. Für einen solchen Subgraphen wird dann $\mathcal{G}' \subseteq \mathcal{G}$ geschrieben. Dieser Subgraph wird weiterhin als *induzierter Subgraph* bezeichnet, sofern E' alle Kanten zwischen den Knoten in V' beinhaltet, die auch in E zwischen diesen

2. Die Funktion `kill()` erlaubt das Senden verschiedener Signale an den gegebenen Prozess. Eines dieser Signale ist `SIGKILL`, welches das besagte Beenden des Prozesses nach sich zieht.



(a) $\mathcal{G} = (V, E)$ **(b)** $\mathcal{G}' = (V', E')$

Abbildung 5.1: Ein Beispielgraph $\mathcal{G}$ mit $V = \{1, \ldots, 10\}$ und $E = \{(1, 3), (1, 6), (1, 10), (2, 1), (4, 1), (5, 1), (6, 1), (7, 1), (8, 1), (9, 1)\}$ und ein zugehöriger Subgraph $\mathcal{G}'$ von $\mathcal{G}$

Knoten existieren. Formal ausgedrückt gilt in einem induzierten Subgraphen $\mathcal{G}' = (V', E')$ von $\mathcal{G} = (V, E)$ für alle geordneten Knotenpaare $(a, b) \in V' \times V'$: Wenn $(a, b) \in E$, dann auch $(a, b) \in E'$. Ein induzierter Subgraph $\mathcal{G}' = (V', E') \subseteq \mathcal{G}$ ist also immer eindeutig durch $\mathcal{G}$ und V' definiert und wird daher auch mit $\mathcal{G}' = \langle \mathcal{G}, V' \rangle$ bezeichnet.

Zwei Graphen $\mathcal{G} = (V, E)$ und $\mathcal{G}' = (V', E')$ sind *isomorph* zueinander, sofern eine bijektive Abbildung $\sigma : V \mapsto V'$ existiert, deren inverse Abbildung σ^{-1} ebenfalls bijektiv ist und mit der zusätzlich für alle Knoten $a, b \in V$ gilt: $(a, b) \in E \Leftrightarrow (\sigma(a), \sigma(b)) \in E'$. Der Isomorphismus zwischen den beiden Graphen wird mit $\mathcal{G} \simeq \mathcal{G}'$ beschrieben. Isomorphe Graphen bilden zusammen genommen eine Isomorphieklasse, wobei nachfolgend zwischen isomorphen Graphen nicht weiter unterschieden wird. Jeder beliebige Graph einer Isomorphieklasse kann als Vertreter dieser Klasse herangezogen werden.

Die Kanten eines gerichteten Graphen $\mathcal{G} = (V, E)$ ermöglichen neben einer Nachbarschaftsbeziehung zwischen zwei durch eine Kante verbundenen Knoten auch eine Vorgänger- und eine Nachfolgerbeziehung. Ein direkter Vorgänger eines Knotens b ist ein Knoten a, sofern die Kante $(a, b) \in E$ existiert. Entsprechend ist b durch eine solche existierende gerichtete Kante ein direkter Nachfolger von a. Die Menge aller direkten Vorgänger eines Knotens $b \in V$ wird mit $Pred(b)$ bezeichnet und ergibt sich aus $Pred(b) = \{a \in V \mid (a, b) \in E\}$. Äquivalent dazu wird die Menge aller direkten Nachfolger eines Knotens $a \in V$ mit $Succ(a) = \{b \in V \mid (a, b) \in E\}$ beschrieben. Die Menge aller Nachbarn $Neig(a)$ eines Knotens $a \in V$ ergibt sich dann aus der Vereinigung der Menge aller seiner Vorgänger mit der Menge aller seiner Nachfolger: $Neig(a) = Pred(a) \cup Succ(a)$. Die Mengen $Pred(a)$ und $Succ(a)$ sind nicht notwendigerweise disjunkt, da ein Nachbarknoten b eines Knotens a möglicherweise sowohl Vorgänger- als auch Nachfolgerknoten von a sein kann. Oftmals wird in diesem Fall auch von einer ungerichteten Kante zwischen den beiden Knoten a und b gesprochen. In der vorliegenden Arbeit werden dafür allerdings zwei getrennte entgegengesetzt gerichtete Kanten verwendet, wie in Abbildung 5.1a zu sehen.

Daraus ergibt sich auch der *Grad* $deg(a)$ eines Knotens $a \in V$. Dieser ist bestimmt durch die Anzahl aller eingehenden Kanten $deg_{in}(a) = |Pred(a)|$ von seinen Vorgängerknoten, summiert mit der Anzahl aller ausgehenden Kanten $deg_{out}(a) = |Succ(a)|$ zu seinen Nachfolgerknoten: $deg(a) = deg_{in}(a) + deg_{out}(a) = |Pred(a)| + |Succ(a)|$.

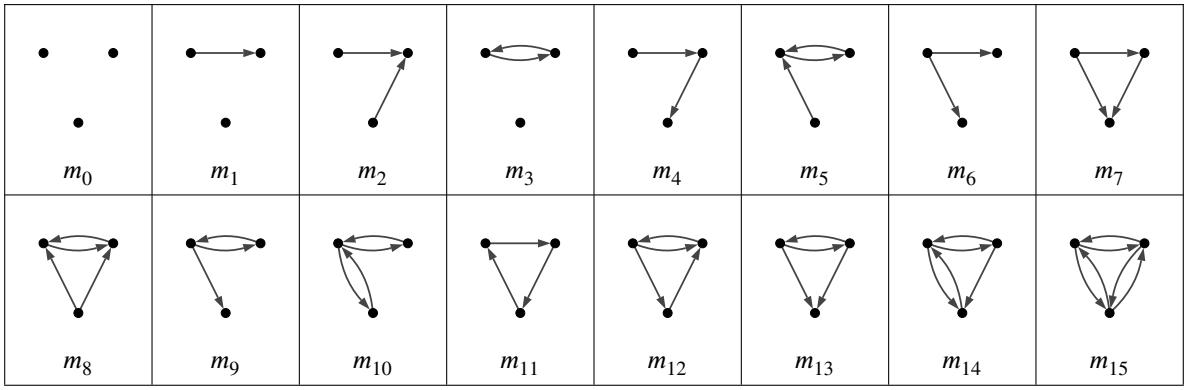


Abbildung 5.2: Alle 16 möglichen 3-Motive

Für detailliertere Betrachtungen der Graphentheorie wird auf das Lehrbuch von Diestel [Die17] verwiesen.

5.1.3 Netzwerkmotive

Eine charakteristische Eigenschaft von Graphen ist die Häufigkeit der im Graph vorkommenden sogenannte *Netzwerkmotive* (engl. network motifs) oder kurz *Motive* [Mil+02]. Sie legen statistisch signifikante Muster in Netzwerken beziehungsweise Graphen offen und erlauben die Identifizierung komplexer Strukturen und ein besseres Verständnis darunter liegender, komplexer Systeme und Vorgänge [MSA03, Abschnitt 8.1]. Es handelt sich bei n-Motiven eines Graphen um alle vorkommenden induzierten Subgraphen (vgl. Abschnitt 5.1.2) mit fester Knotenanzahl n, wobei isomorphe Subgraphen zu einem Motiv beziehungsweise genauer zu einer Isomorphieklasse, in diesem Fall auch Motivklasse genannt, zusammengefasst werden. Die jeweilige Häufigkeit H der vorkommenden n-Motive in einem Graphen wird dabei festgehalten und beschreibt eine charakteristische n-Motiv-Signatur des Graphen. Jeder Subgraph einer Isomorphieklasse ist ein Repräsentant dieser Klasse und beschreibt das gleiche Motiv. Die Begriffe *Motiv*, *Isomorphieklasse eines Motivs* und *Motivklasse* werden in dieser Arbeit synonym verwendet.

Von besonderer Bedeutung für die vorliegende Arbeit sind die 3-Motive, die in der Literatur auch Triaden genannt werden [Jan06, Abschnitt 3.3.2]. Von ihnen existieren insgesamt 16 unterschiedliche Ausprägungen beziehungsweise Isomorphieklassen, die nachfolgend mit $m_0, \ldots, m_{15}$ bezeichnet werden und in Abbildung 5.2 illustriert sind. Ein weiterer Fokus der vorliegenden Arbeit liegt auf jenen 3-Motiven, in denen alle Knoten mindestens einen Grad von 1 haben, also alle Knoten Teil von mindestens einer Kante sind. Somit werden die 3-Motive m_0, m_1 und m_3 aus Abbildung 5.2 nicht weiter betrachtet. Mit diesen Bedingungen ergibt sich für den Graphen $\mathcal{G} = (V, E)$ mit $V = \{1, \ldots, 10\}$ und $E = \{(1, 3), (1, 6), (1, 10), (2, 1), (4, 1), (5, 1), (6, 1), (7, 1), (8, 1), (9, 1)\}$, wie er in Abbildung 5.1 auf Seite 101 visualisiert wurde, die folgende 3-Motiv-Signatur:

m_i:	m_2	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}	m_{11}	m_{12}	m_{13}	m_{14}	m_{15}
$H(m_i)$:	15	6	12	0	0	0	2	1	0	0	0	0	0

Die Autoren Milo u. a. [Mil+02] schlagen eine Bereinigung der Motiv-Signatur eines Graphen vor, um diejenigen Motive aus der Signatur herauszufiltern, die mit hoher Wahrscheinlichkeit zufällig entstanden sind und dadurch keine statistische Relevanz für die Struktur des Graphen und somit auch keinen wichtigen Einfluss auf die Funktion des dahinterliegenden Netzwerks haben. Um das zu erreichen, erzeugen sie ein sogenanntes Nullmodell des Graphen. Dahinter verbergen sich zufällig erzeugte Graphen, die je nach Anforderungen gewisse topologische Merkmale auf niedriger Ebene des Graphen beibehalten. Dazu gehören etwa die Erhaltung von Einzelknoten-Charakteristiken, wie zum Beispiel der jeweilige Grad eines Knotens [MSA03, Abschnitt 8.2]. Mit diesem Nullmodell beziehungsweise genauer mit der durchschnittlichen Anzahl aller vorkommenden n-Motive in diesem Nullmodell des Graphen sowie mit der jeweiligen Standardabweichung errechnen die Autoren die Anzahlen der über- und unterrepräsentierten n-Motive des ursprünglichen Graphen. Durch diesen Vorschlag der Bereinigung wird der Begriff *Motiv* in der Literatur auch teilweise ausschließlich auf diese über- beziehungsweise unterrepräsentierten Subgraphen bezogen, etwa von Wernicke [Wer05] und Kashani u. a. [Kas+09]. Diese sehr aufwendige Bereinigung wird in der vorliegenden Arbeit allerdings umgangen, da die untersuchten Graphen keine mit Zufall behafteten Vorgänge modellieren, deren Rauschen entfernt werden muss. Die hier erzeugten Graphen enthalten hingegen relevante Strukturen, die sich in ihrer Anzahl möglicherweise statistisch nicht signifikant von zufälligen Graphen unterscheiden. Eine Bereinigung würde diese Strukturen aus der Motiv-Signatur entfernen [Mil+02]. Der Begriff *Motiv* wird daher synonym zur einem *induzierten Subgraphen mit fester Knotenzahl* verwendet.

Bei der Erstellung der Motiv-Signatur eines Graphen gibt es zwei Herausforderungen: Erstens das effiziente Finden aller induzierten Subgraphen der festen Knotenanzahl n sowie zweitens die Zuordnung dieser Subgraphen zu einer Isomorphieklasse, also das Finden eines jeweiligen Isomorphismus zwischen den einzelnen gefundenen Subgraphen der festen Knotenzahl n und einem Repräsentanten einer Motivklasse. Die Grundlagen zur Lösung beider Probleme werden für die Signaturberechnung der in diesem Kapitel speziell entwickelten Graphen benötigt (s. Abschnitt 5.5) und daher im Folgenden genauer betrachtet.

5.1.3.1 Auffinden induzierter Subgraphen mit fester Knotenzahl

Ein sehr einfacher Algorithmus, der von den Autoren Milo u. a. [Mil+02, Ref. 18] (vgl. auch Shen-Orr u. a. [She+02]) vorgeschlagen wird, arbeitet sich mit einer vollständigen Suche durch alle Kanten eines Graphen und sucht für jede Kante und deren Knoten solange weitere Kanten, die mit einem der Knoten verbunden sind, bis sich ein induzierter Subgraph mit n Knoten ergibt. Eine Verbesserung dieses Bruteforce-Ansatzes wird von Kashtan u. a. [Kas+04] vorgeschlagen, die nicht alle vorkommenden induzierten Subgraphen mit fester Knotenzahl n in einem Graphen aufsuchen und damit auch nicht die absolute Anzahl an n-Motiven zählen. Ausgehend von einer zufällig ausgewählten Kante des Graphen extrahieren sie stattdessen einen einzelnen induzierten Subgraphen mit fester Knotenzahl n und zählen das damit verbundene Motiv. Mit einer festgelegten Anzahl an Wiederholungen dieses Vorgangs errechnen die Autoren schließlich die Durchschnittswerte aller gezählten Motive, was eine Approximation für die tatsächliche durchschnittliche Anzahl an vorkommenden Motiven darstellt. Problematisch bei diesem Vorschlag ist allerdings die Tatsache, dass trotz der zufälligen Auswahl eines im Graphen vorliegenden Motivs gewisse Motive im Graphen mit einer höheren Wahrscheinlichkeit ausgewählt werden als andere. Diese sogenannte Verzerrung kann für ein gewähltes Motiv anhand des zugrunde liegenden

Graphen berechnet werden und wird von den Autoren kompensiert, indem die Durchschnittswerte der zufällig ausgewählten und gezählten Motive mittels der genannten Wahrscheinlichkeit gewichtet werden. Dieses Vorgehen erfordert allerdings weiteren Berechnungsaufwand.

Eine weitere Effizienzsteigerung wird von Wernicke [Wer05] erreicht, der sowohl einen Algorithmus zur unverzerrten Approximation aller vorkommenden n-Motive, genannt RAND-ESU, als auch einen Algorithmus zum vollständigen Auffinden aller Motive in einem Graphen, genannt ESU, publizierte. Letzterer sieht im Detail folgendermaßen aus:

Algorithmus ENUMERATESUBGRAPHS($\mathcal{G}$, n) (ESU)
Eingabe: Ein Graph $\mathcal{G} = (V, E)$, dessen Knoten ohne Beschränkung der Allgemeinheit mit den Zahlen von 1 bis zur Anzahl aller Knoten $|V|$ benannt sind sowie eine natürliche Zahl n.
Ausgabe: Alle induzierten Subgraphen der Knotenzahl n aus $\mathcal{G}$.

```
1   forall vertices v ∈ V do
2     V' = {u ∈ Neig(v) | u > v}
3     call the function EXTENDSUBGRAPH({v}, V', v)
4
5   EXTENDSUBGRAPH(V_sub, V', v)
6     if |V_sub| = n return ⟨G, V_sub⟩ and terminate the function
7     while V' ≠ ∅ do
8       remove a random vertex w from V'
9       V' = V' ∪ {u ∈ Neig(w) | u > v and Neig(u) ∩ V_sub = ∅}
10      call the function EXTENDSUBGRAPH(V_sub ∪ {w}, V', v)
```

Weitere Entwicklungen bei der probabilistischen sowie der exakten Auszählung von gerichteten induzierten Subgraphen mit fester Knotenzahl wurden von Schreiber und Schwöbbermeyer [SS05], Chen u. a. [Che+06], Grochow und Kellis [GK07], Kashani u. a. [Kas+09] und Ribeiro und Silva [RS10] publiziert, wobei letztere die aktuell effizienteste Entwicklung darstellt.

5.1.3.2 Das Graph-Isomorphismus-Problem

Bei der n-Motiv-Signatur eines Graphen werden isomorphe Subgraphen gesucht und einer gemeinsamen Motivklasse zugeordnet. Anschließend wird deren Häufigkeit für dieses Motiv aufsummiert. Das dabei zu lösende sogenannte *Subgraph-Isomorphismus-Problem* ist im Allgemeinen sehr komplex und wurde als $\mathcal{NP}$-vollständig gezeigt [Coo71]. Mit bereits gefundenen Subgraphen (vgl. Abschnitt 5.1.3.1) ist das verbleibende *Graph-Isomorphismus-Problem* zwar immer noch ein schweres Problem,[3] in der Praxis ist es allerdings insbesondere für sehr kleine Graphen häufig effizient lösbar. Die Arbeiten von Wernicke [Wer05] und Kashani u. a. [Kas+09] verwenden beispielsweise den Algorithmus namens *nauty* von McKay [McK81] zum Lösen dieser Aufgabe.

Für die Isomorphismen von 3-Motiven können aufgrund der relativ geringen Anzahl alle Möglichkeiten vorberechnet, gespeichert und später für einen Äquivalenzvergleich herangezogen werden. Denn in einem Graphen mit $n = 3$ Knoten können $n \cdot (n-1) = 3 \cdot 2 = 6$ unterschiedliche gerichtete Kanten vorkommen, sodass die Potenzmenge dieser Kantenmenge unter Einbezug aller

3. Bisher ist weder ein effizienter, also in polynomieller Zeit arbeitender Algorithmus zur Lösung des Graph-Isomorphismus-Problems noch ein Beweis für eine $\mathcal{NP}$-Vollständigkeit des Problems bekannt [KST12, Introduction]. Somit wird angenommen, dass das Graph-Isomorphismus-Problem in der Komplexitätsklasse $\mathcal{NP}$-intermediär liegen könnte.

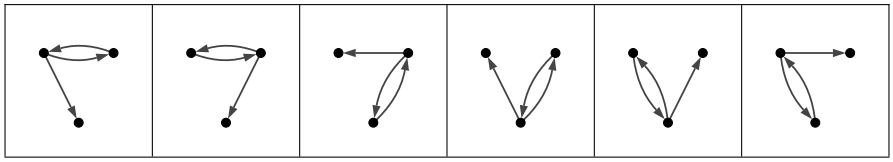


Abbildung 5.3: Alle isomorphen Graphen der Motivklasse m_9

isomorphen Graphen nur zu $2^6 = 64$ verschiedenen gerichtete Graphen mit der Knotenzahl 3 führt. Die jeweiligen Motivklassen können dabei 1, 3 oder 6 isomorphe Graphen enthalten. Beispielsweise ergibt sich die Motivklasse m_9 aus den in Abbildung 5.3 gezeigten 6 isomorphen Graphen.

5.2 Existierende Arbeiten

Die folgenden wissenschaftlichen Arbeiten haben einen engen inhaltlichen Bezug zur vorliegenden Thematik und wurden demnach beim Entwurf und der Umsetzung der Erkennungs- und Abwehrtechnik in diesem Kapitel beachtet.

5.2.1 Graphenbasierte Bedrohungserkennung

Die Autoren Eberle, Graves und Holder [EGH10] erzeugen für verschiedene Szenarien, etwa die E-Mail-Kommunikation in einem Unternehmen oder die Freundschaftsbeziehungen in einem sozialen Netzwerk, unterschiedliche und Szenario-spezifische Graphen und wenden drei bereits in [EH07] publizierte Algorithmen an, die anhand dieser Graphen Insiderangriffe in Form von Anomalien erkennen können. Die Algorithmen basieren auf der von Cook und Holder [CH00] entwickelten, graphenbasierten Strukturerkennung namens *SUBDUE* und werten Substrukturen in den Graphen anhand deren Abweichung zur normativen Substruktur des jeweiligen Graphen aus. Die normative Substruktur eines Graphen wird als diejenige definiert, anhand derer der Graph am besten komprimiert werden kann, indem die Substruktur überall bei dessen Auftreten als einzelner Knoten ersetzt wird. *SUBDUE* wurde bereits zuvor von Noble und Cook [NC03] verwendet, um anhand der gefundenen Substrukturen Anomalien in Graphen zu erkennen. So wird von den Autoren einzelnen Substrukturen Werte zugewiesen, die deren Grad an Anomalie aufzeigen. Weiterhin gehen die Autoren der Hypothese nach, dass Subgraphen, die wenig gemeinsame Substrukturen besitzen, ein höheres Anomaliepotenzial aufweisen. All diese Verfahren legen das sogenannte Minimum Descriptive Length (MDL) Prinzip bei der Auswertung der Substrukturen zugrunde, das darauf beruht, mit wie vielen Bits ein Graph beziehungsweise eine Substruktur minimal beschrieben werden kann.

Henderson u. a. [Hen+10] stellen ein entwickeltes Framework vor, das mithilfe verschiedener Metriken Grapheneigenschaften von dynamischen Graphen berechnet und diese für Analysen von Anomalien verwendet. Dabei haben die Autoren großen Wert auf Effizienz und Skalierbarkeit gelegt. Dabei haben sie ein mehrstufiges Verfahren integriert, das zunächst generelle und einfach zu berechnende Metriken mithilfe unterschiedlicher Analyseverfahren auswertet. Im Fall von interessanten Ergebnissen lokalisiert dieses Verfahren die dafür verantwortlichen Bereiche im Graphen und berechnet dafür speziellere und rechenintensivere Metriken, die genauere Analysen

erlauben. Sowohl die Metriken als auch die Analyseverfahren der Metriken sind ersetz- und erweiterbar. Bereits voreingestellt sind allgemeine Metriken, wie etwa die Anzahl der Knoten oder Kanten, der durchschnittliche Knotengrad oder das durchschnittliche Kantengewicht sowie speziellere Metriken wie etwa der Anteil der Knoten in der größten zusammenhängenden Komponente, die Anzahl der zusammenhängenden Komponenten oder der Eigenwert der Adjazenzmatrix. Die Analyseverfahren der Metriken beinhalten die Erkennung von unnormal hohen oder niedrigen Metrikwerten in einem zeitlichen Verlauf, die Identifizierung von periodischen Metrikwerten mithilfe einer Fourieranalyse sowie verschiedene Anomalieerkennungsverfahren.

In Abschnitt 4.9 wurde bereits auf die Arbeit von Gamachchi, Sun und Boztas [GSB17] eingegangen. Die Autoren bereiten die Interaktionen von Insidern einer Domäne untereinander und mit den Ressourcen der Domäne als Graphen auf und berechnen darauf verschiedene Grapheneigenschaften für einzelne Nutzer. Die resultierenden Ergebnisse werden dann als Grundlage für die Erstellung von Profilen beziehungsweise die Erkennung von Anomalien verwendet. Die Datenbasis, aus der die Graphen erstellt werden, erstreckt sich von Ereignisaufzeichnungen über E-Mail- und HTTP-Logs bis hin zu Informationen aus der Personalabteilung über Persönlichkeitsmerkmale von Mitarbeitern. Anschließend wird für jeden Nutzer der induzierte Subgraph extrahiert und mehrere Subgrapheigenschaften, wie zum Beispiel die Anzahl der Knoten und Kanten, die Dichte oder der Durchmesser des Subgraphen errechnet. Für die Erkennung von unerwünschten Insideraktivitäten wenden die Autoren eine angepasste Variante des von Liu, Ting und Zhou [LTZ08] publizierten *Isolation Forest* Algorithmus an, der explizit Anomalien isoliert anstatt Profile von normalen Instanzen zu erlernen.

Im Kontext der Rechnernetz-basierten Erkennung und automatisierten Klassifizierung von Angriffen stützten sich die Autoren Haas, Wilkens und Fischer [HWF19] ebenfalls auf eine Repräsentation von Aktivitäten anhand von Graphen und verwendeten Netzwerkmotiven, um die Strukturen und Charakteristiken dieser Rechnernetz-Graphen analysieren zu können. Das Vorgehen der Autoren ist zu dem in dieser Dissertation entwickelten Mechanismus sehr ähnlich, wurde allerdings für einen anderen Kontext konzipiert und lässt sich nicht auf die hier anvisierte Erkennung und Abwehr von Insiderbedrohungen übertragen.

5.2.2 Bedrohungserkennung mithilfe von Systemaufrufen

Das Potenzial von Systemaufrufen zur Erkennung von bösartigen Aktivitäten in informationstechnischen Systemen wurde bereits früh erkannt, da für alle Prozesse und damit auch alle von Nutzern gewollt oder ungewollt durchgeführte Vorgänge kein Weg an ihnen vorbeiführt.

Forrest u. a. [For+96] stellen fest, dass normale Ausführungen von Prozessen sehr stabile kurze aufeinanderfolgende Sequenzen der Länge N von Systemaufrufen, sogenannte N*-Gramme*, aufweisen und somit abweichende Sequenzen als Anomalien erkannt werden können, etwa wenn eine Schwachstelle in einem Programm ausgenutzt wird und dadurch ein Angreifer das Verhalten des Programms zur Laufzeit ändert. Die Autoren erzeugen in einer Trainingsphase eine Datenbank solcher Sequenzen für normale Ausführungen von Prozessen. Dabei verwenden sie ein sogenanntes sliding window (deut. Schiebefenster) mit einer festen Sequenzlänge, mit dem sie über die Abfolge aller auftretenden Systemaufrufe laufen und diese in der Datenbank ablegen. In der anschließenden Testphase werden alle Sequenzen von Systemaufrufen eines Prozesses mit denen in der Datenbank verglichen und bei keiner Übereinstimmung als Fehler registriert und gezählt. Die Anzahl beziehungsweise der Anteil an Fehlern im Vergleich zur Anzahl aller Sequenzen gibt dann den Grad der Abweichung vom Normalverhalten. Hofmeyr,

Forrest und Somayaji [HFS98] erweitern diese Arbeit, indem sie die Sequenzen von normalen Prozessabläufen in der Datenbank als Bäume speichern und somit die Effizienz bei der Suche nach Übereinstimmungen mit neuen Systemaufrufsequenzen erhöhen. Weiterhin verwenden die Autoren den Hammingabstand als Distanzmaß zwischen einer nicht-übereinstimmenden Sequenz und den normalen Sequenzen in der Datenbank, um den Grad einer Abweichung zu verfeinern. Weitere Verbesserungen werden von Warrender, Forrest und Pearlmutter [WFP99] untersucht, die zur Anomalieerkennung statistische Methoden und maschinelles Lernen auf die Sequenzen von Systemaufrufen anwenden, sowie von Eskin, Lee und Stolfo [ELS01], die detaillierte Analysen der Sequenzlängen vornehmen. Die Autoren Eskin, Lee und Stolfo [ELS01] führen für ihre Analysen einen sogenannten *Aufrufgraphen* ein, der vom Kontext und der Benennung her sehr ähnlich zu den Systemaufrufgraphen in dieser Dissertation erscheint (s. Abschnitt 5.4), allerdings vollständig anders konstruiert wird und einem anderen Zweck dient. Jüngere Arbeiten mit dem Ansatz der Systemaufrufsequenzen sind beispielsweise die von Assem, Rachidi und Graini [ARG14] sowie die von Mouttaqi, Rachidi und Assem [MRA17].

Die Arbeit von Liao und Vemuri [LV02] basiert ebenfalls auf den Systemaufrufen von Prozessen. Die Autoren betrachten allerdings im Unterschied zu den bisher vorgestellten Veröffentlichungen nicht die Reihenfolge von kurzen Systemaufrufsequenzen, sondern die Häufigkeiten von Systemaufrufen ganzer Prozessabläufe und wenden darauf Techniken der Textkategorisierung an, um Anomalien in Prozessaktivitäten zu erkennen. Dafür fassen die Autoren alle Systemaufrufe eines Prozessablaufs als Dokument und jeden einzelnen vorkommenden Systemaufruf als Wort auf. Dadurch können sie alle bis dato wohluntersuchten Techniken der automatisierten Textanalyse und -kategorisierung auf die Systemaufrufe anwenden. Konkret werden die Systemaufrufhäufigkeiten von den Autoren als Vektoren modelliert und diese Vektoren zwischen zwei Programmaktivitäten mittels der *k-Nearest-Neighbour-Methode* miteinander verglichen. Mit dieser Methode können Dokumente kategorisiert werden, indem nach den k ähnlichsten Dokumenten im Vektorraum gesucht wird. Als Ähnlichkeitsmaß für die Vektoren schlagen Liao und Vemuri den Hammingabstand oder die Kosinusähnlichkeit vor. In einem ähnlichen Ansatz entwickeln Creech und Hu [CH14] ein Verfahren, in dem jeder einzelne Systemaufruf als Buchstabe und alle Systemaufrufsequenzen variabler Länge als Wörter aufgefasst und in einem Wörterbuch gespeichert werden. Aus diesem Wörterbuch generieren die Autoren Sätze der Länge fünf und analysieren die Häufigkeiten dieser Sätze in gegebenen Systemaufrufen von Prozessabläufen mit einem neuronalen Netz.

Genau wie in dieser Dissertation schlagen Nguyen, Reiher und Kuenning [NRK03] einen Echtzeit-Erkennungsmechanismus von bösartigen Insideraktivitäten vor und beziehen dabei nicht nur die reinen Systemaufrufnamen, Abfolgen und Häufigkeiten in ihre Technik zur Erkennung und Abwehr von bösartigen Insideraktivitäten ein, sondern erweitern ihre Analysen auf die Systemaufrufargumente. Damit erfassen sie die Interaktionen zwischen Nutzern, Prozessen und den Ressourcen eines Rechners und erstellen beispielsweise Dateizugriffsmuster von Nutzern und Prozessen. Bei ihren Analysen kommen die Autoren zu dem Schluss, dass Zugriffshäufigkeiten von menschlichen Nutzern auf Dateien keine adäquate Methode zur Erkennung von bösartigen Aktivitäten darstellen. Im Unterschied dazu kann allerdings den Autoren zufolge die Überwachung der Zugriffe und Zugriffshäufigkeiten von Systemnutzern wie etwa *nobody* oder *daemon* auf Dateien zielführend sein, da diese Nutzer festgelegte und strikt abgrenzbare Aktivitäten durchführen und somit eine Modellierung des normalen Verhaltens erleichtern beziehungsweise eine Abweichung von diesem Normalverhalten ein starkes Indiz für eine bösartige Veränderung darstellt. Weiterhin identifizieren Nguyen, Reiher und Kuenning die Dateizugriffsmuster von Prozessen sowie die Prozesshierarchiebäume als vielversprechende Grundlagen für eine Anoma-

lieerkennung. Mit diesen Erkenntnissen entwickeln sie einen Demonstrator zur Erkennung von Pufferüberläufen, der auf einem speziell veränderten Linux-Kernel aufbaut. Der Linux-Kernel beinhaltet angepasste Systemaufrufe, die jeweils beim Eintritt und beim Beenden eines Systemaufrufs eine Ereignisnachricht erzeugen, die aufgezeichnet wird. Der Demonstrator beinhaltet eine Liste an akzeptierten Prozessbeziehungen. Sobald ein Prozess einen anderen Prozess startet, der nicht in seiner Liste steht, wird eine Alarmmeldung an den Systemadministrator erzeugt. Ausgenommen werden müssen allerdings Prozesse, die keine feste Liste an Kindprozessen erzeugen, sondern jeden möglichen anderen Prozess starten können, wie etwa eine Shell. Der um Aufrufargumente und zusätzliche Informationen erweiterte Ansatz bei der Analyse von Systemaufrufen wird auch von Koucham, Rachidi und Assem [KRA15] sowie von Berlin, Slater und Saxe [BSS15] aufgegriffen. Die Autoren sammeln umfangreiche Systemaufrufinformationen mithilfe eines Auditierungsmechanismus und klassifizieren Systemaufrufsequenzen anhand dieser Informationen in eine *normale* oder eine *bösartige* Klasse.

5.3 Linux Audit System

Im Zuge der Zertifizierung von Betriebssystemversionen als Computersicherheitssoftware nach dem internationalen Standard ISO/IEC 15408, besser bekannt unter dem Namen *Common Criteria for Information Technology Security Evaluation* oder einfach *Common Criteria (CC)* [Com17], wurden Mechanismen zur Auditierung sicherheitsrelevanter Abläufe und Handlungen in die jeweiligen Betriebssystemkerne integriert. Diese *Auditierungsmechanismen* erlauben mit entsprechender Konfiguration die Erzeugung von Ereignisnachrichten, die zum Zwecke der Beobachtung von Nutzeraktivitäten und der Bedrohungserkennung ausgewertet werden können und auf die zeitnah reagiert werden kann. Weiterhin sind die Ereignisnachrichten mit Informationen angereichert, sodass die Abläufe und Handlungen rekonstruiert und beteiligten Nutzern zugeordnet werden können.

Das *Linux Audit System* bildet die Basis für den in dieser Dissertation entwickelten Schutzmechanismus zur Erkennung und Abwehr von Insiderbedrohungen. Es erlaubt durch seinen Aufbau die leichte Akquisition von Ereignisnachrichten auf der Ebene der Systemaufrufe (vgl. Abschnitt 5.1.1) und ermöglicht somit die Erfassung aller grundlegenden Operationen, die auf Ressourcen eines Rechners stattfinden. Geeignete Schnittstellen des Frameworks erlauben darüber hinaus eine leichte Weiterverarbeitung der Ereignisnachrichten. Die Nutzbarmachung dieses Auditierungsmechanismus bedarf allerdings spezieller Vorbereitungen, die nachfolgend aufgezeigt und deren Umsetzung erläutert werden.

5.3.1 Überblick des Auditierungsmechanismus

Entwickelt wurde das *Linux Audit System* von den Red Hat Entwicklern Faith und Grubb [FG19] und zum ersten Mal in den Linux Kernel 2.6 integriert [Cor04]. Seitdem ist es fester Bestandteil der allermeisten Linux-Distributionen und Grundlage für die bereits erwähnte CC-Zertifizierung. Eine einführende Beschreibung findet sich in den Produktdokumentationen der Red Hat Enterprise Linux Versionen ab Version 6, aktuell in der Version 8 [Red19, Kapitel 10]. Die beteiligten Komponenten sind in Abbildung 5.4 visualisiert. Dazu gehören die folgenden:

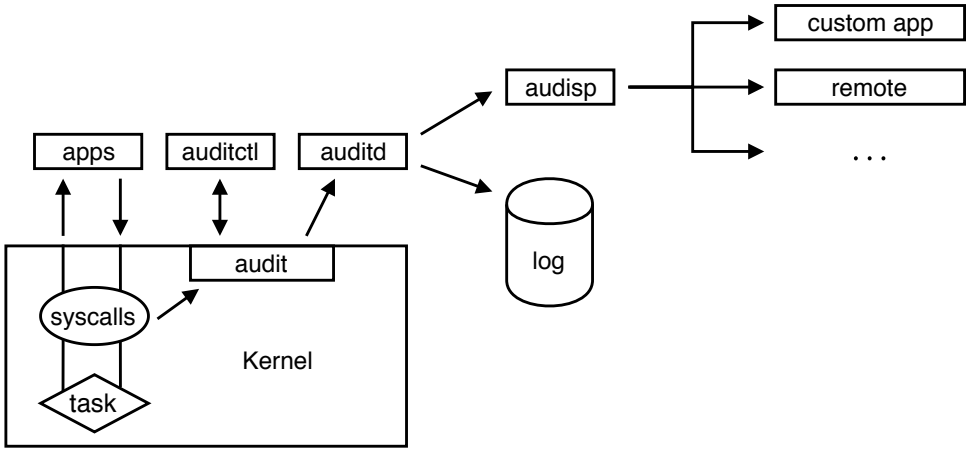


Abbildung 5.4: Die Komponenten des *Linux Audit Systems* sowie deren Relation in Anlehnung an [Gru11] und [Red10, Abbildung 7.1]

- **audit**: Eine Kernelkomponente, die im Systembereich des Betriebssystems (engl. kernel space) arbeitet und dort je nach Konfiguration verschiedene Vorgänge des Betriebssystems in Ereignisnachrichten erfasst und diese über einen sogenannten Netlink-Socket, also eine spezielle Interprozesskommunikationsschnittstelle, an eine Komponente im Benutzerbereich (engl. user space) weiterleitet.
- **auditd**: Ein Hintergrundprozess (engl. daemon), auch *Audit-Daemon* genannt, der im Benutzerbereich arbeitet, die Ereignisnachrichten aus dem Systembereich über den Netlink-Socket abnimmt und je nach Konfiguration sowohl in eine Log-Datei schreiben als auch an eine Verteilerkomponente (engl. dispatcher) weiterleiten kann.
- **audisp**: Eine Verteilerkomponente, die Ereignisnachrichten von *auditd* in Empfang nimmt und diese an benutzerdefinierte Anwendungen weiterleitet.
- **auditctl**: Eine Konfigurationskomponente, mit deren Hilfe die Kernelkomponente *audit* konfiguriert und somit die aufzuzeichnenden Ereignisse definiert werden können.

Die Konfigurationen der Kernelkomponente *audit* mittels *auditctl* erstreckt sich für die vorliegende Arbeit auf die nachfolgend erläuterten Anwendungsfälle [Red19, Abschnitt 10.1].

5.3.1.1 Überwachung von Zugriffen auf Dateisystemobjekte

Der Zugriff und die Modifikation von Dateien oder Ordnern sowie die Änderungen von Datei- und Ordnereigenschaften können durch *audit* aufgezeichnet werden. Dafür könnte einerseits die Überwachung aller dateisystemrelevanten Systemaufrufe, zum Beispiel `read()`, `write()` oder `getxattr()`, einzeln konfiguriert werden. Andererseits bietet die Konfigurationskomponente *auditctl* sogenannte Dateisystemobjektüberwachungen (engl. file system object watches) an. Diese können für einen speziellen Pfad oder für einen Pfad und alle Unterobjekte dieses Pfades konfiguriert werden. Dadurch werden automatisch alle dateisystemrelevanten Systemaufrufe konfiguriert. Ausgenommen davon werden allerdings die `read()` und `write()` Systemaufrufe, da sie eine überwältigende Anzahl an Ereignisnachrichten erzeugen würden. Stattdessen spielt der bereits in Abschnitt 5.1.1 kennengelernte Systemaufruf `openat()` sowie die Dateistatusindikatoren eine zentrale Rolle bei der Erkennung und Aufzeichnung von Ereignisnachrichten, die das Lesen und Schreiben von Daten aus beziehungsweise in Dateien betreffen.

Um Daten in eine Datei schreiben zu können, muss ein Prozess diese Datei in der Regel zunächst über den Systemaufruf `openat()` öffnen. Dieser Systemaufruf benötigt als Funktionsargument neben dem Pfad der Datei, die geöffnet werden soll, auch die sogenannten Dateistatusindikatoren, die vermitteln, auf welche Art und Weise die Datei geöffnet werden soll. Zum Schreiben in eine Datei muss demnach entweder der Indikator `O_WRONLY` oder `O_RDWR` in den übergebenen Dateistatusindikatoren vorhanden sein. Zum Lesen entsprechend entweder `O_RDONLY` oder `O_RDWR`. Diese angeforderten Berechtigungen im `openat()` Systemaufruf sind zwar kein Garant dafür, dass anschließend aus einer Datei gelesen oder in eine Datei geschrieben wird oder beides, aber sie werden in den folgenden Abschnitten als Hinweis darauf verwendet. Damit können Zugriffe insbesondere auf wichtige Dateien erkannt und notwendige Informationen für nachträgliche Untersuchungen festgehalten werden. Zu diesen Informationen gehören:

- der Zeitpunkt des Datei- oder Ordnerzugriffs,
- der Pfad der Datei oder des Ordners,
- die Inode-Nummer der Datei oder des Ordners im Dateisystem,
- der Dateityp, also reguläre Datei, Order oder Verknüpfung,
- die Zugriffsrechte der Datei oder des Ordners,
- den Besitzer und die Gruppe der Datei oder des Ordners sowie
- Systemaufruf-, prozess- und nutzerspezifische Informationen, auf die im folgenden Abschnitt 5.3.1.2 genauer eingegangen wird.

Für die entwickelte Erkennungs- und Abwehrtechnik wird aktuell nur der `openat()` Systemaufruf betrachtet. Weitere Informationen über anderen Möglichkeiten, die das Schreiben von Daten in eine Datei anzeigen, finden sich in den Abschnitten 5.4.1 und 5.8.

5.3.1.2 Überwachung von Programmaufrufen und Befehlen eines Nutzers

Die Ausführungen von Dateien sowie von Befehlen können durch *audit* aufgezeichnet und mit besonderen prozess- und nutzerspezifischen Informationen angereichert werden. Damit lässt sich eine Befehls- und Programmaufrufkette einzelner oder aller Nutzer erstellen. Zu den wichtigen Systemaufrufen bezüglich der Überwachung von Programmaufrufen und Befehlen gehören `clone()` und `execve()`, wobei letzterer der weitaus wichtigere ist. Denn damit kann aus einem laufenden Prozess heraus, beispielsweise einer Shell, eine andere ausführbare Datei gestartet werden, was ein häufiger Ablauf eines Programmaufrufs oder von Befehlen eines Nutzers ist.

Die bereits im vorigen Anwendungsfall angesprochenen prozess- und nutzerspezifischen Informationen beinhalten dabei:

- die Prozess-ID und die Elternprozess-ID des neuen Prozesses, der durch den Programmaufruf oder Befehl angestoßen wurde,
- die effektive und die ursprüngliche Nutzer-ID des Nutzers,[4] der den Prozess angestoßen hat sowie
- der vollständige Programmaufruf oder Terminalbefehl, mit dem der Prozess angestoßen wurde.

4. Damit kann die Durchführung einer Aktivität unter fremder Nutzer-ID festgestellt werden.

5.3.1.3 Überwachung weiterer Systemaufrufe

Weitere von Prozessen verwendete Systemaufrufe können durch *audit* aufgezeichnet und mit besonderen Informationen angereichert werden. Für den vorliegenden Kontext sind neben den Systemaufrufen, die für die bisher angesprochenen Anwendungsfälle wichtig sind, alle Rechnernetz-assoziierten Systemaufrufe von großem Interesse. Dadurch können Verbindungen von und zu externen Ressourcen erkannt und untersucht werden. Die wichtigsten darunter sind `accept4()`, der eine Verbindung von einem Socket annimmt, `connect()`, der eine Verbindung an eine spezielle Adresse über einen Socket initiiert, `sendto()`, der Daten an eine spezielle Adresse sendet sowie `recvfrom()`, der Daten von einer speziellen Adresse empfängt.

Äquivalent zum `openat()` Systemaufruf, wie er mit den Dateistatusindikatoren für die Überwachung von Lese- und Schreibzugriffen auf Dateisystemobjekte in Abschnitt 5.3.1.1 beschrieben wurde, sind diese Rechnernetz-assoziierten Systemaufrufe kein Garant dafür, dass Daten über einen Socket tatsächlich in eine bestimmte Richtung gesendet beziehungsweise empfangen werden. Dafür müssten weitere Systemaufrufe, wie etwa `send()`, `sendmsg()`, `recv()`, `recvmsg()` oder auch `read()` und `write()` überwacht und aufgezeichnet werden. Doch auch diese würden eine überwältigende Anzahl an Ereignisnachrichten erzeugen, weshalb aktuell in der entwickelten Erkennungs- und Abwehrtechnik darauf verzichtet wird. Im Unterschied zum `openat()` Systemaufruf liefern die systemaufrufspezifischen Informationen der genannten wichtigen Rechnernetz-assoziierten Systemaufrufe `accept4()`, `connect()`, `sendto()` und `recvfrom()` allerdings keinen Hinweise auf die Flussrichtung von Daten in Sockets. Eine einmal aufgebaute Verbindung über einen Socket kann immer in beide Richtungen mit Daten beschrieben oder ausgelesen werden. Aus diesem Grund werden bei der Überwachung der Rechnernetz-assoziierten Systemaufrufe die Informationen nur als Verbindungsinitiierungen und nicht als Lese- beziehungsweise Schreibzugriffe interpretiert.

Mit diesen und weiteren Systemaufrufen können demnach zusätzliche Aktivitäten, insbesondere mit externen Ressourcen in einem Rechnernetz, erkannt und notwendige Informationen für Untersuchungen festgehalten werden. Die bereits angesprochenen dabei anfallenden systemaufrufspezifischen Informationen beinhalten dabei:

- die architekturspezifische Systemaufrufnummer,
- die zugrunde liegende Architektur, mit deren Wissen man die Systemaufrufnummer einem korrekten Systemaufruf zuordnen kann,
- die wichtigsten Funktionsargumente des Systemaufrufs,
- der Rückgabewert des Systemaufrufs sowie
- der Erfolg oder Misserfolg bei der Ausführung des Systemaufrufs.

5.3.2 Eignung der Audit-Ereignisnachrichten für eine Echtzeitanwendung

Die Ereignisnachrichten der Kernelkomponente *audit*, die vom Audit-Daemon *auditd* entgegengenommen und in einer Log-Datei abgelegt werden, sind in ihrer Form und ihrem Inhalt noch nicht für die Erkennung und Abwehr von Insiderbedrohungen geeignet. Am nachfolgenden Beispiel mit vier Ereignisnachrichten lassen sich die Gründe dafür veranschaulichen:

```
1 node=mw-host type=SYSCALL msg=audit(1547121809.485:31317): arch=40000003
    syscall=102 success=yes exit=-115 a0=3 a1=19302860 a2=0 a3=0 items=0
    ppid=1734 pid=1750 auid=1000 uid=1000 gid=1000 tty=pts0 ses=2
    comm="WTEpZSFwgb" exe="/home/mw/Downloads/WTEpZSFwgb"
```

```
2 node=mw-host type=SOCKETCALL msg=audit(1547121809.485:31317): nargs=3
    a0=9d a1=19d85988 a2=10
3 node=mw-host type=SOCKADDR msg=audit(1547121809.485:31317): saddr=0200005
    28EBEE145000000000000000
4 node=mw-host type=PROCTITLE msg=audit(1547121809.485:31317): proctitle=2F
    686F6D652F6D772F446F776E6C6F6164732F575445705A5346776762002D6970632E6
    6643D33007363616E
```

Zum ersten wird ein Ereignis aus Effizienzgründen auf mehrere, in ihrer Reihenfolge nicht notwendigerweise sortierte Ereignisnachrichten aufgeteilt. Erst die Vereinigung aller zu einem Ereignis gehörenden Ereignisnachrichten liefern die benötigten Informationen, um das Ereignis korrekt interpretieren und analysieren zu können. Die zu einem Ereignis zugehörigen unterschiedlichen Ereignisnachrichten lassen sich anhand des Zeitstempels (hier: `1547121809.485`) und der Nachrichten-ID (hier: `31317`) zusammenführen.

Zum zweiten können Ereignisnachrichten verloren gehen und daher den Audit-Daemon *audit* nicht erreichen. Darüber hinaus können sie stark verzögert beim Audit-Daemon *audit* eintreffen, sodass bereits nachfolgende Ereignisnachrichten von *audit* verarbeitet und abgelegt wurden. Ein solches Verlorengehen oder Vertauschen von Ereignisnachrichten unterschiedlicher Ereignisse kann dazu führen, dass gewisse Informationen der Ereignisnachrichten fehlinterpretiert werden. Beispielsweise könnte eine Ereignisnachricht verloren gehen, in der ersichtlich werden würde, dass ein externes Speichermedium an einer bestimmten Stelle im Dateisystem eingehängt wurde. Das anschließende Kopieren von Dateien auf dieses Speichermedium würde dann als normales Kopieren an diese Stelle im lokalen Dateisystem interpretiert werden. Das eigentliche Herausschleusen von Informationen aus diesem Rechner würde dadurch unbemerkt bleiben.

Zum dritten liegen viele Informationen in kodierter Form vor, die teilweise nur mit systemspezifischen Zusatzinformationen korrekt dekodiert werden können. Dazu gehören etwa die Nutzer- und Gruppen-IDs (hier: `auid`, `uid` und `gid`), die für eine korrekte Interpretation in die tatsächlichen Nutzer- und Gruppennamen der Nutzerverwaltung des zugrunde liegenden Betriebssystems überführt werden müssen. Auch die prozessspezifischen Informationen (hier unter anderem: `proctitle`) liegen häufig base64-kodiert vor. Diese und weitere kodierte Informationen müssen für eine korrekte Interpretation und Analyse beziehungsweise Weiterverarbeitung dekodiert werden.

Zum vierten werden die Ereignisnachrichten standardmäßig vom Audit-Daemon *auditd* in lokale Log-Dateien abgelegt. Das beschränkt die darauf aufbauenden Analysen ausschließlich auf den lokal vorliegenden Kontext und überlässt die Hoheit der gesammelten Ereignisnachrichten potenziell demjenigen Nutzer eines Systems, der Gegenstand der Untersuchung ist. Das bedeutet, dass ein Nutzer mit vollständigen Zugriffsrechten, unter Unix-artigen Betriebssystemen auch *Rootrechte* genannt, die Möglichkeit sowie auch die Gelegenheit bekommt, die gesammelten Ereignisnachrichten zu manipulieren.

5.3.3 Existierende Lösungen

Für die Lösung einzelner vorgenannter Teilprobleme aus Abschnitt 5.3.2 existieren bereits Softwarelösungen, die entweder als dediziertes Audit-Werkzeug, als Plugin der Verteilerkomponente *audisp* oder als vollständiger Ersatz des Audit-Daemons *auditd* realisiert sind.

Das Auditierungssystem-eigene Werkzeug *ausearch*[5] beherrscht die Zusammenfassung aller bis zum Zeitpunkt des Programmaufrufs abgelegten Ereignisnachrichten, die zu einem Ereignis gehören, sowie die Dekodierung der kodierten Werte. Problematisch ist dabei allerdings der Umstand, dass *ausearch* nur auf den bereits lokal vorhandenen ruhenden Log-Daten arbeiten kann. Das bedeutet einerseits, dass neue Ereignisnachrichten nur durch erneute Aufrufe des Werkzeugs in Analysen einbezogen werden können. Andererseits erfolgt eine Dekodierung der angesprochenen kodierten Informationen nur anhand des lokal vorliegenden Kontextes, in dem *ausearch* aufgerufen wird. Die Nutzer- und Gruppen-IDs etwa werden in die lokalen Nutzer- und Gruppennamen dekodiert, auch wenn es sich potenziell um Ereignisnachrichten und damit IDs aus anderen Geräten handeln kann.

Eine Verbesserung dieser Probleme schaffen existierende Plugins der Verteilerkomponente *audisp*. Dazu gehören die Plugins *audisp-cef* [Des14a] und *audisp-json* [Des14b], die ebenfalls Ereignisnachrichten zu einem Ereignis zusammenfassen und eine Dekodierung vornehmen können. Im Unterschied zu *ausearch* verarbeiten diese Lösungen die Ereignisnachrichten in Echtzeit und sind derart gestaltet, dass die zusammengefassten und dekodierten Ereignisnachrichten in den speziellen Datenformaten *Common Event Format (CEF)* oder *JSON* an einen entfernten Server geschickt werden. Einen Schritt weiter geht das studentische Projekt *audit-go* [Jun+14] und die darauf aufbauende Entwicklung *go-audit* [Hub16] sowie das Überwachungs- und Analysewerkzeug *osquery* [Ree15]. Diese ersetzen jeweils den Audit-Daemon *auditd* und können die Ereignisnachrichten von der Kernelkomponente *audit* über die Netlink-Verbindung direkt entgegennehmen, verarbeiten und anschließend an einen entfernen Server senden beziehungsweise im Fall von *osquery* von einem entfernten Server abfragen.

Für den vorliegenden Anwendungsfall haben allerdings auch diese Lösungen Nachteile. So werden zwar die Ereignisnachrichten in eine bessere und vor allem maschinenlesbarere Form gebracht, als sie von der Kernelkomponente *audit* emittiert werden. Für eine effektive und effiziente Weiterverarbeitung dieser Nachrichten, wie sie im vorliegenden Kontext entwickelt wird, ist das Schema des Nachrichtenformats jedoch ungeeignet. Die benötigten Informationen in den einzelnen Ereignisnachrichten könnten nur durch aufwendige Umformungen und Verknüpfungen von Datenfeldern gewonnen werden. Darüber hinaus selektieren die genannten Lösungen einzelne Datenfelder aus den ursprünglichen Ereignisnachrichten und verwerfen dabei viele der Informationen, die für die in dieser Dissertation entwickelte Erkennungs- und Abwehrtechnik allerdings eine zentrale Rolle spielen.

5.3.4 Entwicklung eines *audisp* Plugins zur Echtzeitverarbeitung

Im Zuge der Dissertation wurde in Anlehnung an die bereits existierenden Lösungen aus Abschnitt 5.3.3 ein eigenes Plugin der Verteilerkomponente *audisp* entwickelt, das sowohl die aufgezeigten Probleme aus Abschnitt 5.3.2 löst, als auch die Defizite der existierenden Lösungen für den vorliegenden Anwendungsfall ausgleicht. Weiterhin wurde die eigene Lösung derart entworfen, dass spätere Anpassungen zur Erhöhung des Datenschutzes möglich sind. Der Datenschutz wird in Kapitel 6 genauer beleuchtet und in Kapitel 7 prototypisch umgesetzt und in die hier beschriebene Softwarelösung integriert. Das neue Plugin namens *audisp-hostmon* erfüllt die folgenden hier relevanten Aufgaben, die in Abbildung 5.5 illustriert sind:

5. Informationen darüber sind auf Unix-ähnlichen Systemen aufrufbar über die Dokumentationsseite von *ausearch* mit dem Kommandozeilenbefehl `man 8 ausearch`.

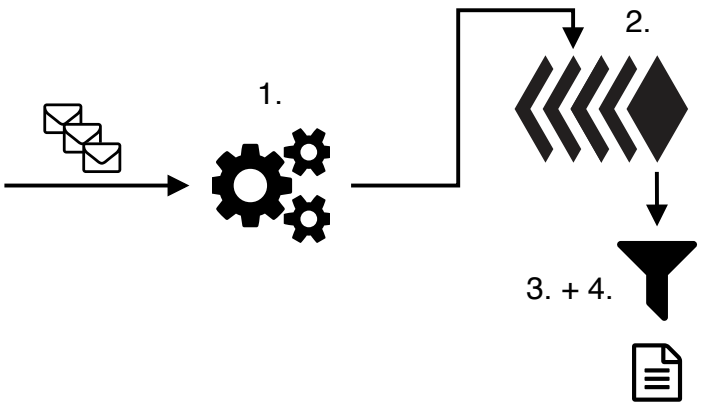


Abbildung 5.5: Die Aufgaben des entwickelten *audisp-hostmon*-Plugins

1. Eintreffende Ereignisnachrichten von der Kernelkomponente *audit*, die vom Audit-Daemon *auditd* weitergeleitet wurden, werden zunächst geparst und anderen gegebenenfalls bereits existierenden Ereignisnachrichten desselben Ereignisses zugeordnet. Weiterhin werden sie in ihre Informationseinheiten zerlegt, die wiederum gegebenenfalls dekodiert werden.

2. Für die Zuordnung zu existierenden Ereignisnachrichten ist ein Puffer eingerichtet, der eine definierte Anzahl an offenen Ereignisnachrichten für eine definierte Zeitspanne vorhält, bis sie für eine Weiterverarbeitung freigegeben werden. Dieser Puffer ermöglicht gleichzeitig eine auf die Puffergröße beschränkte Sortierung der Ereignisnachrichten, sofern sie in verkehrter Reihenfolge vom Audit-Daemon *audit* empfangen wurden, sowie die Erkennung von verloren gegangenen Ereignissen.

3. Ein Filtermechanismus kann sowohl alle Ereignisnachrichten verwerfen, die zu einem ignorierbaren Ereignis gehören, als auch einzelne Informationseinheiten, die für den zu entwickelnden Erkennungs- und Abwehrmechanismus von Insiderbedrohungen keine Relevanz aufweisen. Damit wird eine unnötige Weiterverarbeitung verhindern. Dieser Mechanismus ist gleichzeitig eine Vorbereitung für die in den Abschnitten 6.2.2.2 und 6.2.2.3 geforderte Datenminimierung und De-Identifizierung zur Erhöhung des Datenschutzes (vgl. auch Abschnitt 7.2.3).

4. Die zu einem Ereignis gesammelten Ereignisnachrichten werden korreliert und in ein JSON-Format überführt, bevor dann diese konsolidierte Ereignisnachricht an eine entfernte Datenverarbeitungseinheit gesendet wird, die letztlich die Funktionalität der Erkennung und Abwehr von Insiderbedrohungen umsetzt.

Die unbearbeiteten Beispielereignisnachrichten aus Abschnitt 5.3.2 werden durch *audisp-hostmon* mit den genannten Schritten in die folgende konsolidierte Ereignisnachricht überführt, wobei hier zur besseren Lesbarkeit vom korrekten JSON-Format abgewichen wird:

```
31317: {
  node: mw-host,
  timestamp: 1547121809.485,
  types: {
    PROCTITLE: {
      proctitle: /home/mw/Downloads/WTEpZSFwgb -ipc.fd=3 scan
    },
    SOCKADDR: {
```

```
      addr: 192.168.225.69,
      family: inet,
      port: 82
    },
    SOCKETCALL: {
      a0: 0x9d, a1: 0x19d85988, a2: 0x10,
      nargs: 3
    },
    SYSCALL: {
      a0: 0x19302860, a1: 0x0, a2: 0x0,
      arch: i386,
      auid: mw, uid: mw gid: mw,
      comm: WTEpZSFwgb,
      exe: /home/mw/Downloads/WTEpZSFwgb,
      exit: -115(EINPROGRESS),
      items: 0,
      pid: 1750, ppid: 1734,
      success: yes,
      syscall: connect,
      tty: pts0,
    }
  }
}
```

5.4 Systemaufruf-Graphen

Anhand der vorliegenden Ereignisnachrichten des *Linux Audit Systems* und den darin enthaltenen Informationen können Beziehungen zwischen verschiedenen Ressourcen eines Rechners definiert werden, die Aktivitäten eines Nutzers dieses Rechners abbilden. Die Ressourcen werden dabei anhand ihrer Eigenschaften unterteilt in die drei Typen

- *Prozesse* (P),
- *Dateisystemobjekte* (F) sowie
- externe und interne *Interprozesskommunikationsschnittstellen* oder auch *Sockets* (S) genannt.

Dateisystemobjekte umfassen dabei alle regulären Dateien, Verknüpfungen sowie Ordner. Die Beziehungen zwischen den Ressourcen eines Rechners lassen sich als ein gerichteter Graph modellieren, der nachfolgend als *Systemaufruf-Graph* (SysGraph) bezeichnet wird und folgendermaßen definiert ist:

Definition 5.1 (Systemaufruf-Graph)**.** Ein SysGraph $\mathcal{G}_{sys} = (V, E)$ ist ein gerichteter Graph, bestehend aus einer Menge V von Knoten, welche die Ressourcen eines Rechners repräsentieren sowie aus einer Menge E von Kanten, welche die Beziehungen zwischen diesen Ressourcen repräsentieren. Die Knoten sind für eine bessere Unterscheidbarkeit entsprechend ihrer Ressourcentypen unterschiedlich gefärbt.

Bei der Färbung von SysGraphen handelt es sich nicht um eine Knotenfärbung im Sinne der Graphentheorie, die nur dann gültig ist, wenn je zwei beliebige benachbarte Knoten nicht die gleiche Farbe besitzen [Die17, Kapitel 5]. Tatsächlich gilt dies für F- und für S-Knoten. P-Knoten

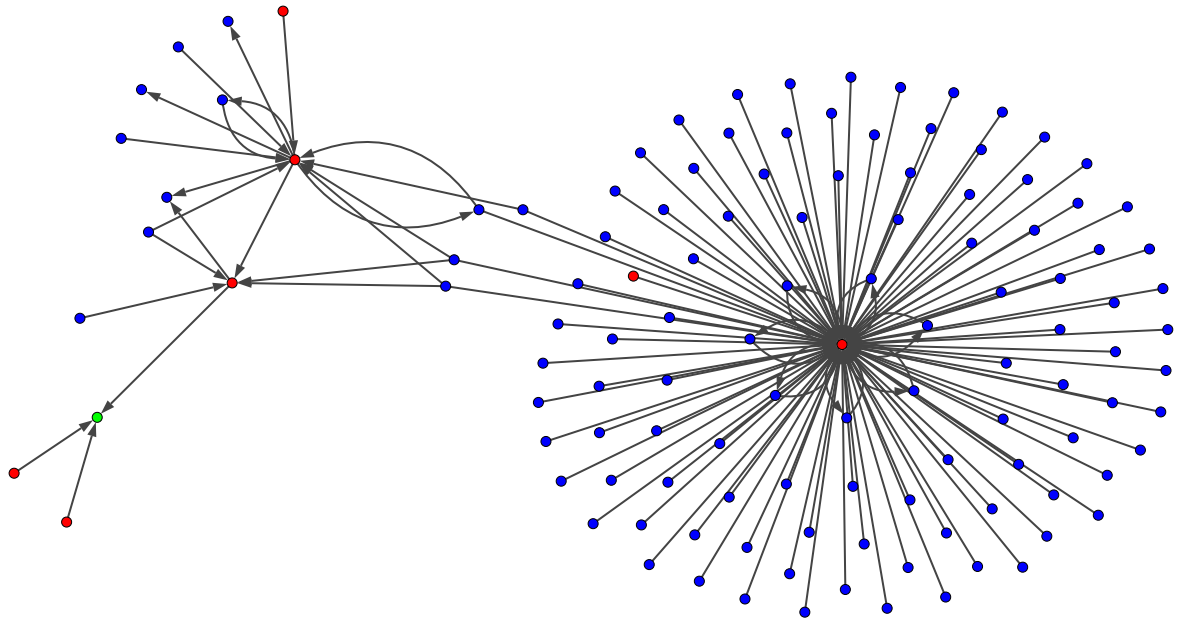


Abbildung 5.6: Beispiel eines SysGraphen $\mathcal{G}_{sys}$. Prozess-Knoten sind rot, Dateisystemobjekt-Knoten sind blau und Socket-Knoten sind grün gefärbt.

können allerdings benachbart sein (s. Abschnitt 5.4.1). Für die Knotenfärbung existiert eine Funktion $f : V \mapsto \{\text{rot}, \text{blau}, \text{grün}\}$ mit $f(v) = \text{rot}$, wenn v vom Ressourcentyp *Prozess* ist (P-Knoten), $f(v) = \text{blau}$, wenn v vom Ressourcentyp *Dateisystemobjekt* ist (F-Knoten) und $f(v) = \text{grün}$, wenn v vom Ressourcentyp *Socket* ist (S-Knoten). SysGraphen ermöglichen, die komplexen Strukturen der Interaktionen von Nutzern mit Ressourcen eines Rechners erfassbar und vor allem analysierbar zu machen. Ein Beispiel-Systemaufruf-Graph ist in Abbildung 5.6 illustriert.

5.4.1 Konstruktion von Systemaufruf-Graphen anhand von Ereignisnachrichten des Linux Audit Systems

Ein SysGraph wird anhand der Auswertung von Systemaufrufen eines Rechners konstruiert. Dabei ist die Domäne eines SysGraphen beschränkt auf die ununterbrochene Laufzeit des Rechner-Betriebssystems. Ein Neustart des Betriebssystems zieht zwangsläufig einen neuen SysGraphen nach sich. Mit den drei definierten Knotentypen P-Knoten, F-Knoten und S-Knoten gibt es fünf verschiedene Knotenpaarkonstellationen, die durch eine Kante miteinander verbunden sein können. Aus welchen Ereignisnachrichten diese Konstellationen hervorgehen und wie sie konstruiert werden, wird im Folgenden aufgezeigt.

5.4.1.1 Prozess • → • Prozess

Anhand der Erfassung aller `fork()` beziehungsweise `clone()` Systemaufrufe könnten die Eltern-Kind-Beziehungen der Prozesse in einen SysGraphen integriert werden. Das *Linux Audit System* liefert allerdings bereits an anderer Stelle die dafür notwendigen Informationen. Die meisten konsolidierten Ereignisnachrichten beinhalten als Teil der prozessspezifischen Informationen die

Prozess-ID des Prozesses, der die Ereignisnachricht angestoßen hat, sowie dessen Elternprozess-ID. In der Ereignisnachricht aus Abschnitt 5.3.4 sind diese Informationen beispielsweise in den Informationseinheiten `pid` und `ppid` ersichtlich:

```
31317: {node: mw-host, timestamp: 1547121809.485, types: {..., SYSCALL:
    {a0: 0x19302860, a1: 0x0, a2: 0x0, arch: i386, auid: mw, exe:
    /home/mw/Downloads/WTEpZSFwgb, exit: -115(EINPROGRESS), gid: mw, pid:
    1750, ppid: 1734, success: yes, syscall: connect, uid: mw}}}
```

Somit können zwei P-Knoten $P_1 = 1734$ und $P_2 = 1750$ in $\mathcal{G}_{sys}$ erzeugt werden, die durch ihre IDs definiert und durch eine Kante $e = (P_1, P_2)$ miteinander verbunden sind. Die Kante beschreibt dabei die Eltern-Kind-Beziehung zwischen den beiden Prozessen. Der Vorteil gegenüber der Erfassung aller `fork()` und `clone()` Systemaufrufe ist die Vermeidung der dadurch zusätzlich entstehenden Ereignisnachrichten und die Fokussierung auf nur diejenigen Prozesse, die Interaktionen mit Ressourcen durchführen, die auch von Relevanz für den zu entwickelnden Schutzmechanismus sind. An dieser Stelle wird in Kauf genommen, dass die Beziehung zwischen einem Prozess und dessen Vorfahren-Prozessen mithilfe gezielter `fork()` Aufrufe verschleiert werden kann (s. Abschnitt 5.8). Es kann außerdem passieren, dass Prozess-IDs wiederverwendet werden, was in der aktuellen Konstruktion der SysGraphen nicht berücksichtigt wird. Ebenfalls unberücksichtigt bleiben die Fälle, in denen sich der Elternprozess im Laufe der Ausführung eines Prozesses ändert. Dadurch werden in den SysGraphen immer nur die ursprüngliche Eltern-Kind-Beziehung zwischen Prozessen festgehalten.

5.4.1.2 Prozess • → • Dateisystemobjekt

Mit den konsolidierten Ereignisnachrichten, die durch die Überwachung von Zugriffen auf Dateisystemobjekte anfallen (vgl. Abschnitt 5.3.1.1), können Schreibzugriffe von Prozessen auf Dateisystemobjekte anhand der Dateistatusindikatoren `O_WRONLY` oder `O_RDWR` des `openat()` Systemaufrufs erkannt werden. Mit diesen Informationen wird ein P-Knoten erzeugt, der den schreibenden Prozess repräsentiert, sowie ein F-Knoten, der sich aus dem Pfad beziehungsweise genauer aus der Inode der zu beschreibenden Datei beziehungsweise des zu ändernden Dateisystemobjektes ergibt. Eine Kante vom P-Knoten zum F-Knoten repräsentiert damit das Beschreiben des Dateisystemobjektes mit Daten.[6] Eine entsprechend konsolidierte Ereignisnachricht sieht beispielsweise folgendermaßen aus:

```
{13212: {node: mw-host, timestamp: 1571919963.301, types: {PATH:
    [{inode: 539074, mode: file,644, name: ~/Desktop/file.txt, ogid: mw,
    ouid: mw}], SYSCALL: {a0: 0xffffff9c, a1: 0x5620d6373cf0, a2:
    O_WRONLY|O_CREAT, a3: 0x1b6, arch: x86_64, auid: mw, exe:
    /usr/bin/vim.tiny, exit: 4, gid: mw, pid: 19696, ppid: 19688,
    success: yes, syscall: openat, uid: mw}}}}
```

Der P-Knoten $P = 19696$ ergibt sich aus der Informationseinheit `pid`. Der F-Knoten $F = 539074$ ist eindeutig durch die Informationseinheit `inode` definiert. Die beiden Knoten sowie die Kante $e = (P, F)$ kann dadurch dem SysGraphen $\mathcal{G}_{sys}$ hinzugefügt werden. Der Pfad

6. Streng genommen repräsentiert eine solche Kante nur das potenzielle Beschreiben des Dateisystemobjektes, da auf die Überwachung von tatsächlichen Schreiboperationen beispielsweise mittels des `write()` Systemaufrufs verzichtet wird (vgl. Abschnitt 5.3.1.1).

~/Desktop/file.txt, der durch den F-Knoten repräsentiert wird, ergibt sich aus der Informationseinheit name. Dieser Pfad ist prinzipiell ebenfalls eindeutig, kann sich allerdings im Laufe der Zeit durch Umbenennung oder Verschiebung des Dateisystemobjektes ändern.

Weitere Ereignisse, die eine Kante von einem P-Knoten zu einem F-Knoten hervorrufen, können beispielsweise das Löschen oder Umbenennen eines Dateisystemobjektes sein.

5.4.1.3 Dateisystemobjekt • → • Prozess

Äquivalent zum vorigen Abschnitt werden Lesezugriffe von Prozessen auf Dateisystemobjekte anhand der Dateistatusindikatoren des openat() Systemaufrufs ersichtlich. Dessen Überwachung durch das *Linux Audit System* liefert Informationen über Dateisystemobjekte, die von Prozessen mit den Indikatoren O_RDONLY oder O_RDWR versucht werden zu öffnen. Bei erfolgreicher Durchführung des Systemaufrufs kann demnach ein F-Knoten für den angegebenen Pfad beziehungsweise der zugehörigen Inode sowie ein P-Knoten für den anstoßenden Prozess erzeugt und zusammen mit der Kante $e = (F, P)$ dem SysGraphen $\mathcal{G}_{sys}$ hinzugefügt werden. Die Kante vom F-Knoten zum P-Knoten repräsentiert damit das Auslesen von Daten aus dem Dateisystemobjekt.[7] Ein Beispiel einer passenden konsolidierten Ereignisnachricht sieht folgendermaßen aus:

```
{13165: {node: mw-host, timestamp: 1571919957.064, types: {PATH:
    [{inode: 525373, mode: file,600, name:
    /home/mw/Desktop/.anotherfile.txt.swp, ogid: mw, ouid: mw}], SYSCALL:
    {a0: 0xffffff9c, a1: 0x5620d6373ca0, a2:
    O_RDONLY|O_RDWR|O_CREAT|O_EXCL|O_NOFOLLOW, a3: 0x180, arch: x86_64,
    auid: mw, exe: /usr/bin/vim.tiny, exit: 3, gid: mw, pid: 19696, ppid:
    19688, success: yes, syscall: openat, uid: mw}}}}
```

Für dieses Beispiel ergibt sich aufgrund des Dateistatusindikators O_RDONLY der F-Knoten $F = 525373$ und der P-Knoten $P = 19696$ sowie eine Kante vom F- zum P-Knoten. Darüber hinaus ergibt sich in diesem Beispiel aufgrund des Dateistatusindikators O_RDWR entsprechend der Anweisungen aus Abschnitt 5.4.1.2 auch eine Kante in die entgegengesetzte Richtung, also vom P-Knoten zum F-Knoten.

5.4.1.4 Prozess • → • Socket

Mit der Überwachung von Rechnernetz-assoziierten Systemaufrufen durch das *Linux Audit System* entsprechend des Anwendungsfalls aus Abschnitt 5.3.1.3 werden Informationen darüber gesammelt, wann und wie ein Prozess eine Verbindung zu einem Socket aufbaut. Zu diesem Zweck können die connect() und die sendto() Systemaufrufe überwacht und aufgezeichnet werden. Aus den darin enthaltenen prozessspezifischen Informationen kann ein P-Knoten für denjenigen Prozess erzeugt werden, der den Systemaufruf angestoßen hat und aus den systemaufrufspezifischen Informationen kann ein S-Knoten erzeugt werden, zu dem der Prozess eine Verbindung aufgebaut hat. Zusammen mit einer Kante vom P- zum S-Knoten können sie dem

7. Eine solche Kante stellt streng genommen nur einen starken Indikator für das Auslesen von Daten aus dem Dateisystemobjekt dar. Eine gesicherte Aussage darüber könnte nur die Überwachung der tatsächlichen read() und ähnlicher Systemaufrufe erlauben, die allerdings aus Effizienzgründen hier nicht durchgeführt wird (vgl. Abschnitt 5.3.1.1).

SysGraphen $\mathcal{G}_{sys}$ hinzugefügt werden. Der P-Knoten ist durch die Prozess-ID bestimmt. Für den S-Knoten enthalten die systemaufrufspezifischen Informationen die Adressfamilie der Verbindung, etwa `AF_LOCAL` für eine rechnerinterne Verbindung oder `AF_INET` für eine IPv4-Verbindung, sowie, je nach der Adressfamilie, die eigentliche Adresse. Diese Informationen definieren den S-Knoten eindeutig. Folgendes aus Abschnitt 5.3.4 bereits bekanntes Beispiel zeigt eine dazu passende konsolidierte Ereignisnachricht:

```
{31317: {node: mw-host, timestamp: 1547121809.485, types: {SOCKADDR:
    {addr: 192.168.225.69, family: AF_INET, port: 82}, SYSCALL: {a0:
    0x19302860, a1: 0x0, a2: 0x0, arch: i386, auid: mw, exe:
    /home/mw/Downloads/WTEpZSFwgb, exit: -115(EINPROGRESS), gid: mw, pid:
    1750, ppid: 1734, success: yes, syscall: connect, uid: mw}}}}
```

Der P-Knoten $P = 1705$ ergibt sich aus der `pid` Informationseinheit. Mit dem `connect()` Systemaufruf beinhaltet die Ereignisnachricht die `SOCKADDR` Informationen, die wiederum den S-Knoten $S = (\text{AF\_INET}, 192.168.225.69)$ ergeben. Die Kante $e = (P, S)$ beschreibt dadurch im vorliegenden Beispiel den Verbindungsaufbau vom P- zum S-Knoten.[8]

5.4.1.5 Socket • → • Prozess

Die Überwachung aller `accept4()` und `recvfrom()` Systemaufrufe liefert äquivalent zum vorigen Abschnitt Verbindungsaufbauten von einem Socket zu einem Prozess. Bei erfolgreicher Durchführung dieser Systemaufrufe kann demnach ein S-Knoten für den beteiligten Socket sowie ein P-Knoten für den entsprechenden Prozess erzeugt und zusammen mit der Kante $e = (S, P)$ dem SysGraphen $\mathcal{G}_{sys}$ hinzugefügt werden. Die Kante beschreibt damit den Verbindungsaufbau vom S- zum P-Knoten.[9] Eine Ereignisnachricht, die einen solchen Verbindungsaufbau von einem Socket zu einem Prozess zeigt, sieht folgendermaßen aus:

```
{11941: {node: mw-host, timestamp: 1571918301.328, types: {SOCKADDR:
    {addr: 192.168.225.69, family: AF_INET, port: 54806}, SYSCALL: {a0:
    0x3, a1: 0x7fffaadc33c0, a2: 0x7fffaadc3354, a3: 0x800, arch: x86_64,
    auid: mw, exe: /bin/nc.openbsd, exit: 4, gid: mw, pid: 19596, ppid:
    19588, success: yes, syscall: accept4, uid: mw}}}}
```

Die resultierenden Knoten sind $S = (\text{AF\_INET}, 192.168.225.69)$ sowie $P = 19596$.

5.4.2 Informationsfluss-Systemaufruf-Graphen

Informationsfluss-SysGraphen sind spezielle knotengenerierte Sub-SysGraphen, die ausgehend von einem Generatorknoten chronologisch zusammenhängende Beziehungen zwischen Ressourcen eines Rechners repräsentieren und damit einzelne Nutzeraktivitäten und Teilabläufe, die im weitesten Sinne als Informationsflüsse aufgefasst werden können, isoliert voneinander darstellen. Knotengenerierte Sub-SysGraphen werden wie folgt definiert:

8. Anders als bei Kanten zwischen Prozessen und Dateisystemobjekten kann hier nicht angenommen werden, dass Informationen vom Prozess zum Socket gesendet werden. Demnach beschreibt eine solche Kante nur die Tatsache, dass eine Verbindung vom P- zum S-Knoten initiiert wurde.
9. Eine solche Kante kann nicht unbedingt als das Senden von Informationen vom S- zum P-Knoten aufgefasst werden. Sie repräsentiert nur die Tatsache, dass eine Verbindung vom S- zum P-Knoten initiiert wurde.

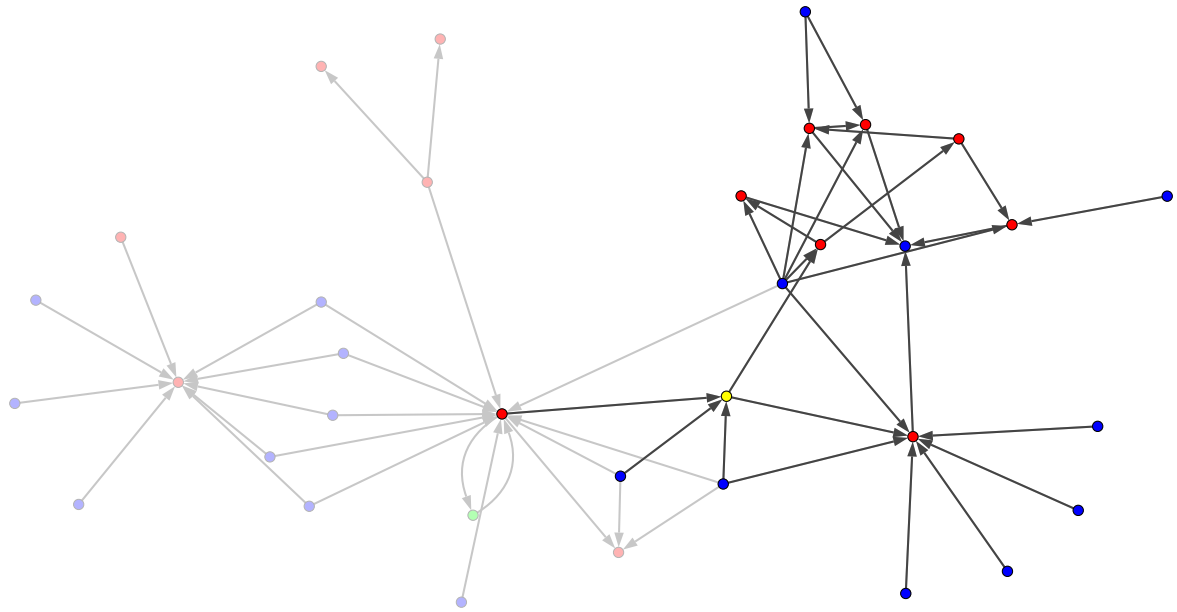


Abbildung 5.7: Beispiel eines Informationsfluss-SysGraphen $\widetilde{\mathcal{G}}_{sys} = \langle \mathcal{G}_{sys}, g \rangle$. Der Generatorknoten ist gelb gefärbt.

Definition 5.2 (Knotengenerierter Sub-SysGraph)**.** Ein von einem beliebigen Knoten $g \in \mathcal{G}_{sys} = (V, E)$ generierter Sub-SysGraph $\mathcal{G}'_{sys} = (V', E')$, nachfolgend mit $\langle \mathcal{G}_{sys}, g \rangle$ bezeichnet, umfasst eine Teilmenge $V' \subseteq V$ der Knoten, die von g aus über einen Pfad $(g, a_1, \ldots, a_n) \in \mathcal{G}_{sys}$ erreichbar sind sowie der Knoten, von denen aus g über einen Pfad $(a_0, \ldots, a_{n-1}, g) \in \mathcal{G}_{sys}$ erreichbar ist. Die Gesamtheit aller Pfadkanten beschreibt E'.

Ein knotengenerierter Sub-SysGraph enthält nicht unbedingt alle Knoten, die von g aus erreichbar sind oder von denen aus g erreichbar ist. Weiterhin handelt es sich nicht unbedingt um einen induzierten Subgraphen (vgl. Abschnitt 5.1.2). Das zeigt sich im Fall von Informationsfluss-SysGraphen $\widetilde{\mathcal{G}}_{sys} = \langle \mathcal{G}_{sys}, g \rangle = (\tilde{V}, \tilde{E})$, die mit speziellen Eigenschaften aus einem knotengenerierten Sub-SysGraphen hervorgehen.

Definition 5.3 (Informationsfluss-Systemaufruf-Graph)**.** Ein Informationsfluss-SysGraph $\widetilde{\mathcal{G}}_{sys} = \langle \mathcal{G}_{sys}, g \rangle = (\tilde{V}, \tilde{E})$ ist ein durch einen Generatorknoten g erzeugter Subgraph des SysGraphen $\mathcal{G}_{sys}$ mit der folgenden Eigenschaft: Alle Pfade in $\mathcal{G}_{sys}$, die in g beginnen und deren Kanten eine lückenlose chronologische Abfolge von Beziehungen zwischen Ressourcen darstellen, sind auch in $\widetilde{\mathcal{G}}_{sys}$ enthalten. Weiterhin sind alle direkten Vorgängerknoten von g sowie die jeweiligen Kanten zu g in $\widetilde{\mathcal{G}}_{sys}$ enthalten.

Ein Beispiel eines Informationsfluss-SysGraphen ist in Abbildung 5.7 illustriert. Für ihre Erzeugung müssen Zeitpunkte als Attribute an allen Kanten eines SysGraphen $\mathcal{G}_{sys}$ eingeführt werden, welche die Zeitpunkte der Beziehungen zwischen den Ressourcen eines Rechners festhalten. Die notwendigen Änderungen an den in Abschnitt 5.4.1 entwickelten SysGraphen sowie die genaue Erzeugung von Informationsfluss-SysGraphen wird nachfolgend spezifiziert.

5.4.2.1 Zeitpunkte der Beziehungen zwischen Ressourcen

Im Zuge der Konstruktion eines SysGraphen $\mathcal{G}_{sys} = (V, E)$ wird allen neu konstruierten Kanten $e \notin E$, die bisher noch nicht in $\mathcal{G}_{sys}$ existieren, eine Liste $t_e = [\texttt{timestamp}]$ als Attribut angefügt, die initial den Zeitpunkt `timestamp` enthält, der in der Ereignisnachricht angegeben ist. Sofern die Kante, die sich aus einer Ereignisnachricht ergibt, bereits in $\mathcal{G}_{sys}$ existiert, wird der Zeitpunkt, der in der Ereignisnachricht angegeben ist, der Liste hinzugefügt: $t_e = [t_0, \ldots, t_m, \texttt{timestamp}]$.

Die Zeitpunkte der Kanten zwischen zwei P-Knoten repräsentieren Annäherungen an die Erzeugungszeitpunkte der jeweiligen Kind-Prozesse. Diese Annäherungen sind allerdings für die vorliegende Arbeit ausreichend. Für eine Erfassung der echten Erzeugungszeitpunkte müssten alle `fork()` und `clone()` Systemaufrufe überwacht werden (vgl. Abschnitt 5.4.1.1). Darüber hinaus werden die Zeitpunkte aller Kanten von F- zu P-Knoten als Lesezugriffszeitpunkte und die Zeitpunkte aller Kanten von P- zu F-Knoten als Schreibzugriffszeitpunkte verstanden. Äquivalent werden die Zeitpunkte aller Kanten von S- zu P-Knoten als Verbindungsannahmezeitpunkte und die Zeitpunkte aller Kanten von P- zu S-Knoten als Verbindungsaufbauzeitpunkte verstanden.

5.4.2.2 Extraktion von Informationsfluss-SysGraphen

Mit den folgenden Anweisungen wird ein Informationsfluss-SysGraph $\widetilde{\mathcal{G}}_{sys} = \langle \mathcal{G}_{sys}, g \rangle = (\tilde{V}, \tilde{E})$ ausgehend von einem Generatorknoten g aus einem SysGraphen $\mathcal{G}_{sys} = (V, E)$ extrahiert:

1) Setze initial:

   - $\tilde{V} = \{g\} \cup Neig(g)$
   - $\tilde{E} = \{(a, g) \mid \forall a \in Pred(g)\} \cup \{(g, b) \mid \forall b \in Succ(g)\}$

   $\widetilde{\mathcal{G}}_{sys}$ enthält somit anfangs alle direkten Nachbarn $Neig(g)$ von g in $\mathcal{G}_{sys}$ sowie g selbst und alle zugehörigen Kanten (a, g) für alle $a \in Pred(g)$ sowie (g, b) für alle $b \in Succ(g)$.

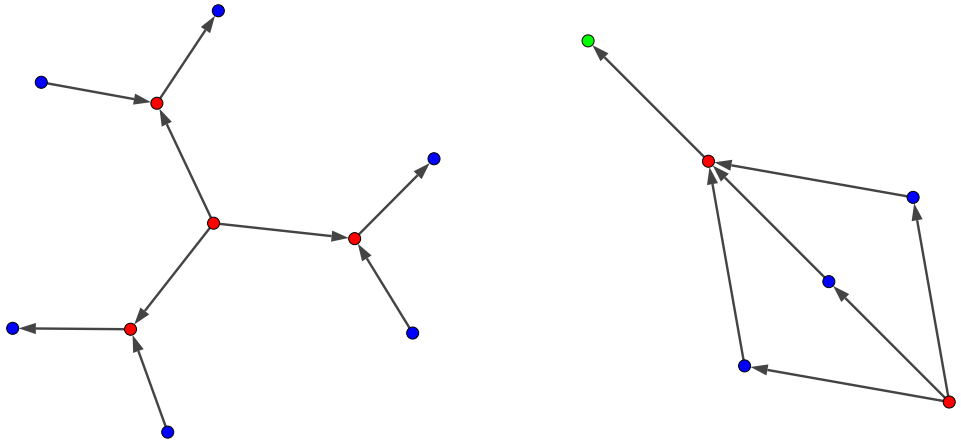
2) Für alle Pfade $(a_0, \ldots, a_{n-1}, a_n = g) \in \mathcal{G}_{sys}$ der Länge $n \geq 2$ gilt: Wenn für alle $i \in \{0, \ldots, n-2\}$ der Zeitpunkt einer Beziehung zwischen den Ressourcen a_i und a_{i+1} vor einem Zeitpunkt einer Beziehung zwischen den folgenden Ressourcen a_{i+1} und a_{i+2} liegt, dann füge alle Knoten des Pfades a_i mit $i \in \{0, \ldots, n-2\}$ der Knotenmenge $\tilde{V}$ und alle Kanten des Pfades (a_i, a_{i+1}) mit $i \in \{0, \ldots, n-2\}$ der Kantenmenge $\tilde{E}$ hinzu:

   - $\tilde{V} = \tilde{V} \cup \{a_0, \ldots, a_{n-2}\}$
   - $\tilde{E} = \tilde{E} \cup \{(a_i, a_{i+1}) \mid i \in \{0, \ldots, n-2\}\}$

3) Für alle Pfade $(a_0 = g, a_1, \ldots, a_n) \in \mathcal{G}_{sys}$ der Länge $n \geq 2$ gilt: Wenn für alle $i \in \{0, \ldots, n-2\}$ der Zeitpunkt einer Beziehung zwischen den Ressourcen a_{i+1} und a_{i+2} nach einem Zeitpunkt einer Beziehung zwischen den vorigen Ressourcen a_i und a_{i+1} liegt, dann füge alle Knoten des Pfades a_i mit $i \in \{2, \ldots, n\}$ der Knotenmenge $\tilde{V}$ und alle Kanten des Pfades (a_i, a_{i+1}) mit $i \in \{2, \ldots, n\}$ der Kantenmenge $\tilde{E}$ hinzu:

   - $\tilde{V} = \tilde{V} \cup \{a_2, \ldots, a_n\}$
   - $\tilde{E} = \tilde{E} \cup \{(a_i, a_{i+1}) \mid i \in \{2, \ldots, n\}\}$

Aus Effizienzgründen kann die Pfadlänge in Schritt 2) beschränkt werden, sodass nur eine begrenzte Anzahl an indirekten Vorgängerknoten in den Informationsfluss-SysGraphen aufgenommen wird. Dadurch kann sich eine Analyse der Graphenstrukturen auf die Vorgänge fokussieren, die vom Generatorknoten ausgehen beziehungsweise indirekt beeinflusst werden.



(a) $\widetilde{\mathcal{G}}_{sys,1}$ **(b)** $\widetilde{\mathcal{G}}_{sys,2}$

m_i:	m_2	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}	m_{11}	m_{12}	m_{13}	m_{14}	m_{15}
$H(m_i)$:	3	6	0	3	0	0	0	0	0	0	0	0	0

Abbildung 5.8: Zwei grundsätzlich verschiedene Informationsfluss-SysGraphen mit jeweils der gleichen 3-Motiv-Signatur: $\{m_2 = 3, m_4 = 6, m_6 = 3\}$

5.5 Signaturen von Systemaufruf-Graphen

Die Analyse der Graphenstrukturen von SysGraphen im Allgemeinen und Informationsfluss-SysGraphen im Speziellen lässt Rückschlüsse auf besondere Nutzeraktivitäten beziehungsweise auf bestimmte Arten von Nutzeraktivitäten zu. Diese Graphenstrukturen können anhand der im Graphen vorkommenden Netzwerkmotive erfasst und interpretiert werden (vgl. Abschnitt 5.1.3). Das Besondere an SysGraphen sind allerdings deren unterschiedliche Knotentypen, welche die drei Ressourcentypen *Prozesse*, *Dateisystemobjekte* und *Sockets* repräsentieren. Dadurch können zwei grundsätzlich verschiedene Nutzeraktivitäten zu identischen Motiv-Signaturen führen.

Zwei Beispiele solcher Informationsfluss-SysGraphen finden sich in Abbildung 5.8. Der SysGraph $\widetilde{\mathcal{G}}_{sys,1}$ in Abbildung 5.8a zeigt einen Prozess, der drei weitere Kindprozesse erzeugt, die jeweils von einer Datei lesen und in eine andere Datei schreiben. Die zugehörige 3-Motiv-Signatur enthält drei m_2-Motive, sechs m_4-Motive und drei m_6-Motive und lautet daher in kompakter Schreibweise: $\{m_2 = 3, m_4 = 6, m_6 = 3\}$. Der SysGraph $\widetilde{\mathcal{G}}_{sys,2}$ in Abbildung 5.8b zeigt einen Prozess, der in drei unterschiedliche Dateien schreibt und einen anderen Prozess, der aus diesen Dateien liest und eine Verbindung mit einem Socket initiiert. Auch dieser SysGraph hat die gleiche genannte 3-Motiv-Signatur, obwohl sich der in der Graphenstruktur modellierte Vorgang eindeutig von $\widetilde{\mathcal{G}}_{sys,1}$ unterscheidet. Aus diesem Grund wurde eine erweiterte 3-Motiv-Signatur entwickelt, die den drei unterschiedlichen Knotentypen von SysGraphen Rechnung trägt.

Definition 5.4 (SysGraph-Signatur)**.** Die Unterteilung aller möglichen klassischen 3-Motive von Graphen in knotentypabhängige SysGraph-Motive ermöglicht eine eindeutige *SysGraph-Signatur* $S_{\widetilde{\mathcal{G}}_{sys}}$, die die Anzahl der im SysGraphen vorkommenden 3-Motive mit unterscheidbaren Knotentypkonstellationen angibt. Die zugehörige Berechnungsfunktion wird mit $Sig(\widetilde{\mathcal{G}}_{sys})$ bezeichnet.

Das Auffinden und Zählen vorliegender SysGraph-Motive in einem SysGraphen basiert zunächst auf dem Auffinden klassischer 3-Motive (vgl. Abschnitt 5.1.3.1) inklusive dem Lösen des dabei entstehenden Graphen-Isomorphismus-Problems (vgl. Abschnitt 5.1.3.2). Dabei können bereits existierende Algorithmen wie etwa von Wernicke [Wer05] oder von Ribeiro und Silva [RS10] verwendet werden, die allerdings bei der Bearbeitung des Graphen-Isomorphismus-Problems an die Besonderheiten von SysGraphen angepasst werden müssen, um eine eindeutige und korrekte Lösung zu finden und damit eine SysGraph-Signatur effizient berechnen zu können. Genaue Erläuterungen dazu finden sich in Abschnitt 5.5.3. Darüber hinaus wird in Abschnitt 5.6 eine spezielle Methode für das Auffinden von SysGraph-Motiven entwickelt, die dem vorliegenden Kontext der Insiderbedrohungserkennung und -prävention gerecht wird und die erwähnten existierenden Algorithmen ersetzt.

Jede der 16 möglichen klassischen Isomorphieklassen eines 3-Motivs in einem Graphen, also jeder induzierte Subgraph mit fester Knotenanzahl 3 ohne Graphenisomorphismen (vgl. Abschnitt 5.1.2 und 5.1.3), lässt sich im Kontext von SysGraphen aufschlüsseln in $3^3 = 27$ spezielle Knotentypkonstellationen, da jeder der drei Knoten jeweils drei unterschiedliche Typen haben kann. Diese 27 zu einem klassischen 3-Motiv gehörenden Sub-SysGraphen werden als *SysGraph-Motivcluster* M_i bezeichnet, wobei $i \in \{0, \dots, 15\}$ der klassische 3-Motiv-Index ist. Damit ergeben sich $27 \cdot 16 = 432$ mögliche induzierte Sub-SysGraphen, aus denen die SysGraph-Motive hervorgehen. Diese Zahl spiegelt allerdings noch nicht die Anzahl aller möglichen SysGraph-Motive wider, sondern reduziert sich dafür auf 65 aus drei Gründen. Zum ersten werden drei der klassischen 3-Motivklassen und somit auch deren aufgeschlüsselte Knotentypkonstellationen ignoriert (s. Abschnitt 5.5.1), zum zweiten gibt es im Kontext von SysGraphen ungültige Knotentypkonstellationen (s. Abschnitt 5.5.2) und zum dritten existieren SysGraph-Isomorphismen, die zusammengefasst werden müssen (s. Abschnitt 5.5.3).

Entsprechend der Bezeichnung $m_0, \dots, m_{15}$ aller möglichen klassischen 3-Motive eines Graphen sollen auch die möglichen SysGraph-Motive eindeutige Bezeichnungen $m_0^{sys}, \dots, m_{64}^{sys}$ erhalten. Damit enthält eine SysGraph-Signatur am Ende die Bezeichnungen eines jeden in einem SysGraphen vorkommenden SysGraph-Motivs sowie dessen Häufigkeit. Die unterschiedlichen Nutzeraktivitäten der in Abbildung 5.8 gezeigten Informationsfluss-SysGraphen resultieren dadurch entsprechend in unterschiedlichen SysGraph-Signaturen:

$$(\mathbf{a}) : S_{\widetilde{G}_{sys,1}} = \{m_0^{sys} = 3, m_8^{sys} = 3, m_{13}^{sys} = 3, m_{26}^{sys} = 3\}$$
$$(\mathbf{b}) : S_{\widetilde{G}_{sys,2}} = \{m_4^{sys} = 3, m_{10}^{sys} = 3, m_{14}^{sys} = 3, m_{29}^{sys} = 3\}$$

5.5.1 Unbedeutende 3-Motivklassen

Von den 432 möglichen SysGraphen mit fester Knotenzahl 3 werden jene nicht betrachtet, bei denen mindestens ein Knoten nicht durch eine Kante mit einem beliebigen anderen Knoten verbunden ist. Sie spiegeln Nicht-Verbindungen wider, die allein schon aufgrund von Designentscheidungen bei der Konstruktion von SysGraphen stark beeinflusst werden (s. Abschnitt 5.5.2). Das trifft auf alle SysGraphen zu, die aus den klassischen 3-Motiven m_0, m_1 und m_3 hervorgehen. Somit werden $3 \cdot 27 = 81$ 3-knotige SysGraphen ignoriert und es verbleiben 351. Die Anzahl kann weiter reduziert werden, wie der folgende Abschnitt zeigt.

5.5.2 Unmögliche Knotentypkonstellationen

Die Konstruktion von SysGraphen anhand der Vorgaben aus Abschnitt 5.4.1 sowie anhand der dahinterliegenden Systeme führt dazu, dass eine Vielzahl der möglichen 3-knotigen SysGraphen keine gültigen Knotentypkonstellationen darstellen und demnach nicht in einem SysGraphen vorkommen können.

- Sofern der Anfangsknoten einer Kante ein F- oder ein S-Knoten ist, kann der Endknoten dieser Kante nur ein P-Knoten sein. Das folgt aus der Tatsache, dass *Dateisystemobjekte* nur von *Prozessen* gelesen werden und *Sockets* nur Verbindungen zu *Prozessen* initiieren können.
- Sofern zwei P-Knoten über eine Kante miteinander verbunden sind, kann es keine entgegengesetzte Kante zwischen diesen beiden P-Knoten geben. Andernfalls wären beide assoziierten Prozesse gegenseitig jeweils gleichzeitig Eltern- und Kindprozess.
- Ein P-Knoten kann keine zwei Vorgängerknoten haben, die jeweils ebenfalls P-Knoten sind, denn ein Prozess kann keine zwei unterschiedlichen Elternprozesse haben.
- Sofern sowohl der Vorgänger als auch der Nachfolger eines P-Knotens ebenfalls P-Knoten sind, dann können die beiden anderen P-Knoten nicht durch eine Kante miteinander verbunden sein. Andernfalls würde das aus der Sicht eines zugehörige Prozesses bedeuten, dass der Kindprozess eines eigenen Kindprozesses gleichzeitig der eigene Elternprozess ist.

Alle 3-knotigen SysGraphen, die mindestens eine der genannten Eigenschaften verletzen, sind ungültig und können demnach im Folgenden ignoriert werden. Dadurch reduziert sich die Anzahl der möglichen induzierten Sub-SysGraphen, die von Interesse sind, auf 80. Da diese Anzahl allerdings noch isomorphe SysGraphen enthält, wird sie im folgenden Abschnitt noch einmal reduziert.

5.5.3 Graphen-Isomorphismen im Kontext von 3-knotigen SysGraphen

Zu einem gegebenen 3-knotigen Sub-SysGraphen, der im Zuge der Erstellung einer SysGraph-Signatur aufgefunden wurde und nun die entsprechende SysGraph-Motiv-Anzahl in dieser Signatur inkrementieren soll, muss zunächst das *SysGraph-Isomorphismus-Problem* gelöst werden. Dieses Problem umfasst das klassische Graph-Isomorphismus-Problem, denn der 3-knotige Sub-SysGraph fällt in genau eine der klassischen 16 möglichen Motivklassen (vgl. Abschnitt 5.1.3.2), die ermittelt werden muss. Das SysGraph-Isomorphismus-Problem geht allerdings darüber hinaus, denn für eine Motivklasse existieren 27 verschiedene Knotentypkonstellationen, die wiederum selbst isomorphe SysGraphen enthalten können. Ein Beispiel ist in Abbildung 5.9 veranschaulicht. Aus den drei isomorphen klassischen Graphen der 3-Motivklasse m_{13} können bis zu sechs isomorphe 3-knotige SysGraphen des SysGraph-Motivclusters M_{13} hervorgehen, die demnach auch jeweils die gleichen SysGraph-Motive repräsentieren. In Abbildung 5.9 sind beispielsweise die beiden zu den Isomorphieklassen zugehörigen SysGraph-Motive m_{61}^{sys} und m_{62}^{sys} dargestellt. Die bereits reduzierten 80 induzierten Sub-SysGraphen mit fester Knotenzahl 3 aus den vorigen Abschnitten reduzieren sich durch die Zusammenfassung der isomorphen SysGraphen somit auf bereits erwähnte verbleibende 65 nicht-isomorphe Sub-SysGraphen.

Für die Lösung des SysGraph-Isomorphismus-Problems wurde eine spezielle Berechnungsvorschrift entwickelt, deren Ergebnis für alle nicht-isomorphen 3-knotigen SysGraphen unterschiedlich, aber für alle isomorphen 3-knotigen SysGraphen invariant ist. Die Berechnungsvorschrift

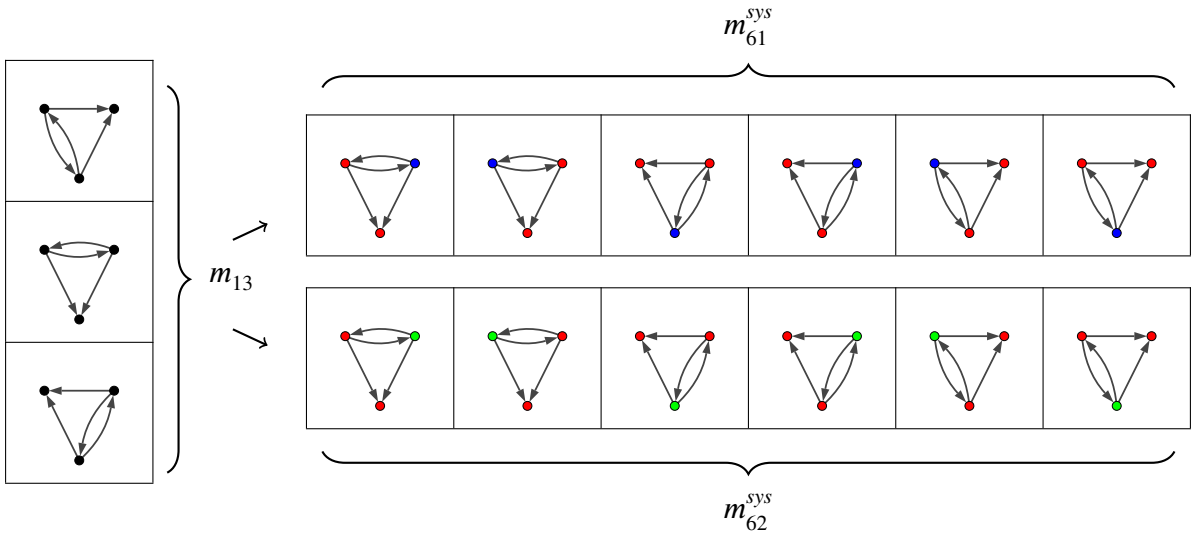
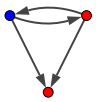


Abbildung 5.9: Zweimal sechs isomorphe 3-knotige SysGraphen, die aus der klassischen Motivklasse m_{13} hervorgehen und die in M_{13} die SysGraph-Motive m_{61}^{sys} und m_{62}^{sys} darstellen.

ist allerdings nicht auf andere feste Anzahlen von Knoten übertrag- oder gar verallgemeinerbar. Sie lautet wie folgt:

Gegeben sei ein 3-knotiger Sub-SysGraph $\mathcal{G}'_{sys} = (V', E')$

Beispiel anhand von 

1) Berechne für jeden Knoten $a_i \in V' = \{a_0, a_1, a_2\}$ einen sogenannten Knotenindex υ_{a_i} mit

$$\upsilon(a_i) = 2 \cdot deg_{in}(a_i) - deg_{out}(a_i).$$

$V' = \{a_0 = F, a_1 = P_1, a_2 = P_2\}$

$\upsilon_{a_0} = \upsilon(a_0) = 2 \cdot 1 - 2 = 0$

$\upsilon_{a_1} = \upsilon(a_1) = 2 \cdot 1 - 2 = 0$

$\upsilon_{a_2} = \upsilon(a_2) = 2 \cdot 2 - 0 = 4$

2) Sortiere die Knotenmenge V' absteigend zu einer Liste $V'_{sort} = [a'_0, a'_1, a'_2]$ primär anhand des jeweiligen Knotenindexes $\upsilon_{a'_i}$ und sekundär anhand der Knotentypwertigkeit t_{a_i}, die sich mithilfe der Funktion $t : V' \mapsto \mathbb{N}$ berechnet:

$$t(a) = \begin{cases} 0 & \text{wenn } a \text{ ein P-Knoten} \\ 1 & \text{wenn } a \text{ ein F-Knoten} \\ 2 & \text{wenn } a \text{ ein S-Knoten.} \end{cases}$$

Sie weist einem Knoten gemäß seines Knotentyps eine Typwertigkeit zu.

$t_{a_0} = t(a_0) = 1$

$t_{a_1} = t(a_1) = 0$

$t_{a_2} = t(a_2) = 0$

$V'_{sort} = [a'_0 = P_2, a'_1 = F, a'_2 = P_1]$

3) Berechne den 3-Motivklassenindex $\kappa_{\mathcal{G}'_{sys}}$ von $\mathcal{G}'_{sys}$ anhand von V'_{sort} und den in Punkt 1) berechneten Knotenindizes υ_{a_i}: $$\kappa_{\mathcal{G}'_{sys}} = \kappa(V'_{sort}) = 12 \cdot \upsilon_{a'_0} + 4 \cdot \upsilon_{a'_1} + \upsilon_{a'_2}.$$	$\kappa(V'_{sort}) = 12 \cdot 4 + 4 \cdot 0 + 0 = 48$
4) Berechne den Knotentypkonstellationsindex $\tau_{\mathcal{G}'_{sys}}$ anhand von V'_{sort}: $$\tau_{\mathcal{G}'_{sys}} = \tau(V'_{sort}) = 9 \cdot t_{a'_0} + 3 \cdot t_{a'_1} + t_{a'_2},$$ mithilfe der bereits in Punkt 2) erwähnten Typwertigkeitsfunktion t.	$\tau(V'_{sort}) = 9 \cdot 0 + 3 \cdot 1 + 0 = 3$
5) Der SysGraph-Index $\mathcal{I}_{\mathcal{G}'_{sys}}$ von $\mathcal{G}'_{sys}$ berechnet sich aus: $$\mathcal{I}_{\mathcal{G}'_{sys}} = 27 \cdot \kappa_{\mathcal{G}'_{sys}} + \tau_{\mathcal{G}'_{sys}}.$$	$\mathcal{I}_{\mathcal{G}'_{sys}} = 27 \cdot 48 + 3 = 1299$

Der errechnete SysGraph-Index $\mathcal{I}_{\mathcal{G}'_{sys}}$ von $\mathcal{G}'_{sys}$ ist für isomorphe 3-knotige SysGraphen immer identisch und für nicht-isomorphe 3-knotige SysGraphen immer unterschiedlich. Belegende Erläuterungen werden nachfolgend ausgeführt. Damit kann das SysGraph-Isomorphismus-Problem für 3-knotige SysGraphen gelöst werden. Anhand einer eineindeutigen Zuordnung kann zu jedem möglichen SysGraph-Index die zugehörige Bezeichnung des SysGraph-Motives $m_0^{sys}, \ldots, m_{64}^{sys}$ ermittelt werden.

Die Knotenindizes eines 3-knotigen Graphen

Der Knotenindex υ_{a_i} eines Knotens $a_i \in V'$ aus einem 3-knotigen Sub-SysGraphen $\mathcal{G}'_{sys} = (V', E')$, wie er in Punkt 1) berechnet wird, ist unabhängig von der Knotentypwertigkeit und bezieht sich damit ausschließlich auf den zugrunde liegenden klassischen Graphen. Die Berechnung ist derart konzipiert, dass er für Graphen aus unterschiedlichen 3-Motivklassen immer eine unterschiedliche Liste von Knotenindizes erzeugt. Darüber hinaus weist die Berechnung der Knotenindizes von äquivalenten Knoten, das heißt Knoten, die vertauscht werden können und durch das Vertauschen ein isomorpher Graph entsteht, den gleichen Knotenindex zu. Nicht-äquivalente Knoten erhalten allerdings unterschiedliche Knotenindizes. Dadurch entstehen für isomorphe 3-knotige Graphen die gleichen Knotenindizes, die in der Liste der Knotenindizes eventuell nur an unterschiedlichen Stellen stehen. Eine Sortierung dieser Listen von Knotenindizes liefert daher für isomorphe 3-knotige Graphen die gleichen und für nicht-isomorphe 3-knotige Graphen unterschiedliche Abfolgen von sortierten Knotenindizes.

Das Sortieren einer 3-knotigen Knotenmenge

Die Sortierung der Knotenmenge V' zu einer Liste V'_{sort} in Punkt 2) basiert primär auf der sortierten Abfolge der zugehörigen Knotenindizes. Das dient als Vorbereitung für die Berechnung in Punkt 3), die auf diese sortierten Knotenindizes zurückgreift. Darüber hinaus werden allerdings auch diejenigen Knoten anhand ihrer Knotentypwertigkeit untereinander sortiert, welche die gleichen Knotenindizes aufweisen und demnach äquivalent in Bezug auf den klassischen Graphenisomorphismus sind. An der Sortierung der Knotenindizes ändert diese sekundäre Sortierung nichts, sodass dadurch die Berechnung in Punkt 3) nicht beeinflusst wird. Diese sekundäre Sortierung hat allerdings einen für die Lösung des SysGraph-Isomorphismus-Problems entscheidenden Einfluss auf die Berechnung des Knotentypkonstellationsindexes in Punkt 4). Knoten mit einem identischen Knotenindex aber mit unterschiedlichen Knotentypen werden dadurch immer in die gleiche Reihenfolge gebracht und resultieren dadurch auch immer im gleichen Knotentypkonstellationsindex.

Der 3-Motivklassenindex eines 3-knotigen Graphen

Der in Punkt 3) berechnete 3-Motivklassenindex ist in Folge der sortierten Knotenindizes unterschiedlich für jede der 16 klassischen 3-Motivklassen und invariant für alle isomorphen Graphen einer 3-Motivklasse. Aus diesem Wert kann also eine eindeutige Zugehörigkeit eines 3-knotigen Graphen zu einer der 16 möglichen 3-Motivklassen abgeleitet werden. Damit wird das zugrunde liegende klassische Graph-Isomorphismus-Problem gelöst.

Die verwendete Formel $\kappa(V'_{sort}) = 12 \cdot v_{a'_0} + 4 \cdot v_{a'_1} + v_{a'_2}$ wurde empirisch ermittelt und auf Korrektheit bezüglich der genannten Eigenschaften untersucht.

Der Knotentypkonstellationsindex eines 3-knotigen SysGraphen

Der in Punkt 4) berechnete Knotentypkonstellationsindex hat zwei besondere Eigenschaften. Erstens, er berechnet einen Wert, der alle drei Knoten anhand ihres Knotenindexes unterscheidet und damit die Typwertigkeiten von nicht-äquivalenten Knoten unterschiedlich gewichtet. Zweitens bringt er äquivalente aber Knoten unterschiedlicher Typen, die jedoch vertauscht werden können und somit zu isomorphen SysGraphen führen, in eine festgelegte Reihenfolge und sorgt so dafür, dass deren berechneter Knotentypkonstellationsindex identisch ist. Zu beachten ist, dass die Knotenindizes nicht direkt in die Berechnung einfließen, sondern nur deren Sortierung einen wichtigen Einfluss auf die Reihenfolge der hier benötigten Knotentypwertigkeiten hat.

Die verwendete Formel $\tau(V'_{sort}) = 9 \cdot t_{a'_0} + 3 \cdot t_{a'_1} + t_{a'_2}$ ergibt sich aus den potenziell möglichen Knotentypwertigkeiten für alle drei Knoten. Dadurch ergeben alle möglichen 3er-Kombinationen dieser Wertigkeiten eindeutig unterschiedliche Knotentypkonstellationsindizes $0, \ldots, 26$.

Der SysGraph-Index eines 3-knotigen SysGraphen

Wie bereits erwähnt stellt der in Punkt 5) berechnete SysGraph-Index einen Wert dar, der für alle 65 möglichen SysGraph-Motive unterschiedlich und für alle isomorphen 3-knotigen SysGraphen innerhalb einer SysGraph-Motivklasse invariant ist. Ein identisch berechneter Wert für zwei 3-knotige SysGraphen zeigt deren Isomorphismus auf.

Die verwendete Formel $\mathcal{I}_{\mathcal{G}'_{sys}} = 27 \cdot \kappa_{\mathcal{G}'_{sys}} + \tau_{\mathcal{G}'_{sys}}$ ergibt sich aus der Tatsache, dass pro klassischer 3-Motivklasse potenziell $3^3 = 27$ unterschiedliche SysGraphen der festen Knotenzahl 3 und damit auch 27 unterschiedliche Knotentypkonstellationsindizes hervorgehen können. Die besagte Formel stellt demnach sicher, dass die SysGraph-Indizes dieser 27 SysGraphen nicht mit den Indizes von SysGraphen anderer 3-Motivklassen kollidieren.

5.6 Insiderbedrohungserkennung und -abwehr via SysGraph-Signaturen

Die Konstruktion von SysGraphen, die Extraktion von Informationsfluss-SysGraphen und die anschließende Berechnung von SysGraph-Signaturen (vgl. Abschnitte 5.4 und 5.5) kann nun genutzt werden, um bösartige Insideraktivitäten erkennen und verhindern zu können. Diese Insideraktivitäten beschränken sich nicht nur auf aktive Schritte, die ein Insiderangreifer durchführen kann, um ein Angriffsziel zu erreichen, sondern umfassen insbesondere auch Schadprogramme, die bewusst oder unbewusst von einem Insiderbedrohungsagenten ausgeführt werden. Entscheidend bei der Erkennung und Abwehr mithilfe der hier vorgestellten Technik ist ein typisches Muster, dass sich im Informationsfluss-SysGraphen niederschlägt.

Eine kurze Einführung in die Anomalieerkennung und -abwehr sowie in die signatur- und regelbasierte Überwachung zur Erkennung und Abwehr von unerwünschtem Verhalten wurde bereits in Abschnitt 4.9 gegeben. Dort findet sich auch eine Einordnung dieser Art von Sicherheitsmechanismen in die in Abschnitt 2.6 vorgestellte Insidertaxonomie, was auf die hier entwickelte Erkennungs- und Abwehrtechnik unmittelbar übertragbar ist.

5.6.1 Bedrohungsszenarien

Die folgenden verschiedenartig gelagerten Bedrohungsszenarien dienen der Veranschaulichung der in dieser Dissertation entwickelten Erkennungs- und Abwehrtechnik. Eine Einordnung und praktische Anwendung der Technik wird in Abschnitt 5.7 vorgenommen.

Szenario 5.1 (*Unerlaubtes Kopieren*)

Ein Mitarbeiter kopiert eine große Menge von Dokumenten und Dateien, auf die er legitimen Zugriff hat, auf ein externes Speichermedium und entwendet diese damit aus dem Unternehmen.

Mit dem Szenario 5.1 wird ein typischer Insiderangriff aufgeführt, der in vielen Unternehmen eine große und häufige Bedrohung darstellt. Hauer [Hau15] schätzt den Anteil von Insideraktivitäten beim Abfluss von Daten aus Unternehmen im Jahre 2014 auf 60 %, während Cheng, Liu und Yao [CLY17] anhand verschiedener Studien aus dem Jahre 2016 auf einen Anteil von mehr als 40 % schließen. Bei dem Szenario ist es irrelevant, ob der Mitarbeiter bewusst handelt und selbst aktiv mit den Ressourcen des Unternehmens interagiert oder ob sich eine externe dritte Person als der Mitarbeiter ausgibt. Entscheidend ist der Zugang zu dem System, das die Daten enthält sowie die Verwendung der zur Verfügung stehenden Zugriffsrechte.

Szenario 5.2 (*Vermeintlich harmloses Websurfen*)

Ein Mitarbeiter surft mit einem verwundbaren Browser auf harmlosen Webseiten. Eine dieser Webseiten hat eine Werbeanzeige geschaltet, die mithilfe von Schadcode die Verwundbarkeit im Browser ausnutzt und dadurch einen Kryptotrojaner auf dem Rechner des Mitarbeiters ausführen kann. Als Resultat werden alle Dateien sowohl des Rechners als auch aller verbundener Netzlaufwerke verschlüsselt und es wird eine Lösegeldsumme für die Herausgabe des Entschlüsselungsschlüssels verlangt.

Szenario 5.2 stellt einen Angriffsfall dar, der zunächst auf einer Eskalation von *Privileges* (vgl. Abschnitt 3.2.3.1) beruht und anschließend mit den erhaltenen Zugriffsrechten die entsprechenden Ressourcen verändert und damit Zugriffe verhindert (vgl. Abschnitt 3.2.4.2).

Szenario 5.3 (*Schwaches Passwort*)

Der Server eines Unternehmens ist mit einem Fernwartungszugang ausgestattet, der mit einem Benutzernamen und einem einfach zu erratenden Passwort geschützt ist. Ein unautorisierter Nutzer findet diesen Wartungszugang über das Internet und versucht eine Vielzahl an wahrscheinlichen Zugangsdaten durch, bis er schließlich eine erfolgreiche Kombination aus Benutzernamen und Passwort findet.

Ein solcher Angriff, wie er in Szenario 5.3 beschrieben ist, benötigt keinerlei *Insidergrad*, um erfolgreich zu sein. Es kann sich dabei also potenziell um einen reinen Outsiderangriff handeln. Das unmittelbare Ziel ist dabei das Erraten der *Credentials* eines Insiders (vgl. Abschnitt 3.3.2.1). Vorhandenes Insiderwissen beispielsweise über aktive Passwortrichtlinien und damit die Struktur von gültigen Passwörtern erlaubt allerdings eine Verbesserung des Angriffs und lässt das Szenario somit zu einer Insiderbedrohung werden (vgl. Abschnitt 3.2.5.2).

Die drei Szenarien werden im folgenden Abschnitt 5.6.2 zur Veranschaulichung der Ähnlichkeiten und Distanzen von SysGraph-Signaturen verwendet. Szenario 5.2 wird für die Evaluation der entwickelten Erkennungs- und Abwehrtechnik in Abschnitt 5.7 zugrunde gelegt.

5.6.2 Regelbasierte Erkennung und Kategorisierung von Insideraktivitäten

Die SysGraph-Signatur eines Informationsfluss-SysGraphen, wie sie in Abschnitt 5.5 eingeführt wurde, ist derart konzipiert, dass sich gleichgelagerte wiederkehrende Relationen zwischen den Ressourcen eines Rechners aufsummieren und unterschiedlich gelagerte Relationen möglichst weit verteilen. Dadurch werden markante Strukturen im Graphen hervorgehoben und lassen sich auf zweierlei Arten miteinander vergleichen. Erstens kann die Ähnlichkeit einer SysGraph-Signatur zu einer bereits vorberechneten Signatur einer speziellen Nutzeraktivität evaluiert werden. Zweitens lässt sich die Abweichung einer SysGraph-Signatur von einer zuvor von der gleichen Nutzeraktivität erfassten Signatur feststellen. Für einen tieferen Einblick in verschiedene Ähnlichkeits- und Distanzmaße, die dafür verwendet werden können, wird auf die Arbeit von Chandola, Banerjee und Kumar [CBK09, Kapitel 5] verwiesen.

5.6.2.1 Ähnlichkeit von SysGraph-Signaturen

Für bekannte und als bedrohlich oder bösartig eingestufte Vorgänge können SysGraph-Signaturen im Zuge einer Trainingsphase bereits vorberechnet und für spätere Referenzen gespeichert werden. Diese Referenz-Signaturen können als Vertreter von Aktivitätskategorien festgelegt

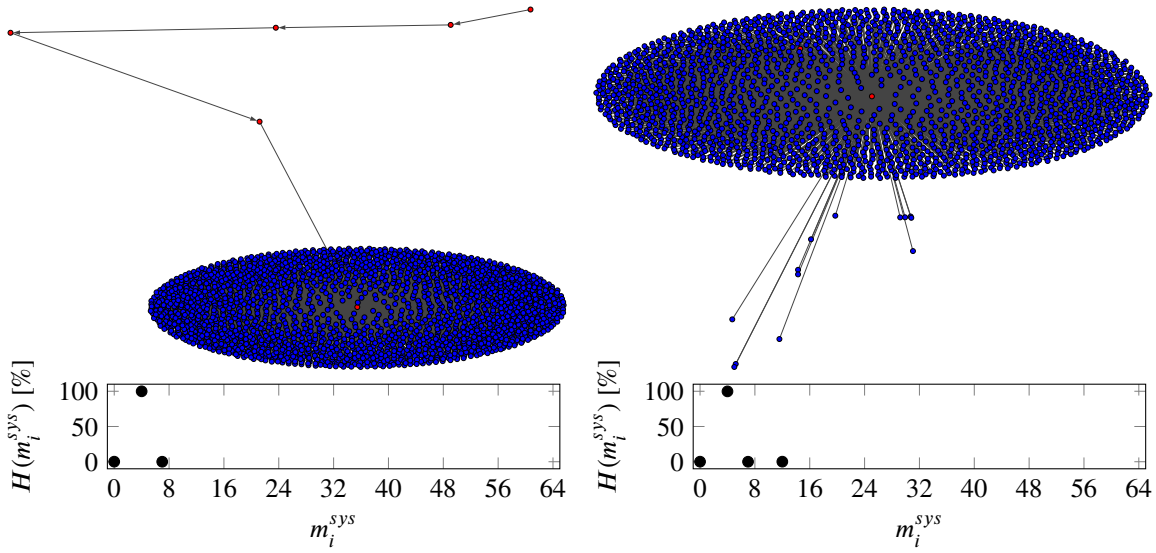
werden. Die SysGraph-Signaturen, die sich anschließend im normalen Betrieb aus den Informationsfluss-SysGraphen ergeben, können dann mit den Referenz-Signaturen auf Ähnlichkeit geprüft werden. Das Ergebnis lässt Rückschlüsse darauf zu, ob beziehungsweise mit welcher Wahrscheinlichkeit eine Aktivität zu einer bereits bekannten Aktivitätskategorie gehört. Sofern eine Ähnlichkeit zu einer bereits bekannten Aktivitätskategorie über einem definierbaren Schwellwert festgestellt wird, kann eine Sicherheitswarnmeldung erzeugt werden, die auf eine potenziell bedrohliche Insideraktivität hinweist. Als Veranschaulichung dienen an dieser Stelle die Insiderbedrohungsszenarien aus Abschnitt 5.6.1, für die jeweils zwei leicht unterschiedliche Angriffe durchgeführt, aufgezeichnet und deren Informationsfluss-SysGraphen extrahiert wurden. Für Szenario 5.1 wurden Dateien von einem Rechner auf einen USB-Stick kopiert und dabei zum einen das Unix-Kommandozeilenprogramm *cp* und zum anderen die graphische Benutzeroberfläche verwendet. Bei den Angriffen für Szenario 5.2 handelt es sich um Aktivitäten der beiden Kryptotrojaner *WannaCry* und *Locky*. Für das Szenario 5.3 wurden zwei SSH-Bruteforce-Angriffe durchgeführt, von denen einer die korrekten Zugangsdaten findet und der zweite nicht. Die Informationsfluss-SysGraphen sowie deren SysGraph-Signaturen sind in Abbildung 5.10 dargestellt. Darin zu erkennen sind die Ähnlichkeiten, die sowohl die SysGraphen als auch die Signaturen innerhalb eines Szenarios zueinander haben. Mit einer der beiden Signaturen als Referenz-Signatur kann demnach die andere Aktivität anhand der Ähnlichkeit erkannt und automatisiert zugeordnet werden.

Als geeignetes Ähnlichkeitsmaß wird hier die sogenannte Kosinusähnlichkeit verwendet, die häufig im Bereich der Textanalyse zur Bewertung von Dokumentenähnlichkeiten zum Einsatz kommt [HPK11, Abschnitt 2.4.7]. Für die Berechnung der Kosinusähnlichkeit werden die SysGraph-Signaturen, die aus den Häufigkeiten der 65 möglichen SysGraph-Motive $m_0^{sys}, \ldots, m_{64}^{sys}$ bestehen (vgl. Abschnitt 5.5), als 65-dimensionale Vektoren $\vec{u}, \vec{v}$ aufgefasst und der Kosinus des Winkels zwischen diesen beiden Vektoren anhand der Formel für das Standardskalarprodukt bestimmt:

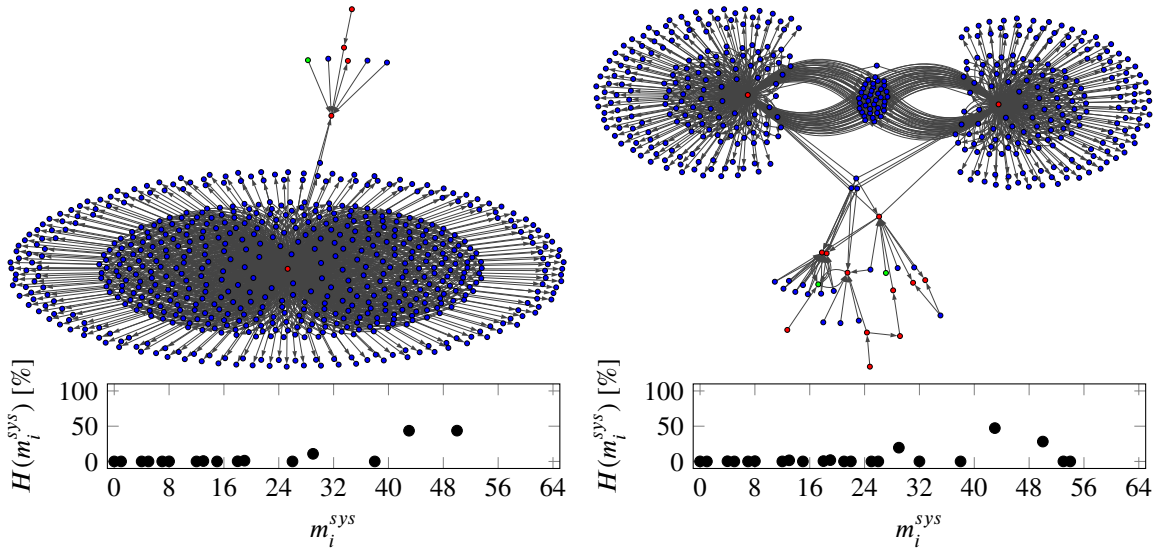
$$\mathtt{sim}(\vec{u}, \vec{v}) = \cos(\vec{u}, \vec{v}) = \frac{\vec{u} \cdot \vec{v}}{\|\vec{u}\| \cdot \|\vec{v}\|} .$$

Dabei stellt $\|\vec{u}\|$ die Euklidische Norm beziehungsweise die Länge des Vektors $\vec{u} = (u_0, \ldots, u_{64})$ dar, die durch $\sqrt{u_0^2 + \cdots + u_{64}^2}$ definiert ist.

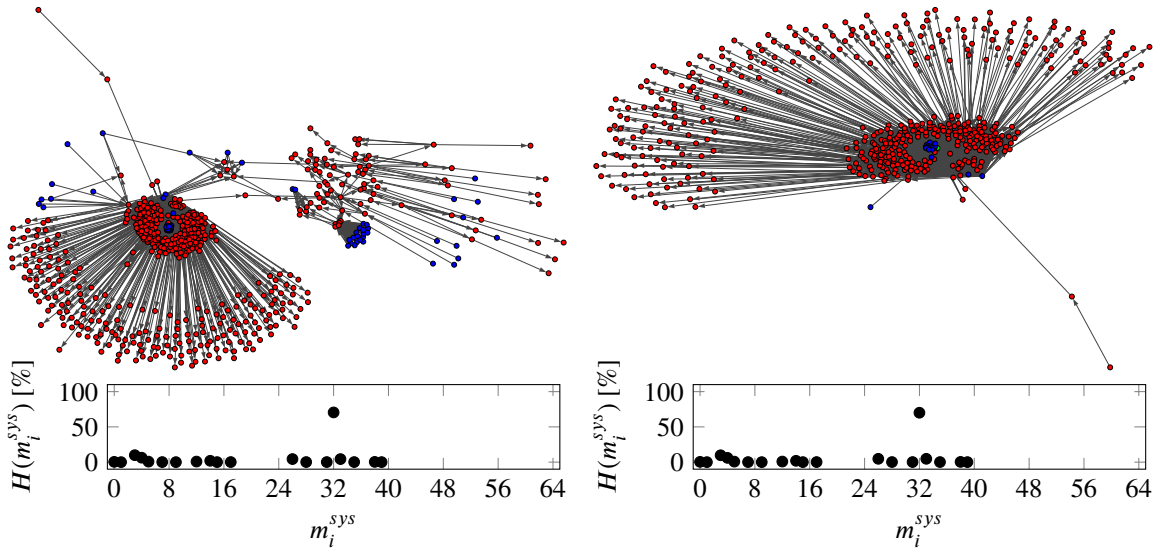
Dieses Ähnlichkeitsmaß erweist sich im vorliegenden Kontext als sehr robust gegenüber kleinen Variationen und hat den weiteren Vorteil, dass es unabhängig von eventuell vorhandenen Größenunterschieden der Informationsfluss-SysGraphen beziehungsweise der Länge der zugehörigen SysGraph-Signaturvektoren ist [XW05]. Da die einzelnen Stellen der Vektoren nur positive Zahlen enthalten können, ergibt sich für die Kosinusähnlichkeit ein Wert im Bereich von $0 \leq \mathtt{sim}(\vec{u}, \vec{v}) \leq 1$, wobei 0 zwei orthogonale Vektoren und damit zwei überschneidungsfreie SysGraph-Signaturen und 1 zwei gleichgerichtete Vektoren und damit zwei vollständig überschneidende Signaturen darstellt. Die berechneten Kosinusähnlichkeiten zwischen den jeweiligen beiden aufgezeichneten Angriffen pro Insiderbedrohungsszenario aus Abschnitt 5.6.1 sind ebenfalls in Abbildung 5.10 aufgeführt. Sowohl die beiden durchgeführten Kopieraktivitäten in Abbildung 5.10a als auch die beiden SSH-Bruteforce-Angriffe in Abbildung 5.10c weisen trotz Abweichungen in den SysGraph-Signaturen eine Kosinusähnlichkeit in Höhe von 100% auf. Die beiden Aktivitäten der Kryptotrojaner in Abbildung 5.10b erreichen dank der gewählten Kosinusähnlichkeit einen immer noch sehr hohen Wert von 95,7%, obwohl die graphischen Darstellungen der Aktivitäten einen größeren Unterschied vermuten lassen. Die Ähnlichkeiten zwischen Angriffsaktivitäten aus unterschiedlichen Szenarien liegen bei maximal 8,77%. Diese



(a) Szenario 5.1: Unerlaubtes Kopieren, links via Terminal und rechts via Dateiexplorer. Ähnlichkeit: 100%.



(b) Szenario 5.2: Kryptotrojaner, links WannaCry und rechts Locky. Ähnlichkeit: 95,7%.



(c) Szenario 5.3: Schwaches Passwort, links erfolgreich und rechts erfolglos. Ähnlichkeit: 100%.

Abbildung 5.10: Informationsfluss-SysGraphen für Beispielangriffe entsprechend der Szenarien aus Abschnitt 5.6.1 mit den jeweiligen SysGraph-Signaturen.

Ähnlichkeit wurde zwischen dem Kopieren mittels *cp* und dem erfolglosen SSH-Bruteforce-Angriff gemessen. Ein Vergleich aller anderen Signaturen aus unterschiedlichen Szenarien ergibt eine geringere Ähnlichkeit. Das zeigt auf, dass die gewählten Bedrohungsszenarien sehr gut anhand ihrer SysGraphen sowie anhand ihrer SysGraph-Signaturen auch automatisiert voneinander abgrenzbar sind.

5.6.2.2 Abweichung von SysGraph-Signaturen

In der zweiten Variante, wie SysGraph-Signaturen bei der Erkennung von Insiderbedrohungen zum Einsatz kommen können, werden bereits bekannte und als unbedrohlich eingestufte beziehungsweise akzeptierte Vorgänge zugrunde gelegt. Für diese Vorgänge können in einer Trainingsphase SysGraph-Signaturen vorberechnet und für spätere Referenzen gespeichert werden. Sobald eine im anschließenden normalen Betrieb erfasste SysGraph-Signatur für eine derartige Aktivität von seiner Referenz-Signatur abweicht beziehungsweise die Abweichung von der zugehörigen Referenz-Signatur über einen definierbaren Schwellwert fällt, kann eine Sicherheitswarnmeldung erzeugt werden, die auf eine potenziell bösartige Insideraktivität hinweist. Dazu gehört auch eine vom Nutzer möglicherweise unbemerkte Veränderung der Rechnerressourcen, die zum Beispiel bei Prozessen auf ausgenutzte Sicherheitslücken und Schadsoftware hinweisen kann. Es handelt sich hierbei also um eine rudimentäre Form der Anomalieerkennung [CBK09]. Für die Bedrohungsszenarien aus Abschnitt 5.6.1 wären unbedrohliche Vorgänge beispielsweise eine Rechnersitzung ohne Kopieren in Szenario 5.1, das Öffnen eines Browsers und das harmlose Websurfen in Szenario 5.2 sowie die autorisierte Fernwartung des Unternehmensservers in Szenario 5.3. Für diese unbedrohlichen Vorgänge wurden beispielhaft die Informationsfluss-SysGraphen sowie deren SysGraph-Signaturen erzeugt und in Abbildung 5.11 dargestellt. Eine solche Signatur kann nun verwendet werden, um die Signaturen von Insideraktivitäten, bei denen mithilfe von Kontextwissen die Art der Aktivität bekannt ist oder erschlossen werden kann,[10] auf ihre Bedrohlichkeit hin zu untersuchen. Wenn zum Beispiel die SysGraph-Signatur einer Browsersitzung eines Insiders von der hier erzeugten Signatur einer harmlosen Browsersitzung in Abbildung 5.11b zu sehr abweicht, kann eine Insiderbedrohung als Grund dafür angenommen und genauere Untersuchungen anberaumt werden.

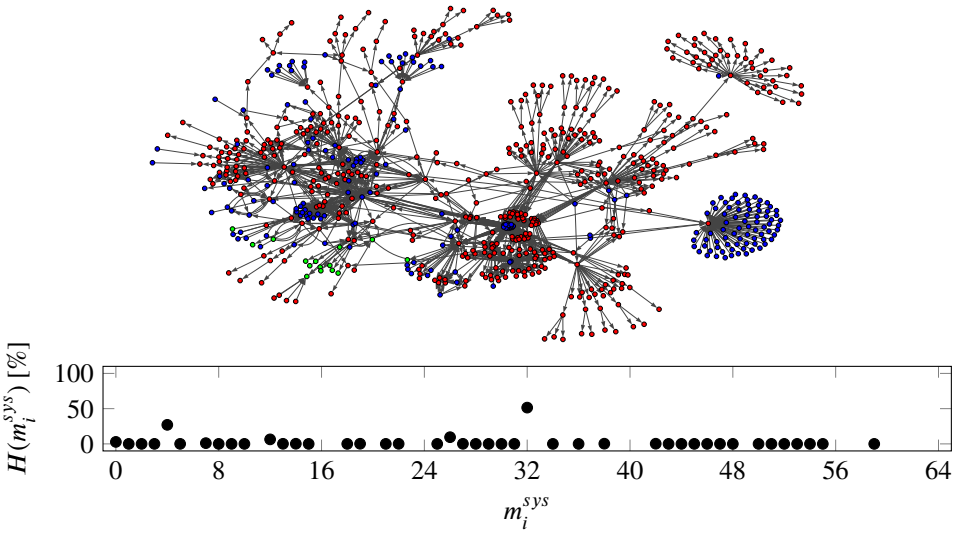
Als geeignetes Maß für diese Abweichung kommt sowohl die Kosinusdistanz, also die Kosinusähnlichkeit, wie sie in Abschnitt 5.6.2.1 eingesetzt wird, von 1 abgezogen, als auch die Euklidische Distanz in Frage. Letztere berechnet sich aus

$$\texttt{dist}(\vec{u}, \vec{v}) = \|\vec{u} - \vec{v}\| = \sqrt{(\vec{u} - \vec{v}) \cdot (\vec{u} - \vec{v})}$$

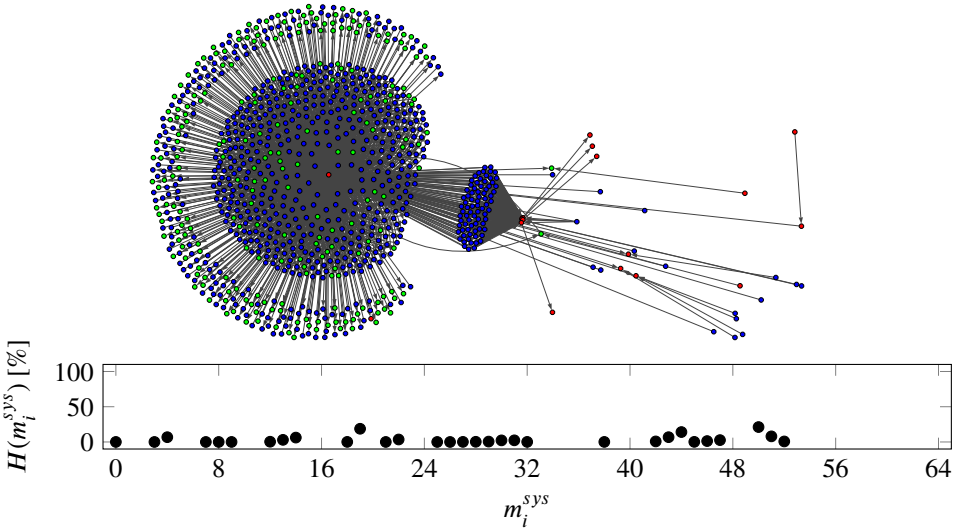
und beschreibt den geradlinigen Abstand zwischen zwei Punkten [HPK11, Abschnitt 2.4.4]. Sofern also auch hier zwei SysGraph-Signaturen als 65-dimensionale Vektoren aufgefasst werden, gibt deren Distanz Aufschluss über das Ausmaß der Abweichung.

Sofern eine Referenz-Signatur einen Informationsfluss-SysGraphen beschreibt, der bei mehrmaligem Auftreten relativ stabil bleibt, ist die Euklidische Distanz vorzuziehen. Abweichungen können dadurch auch dann festgestellt werden, wenn die zugrunde liegenden Relationen zwischen den Ressourcen eines Rechners ausschließlich in ihrer Häufigkeit Unterschiede aufweisen. Die

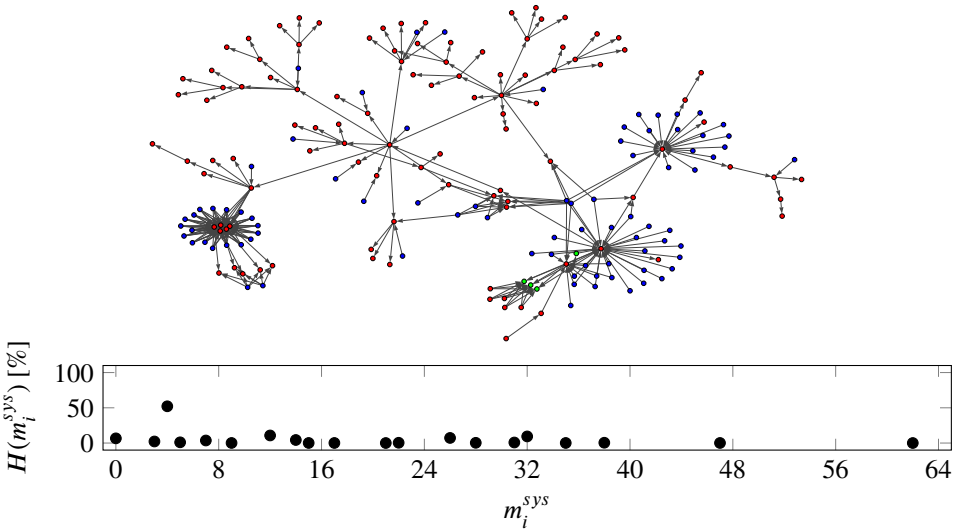
10. Die Ereignisnachrichten des Linux Audit System, die für die Konstruktion der SysGraphen verwendet werden, liefern Informationen, mit denen derartiges Kontextwissen abgeleitet werden kann. Das Starten eines Browsers oder einer Fernwartungsverbindung etwa lassen sich damit direkt erkennen.



(a) Szenario 5.1: Arbeitssitzung ohne Kopieren. Kosinusdistanz zu den Signaturen in Abbildung 5.10a: 54,43%.



(b) Szenario 5.2: Harmloses Websurfen. Kosinusdistanz zu den Signaturen in Abbildung 5.10b: 43,26%.



(c) Szenario 5.3: Legitimer SSH-Zugang. Kosinusdistanz zu den Signaturen in Abbildung 5.10c: 73,19%.

Abbildung 5.11: Informationsfluss-SysGraphen von unbedrohlichem Verhalten in den Szenarien aus Abschnitt 5.6.1 mit den jeweilige SysGraph-Signaturen.

Vektoren der Signaturen haben dann eine unterschiedliche Länge, die sich in der Euklidischen Distanz bemerkbar macht. Sofern allerdings Informationsfluss-SysGraphen auftreten, die bei ausschließlich unterschiedlichen Intensitäten der Relationen zwischen den Ressourcen keine Abweichungen voneinander aufzeigen sollen, ist die Kosinusdistanz geeignet. Die Vektoren der Signaturen haben dann zwar eventuell eine unterschiedliche Länge, aber eine sehr geringe oder keine Winkeldifferenz. Die berechneten Kosinusdistanzen zwischen den harmlosen Aktivitäten und den Angriffsaktivitäten entsprechend der Bedrohungsszenarien aus Abschnitt 5.6.1 sind zur Veranschaulichung in Abbildung 5.11 aufgeführt. Die unerlaubten Kopieraktivitäten in Abbildung 5.10a weisen zur Arbeitssitzung ohne Kopieren in Abbildung 5.11a eine Kosinusdistanz in Höhe von 54,43% auf. Die Kryptotrojaner in Abbildung 5.10b weichen mit einer Kosinusdistanz in Höhe von 43,26% vom harmlosen Websurfen in Abbildung 5.11b ab. Und die SSH-Bruteforce-Angriffe in Abbildung 5.10c ergeben eine Kosinusdistanz in Höhe von 73,19% zum legitimen Fernwartungszugang in Abbildung 5.11c.

5.6.3 Echtzeit SysGraph-Signaturen zur Bedrohungsabwehr

Die bisherige Erzeugung und Verwendung von SysGraph-Signaturen (vgl. Abschnitte 5.5 und 5.6.2) erlauben Momentaufnahmen von bereits vergangenen Systemaktivitäten an einem Rechner und damit die Erkennung von bestimmten Insiderbedrohungen, nachdem sie bereits stattgefunden haben. Für einen Schutz vor Insiderbedrohungen sind sie allerdings aus zwei Gründen bisher noch nicht geeignet, wie nachfolgend erläutert wird.

5.6.3.1 Insiderbedrohungsprävention durch Echtzeit-Analysen

Durch die Erkennung von frühen Anzeichen einer Insiderbedrohung und der Abwehr von weiteren Bedrohungsaktivitäten dieses Insiders können Insiderbedrohungen effektiv verhindert werden. Aus diesem Grund wird fortlaufend ein SysGraph mit allen Aktivitäten eines Nutzers gepflegt, aus dem in Echtzeit alle vorhandenen Informationsfluss-SysGraphen extrahiert sowie deren SysGraph-Signaturen berechnet und auf Ähnlichkeit beziehungsweise Distanz zu Referenz-Signaturen geprüft werden.

Im Detail ergibt sich aus jedem neu erzeugten Prozess-Knoten im SysGraphen, der einen vom Nutzer angestoßenen Prozess darstellt, ein neuer Informationsfluss-SysGraph, der aus dem aktuell vorliegenden SysGraphen extrahiert und in einer Liste von bereits vorhandenen Informationsfluss-SysGraphen gespeichert wird. Jede Änderung im ursprünglichen SysGraphen, ausgelöst durch laufende Aktivitäten des Nutzers, beispielsweise das Öffnen und Lesen einer Datei durch einen laufenden Prozess, zieht ebenfalls eine Änderung in allen davon betroffenen Informationsfluss-SysGraphen nach sich. Betroffen ist ein Informationsfluss-SysGraph aus der gespeicherten Liste genau dann, wenn der Knoten, der im ursprünglichen SysGraphen verändert wird, auch Teil des Informationsfluss-SysGraphen ist.

Für jeden Informationsfluss-SysGraphen dieser Liste wird mit jeder Änderung eine neue SysGraph-Signatur berechnet und mit den vorhandenen Referenz-Signaturen auf Ähnlichkeit beziehungsweise Distanz verglichen, wie es in Abschnitt 5.6.2 beschrieben ist. Sofern die Ähnlichkeit zu vorberechneten und als bedrohlich eingestuften Vorgängen beziehungsweise die Distanz zu zugehörigen unbedrohlich eingestuften Vorgängen einen definierbaren Schwellwert übersteigt und im Laufe der Zeit eventuell noch ansteigt, kann die entsprechende Nutzeraktivität für genauere Untersuchungen markiert und gegebenenfalls gestoppt werden.

5.6.3.2 Optimierung der SysGraph-Signaturberechnung

Alle in einem SysGraphen vorhandenen Informationsfluss-SysGraphen unterliegen permanenten Veränderungen, solange eine zugehörige Nutzeraktivität noch nicht abgeschlossen ist. Jede Veränderung müsste eine erneute Signaturberechnung nach sich ziehen, die, wie in Abschnitt 5.5 beschrieben, sehr aufwendig ist. Aus diesem Grund wurde eine effizientere Berechnung der SysGraph-Signaturen entwickelt, die speziell für die Echtzeit-Analyse von Informationsfluss-SysGraphen aus Abschnitt 5.6.3.1 optimiert ist. Die Berechnung nutzt die Tatsache aus, dass nur eine elementare Kantenänderung im Informationsfluss-SysGraphen, das heißt das Hinzufügen einer einzelnen Kante, Änderungen in der SysGraph-Signatur hervorruft und diese Änderungen zudem anhand der bereits berechneten Häufigkeiten der vorliegenden SysGraph-Motive abgeleitet werden können.[11]

Damit eine neue Kante $e = (a, b)$ mit $a, b \in \tilde{V}$, die einem Informationsfluss-SysGraphen $\widetilde{\mathcal{G}}_{sys} = (\tilde{V}, \tilde{E})$ hinzugefügt werden soll, eine Änderung in dessen SysGraph-Signatur hervorruft, muss eine weitere Kante zu einem Nachbarknoten c von a oder von b bereits existieren. Nur dann entsteht mit der neuen Kante e ein weiteres SysGraph-Motiv m_i^{sys}, dass die Anzahl dieser Motive in der SysGraph-Signatur um eins erhöht. Welches SysGraph-Motiv dabei genau entsteht, hängt davon ab,

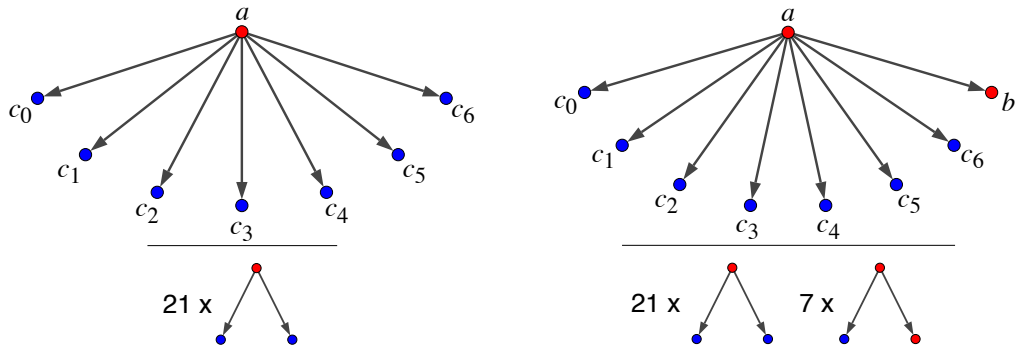
- welchen Knotentyp a, b und c haben,
- ob eine umgekehrte Kante $e^{-1} = (b, a)$ bereits existiert und
- wie genau c mit a oder mit b oder mit beiden verbunden ist.

In den Fällen, in denen c entweder nur mit a oder nur mit b aber nicht mit beiden verbunden ist und auch keine umgekehrte Kante $e^{-1} = (b, a)$ existiert, entsteht mit e ein neues SysGraph-Motiv, wo vorher noch keines war. In allen anderen Fällen haben die drei Knoten a, b und c bereits ein SysGraph-Motiv m_j^{sys} erzeugt, welches nun von der SysGraph-Signatur abgezogen werden muss, bevor mit der neuen Kante das neue SysGraph-Motiv m_i^{sys} der Signatur hinzugefügt wird.

Diese Bestimmung des vorher-nachher SysGraph-Motivs und die entsprechende Änderung der SysGraph-Signatur muss natürlich für jeden Nachbarknoten $c \in Neig(a) \cup Neig(b)$ von a und von b vorgenommen werden. Eine Vereinfachung ergibt sich allerdings aus der Tatsache, dass n gleichartige Nachbarknoten $c_0, \ldots, c_{n-1}$, also vom gleichen Knotentyp und in der gleichen Verbindungskonstellation mit a und mit b, zum gleichen neuen SysGraph-Motiv m_i^{sys} führen und somit die Anzahl von m_i^{sys} in der SysGraph-Signatur um n erhöhen. Äquivalent dazu müssen natürlich auch n SysGraph-Motive m_j^{sys} abgezogen werden, sofern a, b und $c_0, \ldots, c_{n-1}$ bereits vor dem Hinzufügen von e das Motiv m_j^{sys} erzeugten. Ein Beispiel in Abbildung 5.12 veranschaulicht diese Beobachtung.

Die Anzahlen solcher gleichartigen Knoten lassen sich anhand der Schnitt- und Differenzmengen aller Vorgänger und Nachfolger von a sowie aller Vorgänger und Nachfolger von b und dem Filtern nach allen drei möglichen Knotentypen herausfinden. Dabei entstehen pro möglichem Knotentyp für $c_0, \ldots, c_{n-1}$ insgesamt 30 Fallunterscheidungen, wobei sechs davon gänzlich neue SysGraph-Motive erzeugen und 24 davon bereits bestehende SysGraph-Motive in andere Motive ändern.

11. Auch das Wegnehmen einer einzelnen Kante ruft eine Änderung in der SysGraph-Signatur hervor. Diese Operation ist allerdings bei der Extraktion von Informationsfluss-SysGraphen nicht vorgesehen (vgl. Abschnitt 5.4.1).



(a) vorher **(b)** nachher

Abbildung 5.12: Änderungen der SysGraph-Signatur bei neu hinzukommenden Kanten an einem Beispiel-Informationsfluss-SysGraphen.

5.7 Evaluation

In diesem Abschnitt wird die entwickelte und im Detail erläuterte Abwehrtechnik von Insiderbedrohungen anhand eines ausgewählten realen Szenarios und mit realen Daten evaluiert.

5.7.1 Evaluationsszenario und Erzeugung der Datensätze

Für die Evaluation wurde das Bedrohungsszenario 5.2 aus Abschnitt 5.6.1 zugrunde gelegt. Darin ist eine Insiderbedrohung beschrieben, bei der durch vermeintlich harmloses Websurfen eines Mitarbeiters unbemerkt Schadsoftware installiert wird und dadurch alle Dateien des Rechners sowie der angeschlossenen Netzlaufwerke verschlüsselt werden. Es handelt sich dabei also um den Fall einer sogenannten Ransomware,speziell um einen Kryptotrojaner, der durch die Verschlüsselung den Zugriff auf Ressourcen eines Rechners verhindert und eine Lösegeldsumme anfordert, um diesen Zugriff wieder freikaufen zu können. Dieses Szenarios wurden aus folgenden zwei Gründen für die Evaluation ausgewählt. Zum einen handelt es sich um eine aktuell sehr weitverbreitete Bedrohung, die anhand der Ausführungen in Kapitel 3 als Insiderbedrohung einzustufen ist. Im Zeitraum zwischen April 2017 und März 2018 lag laut Kaspersky Lab [Kas18] die Anzahl der mit Verschlüsselungstrojanern konfrontierten Nutzer von Kaspersky Lab Produkten weltweit bei 752 000. Zum anderen existiert eine Vielzahl an unterschiedlichen Kryptotrojanern, die sich bei aller Unterschiedlichkeit anhand ihrer SysGraph-Signaturen nicht signifikant unterscheiden sollten. Denn das Ziel der entwickelten Signaturen ist, dass unterschiedlichste Aktivitäten an einem Rechner, die das gleiche Ziel verfolgen, am Ende auf eine gleiche beziehungsweise sehr ähnliche Signatur hinauslaufen.

Für den Trainingsdatensatz und somit für die Erzeugung und Speicherung von bekannten Signaturen in der Trainingsphase wurden aus fünf verschiedenen öffentlich zugänglichen Schadsoftware-Archiven manuell Proben zusammengetragen, die bekannte Windows-Kryptotrojaner enthalten.[12] Aus allen 19 gefundenen Kryptotrojaner-Proben wurden für den Trainingsdatensatz

12. Bei den Schadsoftware-Archiven handelt es sich um *theZoo* (https://github.com/ytisf/theZoo), *Das Malwerk* (https://dasmalwerk.eu/), *MalwareDatabase* (https://github.com/NTFS123/MalwareDatabase), *malware-samples* (https://github.com/fabrimagic72/malware-samples/) sowie *tutorialjinni* (https://www.tutorialjinni.com/download-free-malware-samples.htm).

nur diejenigen ausgewählt, die unter starker Isolation, also insbesondere ohne Verbindung zu einem gegebenenfalls benötigten Command & Control Server, ihre Arbeit verrichteten und die Dateien und Ordner auf angeschlossenen Netzlaufwerken verschlüsselten. Letzteres Kriterium ist für die Evaluationsumgebung von zentraler Bedeutung und wird im folgenden Abschnitt genauer erläutert. Übrig geblieben sind dabei neun Schadsoftware-Proben von diesen Kryptotrojanern:

- Cerber (1x),
- Jigsaw (1x),
- Locky (1x),
- Mischa (1x),
- Vipasana (3x) sowie
- WannaCry (2x).

Alle Proben wurden nach einer manuellen Ausführung und anhand öffentlich verfügbarer Informationen auf korrekte Bestimmung der Kryptotrojaner-Familie geprüft und bestätigt.

Grundlage für die Erzeugung des Testdatensatzes war eine Menge von 106 349 Schadsoftware-Proben aus den Jahren 2017 bis 2019, die von VirusTotal [Vir20] im Rahmen ihres Programms *Academic Malware Samples* für diese Evaluation zur Verfügung gestellt wurden. Aus dieser Menge wurden ebenfalls diejenigen Proben ausgewählt, die offline und unter starker Isolation arbeiteten und Ressourcen auf Netzlaufwerken verschlüsselten. Dabei sind 6703 geeignete Kryptotrojaner gefunden und nach der Eliminierung von Duplikaten 6199 Proben dem Testdatensatz hinzugefügt worden.

Für die Prüfung auf Eignung einer Schadsoftware-Probe bei der Erzeugung sowohl des Trainings- als auch des Testdatensatzes wurde ein eigenes Verfahren entwickelt und automatisiert angewandt. Jede Schadsoftware-Probe wurde auf einer virtuelle Maschine (VM) mit installiertem Windows-Betriebssystem ausgeführt, welche in einem virtuellen Rechnernetz isoliert nur Zugriff auf ein Netzlaufwerk hatte. Nach Ausführung der Schadsoftware-Probe wurden einerseits die Anzahl der Zugriffe auf das Netzlaufwerk mit der Anzahl der auf dem Netzlaufwerk verfügbaren Dateien abgeglichen und andererseits wurden die Dateien auf dem Netzlaufwerk auf Veränderungen getestet und bei positivem Test dem jeweiligen Datensatz hinzugefügt.

5.7.2 Implementierung und Evaluationsumgebung

Die Implementierung der in diesem Kapitel im Detail vorgestellten Abwehrtechnik zum Schutz vor Insiderbedrohungen umfasst das Parsen und Vorverarbeiten von Linux-Audit-Ereignisnachrichten durch das in Abschnitt 5.3.4 beschriebene Plugin, die Erzeugung von SysGraphen anhand der Linux-Audit-Ereignisnachrichten, wie es in Abschnitt 5.4.1 beschrieben wurde, die Extraktion von Informationsfluss-SysGraphen (vgl. Abschnitt 5.4.2) sowie die Berechnung von Echtzeit-SysGraph-Signaturen dieser Informationsfluss-SysGraphen zur regelbasierten Erkennung und Abwehr von Insiderbedrohungen (vgl. Abschnitte 5.6.2 und 5.6.3). All diese Teilaspekte wurden in Python3 mit 2734 Zeilen Code in neun Objektklassen softwaretechnisch umgesetzt.

Da alle gesammelten Kryptotrojaner-Proben in den Datensätzen ausschließlich unter einem Windows-Betriebssystem lauffähig sind, die in diesem Kapitel entwickelte Erkennungs- und Abwehrtechnik aber auf die Ereignisnachrichten des *Linux Audit Systems* angewiesen ist, wurde für die Evaluation die gängige Praxis ausgenutzt, dass viele der bekannten Windows-Kryptotrojaner versuchen, auch die Ressourcen der erreichbaren Netzlaufwerke zu verschlüsseln. Ein solches Netzlaufwerk wurde daher für die Evaluation von einem Rechner mit Linux-Betriebssystem

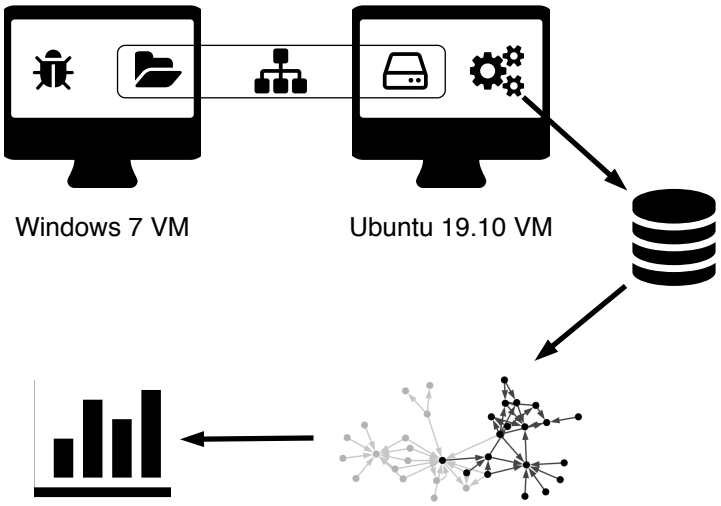


Abbildung 5.13: Vorgehensweise bei der Evaluation der Erkennungs- und Abwehrtechnik von Insiderbedrohungen.

bereitgestellt, wodurch die Verschlüsselungsaktivitäten der Windows-Kryptotrojaner auf den Netzlaufwerkressourcen vom Linux-Audit-Daemon *auditd* erfasst werden konnten.

Für die Ausführung der Kryptotrojaner wurde eine VM mit Windows 7 32Bit Betriebssystem, mit installiertem Service Pack 1 und allen wichtigen Windows-Updates bis zum 01.08.2016, mit deaktivierter Nutzeraccount-Kontrolle (engl. user account control, UAC) sowie mit konfiguriertem Samba-Netzlaufwerk vorbereitet. Als Untersuchungsgerät, dem sogenannten *Target of Evaluation (TOE)*, kommt eine Linux-VM mit installiertem Ubuntu 19.10 Betriebssystem und konfiguriertem Samba in der Version 4.10.7 zum Einsatz. Die Samba-Freigabe wurde mit 2000 JPG-Bilddateien bestückt. Das *Linux Audit System* wurde auf dem TOE derart konfiguriert, dass die Audit-Ereignisnachrichten durch das implementierte *audisp* Plugin verarbeitet, also konsolidiert und in ein Format überführt werden, welches die für die SysGraph-Erzeugung notwendigen Informationen verarbeiten lässt. Durch das *audisp* Plugin werden die Ereignisnachrichten letztendlich in einem mit dem Host-Rechner geteilten Ordner abgelegt.

Der Host-Rechner wurde in der Rolle der Datenverarbeitungseinheit mit einer Python-Laufzeitumgebung in der Version 3.5.3 ausgestattet, um aus den gespeicherten Linux-Audit-Ereignisnachrichten mittels der implementierten Erkennungs- und Abwehrtechnik SysGraph-Signaturen zu erzeugen und mit bereits bekannten Signaturen zu vergleichen. Als Virtualisierungssoftware kommt *VMware Workstation 15.0.0* zum Einsatz. Für eine Parallelisierung der Evaluation wurde das beschriebene Setup auf elf baugleichen Host-Rechnern eingerichtet.

5.7.3 Vorgehensweise

Beim prinzipiellen Vorgehen wurde nicht zwischen dem Trainings- und dem Testdatensatz unterschieden. Eine Illustration findet sich in Abbildung 5.13. Nach dem Starten beider VMs und dem Abwarten des jeweiligen Boot-Vorgangs wurde eine Windows-Kryptotrojaner-Probe zusammen mit einem Kommandozeilenskript auf die Windows-VM geladen. Das Skript stellte die Verbindung zum Netzlaufwerk des TOE sicher, entpackte den Kryptotrojaner und führte die Schadsoftware aus. Auf dem Host-Rechner, der direkten Zugriff auf die Audit-Ereignisnachrichten des TOEs hatte, überwachte ein Skript 170 Sekunden nach dem Starten der Schadsoftware

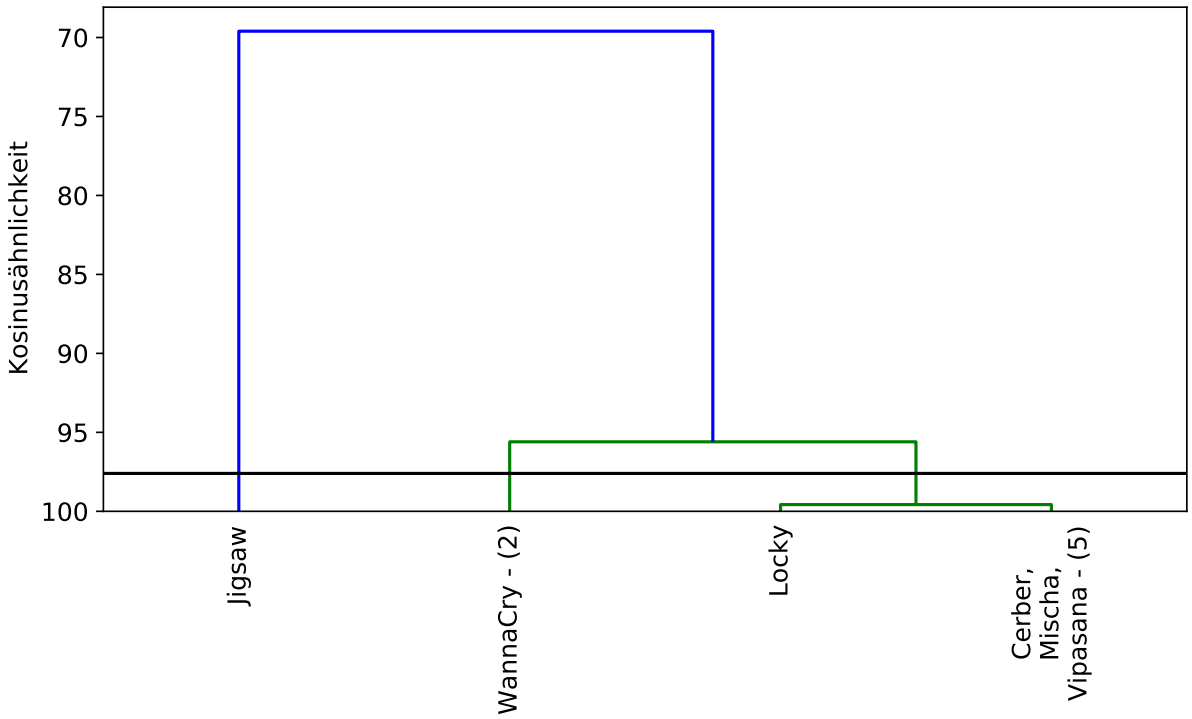


Abbildung 5.14: Cluster-Dendrogram der Kryptotrojaner-SysGraph-Signaturen aus dem Trainingsdatensatz. Die Zahlen in Klammern geben die Größe der Cluster an, deren Zusammenlegung aufgrund der hohen Kosinusähnlichkeit im Dendrogram nicht erkennbar ist.

periodisch die Audit-Ereignisnachrichten und terminierte die beiden VMs, sobald keine Dateizugriffe auf die 2000 Bilddateien des TOEs mehr verzeichnet wurden. Kurz vor der Terminierung wurde in der Windows-VM noch ein Screenshot angefertigt, um bei der späteren Auswertung der Ergebnisse gegebenenfalls eine manuelle Überprüfung der vorliegenden Kryptotrojaner-Familie vornehmen zu können. Anschließend wurde die Kryptotrojaner-Probe zusammen mit den zugehörigen Audit-Ereignisnachrichten und dem Screenshot auf dem Host-Rechner archiviert.

Nach dem Durchlaufen aller Windows-Kryptotrojaner-Proben wurden die archivierten Audit-Ereignisnachrichten einzeln und nacheinander in SysGraphen überführt. Weiterhin wurde aus diesen SysGraphen der Informationsfluss-SysGraph desjenigen Prozesse extrahiert, der für die Dateizugriffe auf die 2000 Bilddateien verantwortlich war. Aus diesen Informationsfluss-SysGraphen wurden wiederum die jeweils zugehörigen SysGraph-Signaturen erzeugt. In einem letzten Schritt wurden in der Trainingsphase alle SysGraph-Signaturen anhand ihrer Ähnlichkeiten geclustert und in der Testphase die Signaturen anhand der gefundenen Cluster klassifiziert.

Weitere Verarbeitung in der Trainingsphase

Als Clusteralgorithmus wurde das hierarchische *Agglomerative Clustering* mit der *Single Linkage* Strategie implementiert [Eve+11, Abschnitt 4.2; MC17]. Aus den SysGraph-Signaturen des Trainingsdatensatzes ergaben sich dabei die folgenden drei Cluster, die in Abbildung 5.14 illustriert sind:

- Jigsaw,

- 2x WannaCry sowie
- Locky, Cerber, Mischa und 3x Vipasana.

Für jedes dieser Cluster wurde eine Mittelwert-SysGraph-Signatur berechnet und als Referenzsignatur gespeichert. Die Kryptotrojaner-Proben innerhalb eines Clusters weisen eine gegenseitige Ähnlichkeit von mindestens 99,57% und die Proben verschiedener Cluster eine Ähnlichkeit von höchstens 95,6% auf. Damit wurde der Ähnlichkeitsschwellwert, an dem die SysGraph-Signaturen aus dem Testdatensatz später klassifiziert werden, auf 97,6% festgelegt (vgl. Abbildung 5.14).

Weitere Verarbeitung in der Testphase

Während der Extraktion der Informationsfluss-SysGraphen aller Windows-Kryptotrojaner wurden in festen Intervallen fortlaufende Statistiken über die Ähnlichkeiten zu den Referenz-SysGraph-Signaturen sowie über die Anzahl der bis dato potenziell verschlüsselten Dateien gesammelt. Für diese Anzahl wurden im Informationsfluss-SysGraphen alle Kanten zwischen einem Prozessknoten und einem Dateisystemobjektknoten gezählt. Als Klassifizierung der SysGraph-Signaturen, die aus dem Testdatensatz hervorgingen, wurde ein *Nearest Neighbour Klassifizierer* implementiert, der eine Zuordnung zu einer der drei Cluster aus der Trainingsphase erlaubte. Dabei erfolgte die Zuordnung zu einem der drei Cluster, sobald die Ähnlichkeit den Schwellwert von 97,6% überstieg. Sofern dieser Ähnlichkeitsschwellwert nicht erreicht wurde, wurden die Kryptotrojaner-Proben als *nicht-klassifiziert* gekennzeichnet.

5.7.4 Auswertungen

Bei den Auswertungen der Ergebnisse wurde einerseits der Frage nachgegangen, wie viele Kryptotrojaner-Proben den nur drei Referenz-Signaturen aus der Testphase zugeordnet werden können, also wie groß die Abdeckung durch diese Referenz-Signaturen ist. Andererseits sollte ermittelt werden, wie schnell eine solche Zuordnung zu einer Referenz-Signatur erfolgt und somit wie viel Schaden bis dahin von dieser speziellen Art der Insiderbedrohung bereits verrichtet wird.

Klassifizierung der Testdaten

Von den 6199 Kryptotrojaner-Proben aus dem Testdatensatz konnten 6197 Proben eindeutig einem der drei Cluster aus dem Trainingsdatensatz zugeordnet werden. 3490 Proben wurden als *Cerber/Locky/Mischa/Vipasana*, 2704 Proben als *Jigsaw* und 3 Proben als *WannaCry* klassifiziert. Die minimale Ähnlichkeit der erfolgreich klassifizierten SysGraph-Signaturen zu den Referenz-SysGraph-Signaturen betrug 98,6%, die maximale Ähnlichkeit betrug 100% und die durchschnittliche Ähnlichkeit betrug 99,9817%. Bei der Interpretation der Klassifizierung ist zu beachten, dass es sich nicht zwangsläufig um einen der konkret genannten Kryptotrojaner handeln muss, auch wenn eine Probe einem bestimmten Cluster zugeordnet werden konnte. Eine Zuordnung bedeutet nur, dass die SysGraph-Signaturen eine sehr hohe Ähnlichkeit aufweisen.

Bei zwei Kryptotrojaner-Proben konnte keine Ähnlichkeit zu einer der drei Referenz-Signaturen über dem Schwellwert von 97,6% festgestellt werden. Eine der beiden *nicht-klassifizierten* Proben wies eine Ähnlichkeit von 93,56% zur WannaCry-Referenz-Signatur, die andere Probe wies eine Ähnlichkeit von 80,07% zur Jigsaw-Referenz-Signatur auf.

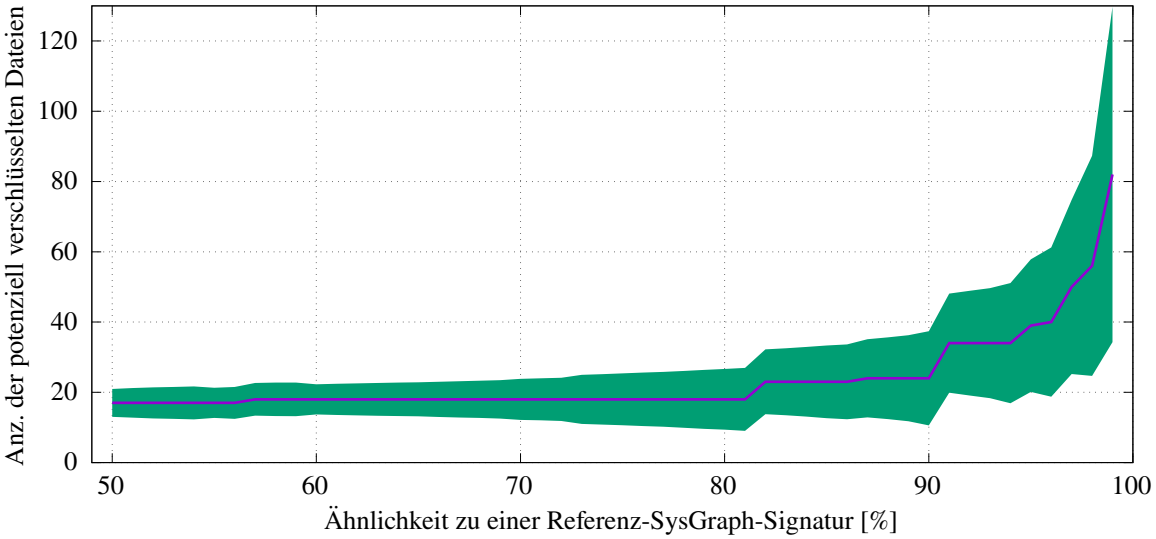


Abbildung 5.15: Medianverlauf und Standardabweichung der potenziell verschlüsselten Dateien durch die Kryptotrojaner-Proben aus dem bereinigten Testdatensatz zum Zeitpunkt einer gemessenen Ähnlichkeit zu einer Referenz-Signatur.

Statistiken über die Anzahl der potenziell verschlüsselten Dateien

Die Anzahl der potenziell verschlüsselten Dateien wurden anhand der im Informationsfluss-SysGraphen vorkommenden Kanten zwischen einem Prozessknoten und einem Dateisystemobjektknoten abgeschätzt. Dieser Heuristik liegt die Annahme zugrunde, dass die Anzahl der tatsächlich verschlüsselten Dateien und die Anzahl der vorhandenen Kanten von der Form *Prozess* • ⟶ • *Dateisystemobjekt* im Informationsfluss-SysGraphen direkt miteinander korrelieren. In Anbetracht der Konstruktion von SysGraphen und der Beschreibung der Fälle, in denen eine Kante zwischen einem Prozess und einem Dateisystemobjekt erzeugt wird, wie sie in Abschnitt 5.4.1.2 beschrieben sind, wird allerdings ersichtlich, dass diese Heuristik nur eine obere Schranke darstellt und die Anzahl der tatsächlich verschlüsselten Dateien weit darunter liegen kann. Eine Kante zwischen einem P-Knoten und einem F-Knoten existiert zum Beispiel auch dann in einem SysGraphen, wenn der zugehörige Prozess das Dateisystemobjekt umbenennt, verschiebt oder löscht. Hinzu kommt, dass ein reiner Lesezugriff auf ein Dateisystemobjekt, der keine Verschlüsselung des Dateisystemobjekts bedeuten kann, mittels des `openat()` Systemaufrufs und dem Dateistatusindikator `O_RDWR` eingeleitet werden kann. Dabei entsteht im zugehörigen SysGraphen allerdings auch eine Kante vom P-Knoten zum F-Knoten, selbst wenn ein Schreiben der Datei tatsächlich gar nicht stattfindet.

Vor diesem Hintergrund ist zunächst eine Bereinigung des Testdatensatzes vorgenommen worden, bei dem 4 Ausreißer-Proben entfernt wurden, die anhand ihrer Statistiken signifikant mehr Dateien potenziell verschlüsselt haben, als die tatsächlich verfügbare Anzahl von 2000. In Abbildung 5.15 ist der Median sowie die Standardabweichung der potenziell verschlüsselten Dateien aller Kryptotrojaner-Proben aus dem bereinigten Testdatensatz zum Zeitpunkt der gemessenen Ähnlichkeiten zu den Referenz-SysGraph-Signaturen eingezeichnet. Dabei wurden die Statistiken auf die Ähnlichkeit bis 99% beschränkt, da alle Proben mindestens eine Ähnlichkeit von 98,7% zu einer Referenz-Signatur erreicht haben. Anhand des Medianverlaufs wird ersichtlich, dass alle Kryptotrojaner-Proben im Median potenziell nur 24 Dateien verschlüsselt haben, bis eine Ähnlichkeit von 90% zu einer Referenz-Signatur erkannt wurde.

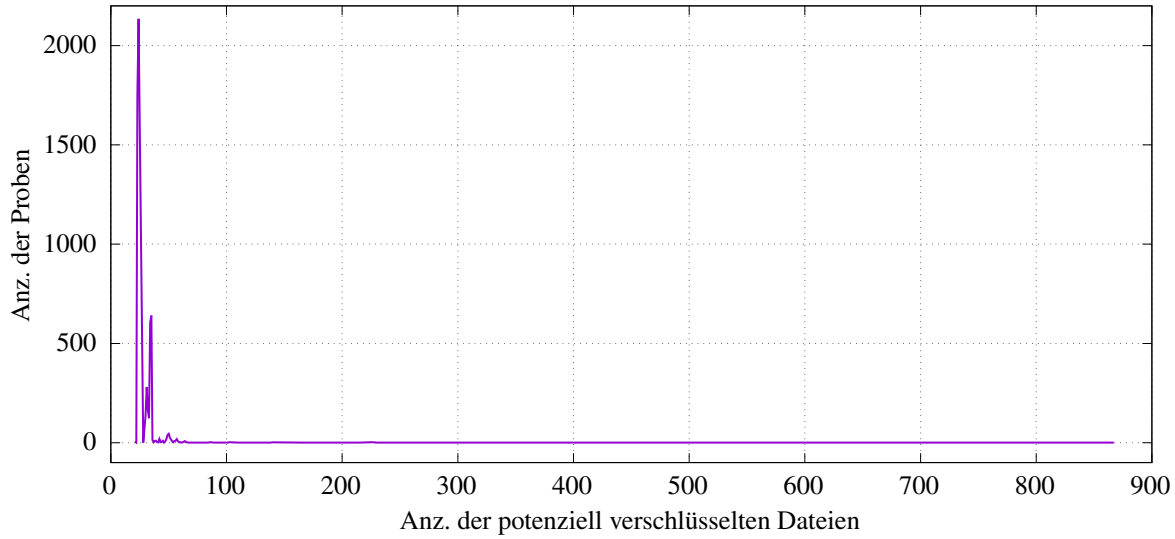


Abbildung 5.16: Verteilung der potenziell verschlüsselten Dateien unter den Kryptotrojaner-Proben aus dem bereinigten Testdatensatzes bei einer festen Ähnlichkeit zu einer Referenz-SysGraph-Signatur in Höhe von 90%.

Die Verteilung der Anzahlen der potenziell verschlüsselten Dateien bis zu dem Zeitpunkt, bei dem eine Ähnlichkeit von 90% zu einer Referenz-Signatur gemessen wurde, ist in Abbildung 5.16 abgebildet. Man erkennt die eindeutige Spitze von 2135 Proben, die bis zu diesem Zeitpunkt potenziell 24 Dateien verschlüsselt haben, wodurch auch der Median an diesem Ähnlichkeitspunkt in Abbildung 5.15 hervorgeht. Vereinzelte Proben haben zu diesem Zeitpunkt auch bereits weit mehr Dateien potenziell verschlüsselt, was allerdings vor dem Hintergrund der bereits erwähnten oberen Schranke gesehen werden muss. An dieser Stelle sind weitere Untersuchungen und genauere Statistiken nötig, um bessere Erkenntnisse zu erzielen.

5.8 Erweiterungsmöglichkeiten

Die nachfolgend beschriebenen Ansätze könnten in zukünftigen Arbeiten eingehender untersucht und umgesetzt werden.

Weiterentwicklung des *audisp-hostmon*-Plugins Im aktuellen Machbarkeitsnachweis der Erkennungs- und Abwehrtechnik von Insiderbedrohungen werden die Auditierungsnachrichten der Kernelkomponente *audit* durch das neu entwickelte und prototypisch implementierte Plugin *audisp-hostmon* der Verteilerkomponente *audisp* verarbeitet (vgl. Abschnitt 5.3). In einem weiteren Schritt könnte *audisp-hostmon* äquivalent zu den Arbeiten aus [Hub16; Jun+14] den Audit-Daemon *auditd* direkt ersetzen und die Ereignisnachrichten über die Netlink-Verbindung ohne Zwischenschritt entgegennehmen und bearbeiten. Dadurch würde eine zentrale Komponente der Erkennungs- und Abwehrtechnik näher an die Ereignisquellen heranrücken und so aktuell noch vorhandene Zwischenkomponenten umgehen. Im Gegenzug müssten allerdings auch die Aufgaben des Audit-Daemon *auditd* auf eine robuste und zuverlässige Art und Weise übernommen werden. Dazu gehört zum Beispiel eine sinnvolle Überlaststeuerung, falls die

Kernelkomponente *audit* zu viele Ereignisnachrichten in zu kurzer Zeit sendet, die dadurch nicht mehr bearbeitet werden können.

Limitierungen der SysGraph-Konstruktion Die Konstruktion von SysGraphen, wie sie in Abschnitt 5.4.1 beschrieben ist, hat die folgenden Limitierungen, die in zukünftigen Arbeiten verringert beziehungsweise umgangen werden könnten:

- Die Eltern-Kind-Prozess-Beziehung wird allein durch die in den Ereignisnachrichten des Linux Audit Systems bereits vorhandenen Prozess-Informationen abgebildet. Dadurch wird ersichtlich, welche Prozess-ID für einen Systemaufruf verantwortlich war und wie dessen Elternprozess-ID lautet. Die Verknüpfungen mit weiteren Elternprozessen ist demnach nur möglich, wenn der Elternprozess ebenfalls Systemaufrufe tätigt, die aufgezeichnet werden. Dadurch wird wiederum dessen Elternprozess-ID offenbart. Durch die gezielte Nutzung des `fork()` Systemaufrufs, der in dieser Dissertation bewusst ignoriert wurde, kann diese Kette der Eltern-Kind-Beziehungen unterbrochen werden, wodurch eine Lücke im SysGraphen entsteht, die auch die Signatur des SysGraphen stark beeinflussen könnte.
- Die gerichteten Kanten von P-Knoten zu F-Knoten und umgekehrt basieren allein auf den Dateistatusindikatoren `O_RDONLY`, `O_WRONLY`, `O_RDWR` sowie `O_APPEND`, des `openat()` Systemaufrufs. Im Prinzip ist dies allerdings keine Garantie dafür, dass der zum P-Knoten zugehörige Prozess auch tatsächlich von dem zum F-Knoten zugehörigen Dateisystemobjekt liest beziehungsweise in dieses schreibt. Dafür müssten die Systemaufrufe `read()` und `write()` aufgezeichnet werden. Das wiederum erzeugt eine sehr große Menge an Ereignisnachrichten in sehr kurzer Zeit und würde die beteiligten Auditierungskomponenten sowie den gesamten Rechner überlasten.
- Die Interaktion von Prozessen mit Dateisystemobjekten werden allein anhand der Systemaufrufe `openat()` sowie `sendfile()` aufgezeichnet. Es existieren allerdings weitere Möglichkeiten, mit denen Daten in Dateisystemobjekte geschrieben beziehungsweise aus diesen herausgelesen werden können. Eine dieser Möglichkeiten ist der Systemaufruf `mmap()`, mit dem spezielle Bereiche von Dateisystemobjekten direkt in den virtuellen Adressraum des aufrufenden Prozesses abgebildet werden können.
- Die gerichteten Kanten von P-Knoten zu S-Knoten und umgekehrt basieren allein auf den Rechnernetz-assoziierten Systemaufrufen `sendto()`, `connect()`, `recvfrom()`, sowie `accept4()`. Sie beschreiben damit lediglich die Initiierungsrichtung einer Interprozesskommunikation. In welche Richtung tatsächlich Daten gesendet werden, wird daran nicht ersichtlich. Selbst eine Überwachung der dafür zuständigen Systemaufrufe, wie etwa `send()`, `sendmsg()`, `recv()`, `recvmsg()` oder auch `read()` und `write()`, würde in den meisten Fällen keinen Aufschluss darüber geben, von welchem Kommunikationsteilnehmer relevante Daten angefordert wurden und welcher Kommunikationsteilnehmer diese angeforderten Daten sendet. Die zuständigen Protokolle, die für eine solche Kommunikation über Sockets von den Teilnehmern verwendet werden, sehen in der Regel Protokollnachrichten in beide Richtungen vor.
- Gerichtete Kanten von einem P-Knoten zu einem S-Knoten, der einen entfernten Rechner repräsentiert, differenzieren nicht zwischen verschiedenen Diensten, die möglicherweise auf demselben entfernten Rechner unter verschiedenen Ports angeboten werden.
- Der Systemaufruf `mount()` wird bisher noch nicht beachtet. Damit könnte allerdings das Schreiben von Dateien auf externe Datenträger, wie es zum Beispiel in Szenario 5.1 in Abschnitt 5.6.1 beschrieben ist, vom Schreiben von Dateien im lokalen Dateisystem

unterschieden werden. Dadurch könnten präzisere SysGraph-Signaturen für diese Art von Szenarien erstellt und analysiert werden.

Multigraphen Die Konstruktion von SysGraphen sieht nur einfach gerichtete Kanten zwischen zwei Knoten vor. Somit existiert nur eine Kante zwischen zwei Knoten, auch wenn die zugehörige Interaktion zwischen den beiden Ressourcen mehrfach in gleicher Art und Weise stattfindet. Die Registrierung solcher Mehrfachinteraktionen wird für die Extraktion der Informationsfluss-SysGraphen benötigt, aber haben keinen Einfluss auf eine SysGraph-Signatur. Allerdings könnten Mehrfachkanten derselben Richtung zu präziseren SysGraph-Signaturen und damit zu einer effizienteren Erkennung und Abwehr von Insiderbedrohungen führen. Ein solcher Graph, dessen Kantenmenge eine Multimenge ist,[13] wird *Multigraph* genannt.

Erfolg-SysGraphen versus Fehler-SysGraphen Alle von der Kernelkomponente *audit* aufgezeichneten Auditierungsnachrichten enthalten Informationen darüber, ob ein aufgezeichneter Systemaufruf erfolgreich durchgeführt oder fehlerhaft abgebrochen wurde. Im Fehlerfall enthält der ebenfalls in den Auditierungsnachrichten enthaltene Rückgabewert Informationen über die Art des Fehlers. So kann beispielsweise erkannt werden, wenn ein Prozess auf eine Datei zugreifen möchte, obwohl keine gültigen Zugriffsrechte vorliegen. Die bisherige Konstruktion der SysGraphen konzentriert sich ausschließlich auf die Auditierungsnachrichten, die eine erfolgreiche Durchführung des aufgezeichneten Systemaufrufs bescheinigen. Dadurch werden auch nur tatsächlich stattfindende Interaktionen zwischen den Ressourcen eines Rechners analysiert. Eine Analyse der fehlerhaften Interaktionen könnte allerdings ebenso aufschlussreich sein.

Weitere Auditierungsmechanismen anderer Betriebssysteme Der entwickelte Schutzmechanismus fokussiert sich auf den Auditierungsmechanismus des *Linux-Kernels* (vgl. Abschnitt 5.3) und ist dadurch auch auf Betriebssysteme beschränkt, die diesen Mechanismus einsetzen. Es gibt allerdings auch Auditierungsmechanismen anderer Betriebssysteme beziehungsweise Softwareprodukte, die Informationen über Systemaufrufe sammeln, aufzeichnen und in Log-Dateien ablegen können. Für *macOS* existiert das *Open Source Basic Security Module* (OpenBSM)[14], welches vom Aufbau und der Funktionsweise dem Linux Audit System sehr ähnlich ist. Für *Windows* existieren mehrere vielversprechende Möglichkeiten. Dazu gehören die *Windows Advanced Audit Policy Configuration*[15], das *Event Tracing for Windows*[16] sowie die Software *Windows Sysmon*[17].

Falsch-Positiv-Rate und Vergleich mit anderen Techniken Die Evaluation hat in Abschnitt 5.7 vielversprechende Ergebnisse in Bezug auf eine spezielle Insiderbedrohung aufgezeigt. Die entwickelte Technik hat allerdings weiteres Potenzial für die Erkennung anderer Insiderbedrohungen und muss dafür im Besonderen mit anderen Erkennungs- und Abwehrtechniken verglichen werden. Darüber hinaus wurde in dieser Dissertation nicht untersucht, wie sensibel die

13. In Multimengen können Mengenelemente mehrfach vorkommen.
14. Informationen verfügbar unter https://github.com/openbsm/openbsm.
15. Informationen verfügbar unter https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/advanced-security-auditing.
16. Informationen verfügbar unter https://go.microsoft.com/fwlink/p/?linkid=213103.
17. Informationen verfügbar unter https://docs.microsoft.com/en-us/sysinternals/downloads/sysmon.

entwickelte Abwehrtechnik auf Nicht-Insiderbedrohungen reagiert und diese fälschlicherweise als Bedrohungen deklariert.

5.9 Fazit

Die in diesem Kapitel vorgestellte Erkennungs- und Abwehrtechnik von Insiderbedrohungen wurde mit dem Ziel entwickelt, die *Uncertainty* von Insidern zu reduzieren. Der grundlegende Sicherheitsmechanismus, der dabei zum Einsatz kommt, ist die Erkennung und Abwehr von Anomalien in der Interaktion von Insidern mit den Ressourcen eines Rechners (vgl. Abschnitt 4.9). Dabei wird speziell die Abweichung einzelner Insideraktivitäten von erwarteten und erlaubten Aktivitäten sowie die Wiedererkennung von bereits unerlaubten und gefährlichen Aktivitäten anvisiert. Damit wird Forschungsbeitrag B3 geleistet und abgeliefert (vgl. Abschnitt 1.3).

Die für diese Analysen benötigten Daten gehen aus den in Abschnitt 5.3.1 beschriebenen Anwendungsfällen hervor, die sich zusammenfassen lassen in die Zugriffe auf Dateisystemobjekte, die Programmaufrufe und Kommandozeilenbefehle eines Nutzers sowie die Netzwerk-assoziierten Verbindungsaktivitäten. Sie wurden unter Zuhilfenahme des Linux Audit System auf der Systemaufrufebene akquiriert und durch die Entwicklung eines eigenen *audisp* Plugins namens *audisp-hostmon* (vgl. Abschnitt 5.3.4), aufbauend auf bereits existierenden Arbeiten (vgl. Abschnitt 5.3.3), für die Weiterverarbeitung aufbereitet. Insbesondere ignorierte Datenfelder in existierenden Arbeiten sowie ungeeignete Datenschemata machten diese Entwicklung notwendig und erlaubten vorbereitende Schritte für die Entwicklungsarbeiten in den Kapiteln 6 und 7.

Die umfassenden Daten, welche die Interaktionen zwischen Prozessen, Dateisystemobjekten und Sockets auf niedriger Ebene enthalten, wurden in Abschnitt 5.4 in neuartige Graphen überführt, die aufgrund ihrer Beschaffenheit SysGraphen genannt wurden. Jeder Knoten in einem SysGraphen repräsentiert eine Ressource in einem Rechner und wurde entsprechend des Ressourcentyps als Prozess-, Dateisystemobjekt- oder Socket-Knoten markiert. Die gerichteten Kanten zwischen den Knoten wurden in Abschnitt 5.4.1 derart festgelegt, dass sie die Informationen aus den aufgezeichneten Systemaufrufen im SysGraphen abbilden. Somit wird, stark vereinfacht ausgedrückt, im Graphen ersichtlich, welcher Prozess von welcher Datei gelesen, in welche Datei geschrieben, welchen Socket initiiert, von welchem Socket kontaktiert oder welchen Kindprozess erzeugt hat. Eine derartige Darstellung von Aktivitäten an einem Rechner erlaubt einerseits die Anwendung von Methoden und Statistiken aus der Graphentheorie und andererseits eine starke Reduktion der Datenmenge bei einem gleichzeitigen Mehrwert für die Analyse und Interpretation von Rohdaten. Die normale Arbeitssitzung beispielsweise, deren Systemaufrufe für Abschnitt 5.6.2.2 aufgezeichnet und in einem Teilausschnitt in Abbildung 5.11a dargestellt wurden, erzeugte insgesamt eine Datenmenge an Ereignisnachrichten in Höhe von 13,48 Megabytes. Mit der Überführung in einen SysGraphen konnte diese Datenmenge auf 788,18 Kilobytes reduziert werden.

Bei dem soeben erwähnten Teilausschnitt handelt es sich um einen speziellen Subgraphen des gesamten SysGraphen, der ausgehend von einem Generatorknoten die chronologisch zusammenhängenden Abfolgen von Aktivitätsschritten betreffend der Ressource des Generatorknotens isoliert und in einem eigenen, als Informationsfluss-SysGraphen bezeichneten Graphen darstellt. Diese Informationsfluss-SysGraphen wurden in Abschnitt 5.4.2 eingeführt und mit einem Algorithmus zur Extraktion versehen. Sie erlauben die Analyse einzelner Aktivitäten getrennt von der

Gesamtheit und lassen somit genauere Rückschlüsse auf die Bedrohlichkeit einzelner Vorgänge zu.

Aus den erwähnten Methoden und Statistiken der Graphentheorie wurden für diese Dissertation die Häufigkeiten vorkommender induzierter Subgraphen der festen Knotenzahl 3 ausgewählt. Diese Häufigkeiten der sogenannten 3-Motive sind charakteristische Eigenschaften, deren Offenlegung komplexe Strukturen im Graphen erfassbar und analysierbar machen. Die insgesamt 16 möglichen klassischen 3-Motive wurden in Abschnitt 5.5 auf 65 mögliche 3-SysGraph-Motive erweitert und dadurch den unterschiedlichen Knotentypen der neuen SysGraphen angepasst. Die Gesamtheit der Häufigkeiten aller SysGraph-Motive ergibt eine SysGraph-Signatur, die sich als typisch für einzelne Insideraktivitäten erweist. Eine besondere Herausforderung war die algorithmische Bestimmung dieser Häufigkeiten, bei der in Abschnitt 5.5.3 das Graphen-Isomorphismus-Problem im Kontext von 3-knotigen SysGraphen gelöst werden musste, dessen Schwierigkeit als $\mathcal{NP}$-intermediär angenommen wird (vgl. Abschnitt 5.1.3.2).

Mit den SysGraph-Signaturen von Informationsfluss-SysGraphen wurde in Abschnitt 5.6.2 anhand von drei unterschiedlichen Insiderbedrohungsszenarien aufgezeigt, dass sich selbst leicht abweichende Bedrohungsaktivitäten desselben Szenarios wiedererkennen lassen. Die Kosinusähnlichkeiten der SysGraph-Signaturen von jeweils zwei Angriffsaktivitäten innerhalb eines Szenarios lagen bei mindestens 95,7%. Andererseits ließen sich auch Abweichungen der Angriffsaktivitäten von unbedrohlichen Aktivitäten als geeignete Anomalieerkennungsmaßnahme nachweisen. Die Mindestdistanz konnte bei den durchgeführten Beispielaktivitäten auf 43,26% festgestellt werden. Legt man diese Zahlen zugrunde, kann also eine im Echtzeitbetrieb aufgezeichnete Insideraktivität, deren SysGraph-Signatur eine Ähnlichkeit im Bereich von 96% zu bereits bekannten und als bedrohlich eingestuften SysGraph-Signaturen aufweist, als Bedrohungsaktivität markiert werden. Durch die Ähnlichkeit können zusätzliche Aussagen darüber gemacht werden, um welche Art von Bedrohungsaktivität es sich handelt. Umgekehrt kann eine im Echtzeitbetrieb aufgezeichnete Insideraktivität, deren SysGraph-Signatur eine Kosinusdistanz im Bereich von 43% zu einer bereits bekannten und als unbedrohlich eingestufte SysGraph-Signatur der gleichen Art von Aktivität aufweist, ebenfalls als Bedrohungsaktivität markiert werden. Letzteres erlaubt beispielsweise die Erkennung von Modifikationen an Prozessabläufen durch Schadsoftware. Anpassungen und Optimierungen der SysGraph-Signaturberechnung für einen solchen Echtzeiteinsatz, sodass Bedrohungen möglichst frühzeitig noch während der Bedrohungsaktion erkannt und gestoppt werden können, wurden in Abschnitt 5.6.3 vorgenommen. Erst mit diesen Änderungen wird aus der hier entwickelten Erkennungs- auch eine Abwehrtechnik.

Eines der drei Insiderbedrohungsszenarien, in dem durch unbedachte Handlungen von Insidern Kryptotrojaner alle Dateien und Ordner eines Rechners verschlüsseln, wurde für die Evaluation in Abschnitt 5.7 zugrunde gelegt. Dabei wurde die gängige Praxis ausgenutzt, dass Kryptotrojaner in der Regel auch alle Dateien und Ordner auf angeschlossenen Netzlaufwerken verschlüsseln. Dadurch konnten Kryptotrojaner auf einem Windows-Rechner ausgeführt werden und deren Aktivitäten auf einem anderen Linux-Rechner mit der entwickelten Software für das Linux Audit System aufgezeichnet werden. Der Linux-Rechner stellte dabei das Netzlaufwerk für den Windows-Rechner zur Verfügung. Insgesamt wurden damit 6199 unbekannte Proben von Kryptotrojanern ausgeführt und konnten anschließend mithilfe der SysGraphen sowie der SysGraph-Signaturen analysiert werden. Die Ergebnisse der Evaluation zeigen, dass alle bis auf zwei Kryptotrojaner-Proben mit einer Mindestähnlichkeit von 98,6% zu einer zuvor unabhängig erzeugten Kryptotrojaner-SysGraph-Signatur zugeordnet werden konnten. Die beiden nicht-klassifizierten Proben wiesen immer noch eine sehr hohe Kosinusähnlichkeit von 93,56% und 80,07%

zu den bekannten SysGraph-Signaturen auf. Darüber hinaus wurde bei der Auswertung der potenziell verschlüsselten Dateien unter allen Kryptotrojanerproben ersichtlich, dass bereits bei einer Ähnlichkeit von 90% zu einer Referenz-SysGraph-Signatur im Median nur 24 Dateien potenziell verschlüsselt wurden. Die Ergebnisse zeigen, wie robust die SysGraph-Signaturen im zugrunde liegenden Bedrohungsszenario sind und wie schnell auf eine solche Bedrohung reagiert und schlimmeres verhindert werden kann, auch wenn die Erweiterungsmöglichkeiten in Abschnitt 5.8 noch Forschungsbedarf aufzeigen.

6 Insiderdatenschutz beim Einsatz von Sicherheitsmaßnahmen

Sicherheitsmaßnahmen gegen Insiderbedrohungen verfolgen primär das Ziel der Erkennung und Abwehr von Insiderbedrohungsaktionen. Davon ausgehend entstehen allerdings für die Insider selbst negative Auswirkungen in Form von Eingriffen in die Privatsphäre und die informationelle Selbstbestimmung. Der Lebensmitteldiscounter Lidl hat beispielsweise im Jahr 2008 tiefgehende Überwachungsmaßnahmen der eigenen Mitarbeiter durchgeführt, um Fehlverhalten am Arbeitsplatz aufzudecken und entsprechend behandeln zu können [AD08]. Dabei sind Informationen über die Mitarbeiter zusammengetragen worden, die weitreichende Einblicke in das Privatleben und in persönliche Präferenzen der Betroffenen erlaubten. Eine solche Bedrohung für Insider durch deren Domäne wurde bereits in Abschnitt 3.4 betrachtet und soll in diesem Kapitel genauer bearbeitet werden. Damit wird der Fokus auf Forschungsfrage 4 aus Abschnitt 1.2 gelegt.

Der für eine Insidermodellierung relevante Bezugspunkt einer Domäne gliedert sich typischerweise in den Kontext eines Unternehmens oder einer Organisation und der jeweils zugehörigen Infrastruktur ein. Die folgenden beiden Kapitel beziehen sich sowohl auf diesen Kontext im Allgemeinen und im Speziellen auf das Arbeitgeber-Arbeitnehmer-Verhältnis. Für eine begriffliche Einordnung der Erkennung und Abwehr von Insiderbedrohungen, auf die in diesem Kapitel mehrfach Bezug genommen wird, wird auf Abschnitt 1.5.4 sowie auf Definition 3.1 in Abschnitt 3.2.1 verwiesen.

Wesentliche Inhalte Mit einer Herausarbeitung der rechtlichen Rahmenbedingungen für den Einsatz von Erkennungs- und Abwehrtechniken im Kontext von Insiderbedrohungen werden die Arbeitnehmerrechte den Rechten und Pflichten der Arbeitgeber gegenübergestellt. Weiterhin werden dabei auftretende Rechts- und Interessenskonflikte diskutiert. Es wird gezeigt, dass der Datenschutz von zentraler Bedeutung für die Arbeitnehmerrechte ist und die Datenschutzgrundverordnung (DSGVO) auch im Beschäftigungskontext Anwendung findet. Die DSGVO schließt allerdings den Einsatz derartiger Techniken nicht vollständig aus, sondern setzt einen Schutzrahmen, in dem die Verarbeitung von personenbezogenen Daten etwa für Zwecke der Wahrung berechtigter Interessen oder der rechtlichen Verpflichtungen im Beschäftigungskontext ermöglicht wird. Abgeleitet von dieser Gegenüberstellung werden darüber hinaus die gegenläufigen Anforderungen an Erkennungs- und Abwehrtechniken beschrieben. Damit wird die Grundlage für den Forschungsbeitrag B4 aus Abschnitt 1.3 gelegt, denn es werden Techniken und Konzepte vorgestellt, die diese Rechts- und Anforderungskonflikte adressieren.

Relevante Veröffentlichungen Die gegenläufigen Anforderungen in Abschnitt 6.2 wurden bereits in [Zim+20] erwähnt. Darüber hinaus wurde das Zusammentragen von existierenden Arbeiten zur vorliegenden Thematik in Abschnitt 6.3 bereits teilweise in [Zim+16; Zim+20] durchgeführt.

Aufbau des Kapitels Das Kapitel beginnt in Abschnitt 6.1 mit einer ausführlichen Betrachtung der Rechtslage in Deutschland bezüglich des Einsatzes von Techniken zur Erkennung und Abwehr von Insiderbedrohungen. Dabei werden beide Seiten eines Arbeitnehmer-Arbeitgeber-Verhältnisses untersucht und gegenübergestellt sowie Konflikte aufgezeigt. Im Anschluss daran erfolgt die Betrachtung von Anforderungen an derartige Techniken in Abschnitt 6.2. Dabei wird ebenfalls die Gegenläufigkeit der beiden Seiten ersichtlich, da sie teilweise einen direkten Bezug zu den vorliegenden Rechten und Pflichten aufweisen. In Abschnitt 6.3 werden existierende Arbeiten zur Erhöhung des Datenschutzes bei der Umsetzung von Sicherheitsmaßnahmen aufgeführt, die für den vorliegenden Kontext eine wichtige Rolle spielen. Ein abschließendes Fazit erfolgt in Abschnitt 6.4.

6.1 Rechtlicher Rahmen

Bei einer Erkennung und Abwehr von Insiderbedrohungen im Unternehmenskontext spielen rechtliche Aspekte eine relevante Rolle. Das ergibt sich aus der Tatsache, dass Unternehmen auf der einen Seite sowohl das Recht auf als auch gegebenenfalls die Pflicht zum Schutz ihrer Wirtschaftsgüter und ihrer Infrastrukturen haben. Andererseits haben jedoch auch die von den Schutzmaßnahmen betroffenen Personen das Recht auf den Schutz ihrer Privatsphäre und ihrer informationellen Selbstbestimmung. Diese mehrseitigen Rechte und Pflichten unterscheiden sich zudem noch je nach der vorliegenden Gerichtsbarkeit. Dieser Abschnitt beleuchtet die rechtlichen Aspekte sowie die aktuelle Rechtsprechung bezüglich des Datenschutzes im Arbeitnehmer-Arbeitgeber-Verhältnis in Deutschland. Für detailliertere Diskussionen und Kommentare wird auf Däubler [Däu17] sowie Franzen, Gallner und Oetker [FGO18] verwiesen.

6.1.1 Rechte von Arbeitnehmern

Anhand der folgenden Abschnitte werden die Rechte von Arbeitnehmern im Beschäftigungskontext aufgezeigt, wenn sie Mechanismen zur Erkennung und Abwehr von Insiderbedrohungen ausgesetzt sind. Im Zentrum der Arbeitnehmerrechte insbesondere in diesem Kontext steht der Datenschutz, der sich anhand der in Deutschland und der Europäische Union (EU) geltenden Grundrechte herleiten lässt. Darüber hinaus haben Interessensvertretungen von Arbeitnehmern die Möglichkeit, auf den Einsatz von Mechanismen zur Bedrohungserkennung und -prävention Einfluss zu nehmen, sofern diese Mechanismen auch zur Leistungskontrolle geeignet sind.

6.1.1.1 Grundrechte

Nach europäischem Recht ist der Schutz personenbezogener Daten und das Recht auf Achtung des Privatlebens und der Kommunikation ein Grundrecht, festgeschrieben in den Artikeln 7 und 8 der EU-Grundrechtecharta (GRCh), die durch den Lissabon-Vertrag 2009 in Deutschland zu geltendem Recht wurde.[1] Dort heißt es:

> „Art. 7 Achtung des Privat- und Familienlebens

1. Bundesgesetzblatt Jahrgang 2009 Teil II Nr. 36 (S. 1223): Bekanntmachung vom 13. November 2009. In Verbindung mit Bundesgesetzblatt Jahrgang 2008 Teil II Nr. 27 (S. 1038): Gesetz zum Vertrag von Lissabon vom 08.10.2008.

> Jede Person hat das Recht auf Achtung ihres Privat- und Familienlebens, ihrer Wohnung sowie ihrer Kommunikation.
>
> Art. 8 Schutz personenbezogener Daten
>
> (1) Jede Person hat das Recht auf Schutz der sie betreffenden personenbezogenen Daten.
>
> (2) Diese Daten dürfen nur nach Treu und Glauben für festgelegte Zwecke und mit Einwilligung der betroffenen Person oder auf einer sonstigen gesetzlich geregelten legitimen Grundlage verarbeitet werden. Jede Person hat das Recht, Auskunft über die sie betreffenden erhobenen Daten zu erhalten und die Berichtigung der Daten zu erwirken.
>
> (3) Die Einhaltung dieser Vorschriften wird von einer unabhängigen Stelle überwacht."

Aus dem national geltenden Grundgesetz (GG) wurde das Grundrecht auf informationelle Selbstbestimmung durch das Urteil des Bundesverfassungsgerichts (BVerfG) über das Volkszählungsgesetz 1983 abgeleitet. Dadurch wurde das durch Art. 2 Abs. 1 in Verbindung mit Art. 1 Abs. 1 GG geschützte allgemeine Persönlichkeitsrecht auf die Preisgabe und Verwendung persönlicher Daten erweitert.[2] In dem Urteil heißt es:

> „Unter den Bedingungen der modernen Datenverarbeitung wird der Schutz des Einzelnen gegen unbegrenzte Erhebung, Speicherung, Verwendung und Weitergabe seiner persönlichen Daten von dem allgemeinen Persönlichkeitsrecht des GG Art. 2 Abs. 1 in Verbindung mit GG Art. 1 Abs. 1 umfasst. Das Grundrecht gewährleistet insoweit die Befugnis des Einzelnen, grundsätzlich selbst über die Preisgabe und Verwendung seiner persönlichen Daten zu bestimmen."

6.1.1.2 EU-Datenschutzgrundverordnung

Vor dem Hintergrund des Grundrechts auf Datenschutz wurde im Mai 2016 mit der EU-Datenschutzgrundverordnung (DSGVO) die Verarbeitung personenbezogener Daten für alle Mitgliedsländer der EU einheitlich geregelt. Sie wurde spätestens ab dem 25. Mai 2018 verbindlich und unmittelbar in Kraft gesetzt und ersetzte damit auch das zu dieser Zeit in Deutschland geltende Bundesdatenschutzgesetz (BDSG) beziehungsweise führte zu dessen Novellierung, mit der die in der DSGVO vorgesehenen Öffnungsklauseln auf nationaler Ebene ausgestaltet wurden.[3] Laut Art. 2 DSGVO gilt die Verordnung (VO) unter anderem „für die ganz oder teilweise automatisierte Verarbeitung personenbezogener Daten". Sofern bei der Erkennung und Abwehr von Insiderbedrohungen also diese Voraussetzungen erfüllt sind und keine Datenschutz-Sonderregelungen vorliegen, findet die DSGVO Anwendung.

2. BVerfG, Urteil des Ersten Senats vom 15. Dezember 1983 - 1 BvR 209/83 - Rn. (1–215), Fundstelle: BVerfGE 65, 1–71.
3. Bundesgesetzblatt Jahrgang 2017 Teil I Nr. 44 (S. 2097): Gesetz zur Anpassung des Datenschutzrechts an die Verordnung (EU) 2016/679 und zur Umsetzung der Richtlinie (EU) 2016/680 (Datenschutz-Anpassungs- und -Umsetzungsgesetz EU – DSAnpUG-EU) vom 30. Juni 2017.

Ganz oder teilweise automatisierte Verarbeitung

Bei der Handhabung von Überwachungsdaten zur Bedrohungserkennung und -prävention muss geprüft werden, ob es sich um eine Verarbeitung im Sinne der DSGVO handelt. Dabei fasst die VO den Begriff der *Verarbeitung* in Art. 4 Abs. 2 durchaus sehr weit, denn er bezeichnet

> „jeden mit oder ohne Hilfe automatisierter Verfahren ausgeführten Vorgang oder jede solche Vorgangsreihe im Zusammenhang mit personenbezogenen Daten wie das Erheben, das Erfassen, die Organisation, das Ordnen, die Speicherung, die Anpassung oder Veränderung, das Auslesen, das Abfragen, die Verwendung, die Offenlegung durch Übermittlung, Verbreitung oder eine andere Form der Bereitstellung, den Abgleich oder die Verknüpfung, die Einschränkung, das Löschen oder die Vernichtung."

Die konkret genannten Verarbeitungsvorgänge haben dabei keinen einschränkenden Charakter, sondern zeigen mögliche Vorgänge oder Vorgangsreihen auf [Däu17, Kapitel 2 Rn. 49a]. Damit wird deutlich, dass in Bezug auf die Erkennung und Abwehr von Insiderbedrohungen faktisch immer eine *Verarbeitung* im Sinne der DSGVO stattfindet, sofern personenbezogene Daten involviert sind.

Personenbezogene Daten

In jedem konkreten Einzelfall eines Mechanismus zur Erkennung und Abwehr von Insiderbedrohungen muss festgestellt werden, welche Daten dabei gesammelt werden und ob diese Daten einen konkreten oder einen potenziellen Personenbezug haben. Laut Art. 4 Abs. 1 DSGVO sind *personenbezogene Daten* „alle Informationen, die sich auf eine identifizierte oder identifizierbare natürliche Person [...] beziehen [...]."

Diese Unterscheidung in *identifiziert* und *identifizierbar* ist für den vorliegenden Fall der Bedrohungserkennung und -prävention relevant. In der Literatur wird diese Unterscheidung auch mit *personenbezogenen* und *personenbeziehbaren* Daten betitelt [MW02, Abschnitt 3.1]. Dabei geht es einerseits um Informationen, die für sich genommen allein eine Verknüpfung mit einer Identität erlauben, wie etwa ein Name. Andererseits geht es um Informationen, die für sich genommen möglicherweise noch keinen identifizierenden Charakter haben, aber in Kombination mit anderen Informationen eine Identifizierung der zugehörigen Person erlauben. Denn weiter wird in Art. 4 Abs. 1 DSGVO eine natürliche Person als *identifizierbar* bezeichnet, wenn sie

> „direkt oder indirekt, insbesondere mittels Zuordnung zu einer Kennung wie einem Namen, zu einer Kennnummer, zu Standortdaten, zu einer Online-Kennung oder zu einem oder mehreren besonderen Merkmalen identifiziert werden kann, die Ausdruck der physischen, physiologischen, genetischen, psychischen, wirtschaftlichen, kulturellen oder sozialen Identität dieser natürlichen Person sind."

Insbesondere im Kontext der Bedrohungserkennung und -prävention, in dem Daten verschiedenen Ursprungs gesammelt und gegebenenfalls korreliert werden, um untypische Muster (Anomalien) zu erkennen und anzuzeigen, können einzelne Daten für sich genommen möglicherweise als wenig sensibel angesehen werden. In der Gesamtheit und Kombination allerdings können Erkenntnisse geschaffen werden, die einen vormals nicht erkennbaren Personenbezug nun herstellen und offenlegen. Das BVerfG hat diese Möglichkeit bereits in seinem Urteil zum Volkszählungsgesetz erkannt und sich dahingehend geäußert, dass es „unter den Bedingungen der automatischen

Datenverarbeitung kein *belangloses* Datum mehr“ gibt.[4] Das hat auch Auswirkungen auf Methoden zur Pseudonymisierung. Bestätigt wird dies durch die DSGVO im Erwägungsgrund[5] 26 Satz 2, in dem es heißt:

> „Einer Pseudonymisierung unterzogene personenbezogene Daten, die durch Heranziehung zusätzlicher Informationen einer natürlichen Person zugeordnet werden könnten, sollten als Informationen über eine identifizierbare natürliche Person betrachtet werden.“

Datenschutz-Sonderregelungen

Bei einer Verarbeitung personenbezogener Daten im Zuge der Erkennung und Abwehr von Insiderbedrohungen, die nicht in den sachlichen Anwendungsbereich der DSGVO fällt (vgl. Art. 2 DSGVO) oder explizit von der DSGVO ausgenommen wird, gelten andere Datenschutzbestimmungen für die Betroffenen der Datenverarbeitung [Däu17, Kapitel 2 Rn. 43–45]. Dazu zählt unter anderem laut Art. 2 Abs. 2a DSGVO der Datenschutz bei Tätigkeiten, die nicht in den Anwendungsbereich des Unionsrechts fallen, wie beispielsweise der Bereich der inneren Sicherheit der Mitgliedstaaten.[6] Weiterhin gelten laut Art. 2 Abs. 2d DSGVO spezielle Datenschutzbestimmungen für den Bereich der Strafverfolgung. Dieser wird mit der Richtlinie (EU) 2016/380 des Europäischen Parlaments und des Rates vom 27. April 2016 geregelt[7] und mit dem BDSG in nationales Recht überführt.[8]

6.1.1.3 Betroffenenrechte und Grundsätze für die Verarbeitung

Die Betroffenen einer Datenverarbeitung haben nach Art. 12–22 DSGVO weitreichende Rechte:

- die Transparenz, Verständlichkeit und Kosten (Art. 12),
- die Benachrichtigung und die Informationspflicht (Art. 13 und 14),
- das Auskunftsrecht (Art. 15),
- das Recht auf Berichtigung (Art. 16),
- das Recht auf Löschung („Recht auf Vergessenwerden“) (Art. 17),
- das Recht auf Einschränkung der Verarbeitung (Art. 18),
- die Mitteilungspflicht im Zusammenhang mit der Berichtigung oder Löschung personenbezogener Daten oder der Einschränkung der Verarbeitung (Art. 19),
- das Recht auf Datenübertragbarkeit (Art. 20),

4. BVerfG, Urteil des Ersten Senats vom 15. Dezember 1983 - 1 BvR 209/83 - Rn. (1–215), Fundstelle: BVerfGE 65, 1–71.
5. Erwägungsgründe sind mit der DSGVO gemeinsam veröffentlichte explizite Ziele, die mit der VO verfolgt werden (vgl. [DSGVO16]).
6. Vergleiche Erwägungsgrund 16 der DSGVO.
7. Amtsblatt Nr. L 119 vom 04. Mai 2016 (S. 89–131): Richtlinie (EU) 2016/680 des Europäischen Parlaments und des Rates vom 27. April 2016 zum Schutz natürlicher Personen bei der Verarbeitung personenbezogener Daten durch die zuständigen Behörden zum Zwecke der Verhütung, Ermittlung, Aufdeckung oder Verfolgung von Straftaten oder der Strafvollstreckung sowie zum freien Datenverkehr und zur Aufhebung des Rahmenbeschlusses 2008/977/JI des Rates.
8. Bundesgesetzblatt Jahrgang 2017 Teil I Nr. 44 (S. 2097): Gesetz zur Anpassung des Datenschutzrechts an die Verordnung (EU) 2016/679 und zur Umsetzung der Richtlinie (EU) 2016/680 (Datenschutz-Anpassungs- und -Umsetzungsgesetz EU – DSAnpUG-EU) vom 30. Juni 2017.

- das Widerspruchsrecht (Art. 21) sowie
- die automatisierten Entscheidungen im Einzelfall einschließlich Profiling (Art. 22).

Auf die Ausübung der Betroffenenrechte wird hier nicht näher eingegangen. Es sei aber darauf hingewiesen, dass diese Betroffenenrechte nach der Öffnungsklausel in Art. 23 DSGVO auf nationaler Ebene eingeschränkt werden können, etwa im Fall der Verhütung, Aufdeckung, Ermittlung und Verfolgung von Verstößen gegen die berufsständischen Regeln reglementierter Berufe (vgl. Art. 23 Abs. 1g DSGVO). Das BDSG macht in den §§ 29–37 davon Gebrauch und schränkt in § 29 Abs. 1 Satz 2 beispielsweise das Recht auf Auskunft (Art. 15 DSGVO) ein, sofern durch die Auskunft Informationen offengelegt würden, die aufgrund von Rechtsvorschriften oder überwiegenden berechtigten Interessen Dritter geheim bleiben müssen.

Bei jeglicher Verarbeitung personenbezogener Daten ist die verarbeitende Stelle laut Art. 5 Abs. 2 DSGVO verpflichtet, die in Art. 5 Abs. 1 a)–f) genannten Grundsätze für die Verarbeitung personenbezogener Daten einzuhalten und nachzuweisen. Die Grundsätze umfassen unter anderem die Rechtmäßigkeit, die Verarbeitung nach Treu und Glauben, die Datenminimierung sowie die Speicherbegrenzung. Auf einige dieser Grundsätze wird in Abschnitt 6.2.2 zurückgegriffen.

6.1.1.4 Anwendungsbereich der DSGVO im Beschäftigungskontext

Die Verarbeitung personenbezogener Beschäftigtendaten im Beschäftigungskontext wird in der DSGVO als besondere Verarbeitungssituation gehandhabt und in Art. 88 Abs. 1 DSGVO mit einer Öffnungsklausel versehen, sodass Mitgliedstaaten *spezifischere Vorschriften* vorsehen können. Wie von dieser Öffnungsklausel im deutschen Datenschutzrecht Gebrauch gemacht wird und welche Rechte dadurch der Arbeitgeberseite eingeräumt werden, wird in Abschnitt 6.1.2.4 näher betrachtet. Der Eindruck, der an dieser Stelle entstehen könnte, dass die DSGVO bei der Datenverarbeitung im Beschäftigungskontext keine Anwendung mehr findet, ist allerdings laut Franzen, Gallner und Oetker [FGO18, Abschnitt DS-GVO Rn. 5–14] nicht haltbar. Die Autoren begründen dies anhand der folgenden Punkte:

- Art. 2 DSGVO beschreibt den Geltungsbereich der VO und enthält keine Geltungsbereichsreduktion für den Beschäftigungskontext.
- Verschiedene Vorgaben der DSGVO nehmen expliziten Bezug zur Verarbeitung personenbezogener Beschäftigtendaten im Beschäftigungskontext. Dazu zählt Erwägungsgrund 48 oder auch Art. 9 Abs. 2b DSGVO, in dem es um die Erlaubnis der Verarbeitung besonderer Kategorien personenbezogener Daten geht, wenn damit „der Verantwortliche oder die betroffene Person die ihm beziehungsweise ihr aus dem Arbeitsrecht und dem Recht der sozialen Sicherheit und des Sozialschutzes erwachsenden Rechte ausüben und seinen beziehungsweise ihren diesbezüglichen Pflichten nachkommen kann [...]“.
- Die Bezeichnung „spezifischere Vorschriften“ in Art. 88 Abs. 1 DSGVO lautet im Englischen „more specific rules“. Mit diesem Wortlaut kann man von einer Art Spezialitätsverhältnis zwischen den Vorgaben der VO und den aufgrund der Öffnungsklausel zugelassenen nationalen Bestimmungen zum Beschäftigungsdatenschutz ausgehen. Damit wären auch die allgemeinen Regeln der DSGVO im Beschäftigungskontext zu beachten.
- Die Entstehungsgeschichte von Art. 88 DSGVO deutet darauf hin, dass sich nationale Regelungen in den Grenzen der VO halten müssen.

Für die Arbeitnehmerseite von Bedeutung ist in Art. 88 DSGVO der Abs. 2, der vorgibt, dass diese spezifischen nationalen Vorschriften

> „geeignete und besondere Maßnahmen zur Wahrung der menschlichen Würde, der berechtigten Interessen und der Grundrechte der betroffenen Person, insbesondere im Hinblick auf die [...] Überwachungssysteme am Arbeitsplatz“

enthalten müssen. Damit wird der Rahmen abgesteckt, in dem von dieser Öffnungsklausel Gebrauch gemacht werden kann. Nationale Bestimmungen zur Verarbeitung personenbezogener Daten im Beschäftigungskontext und insbesondere durch die erwähnten Überwachungssystemen am Arbeitsplatz sind damit grundsätzlich beschränkt und müssen im Einklang mit den allgemeinen Vorgaben der DSGVO und mit dem in Art. 88 Abs. 2 DSGVO vorgegebenen Schutzrahmen sein. Man spricht dabei auch von einem sogenannten *vollharmonisierenden* Ansatz [FGO18, Abschnitt DS-GVO Art. 88 Rn. 8].

Es ist also davon auszugehen, dass die Rechtmäßigkeit von Erkennungs- und Abwehrmaßnahmen von Insiderbedrohungen, die faktisch als Überwachungssysteme am Arbeitsplatz aufgefasst werden könnten (s. Abschnitt 6.1.1.5), auch an der Existenz derartiger geeigneter und besonderer Maßnahmen sowie an der Vereinbarkeit mit den allgemeinen Vorgaben der DSGVO gemessen wird. Derartige oder ähnliche Vorlagefragen zu Art. 88 Abs. 2 sowie dessen weite oder strenge Auslegung wurden allerdings noch nicht vom Europäischen Gerichtshof (EuGH) bearbeitet [FGO18, Abschnitt DS-GVO Art. 88 Rn. 12].

6.1.1.5 Mitbestimmungsrecht der Interessensvertretungen

Ein gegebenenfalls vorhandener Betriebsrat hat als Vertretung der Arbeitnehmerschaft nach § 87 Abs. 1 Nr. 6 Betriebsverfassungsgesetz (BetrVG) ein Mitbestimmungsrecht bei der

> „Einführung und Anwendung von technischen Einrichtungen, die dazu bestimmt sind, das Verhalten oder die Leistung der Arbeitnehmer zu überwachen.“

Diesbezüglich hat das Bundesarbeitsgericht (BAG) in einem Beschluss vom 09. September 1975 den uneindeutigen Wortlaut einer *Bestimmung zur Überwachung* spezifiziert:[9]

> „Eine technische Einrichtung i.S. des § 87 Abs. 1 Nr. 6 BetrVG ist dann dazu bestimmt, das Verhalten oder die Leistung der Arbeitnehmer zu überwachen, wenn die Einrichtung zur Überwachung objektiv und unmittelbar geeignet ist, ohne Rücksicht darauf, ob der Arbeitgeber dieses Ziel verfolgt und die durch die Überwachung gewonnenen Daten auch auswertet.“

Insoweit wird die Einrichtung und Anwendung einer Technik zur Erkennung und Abwehr von Insiderbedrohungen unter das Mitbestimmungsrecht eines Betriebsrates fallen, sofern, wie das BAG weiter ausführte, nicht erst noch „zusätzliche anderweitige Anordnungen oder bestimmte Gestaltungen“ für eine solche Überwachung umgesetzt werden müssen. Dem BetrVG entsprechende Mitbestimmungsrechte existieren ebenfalls für den öffentlichen Dienst der Länder, die jeweils in den Personalvertretungsgesetzen festgeschrieben sind.[10]

9. BAG, Beschluss vom 09. September 1975 – 1 ABR 20/74.
10. Vergleiche beispielsweise § 88 Abs. 1 Nr. 32 Hamburgisches Personalvertretungsgesetz (HmbPersVG).

6.1.2 Rechte und Pflichten von Arbeitgebern

Die DSGVO setzt der Verarbeitung personenbezogener Daten, wie sie im Zuge der Erkennung und Abwehr von Insiderbedrohungen stattfinden kann, enge Grenzen. Nach Art. 6 Abs. 1 Satz 1 DSGVO gilt das Verbotsprinzip mit Erlaubnisvorbehalt. Danach ist eine solche Verarbeitung nur zulässig, sofern die DSGVO oder eine andere Rechtsvorschrift dies erlaubt oder anordnet oder sofern der Betroffene eingewilligt hat. Die Rechtmäßigkeit einer Verarbeitung wird in Art. 6 Abs. 1 Satz 1 auf sechs Punkte zurückgeführt, von denen mindestens einer erfüllt sein muss:

a) Einwilligung der betroffenen Person für bestimmte Zwecke,

b) Erfüllung eines Vertrages oder Durchführung vorvertraglicher Maßnahmen,

c) Erfüllung einer rechtlichen Verpflichtung,

d) Schützen von lebenswichtigen Interessen der betroffenen oder einer anderen Person,

e) Wahrnehmung einer im öffentlichen Interesse liegenden oder in Ausübung öffentlicher Gewalt erfolgenden Aufgabe oder

f) Wahrung berechtigter Interessen der verarbeiteten Stelle oder eines Dritten, sofern nicht die Interessen und Rechte beziehungsweise Freiheiten der betroffenen Person überwiegen.

Es ist somit sehr kontextabhängig, ob eine Verarbeitung rechtmäßig ist oder nicht. Auf die wichtigsten Punkte für den vorliegenden Kontext, nämlich die *Einwilligung der betroffenen Person*, die *Erfüllung einer rechtlichen Verpflichtung* sowie die *Wahrung berechtigter Interessen* wird in den folgenden Abschnitten 6.1.2.1 bis 6.1.2.3 näher eingegangen. Weiter heißt es in Art. 6 Abs. 2 DSGVO, dass Mitgliedstaaten

> „[…] spezifische Anforderungen für die Verarbeitung sowie sonstige Maßnahmen präziser bestimmen [können], um eine rechtmäßig und nach Treu und Glauben erfolgende Verarbeitung zu gewährleisten, einschließlich für andere besondere Verarbeitungssituationen […].“

Zu den *besonderen Verarbeitungssituationen* gehört die für diese Dissertation relevante Datenverarbeitung im Beschäftigungskontext, die von der DSGVO in Art. 88 mit einer Öffnungsklausel für spezifischere nationale Bestimmungen behandelt wird. Abschnitt 6.1.2.4 geht auf diese besondere Verarbeitungssituation sowie die nationalen Bestimmungen näher ein und benennt die Rechte für Arbeitgeber, die daraus entstehen.

6.1.2.1 Einwilligung der betroffenen Person

Die Verarbeitung personenbezogener Daten ist laut Art. 6 Abs. 1a DSGVO rechtmäßig, wenn die betroffene Person

> „ihre Einwilligung zu der Verarbeitung der sie betreffenden personenbezogenen Daten für einen oder mehrere bestimmte Zwecke gegeben“

hat. Diese Einwilligung in die Verarbeitung personenbezogener Daten ist ein zentraler Erlaubnistatbestand und entspricht der Zulässigkeitsvoraussetzung für eine rechtmäßige Datenverarbeitung nach Art. 8 Abs. 2 der GRCh (vgl. [FGO18, Abschnitt DS-GVO Art. 6 Rn. 4]). Wichtig zu beachten ist dabei allerdings, dass die Einwilligung für einen oder mehrere Zwecke gegeben sein muss und dass eine Einwilligung nach Art. 7 Abs. 3 DSGVO jederzeit widerrufen werden kann.

Mit der Bestimmung des Begriffs *Einwilligung* in Art. 4 Nr. 11 DSGVO wird deutlich, dass es für eine wirksame Einwilligung auf eine freiwillige sowie unmissverständliche Willensbekundung ankommt. *Unmissverständlich* ist hier zunächst nicht mit einer *expliziten* Willensbekundung gleichzusetzen. Dennoch hat der Verantwortliche die Pflicht, die Einwilligung nachweisen zu können (vgl. Art. 7 Abs. 1 sowie Erwägungsgrund 42 Satz 1 DSGVO). Somit kann ein Schweigen der betroffenen Person oder ein fehlender Widerspruch gegen die Datenverarbeitung nicht als Einwilligungserklärung gedeutet werden (vgl. [FGO18, Abschnitt DS-GVO Art. 4 Rn. 19] und Erwägungsgrund 32 Satz 3 DSGVO). Etwas strikter verhält es sich mit der Einwilligung zur automatisierten Entscheidung einschließlich Profiling (vgl. Art. 22 Abs. 2c DSGVO), welches im vorliegenden Kontext hohe Relevanz hat, sowie zur Verarbeitung besonderer Kategorien personenbezogener Daten (vgl. Art. 9 Abs. 2a DSGVO). Zu Letzterem gehören etwa genetische, biometrische oder Gesundheitsdaten. Eine jeweils diesbezügliche Einwilligung muss tatsächlich *explizit* erfolgen, das heißt sie muss sich nach der Auffassung von Franzen, Gallner und Oetker [FGO18, Abschnitt DS-GVO Art. 9 Rn. 7] auf die Verarbeitung der konkret benannten besonders geschützten Daten beziehen und schließt eine konkludente[11] Einwilligung aus.

Bei der Beurteilung, ob eine Einwilligung *freiwillig* abgegeben wurde, müssen die Ausführungen in Art. 7 Abs. 4 DSGVO sowie Erwägungsgrund 43 Satz 2 DSGVO beachtet werden. Darin wird ein sogenanntes *Koppelungsverbot* festgeschrieben [FGO18, Abschnitt DS-GVO Art. 7 Rn. 9], mit dem eine Einwilligung in die Verarbeitung personenbezogener Daten als nicht freiwillig gelten kann, sofern die Einwilligung als Bedingung für einen Vertragsabschluss verlangt wird und die überlassenen Daten für die Erfüllung des Vertrages nicht erforderlich sind. Weiterhin führt Erwägungsgrund 43 Satz 1 DSGVO aus, dass eine Einwilligung nicht freiwillig gegeben werden kann, sofern zwischen dem Betroffenen und dem Verantwortlichen ein klares Ungleichgewicht besteht.

Dennoch lässt sich damit nicht darauf schließen, dass eine Einwilligung im Sinne der DSGVO im Beschäftigungskontext aufgrund des Ungleichgewichts zwischen dem Arbeitgeber und dem Arbeitnehmer ausgeschlossen ist [FGO18, Abschnitt DS-GVO Art. 7 Rn. 10]. Das ergibt sich aus der Tatsache, dass der Erwägungsgrund 155 DSGVO den Mitgliedstaaten explizit die Möglichkeit einräumt,

> „Vorschriften über die Bedingungen, unter denen personenbezogene Daten im Beschäftigungskontext *auf der Grundlage der Einwilligung* des Beschäftigten verarbeitet werden dürfen,“

vorzusehen. Im BDSG wurde von dieser Möglichkeit Gebrauch gemacht und eine Einwilligung in die Verarbeitung personenbezogener Beschäftigtendaten in § 26 Abs. 2 als Erlaubnistatbestand eingeräumt. Laut Satz 2 dieses Absatzes kann Freiwilligkeit insbesondere dann vorliegen,

11. Als *konkludent* wird ein Verhalten des Betroffenen bezeichnet, dass aus Sicht des Verantwortlichen indirekt auf eine Einwilligung schließen lässt [Kro16].

> „wenn für die beschäftigte Person ein rechtlicher oder wirtschaftlicher Vorteil erreicht wird oder Arbeitgeber und beschäftigte Person gleichgelagerte Interessen verfolgen."

6.1.2.2 Erfüllung einer rechtlichen Verpflichtung

Die Verarbeitung personenbezogener Daten ist laut Art. 6 Abs. 1c DSGVO rechtmäßig, wenn sie

> „zur Erfüllung einer rechtlichen Verpflichtung erforderlich [ist], der der Verantwortliche unterliegt."

Dieser sehr weit gefasste Rahmen, der laut Erwägungsgrund 41 Satz 1 DSGVO „nicht notwendigerweise einen von einem Parlament angenommenen Gesetzgebungsakt" verlangt, also auch die Rechtsprechung, Betriebsvereinbarungen, Tarifverträge u.ä. einschließt [FGO18, Abschnitt DS-GVO Art. 6 Rn.6, 7], muss nach Art. 6 Abs. 2 DSGVO in Verbindung mit Erwägungsgrund 45 Satz 1 DSGVO von den Mitgliedstaaten durch Unionsrecht oder durch nationale Bestimmungen spezifiziert werden. Dabei sollen besondere Verarbeitungssituationen, wie die von Beschäftigtendaten im Beschäftigungskontext gemäß Art. 88 DSGVO, eingeschlossen werden (s. Abschnitt 6.1.2.4).

Exemplarisch werden an dieser Stelle zwei rechtliche Verpflichtungen in Deutschland beleuchtet, die jeweils einen starken Bezug zur Erkennung und Abwehr von Insiderbedrohungen haben. Zum einen sind das unternehmensinterne Maßnahmen, die von bestimmten Unternehmen zur Einhaltung einer *Legalitätspflicht* umgesetzt werden sollen und mit dem Begriff *Compliance* zusammengefasst werden. Zum anderen ist das die besonders im Kontext kritischer Infrastrukturen auferlegte Pflicht, organisatorische und technische Maßnahmen zum Schutz der Informationstechnik umzusetzen. Eine Vielzahl an weiteren rechtlichen Verpflichtungen liegt vor, auf die im Rahmen dieser Dissertation nicht weiter eingegangen wird.

Compliance

Der Begriff *Compliance* meint „die Gesamtheit aller wirtschaftlich zumutbaren Maßnahmen, mit denen ein Unternehmen die Einhaltung gesetzlicher Ge- und Verbote durch Organmitglieder und Mitarbeiter sicherstellt" [Kam09]. Die Compliance ist beispielsweise laut § 25a Kreditwesengesetz (KWG) und §32 Wertpapierhandelsgesetz (WpHG) für den Finanzsektor rechtlich vorgeschrieben und laut Ziff. 4.1.3 im Deutschen Corporate Governance Kodex (DCGK)[12] für börsennotierte Gesellschaften vorgegeben. Sie lässt sich weiterhin aus der *Sorgfaltspflicht* ableiten, wie sie beispielsweise das Aktienrecht im Aktiengesetz (AktG) § 93 Abs. 1 Satz 1, das Gesellschaftsrecht im Gesetz betreffend die Gesellschaften mit beschränkter Haftung (GmbHG) § 43 Abs. 1 sowie das Genossenschaftsrecht im Gesetz betreffend die Erwerbs- und Wirtschaftsgenossenschaften (GenG) § 34 Abs. 1 Satz 1 für Vorstandsmitglieder und Geschäftsführer

12. Bundesanzeiger Amtlicher Teil 24. April 2017 B2: Bekanntmachung des „Deutschen Corporate Governance Kodex" (in der Fassung vom 7. Februar 2017).

vorschreibt.[13] Die Sorgfaltspflicht unterteilt sich in die Pflicht, im Einklang mit der Rechtsordnung zu handeln (*Legalitätspflicht*) und in die Pflicht, die Einhaltung der Rechtsordnung zu überwachen (*Überwachungspflicht*) [Kam09]. Motiviert wird sie durch die Abwehr von rechtlichen sowie von faktischen Folgen [Thü+14, Abschnitt 2 Rn. 20–34]. Eine rechtliche Folge ist beispielsweise eine drohende Ordnungswidrigkeit nach § 130 im Gesetz über Ordnungswidrigkeiten (OWiG) bei der Verletzung der Aufsichtspflicht in Betrieben und Unternehmen oder die Haftung des Vorstandes gegenüber der Gesellschaft (vgl. § 93 Abs. 2 AktG und § 43 Abs. 2 GmbHG). Faktische Folgen umfassen wirtschaftliche Auswirkungen aufgrund des Bekanntwerdens von Pflichtverletzungen und dem damit einhergehenden potenziellen Reputationsverlust.

Sicherheit in der Informationstechnik Kritischer Infrastrukturen

Zu Kritischen Infrastrukturen gehören aktuell die Sektoren Energie, Wasser, Ernährung, Informationstechnik und Telekommunikation, Gesundheit, Finanz- und Versicherungswesen sowie Transport und Verkehr.[14] Im Juli 2015 wurde vom deutschen Bundestag das Gesetz zur Erhöhung der Sicherheit informationstechnischer Systeme (IT-Sicherheitsgesetz)[15] beschlossen. Damit wurde eine Reihe von Gesetzen dahingehend geändert, dass die IT-Systeme und digitalen Infrastrukturen, einschließlich der Kritischen Infrastrukturen, besser vor Bedrohungen geschützt und Sicherheitsvorfälle an bestimmte Stellen kommuniziert werden. Eine dieser Änderungen betrifft das Gesetz über das Bundesamt für Sicherheit in der Informationstechnik (BSIG), in dem unter anderem der neue § 8a eingefügt wurde. In dessen Abs. 1 heißt es:

> „Betreiber Kritischer Infrastrukturen sind verpflichtet, [...] angemessene organisatorische und technische Vorkehrungen zur Vermeidung von Störungen der Verfügbarkeit, Integrität, Authentizität und Vertraulichkeit ihrer informationstechnischen Systeme, Komponenten oder Prozesse zu treffen, die für die Funktionsfähigkeit der von ihnen betriebenen Kritischen Infrastrukturen maßgeblich sind. Dabei soll der Stand der Technik eingehalten werden. Organisatorische und technische Vorkehrungen sind angemessen, wenn der dafür erforderliche Aufwand nicht außer Verhältnis zu den Folgen eines Ausfalls oder einer Beeinträchtigung der betroffenen Kritischen Infrastruktur steht."

Diese rechtliche Verpflichtung könnte dem Wortlaut nach Vorkehrungen zur Erkennung und Abwehr von Bedrohungen im Allgemeinen und von Insiderbedrohungen im Speziellen umfassen. Bisher besteht dazu allerdings Rechtsunsicherheit, da derartige Verfahren keine ausdrückliche Erwähnung im Gesetz oder der Rechtsprechung finden [KS19]. Die Situation ändert sich allerdings mit dem im April 2019 öffentlich gewordenen Entwurf zum IT-Sicherheitsgesetz 2.0 in

13. Von der rechtlich vorgeschriebenen Sorgfaltspflicht auf eine dadurch ebenfalls rechtlich vorgeschriebene Compliance, also eine Umsetzung konkreter Methoden, Maßnahmen und Prozesse, zu schließen ist allerdings umstritten. Während Thüsing u. a. [Thü+14, Abschnitt § 2 Rn. 3 ff.] die Compliance-Pflicht im Einzelnen herleiten, argumentieren Fleischer und Goette [FG15, Abschnitt § 43 Rn. 145] besonders bei kleinen Gesellschaften für eine gewisse Zurückhaltung.
14. Bundesgesetzblatt Jahrgang 2016 Teil I Nr. 20 (S. 958): Verordnung zur Bestimmung Kritischer Infrastrukturen nach dem BSI-Gesetz (BSI-Kritisverordnung) vom 22. April 2016. In Verbindung mit Bundesgesetzblatt Jahrgang 2017 Teil I Nr. 40 (S. 1903): Erste Verordnung zur Änderung der BSI-Kritisverordnung vom 21. Juni 2017.
15. Bundesgesetzblatt Jahrgang 2015 Teil I Nr. 31 (S. 1324): Gesetz zur Erhöhung der Sicherheit informationstechnischer Systeme (IT-Sicherheitsgesetz) vom 24. Juli 2015.

der Fassung vom 27.03.2019,[16] mit dem weitere Änderungen an bestehenden Vorschriften vorgeschlagen werden. Unter anderem sollen Vorkehrungen zur Angriffserkennung als Bestandteil einer effektiven Cyberabwehrstrategie im Gesetz aufgeführt werden. Darüber hinaus enthält der Entwurf Vorgaben zur Datenverarbeitung sowie zu Aufbewahrungs- und Löschfristen. Dem Entwurf nach wird in § 8a BSIG nach Abs. 1 ein neuer Abs. 1a eingefügt, in dem beschrieben wird, dass die technischen Vorkehrungen

> „auch den Einsatz von Systemen zur Angriffserkennung [umfassen]. Die Betreiber Kritischer Infrastrukturen dürfen die hierzu erforderlichen Daten verarbeiten. Die im Rahmen des Einsatzes von Systemen zur Angriffserkennung erhobenen Daten sind unverzüglich zu löschen, wenn sie nicht für die Vermeidung von Störungen nach Absatz 1 Satz 1 erforderlich sind. Die übrigen Daten dürfen nicht länger als zehn Jahre gespeichert werden. Die Ausgestaltung des Einsatzes von Systemen zur Angriffserkennung legt das Bundesamt in einer Technischen Richtlinie fest."

Es bleibt somit abzuwarten, wie die Ausgestaltung des Einsatzes von Systemen zur Bedrohungserkennung im Detail aussehen wird.

6.1.2.3 Wahrung berechtigter Interessen

Die Verarbeitung personenbezogener Daten ist laut Art. 6 Abs. 1f DSGVO rechtmäßig, wenn sie

> „zur Wahrung der berechtigten Interessen des Verantwortlichen oder eines Dritten erforderlich [ist], sofern nicht die Interessen oder Grundrechte und Grundfreiheiten der betroffenen Person, die den Schutz personenbezogener Daten erfordern, überwiegen, insbesondere dann, wenn es sich bei der betroffenen Person um ein Kind handelt."

Damit ist bei diesem Erlaubnistatbestand stets eine Abwägung der berechtigten Interessen des Verantwortlichen gegenüber den Interessen und Grundrechten der betroffenen Person erforderlich. Die Interessen und Grundrechte der betroffenen Person wurden bereits in Abschnitt 6.1.1 herausgearbeitet. Von Seiten der verarbeitenden Stelle liefert Erwägungsgrund 47 DSGVO einige Beispiele berechtigter Interessen. In Satz 2 und Satz 6 heißt es:

> „[2]Ein berechtigtes Interesse könnte beispielsweise vorliegen, wenn eine maßgebliche und angemessene Beziehung zwischen der betroffenen Person und dem Verantwortlichen besteht, zum Beispiel wenn die betroffene Person ein Kunde des Verantwortlichen ist oder in seinen Diensten steht. [...] [6]Die Verarbeitung personenbezogener Daten im für die Verhinderung von Betrug unbedingt erforderlichen Umfang stellt ebenfalls ein berechtigtes Interesse des jeweiligen Verantwortlichen dar."

Mit Satz 2 ist somit auch eine Verarbeitung personenbezogener Daten im Sinne der DSGVO erlaubt, wenn zwischen dem Betroffenen und der datenverarbeitenden Stelle eine vertragliche Beziehung besteht [FGO18, Abschnitt DS-GVO Art. 6 Rn. 10]. Satz 6 legitimiert das berechtigte Interesse der Verhinderung von Schädigungen. Letzteres wurde durch die Rechtsprechung bereits

16. Der vollständige Gesetzentwurf ist abrufbar unter http://intrapol.org/wp-content/uploads/2019/04/IT-Sicherheitsgesetz-2.0-_-IT-SiG-2.0.pdf.

bestätigt. Das Verwaltungsgericht Lüneburg hat in seinem Teilurteil[17] zur datenschutzrechtlichen Zulässigkeit eines Ortungssystems im Beschäftigungskontext vom 19.03.2019 in Rn. 33 festgehalten, „dass dem Arbeitgeber [bei der Wahrnehmung seiner gesetzlichen oder vertraglichen Rechte] die nach Art. 12 GG verbriefte unternehmerische Freiheit zusteht, zu entscheiden, wie er seinen Betrieb organisiert“. In Art. 12 Abs. 1 GG heißt es wörtlich:

> „Alle Deutschen haben das Recht, Beruf, Arbeitsplatz und Ausbildungsstätte frei zu wählen. Die Berufsausübung kann durch Gesetz oder auf Grund eines Gesetzes geregelt werden.“

Diese im GG verankerte Berufsfreiheit findet sich in ähnlicher Weise auch in den Art. 15 Abs. 1 und Art. 16 der GRCh. Dort heißt es:

> „Art. 15 Berufsfreiheit und Recht zu arbeiten
>
> (1) Jede Person hat das Recht, zu arbeiten und einen frei gewählten oder angenommenen Beruf auszuüben.
>
> [...]
>
> Art. 16 Unternehmerische Freiheit
>
> Die unternehmerische Freiheit wird nach dem Unionsrecht und den einzelstaatlichen Rechtsvorschriften und Gepflogenheiten anerkannt.“

Ein Präzedenzfall diesbezüglich ist derzeit zwar nicht bekannt, aber das Recht auf freie Wahl und Ausübung des Berufs kann bedeuten, dass Verarbeitungsschranken von personenbezogenen Daten nur so weit gehen können, dass sie nicht die Fortführung eines Unternehmens gefährden [Däu17, Kapitel 3 Rn. 117]. Aus Arbeitgebersicht würde ansonsten die Berufswahlfreiheit verletzt werden. Insofern können Maßnahmen zur Erkennung und Abwehr von Insiderbedrohungen unter diesen Umständen als Wahrung berechtigter Interessen aufgefasst werden.

6.1.2.4 Datenverarbeitung im Beschäftigungskontext

Wie in Abschnitt 6.1.1.4 bereits ausgeführt, wird die Verarbeitung personenbezogener Beschäftigtendaten im Beschäftigungskontext in der DSGVO als besondere Verarbeitungssituation gehandhabt und in Art. 88 Abs. 1 DSGVO mit einer Öffnungsklausel versehen, sodass Mitgliedstaaten

> „durch Rechtsvorschriften oder durch Kollektivvereinbarungen spezifischere Vorschriften zur Gewährleistung des Schutzes der Rechte und Freiheiten hinsichtlich der Verarbeitung personenbezogener Beschäftigtendaten im Beschäftigungskontext [...]“

vorsehen können. Von dieser Öffnungsklausel wurde in § 26 BDSG Gebrauch gemacht. Dort heißt es in Abs. 1:

17. VG Lüneburg 4. Kammer, Teilurteil vom 19.03.2019, 4 A 12/19.

> „[1]Personenbezogene Daten von Beschäftigten dürfen für Zwecke des Beschäftigungsverhältnisses verarbeitet werden, wenn dies für die Entscheidung über die Begründung eines Beschäftigungsverhältnisses oder nach Begründung des Beschäftigungsverhältnisses für dessen Durchführung oder Beendigung oder zur Ausübung oder Erfüllung der sich aus einem Gesetz oder einem Tarifvertrag, einer Betriebs- oder Dienstvereinbarung (Kollektivvereinbarung) ergebenden Rechte und Pflichten der Interessenvertretung der Beschäftigten erforderlich ist. [2]Zur Aufdeckung von Straftaten dürfen personenbezogene Daten von Beschäftigten nur dann verarbeitet werden, wenn zu dokumentierende tatsächliche Anhaltspunkte den Verdacht begründen, dass die betroffene Person im Beschäftigungsverhältnis eine Straftat begangen hat, die Verarbeitung zur Aufdeckung erforderlich ist und das schutzwürdige Interesse der oder des Beschäftigten an dem Ausschluss der Verarbeitung nicht überwiegt, insbesondere Art und Ausmaß im Hinblick auf den Anlass nicht unverhältnismäßig sind."

Demnach dürfen personenbezogene Beschäftigtendaten im Beschäftigungskontext verarbeitet werden, sofern dies für alle zeitlichen Etappen eines Beschäftigungsverhältnisses oder in begründeten Verdachtsfällen zur Aufdeckung von bereits begangenen Straftaten erforderlich ist. Erwähnenswert für den vorliegenden Kontext ist auch der letzte Teil von Satz 1, mit dem Interessensvertretungen der Beschäftigten zur Verarbeitung bemächtigt werden (vgl. auch § 26 Abs. 4 BDSG). Für den Fall einer Insiderbedrohungserkennung und -prävention könnten somit auch die Interessensvertretungen rechtlich in die Pflicht zur Durchführung derartiger Maßnahmen oder zur Beteiligung an derartigen Maßnahmen genommen werden. Die Verarbeitung personenbezogener Daten zur Aufdeckung von Straftaten ist allerdings abzugrenzen von der Verhinderung von Straftaten durch Maßnahmen der Bedrohungserkennung und -prävention. Letzteres lässt sich nicht durch § 26 Abs. 1 Satz 2 BDSG rechtfertigen, sondern beurteilt sich Riesenhuber [Rie19, Kapitel BDSG § 26 Rn. 138] zufolge nach § 26 Abs. 1 Satz 1 BDSG. Es gehört schließlich zur Durchführung des Arbeitsverhältnisses, dass der Arbeitgeber Maßnahmen zur Verhinderung von Pflichtverletzungen einsetzen muss. Zu dieser Auffassung kommt auch das BAG in seinem Urteil vom 17. November 2016 bezüglich einer Pflicht zur Teilnahme an einem elektronischen Warn- und Berichtssystem und dem davon berührten Arbeitnehmerdatenschutz.[18]

§ 26 Abs. 2 BDSG spezifiziert den in Abschnitt 6.1.2.1 bereits eingehend behandelten Erlaubnistatbestand der *Einwilligung*:

> „[1]Erfolgt die Verarbeitung personenbezogener Daten von Beschäftigten auf der Grundlage einer Einwilligung, so sind für die Beurteilung der Freiwilligkeit der Einwilligung insbesondere die im Beschäftigungsverhältnis bestehende Abhängigkeit der beschäftigten Person sowie die Umstände, unter denen die Einwilligung erteilt worden ist, zu berücksichtigen. [2]Freiwilligkeit kann insbesondere vorliegen, wenn für die beschäftigte Person ein rechtlicher oder wirtschaftlicher Vorteil erreicht wird oder Arbeitgeber und beschäftigte Person gleichgelagerte Interessen verfolgen. [3]Die Einwilligung bedarf der Schriftform, soweit nicht wegen besonderer Umstände eine andere Form angemessen ist. [4]Der Arbeitgeber hat die beschäftigte Person über den Zweck der Datenverarbeitung und über ihr Widerrufsrecht nach Artikel 7 Absatz 3 der Verordnung (EU) 2016/679 in Textform aufzuklären."

18. BAG, Urteil vom 17. November 2016 – 2 AZR 730/15, Rn. 28–35.

Eine Einwilligung in die Verarbeitung personenbezogener Daten im Zuge der Erkennung und Abwehr von Insiderbedrohungen ist also grundsätzlich möglich. Allerdings heißt es in der Gesetzesbegründung[19] zu den in Satz 1 genannten Umständen:

> „Neben der Art des verarbeiteten Datums und der Eingriffstiefe ist zum Beispiel auch der Zeitpunkt der Einwilligungserteilung maßgebend. Vor Abschluss eines (Arbeits-)Vertrages werden Beschäftigte regelmäßig einer größeren Drucksituation ausgesetzt sein, eine Einwilligung in eine Datenverarbeitung zu erteilen."

Demnach ist eine Einwilligung in die Verarbeitung personenbezogener Daten bei Vertragsabschluss abzulehnen. Mit § 26 Abs. 3 BDSG wurde Art. 9 Abs. 2b der DSGVO umgesetzt. Dieser erlaubt die Verarbeitung besonderer Kategorien personenbezogener Daten (vgl. Art. 9 Abs. 1 DSGVO) bei einer ausdrücklichen Einwilligung der Beschäftigten oder „zur Ausübung von Rechten oder zur Erfüllung rechtlicher Pflichten aus dem Arbeitsrecht, dem Recht der sozialen Sicherheit und des Sozialschutzes". In § 26 Abs. 5 heißt es:

> „Der Verantwortliche muss geeignete Maßnahmen ergreifen, um sicherzustellen, dass insbesondere die in Artikel 5 der Verordnung (EU) 2016/679 dargelegten Grundsätze für die Verarbeitung personenbezogener Daten eingehalten werden."

Auf die Beteiligungsrechte der Interessenvertretungen der Beschäftigten, die nach § 26 Abs. 6 BDSG unberührt bleiben, wurde in Abschnitt 6.1.1.5 bereits näher eingegangen.

6.1.2.5 Profiling

Die detaillierte Erfassung, Analyse und Bewertung von Nutzeraktivitäten, bei denen personenbezogene Daten anfallen und verarbeitet werden, um Insiderbedrohungen erkennen und verhindern zu können, wird durch die DSGVO nicht grundsätzlich verboten [Däu17, Kapitel 3 Rn. 134e f.]. Die VO spricht dabei von einem *Profiling* und definiert es in Art. 4 Nr. 4 genauer. Danach bezeichnet der Ausdruck

> „*Profiling* jede Art der automatisierten Verarbeitung personenbezogener Daten, die darin besteht, dass diese personenbezogenen Daten verwendet werden, um bestimmte persönliche Aspekte, die sich auf eine natürliche Person beziehen, zu bewerten, insbesondere um Aspekte bezüglich Arbeitsleistung, wirtschaftliche Lage, Gesundheit, persönliche Vorlieben, Interessen, Zuverlässigkeit, Verhalten, Aufenthaltsort oder Ortswechsel dieser natürlichen Person zu analysieren oder vorherzusagen."

Sofern also die Verarbeitung personenbezogener Daten nach den Vorgaben der DSGVO zulässig ist, ist auch ein Profiling einschließlich einem Prognostizieren von künftigem Verhalten zulässig.

19. Deutscher Bundestag Drucksache 18/11325 vom 24. Februar 2017, S. 97.

6.1.3 Rechts- und Interessenskonflikte

Da beim Einsatz von Maßnahmen zur Erkennung und Abwehr von Insiderbedrohungen sowohl Grundrechte und Interessen des Arbeitnehmers (vgl. Abschnitt 6.1.1) als auch Grundrechte und Interessen sowie Pflichten des Arbeitgebers (vgl. Abschnitt 6.1.2) berührt werden, gilt es diese Rechts- und Interessenskonflikte durch entsprechende Abwägungen aufzulösen. Dies geht auch aus Erwägungsgrund 4 der DSGVO hervor:

> „[1]Die Verarbeitung personenbezogener Daten sollte im Dienste der Menschheit stehen. [2]Das Recht auf Schutz der personenbezogenen Daten ist kein uneingeschränktes Recht; es muss im Hinblick auf seine gesellschaftliche Funktion gesehen und unter Wahrung des Verhältnismäßigkeitsprinzips gegen andere Grundrechte abgewogen werden. [3]Diese Verordnung steht im Einklang mit allen Grundrechten und achtet alle Freiheiten und Grundsätze, die mit der Charta anerkannt wurden und in den Europäischen Verträgen verankert sind, insbesondere [...] Schutz personenbezogener Daten, [...] unternehmerische Freiheit [...]."

Das in Satz 2 angesprochene *Verhältnismäßigkeitsprinzip* wurde bereits von der Rechtsprechung zugrunde gelegt. Im Zuge eines Rechtsstreits wurde geprüft, ob die Verarbeitung personenbezogener Daten zur Erfüllung eines oder mehrerer spezifischer Zwecke als *erforderlich* sowie diesem Zweck auf der einen Seite und der Eingriffstiefe in die Privatsphäre und das Recht auf Datenschutz auf der anderen Seite als *angemessen* bewertet werden kann.[20] Der Einschätzung von Däubler [Däu17, Kapitel 2 Rn. 49b] zufolge gilt diese *Erforderlichkeit* für alle Erlaubnistatbestände der Verarbeitung personenbezogener Daten im Sinne der DSGVO mit Ausnahme der *Einwilligung* (vgl. Abschnitt 6.1.2.1).

Aus Sicht des Arbeitgebers ergibt sich zusammenfassend ein schwieriges Bild bei der Umsetzung von Maßnahmen zur Insiderbedrohungserkennung und -prävention. Einerseits drohen Pflichtverletzungen, sofern eine Abwehr von Schäden nicht umgesetzt wird. Andererseits drohen unrechtmäßige Eingriffe in das Recht auf Datenschutz gegenüber den Arbeitnehmern. Sofern dieser Konflikt nicht eindeutig aufzulösen ist und Rechtsunsicherheit besteht, „ist der Rechtsunterworfene nach dem Grundsatz der rechtfertigenden Pflichtenkollision von jedem Vorwurf freizusprechen, wenn er sich nach sorgfältiger Abwägung für eine der Handlungsvarianten entschieden hat" [Thü+14, Kapitel 2 Rn. 4].

6.2 Gegenläufige Anforderungen

Die Anforderungen, die sich an einen datenschutzfreundlichen Mechanismus zur Erkennung und Abwehr von Insiderbedrohungen stellen, lassen sich aus zwei unterschiedlichen zumeist entgegengesetzten Perspektiven betrachten. Zum einen ist das die Perspektive der Domäne und zum anderen die der von der Überwachung betroffenen Insider.

20. Vergleiche BAG, Urteil vom 17. November 2016 – 2 AZR 730/15, Rn. 28–35.

6.2.1 Anforderungen aus Sicht der Domäne

Im Fokus einer Domäne stehen die folgenden Anforderungen an Mechanismen zur Erkennung und Abwehr von Insiderbedrohungen, die den Datenschutzanforderungen der Betroffenen generell entgegenstehen.

6.2.1.1 Verkettbarkeit

Eine Bedrohungsaktion ist in der Regel nicht anhand eines einzigen Ereignisses erkennbar, sondern wird erst durch die Verkettung mehrerer Ereignisse als solche sichtbar. Beispielsweise ist ein fehlgeschlagener Loginversuch eines Mitarbeiters an seinem Nutzeraccount für sich genommen kein ungewöhnliches oder besorgniserregendes Ereignis. Sofern allerdings mehrere fehlgeschlagene Loginversuche innerhalb einer sehr kurzen Zeit demselben Mitarbeiteraccount zugeordnet werden können, ist von einem Angriff durch Ausprobieren beziehungsweise Bruteforce auszugehen. Auch die in Abschnitt 3.3.1 vorgestellte Angriffskette zeigt die häufig vielen Ereignisse, die zwar teilweise für sich genommen bereits als Angriff wahrnehmbar sind, die allerdings erst in ihrer Verkettung den gesamten Angriff widerspiegeln.

Verkettbarkeit mehrerer Ereignisse bedeutet, dass eine hinreichend genaue Unterscheidung getroffen werden kann, ob diese Ereignisse miteinander in Verbindung stehen oder nicht [PH10]. Sie bereitet den notwendigen Kontext, den es braucht, um individuelle Ereignisse zu verknüpfen und schafft somit die Basis für eine Korrelation und umfassende Interpretation dieser Ereignisse.

6.2.1.2 Globale Konsistenz

Die korrekte Verkettung und Auswertung von Ereignissen setzt voraus, dass Ereignisse von ein und demselben Urheber auch über Systemgrenzen hinweg und insbesondere in einem physisch verteilten System als solche erkennbar sind. Mit dieser *globalen Konsistenz* wird weiterhin sichergestellt, dass Ereignisse von verschiedenen Urhebern nicht fälschlicherweise kollidieren, also demselben Urheber zugerechnet werden. Andernfalls würden Analysen und Statistiken aufbauend auf gesammelten Daten und korrelierten Ereignissen verfälscht. Für diese Anforderung ist nicht zwingend eine Zuordnung zu einer natürlichen Person oder gar die Zurechenbarkeit zum Urheber notwendig (s. Abschnitt 6.2.1.4).

6.2.1.3 Digital Forensic Readiness

Sofern es technisch und finanziell möglich ist, sollten die erhobenen Ereignisse, idealerweise in ihrer Rohform oder wenigstens unter minimalem Informationsverlust, komprimiert und so lange wie möglich sowie in strukturierter Form gespeichert werden. Diese konsequente Vorbereitung einer potenziellen *Digitalen Forensik* (engl. Digital Forensic Readiness [Row04]) ist ein Schlüssel bei der Erkennung und Aufklärung vergangener Insiderbedrohungsaktionen. Dadurch können gut verdeckte Insiderbedrohungsaktionen mithilfe neuer Kenntnisse und Technologien auch lange nach deren Durchführung möglicherweise noch erkannt und aufgeklärt werden.

6.2.1.4 Attribution und Zurechenbarkeit

Sowohl die Erkennung von laufenden als auch die Rekonstruktion vergangener Bedrohungsaktionen basieren auf der Aufzeichnung und Auswertung von Ereignissen, die durch eine Vielzahl von Nutzern eines Systems oder einer Infrastruktur erzeugt werden. Dabei ist eine Zuordnung der Ereignisse zu ihren jeweiligen Urhebern von zentraler Bedeutung. Diese sogenannte *Attribution* [CL11] ist besonders im Kontext von Informationstechnologien sehr komplex.

Einerseits müssen ausreichend Informationen über die Urheberschaft eines Ereignisses zur Verfügung stehen und aufgezeichnet werden. Das stellt sich genau dann als schwierig heraus, wenn diese Informationen über mehrere Technologieebenen verknüpft werden müssen. Beispielsweise kann ein Ereignis einem Prozess zugeordnet werden, dieser Prozess muss einem Nutzeraccount zugeordnet werden, dieser Nutzeraccount muss einem Gerät zugeordnet werden und schließlich muss dieses Gerät noch einer Abteilung zugeordnet werden. Erst dann ist möglicherweise eine Attribution möglich. Andererseits kann die Urheberschaft eines Ereignisses verschleiert oder gefälscht werden, indem beispielsweise eine Kette von Urhebern geschaffen wird, die zu durchdringen technisch schwierig oder organisatorisch gar unmöglich ist. In diesem Zusammenhang braucht es für die belastbare Erkennung und Abwehr von Bedrohungsaktionen die *Zurechenbarkeit* [Pfi06], das heißt in diesem Fall eine fälschungssichere und beweisbare Attribution.

6.2.2 Anforderungen aus Sicht der Betroffenen

Die Betroffenen eines Mechanismus zur Bedrohungserkennung und -prävention befinden sich nicht nur in der Position, durch einen solchen Mechanismus unter Generalverdacht gestellt zu werden. Sie sind darüber hinaus auch einem potenziellen Missbrauch der anfallenden Daten und damit einem tiefen und oftmals illegalen Eingriff in die informationelle Selbstbestimmung ausgesetzt. Die Anforderungen aus Sicht der Betroffenen sind demnach prinzipiell entgegengesetzt zu denen der Domäne.

6.2.2.1 Schutz vor unrechtmäßiger Verarbeitung

Eine Domäne könnte neben dem berechtigten Interesse der Bedrohungserkennung und -prävention auch unrechtmäßige Ziele bei der Analyse und Auswertung der erhobenen Ereignisse verfolgen. Dazu zählen etwa eine ungenehmigte Leistungs- und Verhaltenskontrolle von Betroffenen [AD08] sowie das Tracken von logischen und physischen Bewegungen und Aktivitäten [Sol17].

Unter der Leistungs- und Verhaltenskontrolle (engl. employee performance monitoring [HLS16]) werden Maßnahmen zur Dokumentation von Arbeitsabläufen und zur Überwachung von Betroffenen zusammengefasst, die es einem Arbeitgeber erlauben, die Sorgfalt, Korrektheit und Produktivität der Mitarbeiter zu überprüfen und auszuwerten. Je nach Gerichtsbarkeit kann eine solche Überwachung ohne angemessene Verhältnismäßigkeit, konkretem Verdachtsfall, Einwilligung der Arbeitnehmervertretungen oder expliziter Einwilligung der Betroffenen unrechtens sein (vgl. Abschnitt 6.1).

Aus Sicht der Betroffenen müssen derartige unrechtmäßige Verarbeitungen verhindert werden. Dabei ist der Grat zwischen legitimer und unrechtmäßiger Verarbeitung allerdings sehr schmal

und letztendlich wohl nur in der *Intention* der Domäne unterscheidbar. Aufgrund der Überlegungen aus Abschnitt 3.2.1.3, in dem anhand von Beispielen aufgezeigt wird, dass die Bestimmung der *Intention* einer Aktivität, wenn überhaupt, in der Regel erst nach der Durchführung der Aktivität festgestellt werden kann, ist es also keine zielführende Lösung, die legitime Verarbeitung zu ermöglichen und diese im Falle einer Unrechtmäßigkeit zu verhindern. Vielmehr muss letzteres unabhängig von einer *Intention* von vornherein unmöglich sein. Eine Möglichkeit dafür ist eine äußerst umfassende und strikte *Datenminimierung* bis hin zu einer konsequenten sogenannten *De-Identifizierung*, auf die jeweils nachfolgend näher eingegangen wird.

6.2.2.2 Datenminimierung

Um die schweren Implikationen für die Privatsphäre der Betroffenen zu mindern, muss das Prinzip der Datenminimierung Anwendung und Durchsetzung finden. Das ergibt sich unmittelbar beispielsweise aus Art. 5 Abs. 1c der DSGVO:

> „Personenbezogene Daten müssen dem Zweck angemessen und erheblich sowie auf das für die Zwecke der Verarbeitung notwendige Maß beschränkt sein."

Dieses Prinzip wurde bereits von Gürses, Troncoso und Diaz [GTD11] diskutiert und von Hoepman [Hoe14] als Datenschutzdesignstrategie definiert:

> „The amount of personal data that is processed should be restricted to the minimal amount possible. [...] By ensuring that no, or no unnecessary, data is collected, the possible privacy impact of a system is limited. "

Ein effektiver Mechanismus zur Datenminimierung im Kontext der Bedrohungserkennung und -prävention ist *Select before you Collect* [Jac05]. Damit wird bereits im Design einer Technologie ausgeschlossen, dass gewisse Daten, die für die Erkennung und Abwehr von Insiderbedrohungen nicht unbedingt notwendig sind, deren Existenz wohl aber das Risiko eines Missbrauchs erhöht, überhaupt erst erhoben und verarbeitet werden.

6.2.2.3 De-Identifizierung

Der Prozess der Entfernung oder Verschleierung von personenbezogenen Daten (engl. Personal Identifiable Information (PII)) in gesammelten Daten, die gespeichert oder für eine weitere Verarbeitung verwendet werden, wird als *De-Identifizierung* bezeichnet [PTF16]. Auf die ebenfalls häufig anzutreffende Bezeichnung der *Anonymisierung* wird hier verzichtet, da sie in der Literatur sehr umstritten ist [Swe97; Ohm09, Abschnitt II.C.2]. Darüber hinaus ruft die Bezeichnung der Anonymisierung eine für den Kontext der vorliegenden Arbeit irreführende Erwartung eines absoluten Datenschutzes hervor. Geeignete Mechanismen, die beim Prozess einer De-Identifizierung zum Einsatz kommen können, sind die folgenden, die teilweise von Shabtai, Elovici und Rokach [SER12, Abschnitt 6.2] als elementare De-Identifizierungsoperationen bezeichnet werden.

- **Generalisierung:** Mithilfe der *Generalisierung* wird PII durch jeweils datenspezifisch verallgemeinerte Werte ersetzt [Swe02]. Dadurch können im Nachhinein die Datensätze verschiedener Einträge nicht mehr unterschieden werden, da die vormals unterscheidbaren und damit personenbezogenen Werte durch die Generalisierung potenziell identische

Ersetzungen erhalten und somit in einer übergeordneten Kategorie zusammenfallen. Die möglichen Kategorien bilden dabei eine natürliche baumartige Hierarchie. Beispielsweise können die genauen Geburtsdaten einzelner Personen in einem Datensatz mit den jeweiligen Geburtsmonaten, Geburtsjahren oder Geburtsjahrzehnten ersetzt werden. Der Grad der Generalisierung reicht dabei in einem konkreten Datensatz von keiner Generalisierung, bei der alle Werte unverändert bleiben, bis hin zu einer maximalen Generalisierung, bei der alle Werte einer speziellen PII in eine Kategorie fallen.

- **Unterdrückung:** Die *Unterdrückung* bedeutet schlicht eine Entfernung der PII aus den vorhandenen Daten [Swe02]. Es ist somit äquivalent zu einer Generalisierung mit maximal möglichem Generalisierungsgrad einer PII. Es kann unterschieden werden zwischen der Unterdrückung einer Datenzeile, also dem Entfernen eines konkreten Ereignisses, der Unterdrückung einer Datenspalte, also dem Entfernen eines speziellen Datenfeldes in allen gesammelten Ereignissen, und der Unterdrückung einer speziellen Datenzelle, also dem Entfernen einzelner Datenfelder in einzelnen Ereignissen.
- **Permutation und Aggregation:** Bei der *Permutation* werden einzelne Ereignisse in Gruppen aufgeteilt und innerhalb dieser Gruppen die Werte einzelner ausgewählter Datenspalten permutiert. Dabei geht der Bezug der potenziell sensiblen Werte dieser Datenspalten zu den zugehörigen Identitäten verloren. Einzelne Datenzeilen bilden dann also einen verfälschten Zustand der gesammelten Ereignisse ab. Wird anschließend eine *Aggregation* der Werte dieser permutierten Datenspalte vorgenommen, bei der die Werte beispielsweise in einer Summe oder einem Mittelwert zusammengefasst werden, bleibt das Ergebnis dieser Aggregation allerdings korrekt und spiegelt den tatsächlichen (aggregierten) Zustand wider [Zha+07]. Auch die Aggregation allein ist ein probates Mittel zu De-Identifizierung insbesondere sensibler numerischer Daten, da diese Einzeldaten im Ergebnis der Berechnung nicht mehr isoliert werden können und somit ihren Bezug zu den zugehörigen Identitäten verlieren.
- **Verrauschen:** Das *Verrauschen* von Daten bezeichnet das Ersetzen der Originalwerte mit synthetisch erzeugten Werten. Die synthetischen Werte werden derart erzeugt, dass sich statistische Analysen der Daten vor und nach der Ersetzung nicht signifikant unterscheiden. Auch hier werden einzelne Ereignisse verfälscht. Dieses sorgfältig kalibrierte und zufällig applizierte Rauschen kann allerdings bei einer statistischen Analyse einer Vielzahl von Daten isoliert und entfernt werden, wobei anschließend nur noch eine vordefinierte Aggregation der Originalwerte vorliegt. Eine bekannte Umsetzung dieses Mechanismus' wurde von Dwork [Dwo06] mit *Differential Privacy* benannt.
- **Pseudonymisierung:** Die *Pseudonymisierung* ersetzt PII mit einer zufälligen oder pseudozufälligen Zeichenfolge, einem sogenannten Pseudonym, welches gegebenenfalls mithilfe einer geheim zu haltenden Vorschrift wieder aufgedeckt werden kann [PH10]. Das Besondere an Pseudonymisierung ist die Entfernung des Personenbezugs bei gleichzeitiger Erhaltung der Korrektheit gesammelter Ereignisse. Eine Aufdeckung beziehungsweise Umkehrung der Pseudonymisierung soll unmöglich oder zumindest impraktikabel sein, sofern die möglicherweise vorhandene Umkehrvorschrift nicht bekannt ist.

Problematisch bei der De-Identifizierung sind im Allgemeinen jedoch die folgenden Erkenntnisse.

1. Die alleinige Entfernung oder Verschleierung von PII bedeutet noch keine effektive De-Identifizierung und somit auch keinen ausreichenden Datenschutz. Denn auch die Existenz von personenbeziehbaren Daten (vgl. Abschnitt 6.1.1.2) stehen einer De-Identifizierung entgegen und müssen daher zunächst bestimmt und dann entsprechend bearbeitet werden.

Derartige Daten werden in der Literatur auch als Quasi-Identifier (QIDs) bezeichnet und umfassen sowohl PII als auch Merkmale, die für sich genommen möglicherweise noch keinen identifizierenden Charakter haben, die allerdings in Kombination miteinander eine Verknüpfung eines Ereignisses mit einer Identität erlauben [BWJ08].

2. Die Bestimmung von QIDs in einem Datensatz ist eine äußerst komplexe und kontextsensitive Aufgabe. Es gibt hauptsächlich zwei Gründe, warum man nicht einfach eine Liste von vordefinierten QIDs abarbeiten kann. Zum einen muss ein unscheinbarer Wert, der allerdings in einem vorliegenden Datensatz einzigartig ist, als QID bewertet werden. Das bedeutet, dass QIDs datensatzspezifisch zu verstehen sind und eine Analyse des vorliegenden Datensatzes voraussetzen. Zum anderen sind QIDs abhängig von möglicherweise vorhandenem oder beschaffbarem Hintergrundwissen. Beispielsweise kann ein Bezug von einer Person zu einem de-identifizierten Datensatzeintrag mit Zeitstempeln hergestellt werden, sofern man selbst eine Person bei der Durchführung von Aktionen beobachten kann und sich die Zeitpunkte der Aktionen notiert. Mit diesen Notizen als Hintergrundwissen sind die Zeitstempel im Datensatz als QIDs zu behandeln.

3. In der Vergangenheit konnte immer wieder gezeigt werden, dass selbst sorgfältig de-identifizierte Daten wieder auf bestimmte Personen zurückgeführt werden konnten [Ohm09; NF14], hauptsächlich aufgrund von Hintergrund- oder öffentlich zugänglichem Wissen, welches bei der De-Identifizierung nicht bedacht wurde. Dieser Umkehrprozess wird auch als *Re-Identifizierung* bezeichnet. Beispielsweise konnten Personen in einem de-identifizierten und publizierten Datensatz von Taxifahrtverläufen in New York eindeutig identifiziert werden [Dou+16], Patienten konnten in einem vom australischen Gesundheitsministerium de-identifizierten Medizindatensatz re-identifiziert werden [CRT17] und Journalisten haben sich mithilfe einer Scheinfirma von einem Unternehmen gesammelte und de-identifizierte Browserverläufe seiner Kunden beschafft und diese nicht nur wieder mit Personen verknüpft, sondern dabei zusätzlich noch brisante Informationen gefunden [ED17; Her17]. Weitere Beispiele einer erfolgreichen Re-Identifizierung sind in [Swe97; LDM10; Rot10; Mon+13; Sid14; Mon+15; LP16] zu finden.

4. Um einer Re-Identifizierung entgegen zu wirken, werden de-identifizierte Datensätze häufig nur unvollständig veröffentlicht oder für eine Verarbeitung freigegeben. Dabei wird argumentiert, dass eine Verknüpfung einer Person mit einem Datensatzeintrag, beispielsweise mithilfe von Hintergrundwissen, noch keine Re-Identifizierung darstellt, da die gefundene Verknüpfung noch auf weitere unveröffentlichte Datensatzeinträge zutreffen könnte. Diese Argumentation wurde in der Arbeit von Rocher, Hendrickx und Montjoye [RHM19] entkräftet. Die Autoren fanden heraus, dass eine Verknüpfung einer Person mit einem Eintrag aus einem unvollständigen Datensatz dennoch mit sehr hoher Wahrscheinlichkeit eine korrekte Re-Identifizierung darstellt.

Diese Erkenntnisse legen den Schluss nahe, dass eine vollständige De-Identifizierung von gesammelten Ereignisdaten zur Erkennung und Abwehr von Insiderbedrohungen grundsätzlich nicht möglich ist. Sofern allerdings der Begriff der *De-Identifizierung* nicht mit einer Vollständigkeit in Verbindung gebracht wird, sondern im Sinne einer Reduzierung und im besten Fall einer Minimierung von identifizierenden oder quasi-identifizierenden Merkmalen verstanden wird, so wie bereits von Sweeney [Swe97] und Ohm [Ohm09, Abschnitt II.C.2] gefordert, ergibt sich daraus ein valides Werkzeug zur Erhöhung des Datenschutzes. Dabei muss konsequent von einer binären Unterscheidung zwischen einem vollständigen Datenschutz einerseits und keinem

Datenschutz andererseits abgerückt werden und der Datenschutz als steigendes und fallendes Kontinuum verstanden werden.

Für den vorliegenden Anwendungsfall der gesammelten Daten zur Insiderbedrohungserkennung und -prävention kann eine De-Identifizierung also eine Erhöhung des Aufwandes bedeuten, den man aufbringen muss, um einen Bezug einzelner Dateneinträge zu einzelnen Personen einer Domäne herstellen zu können, der wiederum ohne die De-Identifizierung ohne Weiteres möglich wäre. Bei der Applikation von De-Identifizierungstechniken steht also nicht die Frage im Mittelpunkt, welche Datenfelder PII beziehungsweise QIDs sind und entfernt oder verschleiert werden müssen. Es geht vielmehr darum, welche Datenfelder, die potenziell alle QIDs sein könnten, für die Bedrohungserkennung und -prävention nicht benötigt werden und somit entfernt oder verschleiert werden können. Das Resultat ist kein vollständiger Schutz vor einer Re-Identifizierung, aber ein erhöhter Datenschutz.

6.2.2.4 Speicherbegrenzung

Die Sammlung von Ereignissen muss in ihrer Speicherdauer begrenzt werden. Die DSGVO beschreibt diese Anforderung in Art. 5 Abs. 1e wie folgt:

> „Personenbezogene Daten müssen in einer Form gespeichert werden, die die Identifizierung der betroffenen Personen nur so lange ermöglicht, wie es für die Zwecke, für die sie verarbeitet werden, erforderlich ist [...]“

Die Ausführungen in Abschnitt 6.2.2.3 zeigen auf, dass gesammelte Daten für die Erkennung und Abwehr von Insiderbedrohungen auch nach einer sorgfältig durchgeführten De-Identifizierung grundsätzlich als personenbeziehbar aufzufassen sind. Demnach müssen Wege gefunden werden, die Speicherdauer oder die allgemeinen Zugriffsmöglichkeit auch auf diese Daten zu begrenzen.

6.3 Existierende Arbeiten

Die folgenden existierenden Arbeiten haben einen engen Bezug zur vorliegenden Thematik und wurden demnach bei der Konstruktion einer datenschutzfreundlichen Insiderbedrohungserkennung in Kapitel 7 beachtet.

6.3.1 Dedizierter Datenschutz bei der Insiderbedrohungserkennung und -abwehr

Die datenschutzfreundliche Erkennung und Abwehr von Insiderbedrohungen ist ein bisher wenig erforschtes Wissenschaftsfeld und wird deshalb als noch offene Herausforderung genannt [Pfl08; SHS08; Fun11; Zim+16]. Eine der wenigen Arbeiten zu diesem Thema ist ein im Jahr 2019 angenommenes US-Patent, dass sich auf den Schutz personenbezogener Daten bei der Überwachung von Insiderbedrohungen spezialisiert [For+19]. Dabei wird das Nutzerverhalten überwacht und dabei anfallende personenbezogene Daten erkannt, gekennzeichnet und durch Einwegfunktionen verschleiert.

6.3.2 Pseudonymisierung im Bereich von Intrusion Detection Systems

Die fehlende Präsenz der Datenschutzthematik im Forschungsgebiet der Erkennungs- und Abwehrmaßnahmen von Insiderbedrohungen liegt womöglich vor allem daran, dass Erkennungs- und Abwehrtechniken von Bedrohungen häufig nicht dediziert auf den Kontext von Insiderbedrohungen abgerichtet sind. Starke Überschneidungen gibt es zum Beispiel mit dem Bereich der *Intrusion Detection Systems (IDSs)*, in dem auch Techniken zur Ausbalancierung von Anforderungen der Anomalieerkennung mit den Datenschutzanforderungen untersucht wurden. Dabei steht insbesondere die Pseudonymisierung im Fokus der Forschung.

Bereits 1997 diskutieren Sobirey, Fischer-Hübner und Rannenberg [SFR97] den Einfluss, den das Sammeln und Analysieren von Auditierungsereignissen auf die Privatsphäre und den Datenschutz von betroffenen Nutzern haben könnte. Sie schlagen Pseudonymisierung als praktikablen Weg vor, um die Datenschutzinteressen von Arbeitnehmern und Nutzern eines derart überwachten Computersystems vor dem aufkommenden Trend der automatisierten Intrusion Detection und Betriebssystemauditierung zu wahren. Die Autoren präsentieren ein verteiltes IDS namens *AID*, das sensible Informationen deterministisch verschlüsselt und diese Schlüsseltexte als Pseudonyme an die Stelle der sensiblen Informationen setzt. Damit werden konsistente Pseudonyme auch bei verteilten Pseudonymisierungsagenten erreicht.

Büschkes und Kesdogan [BK99] beschreiben die gegenläufigen Interessen eines IDS-Betreibers und der betroffenen Nutzer. Die Autoren fordern die Umsetzung der Konzepte *Datenvermeidung* sowie *Datenminimierung* und schlagen ebenfalls eine Pseudonym-basierte Lösung vor. Diese involviert eine vertrauenswürdige dritte Partei, die jeweils die Identitäten der Nutzer kennt und diese durch Pseudonyme ersetzt. Um die Auswirkungen einer potenziellen Profilbildung auch unter der Verwendung von Pseudonymen zu reduzieren, führen die Autoren das Konzept der Gruppenreferenzpseudonyme ein, mit denen nicht die Identitäten einzelner Nutzer, sondern Nutzergruppen verwendet werden.

Ebenfalls um Pseudonymisierung, allerdings im Kontext von kollaborativen IDS, bei denen kooperierende Standorte sicherheitsrelevante Warnmeldungen und Ereignisse austauschen, um insgesamt einen umfassenderen Überblick über die Sicherheitslage zu bekommen, geht es in der Arbeit von Lincoln, Porras und Shmatikov [LPS04]. Der kollaborative Ansatz ist an sich nicht neu. Die Autoren fokussieren sich aber auf den Umgang mit potenziell sensiblen Informationen, die meist nicht ohne weiteres einem solchen Austausch unterzogen werden sollten, da sie weitere Informationen etwa über interne und externe Kommunikationsbeziehungen sowie Rechnernetz-Infrastrukturen gegenüber den kooperierenden Standorten offenlegen. Die Autoren schlagen eine Reihe von Maßnahmen vor, um den Datenschutz zu wahren. Dazu gehört das Unterdrücken von sensiblen Informationen vor der Veröffentlichung, das Verschleiern von IP-Adressen durch Hashfunktionen teilweise gepaart mit einem geheimen Schlüssel (engl. keyed hash functions) [KL14, Abschnitt 5.3.2] sowie der Einsatz von Senderanonymisierungstechniken beim Veröffentlichen von Warnmeldungen, um dabei als kooperierender Standort anonym zu bleiben.

Einen anderen Ansatz als den der Pseudonymisierung verfolgen Locasto u. a. [Loc+05]. Sie schlagen ebenfalls ein kollaboratives IDS vor. Die Autoren konzentrieren sich auch hier speziell auf IP-Adressen, die interne und externe Kommunikationsbeziehungen gegenüber den kooperierenden Standorten offenlegen können. Um die realen IP-Adressen zu schützen, verwenden die Autoren sogenannte *Bloom Filter* [Blo70], die eine Einwegdatenstruktur darstellen und mit denen eine Überwachungsliste von IP-Adressen und Portnummern erstellt und geteilt werden

kann. Die in einem Bloom Filter gespeicherten Informationen können nicht extrahiert werden, aber man kann einen probabilistischen Test durchführen, der offenlegt, ob ein spezielles Datum Teil des Bloom Filters ist.

6.3.3 Re-Identifizierung durch Schwellwertschemata

Biskup und Flegel [BF00a; BF00b] lassen identifizierende Merkmale in Auditierungsereignisnachrichten durch eine zentrale Verarbeitungskomponente mit kurzlebigen Pseudonymen ersetzen, die wiederum mittels des sogenannten *Shamir's Secret Sharing* [Sha79] von langlebigeren Pseudonymen abgeleitet werden. Das erlaubt die Aufdeckung von Pseudonymen, sobald die Anzahl von Auditierungsereignissen und damit die Anzahl von ausgegebenen Geheimnisanteilen (engl. secret shares), die sich auf dieselbe Identität beziehen, einen festgelegten Schwellwert überschreiten. Ein Auditor kann damit die Identitäten hinter den Pseudonymen mithilfe der Geheimnisanteile rekonstruieren.

Eine Weiterentwicklung dieser sogenannten *Schwellwertschemata* (engl. threshold cryptosystems) bietet, anders als bei Shamir's Secret Sharing, die Möglichkeit, ein auf mehrere Geheimnisträger verteiltes Geheimnis zu verwenden, ohne dass die Geheimnisträger ihr Geheimnisanteil aufdecken müssen und dieses somit darauffolgend unbrauchbar wird [DF90].

Ein praktischer Einsatz dieser Verfahren zur kooperationsgezwungenen Aufdeckung einer Identität, die zum Zweck des Datenschutzes verschleiert wurde, wurde von Armknecht und Dewald [AD15] im Kontext der Digitalen Forensik von sensiblen E-Mail-Daten entwickelt.

6.4 Fazit

Die Rechtslage beim Einsatz von Maßnahmen zur Erkennung und Abwehr von Insiderbedrohungen im Beschäftigungskontext zeigt sich als insgesamt sehr schwierig. Dass es sich dabei um die Verarbeitung von personenbezogenen Daten im Sinne der DSGVO handelt, konnte in Abschnitt 6.1.1.2 gezeigt werden. Daraus ergibt sich die Anwendbarkeit der DSGVO, sofern keine bereichsspezifischen Datenschutzsonderregelungen vorliegen. Ebenfalls diskutiert wurde die Anwendbarkeit der DSGVO trotz vorliegender Öffnungsklausel für den Beschäftigungskontext in Abschnitt 6.1.1.4. Aus diesen Betrachtungen geht hervor, dass der Einsatz von Erkennungs- und Abwehrmaßnahmen prinzipiell nicht ausgeschlossen ist, sich aber in einem sehr engen Schutzrahmen bewegen und die Privatsphäre der betroffenen Arbeitnehmer so weit wie möglich schützen muss. Darüber hinaus können Arbeitnehmer unter gewissen Voraussetzungen indirekte Mitbestimmungsrechte beim Einsatz derartiger Maßnahmen mittels der Arbeitnehmervertretungen geltend machen, wie in Abschnitt 6.1.1.5 aufgezeigt wurde. Im Hinblick auf die Rechte und Pflichten der Arbeitnehmer wurde aufgezeigt, unter welchen Bedingungen der Einsatz von Erkennungs- und Abwehrmaßnahmen im Rahmen der DSGVO und des BDSG erlaubt ist. Insbesondere die Wahrung berechtigter Interessen, zu denen unter anderem auch in der GRCh festgeschriebene Grundrechte gehören (vgl. Abschnitt 6.1.2.3), sowie rechtliche Verpflichtungen stellen für diesen Kontext zentrale Erlaubnistatbestände dar. Für den letzteren Fall wurden exemplarisch in Abschnitt 6.1.2.2 die Compliance sowie die Sicherheit in der Informationstechnik Kritischer Infrastrukturen betrachtet. Letztendlich ergibt sich aus den Betrachtungen ein Rechts-

und Interessenskonflikt, der in Abschnitt 6.1.3 aufgezeigt wurde und mit dem Verhältnismäßigkeitsprinzip, also der strikten Prüfung auf Erforderlichkeit und Angemessenheit, für alle Beteiligten zufriedenstellend aufgelöst werden kann.

Aus den Betrachtungen des rechtlichen Rahmens heraus wurden die gegenläufigen Anforderungen identifiziert und erläutert. Aus der Sicht der Domäne beziehungsweise des Arbeitgebers können Maßnahmen zur Erkennung und Abwehr von Insiderbedrohungen nur effektiv umgesetzt werden, wenn mehrere Aktivitätsschritte eines Insiders miteinander verkettet werden und für spätere Analysen sowie Aufklärungen archiviert werden können. Darüber hinaus ist es meist notwendig, die Ereignisse, anhand derer Insiderbedrohungen erkannt wurden, auch dem entsprechenden Urheber der Ereignisse zuordnen zu können. Diese Anforderungen wurden in den Abschnitten 6.2.1.2 bis 6.2.1.4 erläutert. Dem entgegengesetzt richten sich die Anforderungen aus Sicht der Insider beziehungsweise der Arbeitnehmer nach den Grundsätzen der Datenverarbeitung personenbezogener Daten aus der DSGVO. Dazu zählt die Datenminimierung, die Speicherbegrenzung sowie die möglichst weitreichende De-Identifizierung der Daten, die unter anderem in den Abschnitten 6.2.2.1 bis 6.2.2.4 betrachtet wurden.

Mit der Herausarbeitung der Rechte und Pflichten aller Beteiligten sowie der mehrseitigen Anforderungen an datenschutzfreundliche Erkennungs- und Abwehrmaßnahmen von Insiderbedrohungen in diesem Kapitel wurde der erste Teil des Forschungsbeitrags B4 (vgl. Abschnitt 1.3) bearbeitet. Damit wurden Vorbereitungen für die Entwicklung einer Plattform zur Erhöhung des Datenschutzes von Insidern getroffen, die in Kapitel 7 aufgenommen und praktisch umgesetzt werden.

7 Technik zum Insiderdatenschutz

Der Einsatz von Techniken zur Erkennung und Abwehr von Insiderbedrohungen kann aus Sicht eines Arbeitgebers rechtlich erlaubt sein und ist teilweise sogar verpflichtend geboten. Aus Sicht der betroffenen Insider bedeuten sie allerdings immer einen Eingriff in die Privatsphäre und informationelle Selbstbestimmung und beinhalten darüber hinaus Bedrohungspotenzial durch die Möglichkeit der unerlaubten Zweckentfremdung der anfallenden Daten etwa für Produktivitätsbewertungen oder Profilbildungen (vgl. Abschnitt 3.4). Beide Standpunkte wurden in Kapitel 6 eingehend betrachtet. Der Entwurf einer Technik zum Datenschutz bei der Erfassung und Analyse von Insideraktivitäten im Sinne einer mehrseitigen Würdigung von Interessen bietet eine Lösung für diesen Rechts- und Interessenskonflikt. Daher wird ein solcher Entwurf in diesem Kapitel in den Fokus gestellt und die in Abschnitt 6.2 beschriebenen Anforderungen sowohl aus Sicht der Arbeitgeber als auch aus Sicht der betroffenen Insider ausbalanciert und praktisch umgesetzt. Damit wird Forschungsfrage 4 aus Abschnitt 1.2 abschließend bearbeitet.

Wesentliche Inhalte Als Basis für die in dieser Dissertation entwickelte Technik zum Insiderdatenschutz dient die Pseudonymisierung. Mit ihr können die in Abschnitt 6.2 beschriebenen wesentlichen Anforderungen umgesetzt werden. Dazu zählen etwa die Verkettbarkeit von Ereignissen oder die De-Identifizierung von PII beziehungsweise QIDs bei gleichzeitiger Möglichkeit der Re-Identifizierung in gewünschten Fällen. Mithilfe kryptographischer Funktionen wird die Technik der Pseudonymisierung erweitert und der damit anvisierte Datenschutz erhöht, ohne dabei die gegenläufigen Anforderungen für eine effektive Bedrohungserkennung und -abwehr zu verhindern. Unter Berücksichtigung zweier Systemmodelle, einmal dezentralisiert und einmal zentralisiert, wird eine datenschutzfreundliche Pseudonymisierung von Ereignisnachrichten entworfen und softwaretechnisch umgesetzt, bei der neu definierte Pseudonymisierungs-Schutzziele verfolgt werden. Diese neuen Schutzziele ergeben sich aus der Beobachtung, dass die naive Umsetzung einer Pseudonymisierung Informationen erzeugt beziehungsweise für beteiligte Pseudonymisierungskomponenten zur Verfügung stellt, die bei näherer Betrachtung unnötig und vermeidbar sind. Ein Beispiel sind die Informationen, die eine Re-Identifizierung von Pseudonymen erlauben. Diese Informationen müssen nicht bei einer einzelnen Instanz vorliegen, sondern können und sollten zum Zweck der Missbrauchsprävention auf mehrere Instanzen verteilt und eine notwendige Kooperation dieser Instanzen für eine erfolgreiche Re-Identifizierung technisch erzwungen werden. Das im Rahmen dieser Dissertation entworfene und softwaretechnisch umgesetzte Pseudonymisierungsverfahren wird darüber hinaus einer Evaluation unterzogen, die verschiedene Datenschutzstufen miteinander vergleicht und den jeweiligen Einfluss aufzeigt, den das Verfahren auf die Performanz bei der Verarbeitung von Ereignisnachrichten hat.

Relevante Veröffentlichungen Das Anwendungsszenario sowie die zugehörigen Erläuterungen in Abschnitt 7.2.1 wurden bereits in [Zim+16] beschrieben. Die Herleitung der Pseudonymisierungs-Schutzziele anhand der Datenschutzbedrohungen in Abschnitt 7.4 sowie deren Umsetzung im zentralisierten Systemmodell in Abschnitt 7.5.2 basieren auf Vorarbeiten, die bereits in [Zim+20] zusammengefasst und veröffentlicht wurden.

Aufbau des Kapitels Zunächst erfolgt die Einführung in die für dieses Kapitel notwendigen Grundlagen in Abschnitt 7.1. Diese umfassen eine Definition des Pseudonymisierungsbegriffs sowie kryptographische Funktionen, mit denen der erreichbare Datenschutz bei der praktischen Umsetzung von Pseudonymisierung erhöht wird. Der Kontext, für den das neue Pseudonymisierungsverfahren entwickelt wurde, sowie weitere Anforderungen, die in dieser Dissertation außer Acht gelassen wurden, finden sich in Abschnitt 7.2. Das Pseudonymisierungsverfahren unterstützt zwei Systemmodelle, in denen die Pseudonymisierung zum einen dezentral aber dennoch global konsistent und zum anderen zentral und dadurch echt zufällig ermöglicht wird. Diese beiden Systemmodelle werden in Abschnitt 7.3 beschrieben. Die Identifizierung von Bedrohungen für den Datenschutz durch die praktische Umsetzung von Pseudonymisierung in den beiden Sytemmodellen erfolgt in Abschnitt 7.4. Daraus ergibt sich die Herausarbeitung von Pseudonymisierungs-Schutzzielen, deren Definition und Umsetzung getrennt für die zwei Systemmodelle in Abschnitt 7.5 im Detail beschrieben wird. Im Anschluss daran wird in Abschnitt 7.6 die softwaretechnische Realisierung des neuen Pseudonymisierungsverfahrens kurz erläutert und in verschiedenen Konfigurationen entsprechend der Systemmodelle und Pseudonymisierungs-Schutzziele evaluiert. Die für zukünftige Arbeiten offenen Probleme beziehungsweise Verbesserungsmöglichkeiten werden in Abschnitt 7.7 aufgezeigt, bevor das Kapitel in Abschnitt 7.8 mit einem zusammenfassenden Fazit abschließt.

7.1 Grundlagen der datenschutzfreundlichen Insiderbedrohungserkennung

Die folgenden Abschnitte behandeln kurz diejenigen Grundlagen, die als Basisbausteine für das Design und die Implementierung der datenschutzfreundlichen Insiderbedrohungserkennung und -prävention benötigt werden.

7.1.1 Pseudonymisierung

Die Definition von Pseudonymisierung, wie sie nachfolgend verwendet wird, basiert auf der in Art. 4 Satz 5 DSGVO gegebenen Begriffsbestimmung [DSGVO16]. Diese bezeichnet mit Pseudonymisierung

> „die Verarbeitung personenbezogener Daten in einer Weise, dass die personenbezogenen Daten ohne Hinzuziehung zusätzlicher Informationen nicht mehr einer spezifischen betroffenen Person zugeordnet werden können, sofern diese zusätzlichen Informationen gesondert aufbewahrt werden und technischen und organisatorischen Maßnahmen unterliegen, die gewährleisten, dass die personenbezogenen Daten nicht einer identifizierten oder identifizierbaren natürlichen Person zugewiesen werden“.

Der Definition zufolge existieren zusätzliche Informationen, zum Beispiel eine Zuordnungstabelle oder eine Berechnungsvorschrift, mit denen Pseudonyme den korrespondierenden Identifizierungsmerkmalen zugeordnet werden können. Damit kann der Prozess der Pseudonymisierung wieder umgekehrt und eine *Re-Identifizierung* durchgeführt werden. Diese zusätzlichen Informationen werden nachfolgend zusammengefasst als Pseudonymzuordnung bezeichnet. Ohne Kenntnis über Inhalte dieser Zuordnung soll es praktisch unmöglich sein, eine Re-Identifizierung durchführen zu können.

Die Wirksamkeit von Pseudonymisierung wird signifikant durch die Entscheidung bestimmt, welche Daten identifizierende oder quasi-identifizierende Merkmale (QIDs, vgl. Abschnitt 6.2.2.3) darstellen und demnach durch die Pseudonymisierung bearbeitet werden müssen. Dieser Prozess der QID-Festlegung ist hochgradig anwendungs- und datensatzspezifisch.

Zu beachten ist, dass die Literatur die Pseudonymzuordnung zwischen Pseudonymen und Identifizierungsmerkmalen als Verknüpfung (engl. link) bezeichnet, was nicht mit dem Begriff der Verkettbarkeit (engl. linkability), wie er in Abschnitt 6.2.1.1 eingeführt und erläutert wurde, verwechselt werden darf.

7.1.2 Hashfunktionen und HMACs

Eine Hashfunktion ist eine surjektive aber im Allgemeinen nicht injektive Abbildung $H : M^* \mapsto N^n$ von einem Eingabewert aus einer Menge M mit beliebiger Länge zu einem Ausgabewert aus einer Menge N mit fester Länge $n \in \mathbb{N}$ [KL14, Abschnitt 5.1]. Der Ausgabewert wird Hashwert der Eingabe genannt. Die Abbildung wird als eindeutig angenommen, sofern die folgenden Eigenschaften gelten:

1) *Unumkehrbarkeit*: Zu einem gegebenen Hashwert $y = H(x)$ ist es praktisch unmöglich, den Eingabewert x zu finden. Diese Eigenschaft wird auch *Einwegeigenschaft* genannt.

2) *Schwache Kollisionsresistenz*: Zu einem gegebenen Eingabewert x ist es praktisch unmöglich, einen weiteren Eingabewert $x' \neq x$ zu finden, sodass $H(x') = H(x)$ gilt.

3) *Starke Kollisionsresistenz*: Es ist praktisch unmöglich, zwei beliebige Eingabewerte x und x' mit $x \neq x'$ zu finden, sodass $H(x) = H(x')$ gilt.

Abgesehen von Schwachstellen praktisch umgesetzter Hashfunktionen, die sich auf die Konstruktion und die unterliegende Mathematik beziehen, leidet die Unumkehrbarkeit aller Hashfunktionen daran, dass zu gegebenen Hashwerten einfach alle möglichen Eingabewerte durchprobiert werden können, die in der Regel in ihrer Anzahl beschränkt sind. Solche Bruteforce-Angriffe sind insbesondere dann effektiv, wenn die Urbildmenge sehr klein ist, so wie im Fall von QIDs. Eine Unumkehrbarkeit allein durch die Anwendung von Hashfunktionen ist dann nicht mehr hinreichend [Mar+18].

Eine mögliche Lösung ist die Vergrößerung der Urbildmenge durch das Hinzufügen von zusätzlicher Entropie zu den Eingabewerten in Form eines geheimen Schlüssels. Eine derartige Konstruktion wird *schlüsselbasierte Hashfunktion* (engl. keyed hash function) genannt und wurde als Keyed-Hash Message Authentication Code (HMAC) derart standardisiert, dass praktische Schwächen bei der einfachen Konkatenation von ursprünglichem Eingabewert und Schlüssel gelöst werden [KBC97]. Wie jeder MAC nimmt auch ein HMAC einen zufällig gewählten geheimen Schlüssel k von ausreichender Länge und eine Nachricht m von beliebiger Länge als Eingabe und gibt $\texttt{Mac}(k, m)$ aus. Dabei handelt es sich um ein sogenanntes Kennzeichen (engl. tag) der Nachricht, das praktisch unfälschbar ist, solange der geheime Schlüssel k nicht bekannt ist [KL14, Abschnitt 5.3.2].

7.1.3 ElGamal-Kryptosystem

Auf Grundlage des Diffie-Hellman-Schlüsselaustauschs [DH76] wurde von Elgamal [Elg85] ein asymmetrisches kryptographisches Verfahren zum Ver- und Entschlüsseln von Nachrichten entwickelt. Die Sicherheit des *ElGamal-Kryptosystems* hängt mit dem schwer zu lösendem Problem des sogenannten *Diskreten Logarithmus* zusammen [KL14, Abschnitt 8.3.2].

Zugrunde liegt eine multiplikative zyklische Gruppe $\mathbb{Z}_p$ mit der Ordnung einer Primzahl p sowie ein zugehöriger Generator $g \in \mathbb{Z}_p$ dieser Gruppe. Alle Operationen werden modulo p ausgeführt. Der Empfänger $\mathcal{R}$ einer Nachricht wählt zufällig ein Element $a \in \mathbb{Z}_p$ dieser Gruppe als geheimen Schlüssel und berechnet $z = g^a$ als Teil seines öffentlichen Schlüssels (p, g, z).

Der Sender $\mathcal{S}$ einer Nachricht $m \in \mathbb{Z}_p$ wählt zunächst ebenfalls ein zufälliges Element $b \in \mathbb{Z}_p$ und berechnet den Sitzungsschlüssel $k = z^b = (g^a)^b$ und damit dann den Schlüsseltext $(c_1, c_2) = (g^b, k \cdot m)$. $\mathcal{S}$ sendet den Schlüsseltext an $\mathcal{R}$, der zur Entschlüsselung zunächst mit seinem geheimen Schlüssel a den Sitzungsschlüssel $k = c_1^a = (g^b)^a$ berechnet und damit die Nachricht $m = c_2 \cdot k^{-1}$ entschlüsseln kann.

Bei der Wahl der zugrunde liegenden zyklischen Gruppe sowie bei der Menge der möglichen Nachrichten gibt es Besonderheiten, auf die hier nicht näher eingegangen wird, die es aber zu beachten gilt. Für weitere Informationen sowie für Betrachtungen der Sicherheit des Verfahrens wird auf [KL14, Abschnitt 11.4.1] verwiesen.

Zwei Erweiterungen des ElGamal-Kryptosystems werden an dieser Stelle noch betrachtet. Zum einen betrifft das die eigentliche Verschlüsselung der Nachricht m mit dem Sitzungsschlüssel k, die hier als Multiplikation beschrieben wurde. Zum anderen kann das ElGamal-Verfahren auf Basis elliptischer Kurven realisiert werden.

7.1.3.1 Hybride Verschlüsselung

Das ursprünglich entwickelte ElGamal-Kryptosystem setzt voraus, dass die zu verschlüsselnden Nachrichten selbst Elemente der zugrunde liegenden Gruppe $\mathbb{Z}_p$ sind. Die Verschlüsselung kann dadurch als Multiplikation mit dem errechneten Sitzungsschlüssel $k = g^{a \cdot b}$ realisiert werden. Elgamal [Elg85] merkt in seiner Veröffentlichung an, dass die Multiplikation auch mit einer anderen invertierbaren Operation ersetzt werden kann. Ein sehr effizienter Ersatz an dieser Stelle ist die Verwendung eines symmetrischen Verschlüsselungsverfahrens $\mathtt{Enc}(\mathtt{kdf}(k), m)$, bei dem aus k mittels einer geeigneten Schlüsselableitungsfunktion (engl. key derivation function) $\mathtt{kdf} : \mathbb{Z}_p \mapsto \{0,1\}^n$ ein n-Bit symmetrischer Schlüssel abgeleitet und die Nachricht symmetrisch verschlüsselt wird (vgl. [KL14, Abschnitte 11.4.2 und 5.6.4]). Dieser Ansatz wird auch als *hybrides Kryptosystem* bezeichnet. Die Nachricht muss dadurch weder ein Element der zugrunde liegenden Gruppe $\mathbb{Z}_p$ sein noch als solches Element kodiert werden. Effizient ist der Ansatz vor allem dann, wenn es sich um eine sehr lange Nachricht handelt.

7.1.3.2 Elliptische Kurven

Die dem ElGamal-Kryptosystem zugrunde liegende Gruppe mit ihrer modularen Arithmetik kann durch diskrete Punkte auf *elliptischen Kurven* (engl. elliptic curves, EC) zusammen mit

einer speziell definierten Additionsoperation ersetzt werden. Dadurch ergibt sich eine erhebliche Effizienzsteigerung, da die sicherheitsrelevanten Parameter bei vergleichbarer Sicherheit wesentlich kleiner gewählt werden können [KL14, Abschnitt 8.3.4]. Da die Theorie sowie die Eigenschaften von elliptischen Kurven sehr schnell sehr komplex werden, konzentrieren sich die folgenden Darstellungen nur auf das Wesentliche. Für weitere Informationen wird auf [KL14, Abschnitt 8.3.4] verwiesen.

Eine elliptische Kurve wird beschrieben durch eine Primzahl p sowie eine Kurvengleichung mit den Variablen x und y der Form

$$y^2 = x^3 + A \cdot x + B \bmod p,$$

wobei für $A, B \in \mathbb{Z}_p$ die Bedingung $4 \cdot A^3 + 27 \cdot B^2 \neq 0 \bmod p$ gelten muss. Alle Punkte $(x, y) \in \mathbb{Z}_p \times \mathbb{Z}_p$, die diese Gleichung erfüllen, zusammen mit dem sogenannten *Punkt im Unendlichen* $\mathcal{O}$, ergeben die elliptische Kurve

$$\mathcal{E}(p, A, B) = \{(x, y) \mid x, y \in \mathbb{Z}_p \text{ und } y^2 = x^3 + A \cdot x + B\} \cup \{\mathcal{O}\}.$$

Mithilfe einer speziell definierten Addition von Punkten elliptischer Kurven lassen sich additive abelsche Gruppen definieren, in denen für alle Punkte $P, Q, R \in \mathcal{E}$ gilt

- $P + Q \in \mathcal{E}$ (Abgeschlossenheit),
- $P + \mathcal{O} = P$ (neutrales Element),
- $P + (-P) = \mathcal{O}$ (inverses Element),
- $P + Q = Q + P$ (Kommutativität) sowie
- $(P + Q) + R = P + (Q + R)$ (Assoziativität).

Die Addition zweier Punkte $P, Q \in \mathcal{E}$ lässt sich geometrisch als dritter Schnittpunkt der Geraden durch P und Q mit der Kurve $\mathcal{E}$ beschreiben, der wiederum noch an der x-Achse gespiegelt wird. Ein solcher dritter Schnittpunkt existiert durch die Hinzunahme des Punktes im Unendlichen $\mathcal{O}$ immer. Auf die Details der Berechnung des Punktes $R = P + Q$, die sich kalkulatorisch auf die Steigung der Geraden stützen kann, wird an dieser Stelle verzichtet. Wichtig für die Anwendung von elliptischen Kurven im Kontext des ElGamal-Kryptosystems ist vielmehr die Erkenntnis, dass die Mehrfachaddition eines Punktes mit sich selbst $R = P + \cdots + P$ verstanden werden kann als die Multiplikation des Punktes P mit einer Konstanten $c \in \mathbb{Z}_p$: $R = c \cdot P$. Die so berechneten Punkte R für alle $c \in \mathbb{Z}_p$ beschreiben dadurch eine durch P erzeugte zyklische Untergruppe von $\mathcal{E}(p, A, B) : \langle P \rangle = \{\mathcal{O}, 1 \cdot P, 2 \cdot P, 3 \cdot P, \ldots, (n-1) \cdot P\}$, die die Ordnung n besitzt. Diese Ordnung ist die kleinste natürliche Zahl $n > 0$ für die gilt: $n \cdot P = \mathcal{O}$. Mit einem beliebigen Element $Q \in \langle P \rangle$ sowie dem Generatorpunkt P dieser Untergruppe kann das schwer zu lösende Problem des *Diskreten Logarithmus* auch auf elliptischen Kurven definiert werden. Es besteht darin, die Konstante $c \in \mathbb{Z}_p$ zu bestimmen, mit der $Q = c \cdot P$ gilt.

Darauf aufbauend lässt sich das ElGamal-Kryptosystem auf Basis elliptischer Kurven beschreiben. Zugrunde liegt eine ausreichend große elliptische Kurve $\mathcal{E}(p, A, B)$ mit p als Primzahl, $A, B \in \mathbb{Z}_p$ und $4 \cdot A^3 + 27 \cdot B^2 \neq 0 \bmod p$ sowie ein Generatorpunkt $P \in \mathcal{E}$, der eine ausreichend große zyklische Untergruppe $\langle P \rangle$ erzeugt. Alle Operationen werden modulo p ausgeführt. Der Empfänger $\mathcal{R}$ einer Nachricht wählt zufällig ein Element $a \in \mathbb{Z}_p \backslash \{0\}$ als geheimen Schlüssel und berechnet $R = a \cdot P$ als Teil seines öffentlichen Schlüssels $(\mathcal{E}, P, R)$.

Der Sender $\mathcal{S}$ einer Nachricht $M \in \mathcal{E}$ wählt zunächst ebenfalls ein zufälliges Element $b \in \mathbb{Z}_p$ und berechnet den Sitzungsschlüssel $K = b \cdot R = b \cdot (a \cdot P)$ und damit dann den Schlüsseltext

$(C_1, C_2) = (b \cdot P, K + M)$.[1] S sendet den Schlüsseltext an $\mathcal{R}$, der zur Entschlüsselung zunächst mit seinem geheimen Schlüssel a den Sitzungsschlüssel $K = a \cdot C_1 = a \cdot (b \cdot P)$ berechnet und damit die Nachricht $M = C_2 + (-K)$ entschlüsseln kann.

7.1.4 Schwellwertschemata

Mit einem *Schwellwertschema* kann man ein Geheimnis auf verschiedene Geheimnisträger verteilen, sodass keiner dieser Geheimnisträger das ursprüngliche Geheimnis in Erfahrung bringen kann. Erst durch die Kooperation von einer fest vorher definierten Mindestanzahl an Geheimnisträgern kann das ursprüngliche Geheimnis wieder rekonstruiert werden. Man spricht dabei auch von einem (t, n)-Schwellwertschema, wobei n die Anzahl der Geheimnisträger (engl. shares) und t die Mindestanzahl an Geheimnisträgern ist, die kooperieren müssen. Sofern nur $t - 1$ Geheimnisträger kooperieren, bleibt das Geheimnis komplett unbestimmt, das heißt all seine möglichen Ausprägungen bleiben gleichwahrscheinlich.

7.1.4.1 Shamir's Secret Sharing

Ein auf der Polynominterpolation basierendes Schwellwertschema wurde von Shamir [Sha79] entwickelt und *Shamir's Secret Sharing* benannt. Dabei wird ein Geheimnis D, ohne Beschränkung der Allgemeinheit repräsentiert durch eine positive Ganzzahl, in n Geheimnisteile $D_1, \ldots, D_n$ zerlegt, wobei alle Berechnungen modulo einer Primzahl p getätigt werden, die sowohl größer als D als auch größer als n sein muss. Zunächst wird ein Polynom $q(x) = c_0 + c_1 \cdot x + \cdots + c_{t-1} \cdot x^{t-1}$ vom Grad $t - 1$ mit zufälligen Koeffizienten $c_i \in \mathbb{Z}_p$ und $c_0 = D$ gewählt. Die einzelnen Geheimnisteile berechnen sich dann als $D_1 = (x_1, y_1 = q(x_1)), \ldots, D_i = (x_i, y_i = q(x_i)), \ldots, D_n = (x_n, y_n = q(x_n))$, wobei alle $x_i \in \mathbb{Z}_p \backslash \{0\}$ paarweise verschieden aber nicht unbedingt zufällig gewählt sein müssen. Sie können somit auch einfach die Indizes der jeweiligen Geheimnisteile darstellen. Mit diesen Bedingungen lässt sich das Geheimnis D durch die Auswertung des Polynoms an der Stelle $x = 0$ errechnen: $D = q(0)$.

Mit mindestens t unterschiedlichen Geheimnisteilen $D'_1 = (x'_1, y'_1), \ldots, D'_t = (x'_t, y'_t)$ lässt sich das Polynom $q(x)$ mithilfe der Summe der im Jahre 1795 publizierten Lagrange-Polynome interpolieren [Lag95].[2] Die Lagrange-Polynome

$$\ell_i(x) = \prod_{j=1, j \neq i}^{t} \frac{x - x'_j}{x'_i - x'_j} \text{ für alle } i \in \{1, \ldots, t\},$$

sind ebenfalls vom Grad $t - 1$ und es gilt $\ell_i(x'_i) = 1$, da in diesem Fall der Zähler und der Nenner identisch sind. Weiterhin gilt $\ell_i(x'_j) = 0$ für alle $j \in \{1, \ldots, t\}$ und $j \neq i$, da in diesem Fall einer der Faktoren im Zähler $x'_j - x'_j = 0$ ergibt. Das ursprüngliche Polynom $q(x)$ lässt sich nun also mithilfe der t unterschiedlichen Geheimnisteile und der Lagrange-Polynome beschreiben als

$$q(x) = \sum_{i=1}^{t} y'_i \cdot \ell_i(x).$$

1. Die Verschlüsselung von einer beliebigen Nachricht m, die nicht unbedingt einen Punkt der elliptischen Kurven darstellt oder in einen solchen überführt werden muss, kann auch hier durch einen hybriden Ansatz stark vereinfacht werden (vgl. Abschnitt 7.1.3.1).
2. Shamir [Sha79] macht in einer Fußnote deutlich, dass die Lagrange-Polynominterpolation nicht die einzige Möglichkeit ist, das ursprüngliche Geheimnis wiederherzustellen. Die Polynome können auch mit anderen berechenbaren und interpolierbaren Funktionen ersetzt werden.

Damit gilt für alle Geheimnisteile $D'_1 = (x'_1, y'_1), \ldots, D'_t = (x'_t, y'_t)$:

$$q(x'_1) = y'_1 = \sum_{i=1}^{t} y'_i \cdot \ell_i(x'_1),$$

$$\vdots$$

$$q(x'_t) = y'_t = \sum_{i=1}^{t} y'_i \cdot \ell_i(x'_t).$$

Für die Rekonstruktion des ursprünglichen Geheimnisses D muss nur noch die Lagrange-Polynominterpolation an der Stelle $x = 0$ ausgewertet werden:

$$D = q(0) = \sum_{i=1}^{t} y'_i \cdot \ell_i(0).$$

Die Lagrange-Polynome $\ell_i(0)$ sind ausschließlich abhängig von den einzelnen x'_i der an der Rekonstruktion beteiligten Geheimnisteile, aber unabhängig von den einzelnen Polynomwerten y'_i und können somit bereits vorberechnet werden.

7.1.4.2 ElGamal-basiertes Schwellwertschema

Shamir's Secret Sharing hat die Eigenschaft, dass ein einmal aufgedecktes Geheimnis auch für die Zukunft für denjenigen offenliegt, der die Teilgeheimnisse der Geheimnisträger eingesammelt und damit das ursprüngliche Geheimnis rekonstruiert hat. Es ist somit für den vorliegenden Anwendungsfall ungeeignet, denn im Kontext dieser Arbeit sollen mehrere Informationen verschlüsselt werden, die dann jeweils für sich genommen nur durch die Kooperation mehrerer Akteure wieder entschlüsselt werden können. Dafür ist ein neues Geheimnis für die Verschlüsselung jeder neuen Information notwendig, wobei das Secret Sharing idealerweise nicht für jedes Geheimnis neu durchgeführt werden muss.

Eine geeignete Weiterentwicklung ist das von den Autoren Desmedt und Frankel [DF90] vorgeschlagene *ElGamal-basierte Schwellwertschema*. Dabei wird ein geheimer sowie ein öffentlicher Schlüssel eines ElGamal-Kryptosystems erzeugt (vgl. Abschnitt 7.1.3). Der geheime Schlüssel $sk = (p, g, a)$ bestehend aus einer Primzahl p, einem Generator g der zyklischen Gruppe $\mathbb{Z}_p$ und einem zufällig gewählten Element $a \in \mathbb{Z}_p$. Der öffentliche Schlüssel $pk = (p, g, z)$ enthält neben p und g das Verschlüsselungselement $z = g^a$. Alle Rechenoperationen werden modulo p ausgeführt.[3]

Aus dem geheimen Entschlüsselungselement a werden entsprechend des in Abschnitt 7.1.4.1 bereits beschriebenen Secret Sharings nach Shamir [Sha79] n Geheimnisteile $D_1 = (x_1, y_1 = q(x_1)), \ldots, D_i = (x_i, y_i = q(x_i)), \ldots, D_n = (x_n, y_n = q(x_n))$ erzeugt und an die Geheimnisträger verteilt. Das Element a wird danach vernichtet. Der öffentliche Schlüssel pk wird unverändert zur Verschlüsselung einer Nachrichten m verwendet. Dabei wählt der Sender S ein zufälliges Element $b \in \mathbb{Z}_p$, erzeugt daraus den geheimen Sitzungsschlüssel $k = z^b = (g^a)^b$ und errechnet dann den Schlüsseltext $(c_1, c_2) = (g^b, k \cdot m)$.

3. In Abschnitt 7.1.3 sind hybride und EC-basierte Varianten des ElGamal-Kryptosystems beschrieben, die eine wesentliche Effizienzsteigerung erlauben. Sie können auch an dieser Stelle entsprechend eingesetzt werden.

Für die Entschlüsselung muss nun nicht das geheime Entschlüsselungselement a rekonstruiert werden, sondern können t partielle Entschlüsselungsteile erzeugt und von einem dedizierten Individuum eingesammelt werden. Diese partiellen Entschlüsselungsteile (x'_i, d_i) ergeben sich aus den Geheimnisteilen $D'_1 = (x'_1, y'_1), \ldots, D'_t = (x'_t, y'_t)$ mit $d_i = c_1^{y'_i} = (g^b)^{y'_i}$ für alle $i \in \{1, \ldots, t\}$. Mit allen t partiellen Entschlüsselungsteilen und der Lagrange-Polynominterpolation kann das dedizierte Individuum dann den Sitzungsschlüssel $g^{a \cdot b}$ mit

$$g^{a \cdot b} = \prod_{i=1}^{t} d_i^{\ell_i(0)} = \prod_{i=1}^{t} (g^b)^{y'_i \cdot \ell_i(0)} = g^{b \cdot \sum_{i=1}^{t} y'_i \cdot \ell_i(0)} = g^{b \cdot a}$$

berechnen. Der letzte Gleichungsschritt folgt aus der in Abschnitt 7.1.4.1 erläuterten Polynominterpolation.

7.1.5 Bloom-Filter

Bloom-Filter sind probabilistische Einwegdatenstrukturen, die verwendet werden, um Zugehörigkeitsinformationen zu einer Menge in einer sehr speichereffizienten Art und Weise abzuspeichern und abzufragen [Blo70]. Zum Einsatz kommen dabei die in Abschnitt 7.1.2 bereits eingeführten Hashfunktionen. Der Filter selbst besteht aus einem Bit-Array fester Länge m, das initial überall mit dem Wert 0 besetzt wird. Mit r verschiedenen Hashfunktionen $H_i : M^* \mapsto \{0, \ldots, m-1\}, i \in \{1, \ldots, r\}$, die jeweils alle Eingaben auf eine einzige Position im besagten Bit-Array abbilden, können neue Daten in diesem Filter eingefügt werden, indem die r Hashwerte, also r Positionen, dieser Daten berechnet und die Bits im Bit-Array an den jeweils berechneten Positionen auf den Wert 1 gesetzt werden.

Um nach der Konstruktion des Bloom-Filters die Zugehörigkeit einer beliebigen Eingabe zu testen, werden die r Hashwerte dieser Eingabe genau wie im Konstruktionsschritt berechnet und die resultierenden Positionen im Bloom-Filter nachgeschlagen. Wenn alle Bits an diesen r Positionen mit dem Wert 1 belegt sind, dann ist die Eingabe wahrscheinlich im Bloom-Filter enthalten.

Die Speichereffizienz wird auf Kosten von möglichen falsch-positiven Ergebnissen bei der Abfrage der Zugehörigkeit erreicht. Die durch die r Hashfunktionen berechneten Positionen einer nicht in einem Bloom-Filter enthaltenen Eingabe könnten durch andere bereits hinzugefügte Daten im Bloom-Filter auf den Wert 1 gesetzt worden sein. Dadurch würde der Zugehörigkeitstest fälschlicherweise eine Zugehörigkeit für dieses Nicht-Mitglied angeben. Auf der anderen Seite gibt ein Filter, bei dem mindestens ein Bit an den Positionen der Hashwerte einer Eingabe auf dem Wert 0 steht, eine definitive Unzugehörigkeit für diese Eingabe an.

Die Falsch-Positiv-Wahrscheinlichkeit für ein Nicht-Mitglied hängt von der Größe m des Bloom-Filters, von der Anzahl r der Hashfunktionen sowie von der Anzahl n an bereits eingefügten Datenwerten in den Bloom-Filter ab. Unter der Annahme, dass die genutzten Hashfunktionen die Hashwerte gleichverteilt über die Bit-Positionen des Bloom-Filters berechnen, lässt sich die Falsch-Positiv-Wahrscheinlichkeit abschätzen durch:

$$\left(1 - \left(1 - \frac{1}{m}\right)^{r \cdot n}\right)^r \approx \left(1 - e^{-r \cdot n/m}\right)^r$$

Eingehende Untersuchungen der Falsch-Positiv-Raten von Bloom-Filtern werden von Mitzenmacher und Upfal [MU05, Abschnitt 5.5.3] durchgeführt.

7.1.6 Secure Indexes

Secure Indexes bieten die Möglichkeit, in verschlüsselten Dokumenten nach Schlüsselwörtern zu suchen, indem speziell gefertigte Indizes befragt werden, die die Vertraulichkeit der indizierten Schlüsselwörter aufrechterhalten [Goh03]. Jeder Secure Index basiert auf einem Bloom-Filter (vgl. Abschnitt 7.1.5), der die Schlüsselwörter eines zugehörigen Dokumentes kodiert. Schlüsselwörter werden einer zweistufigen Kodierung unterzogen, bevor sie in den Bloom-Filter eingefügt werden.

1) Ein Verbergen des Schlüsselwortes durch die Anwendung einer Pseudozufallsfunktion f auf sowohl das Schlüsselwort w als auch einen geheimen Schlüssel $K = (k_1, \ldots, k_r)$. Die Ausgabe $x = (x_1 = f(w, k_1), \ldots, x_r = f(w, k_r))$ wird Trapdoor (deut. Falltür) genannt.
2) Eine Personalisierung jeder Trapdoor zum zugehörigen Dokument durch die nochmalige Anwendung der Pseudozufallsfunktion f auf sowohl die Trapdoor x als auch den eindeutigen Dokumentenkennzeichner D_{id}. Das Ergebnis $y = (y_1 = f(x_1, D_{id}), \ldots, y_r = f(x_r, D_{id}))$ wird Codewort genannt, dessen Elemente y_i in den Bloom-Filter eingefügt werden.

Der zweite Schritt wird durchgeführt, um verschiedene Codewörter für identische Schlüsselwörter in unterschiedlichen Dokumenten zu erreichen. Andernfalls wären Schnittmengenanalysen mit häufigen Schlüsselwörtern über verschiedene Dokumente möglich.

Der Bloom-Filter wird zusammen mit dem verschlüsselten Dokument als dessen Secure Index abgespeichert. Um abzufragen, ob ein Dokument ein spezielles Schlüsselwort enthält, muss die Trapdoor des Schlüsselwortes sowie dessen Personalisierung mit dem Dokumentenkennzeichen berechnet werden. Mit dem resultierenden Codewort kann getestet werden, ob der zugehörige Bloom-Filter dieses Codewort enthält.

7.1.7 1-out-of-N Oblivious Transfer

1-out-of-N Oblivious Transfer (OT) bezeichnet eine kryptographische Funktion, die eine Lösung für das folgende Problem liefert: Ein Sender $\mathcal{S}$ hat N Nachrichten $M_0, \ldots, M_{N-1}$ und ein Empfänger $\mathcal{R}$ möchte die i-te Nachricht abrufen, ohne dass $\mathcal{S}$ Information darüber gewinnt, für welche Nachricht sich $\mathcal{R}$ interessiert. Zusätzlich soll $\mathcal{R}$ allerdings auch keine Information über irgendeine der nicht angefragten Nachrichten $M_j \neq M_i$ erhalten.

Ein sehr einfaches 1-out-of-N OT-Protokoll basierend auf dem sogenannten *Computational-Diffie-Hellman-Problem*[4] wurde von Naor und Pinkas [NP01] vorgeschlagen und von Chou und Orlandi [CO15] in seiner Effizienz stark verbessert. Die grundlegende Funktionsweise lässt sich wie folgt beschreiben, wobei die Operationen auch entsprechend auf elliptische Kurven übertragen werden können [CO15]:

4. Das Computational-Diffie-Hellman-Problem besteht darin, aus einer gegebenen zyklischen Gruppe $\mathbb{G}$ und einem Generator $g \in \mathbb{G}$ sowie zwei Elementen $r, s \in \mathbb{G}$, die berechnet wurden mit $r = g^x$ und $s = g^y$, den Wert $g^{x \cdot y} = r^y = s^x$ zu berechnen, wobei x und y nicht bekannt sind [KL14, Abschnitt 8.3.2].

- **Vorbereitung**: Das Protokoll verwendet eine Schlüsselableitungsfunktion (engl. key derivation function) `kdf` sowie eine Gruppe $\mathbb{Z}_p$ mit der Ordnung einer Primzahl p und einem zugehörigen Generator g dieser Gruppe. Alle Operationen werden modulo p ausgeführt.
- **Initialisierung** (nur einmalig; wird für alle folgenden Schritte verwendet):
  1) $\mathcal{S}$ wählt ein zufälliges Geheimnis $y \in \mathbb{Z}_p$ und berechnet $s = g^y$ sowie $u = s^y$.
  2) $\mathcal{S}$ sendet s an $\mathcal{R}$, der das Protokoll abbricht, sofern $s \notin \mathbb{Z}_p$.
- **Eingaben/Ausgabe**: $\mathcal{R}$'s Eingaben sind der Index $i \in \{0, \ldots, N-1\}$ der gewünschten Nachricht sowie der von $\mathcal{S}$ erhaltene öffentliche Schlüssel s. $\mathcal{S}$'s Eingaben sind die Nachrichten $M_0, \ldots, M_{N-1}$. Am Ende des Protokolls ist die Ausgabe von $\mathcal{R}$ ausschließlich die gewünschte Nachricht M_i, aber keine Informationen über die anderen Nachrichten, während $\mathcal{S}$ nichts über i lernt.
- **Schlüsselberechnung** (für jeden Index, der für $\mathcal{R}$ von Interesse ist; auch parallel möglich):
  1) $\mathcal{R}$ wählt ein zufälliges Geheimnis $x \in \mathbb{Z}_p$ und berechnet $r = s^i \cdot g^x$ sowie $K_i = \mathtt{kdf}(s^x) = \mathtt{kdf}(g^{y \cdot x})$.[5]
  2) $\mathcal{R}$ sendet r an $\mathcal{S}$, der das Protokoll abbricht, sofern $r \notin \mathbb{Z}_p$.
  3) $\mathcal{S}$ berechnet für alle $j \in \{0, \ldots, N-1\}$: $K_j = \mathtt{kdf}(r^y / u^j) = \mathtt{kdf}(g^{(y \cdot i + x) \cdot y} / g^{y \cdot y \cdot j})$.
- **Übertragung** (für jeden Index, der für $\mathcal{R}$ von Interesse ist; auch parallel möglich):
  1) $\mathcal{S}$ verschlüsselt alle M_j für $j \in \{0, \ldots, N-1\}$, indem er $C_j = \mathtt{Enc}(K_j, M_j)$ berechnet und sendet diese Verschlüsselungen $(C_0, \ldots, C_{N-1})$ an $\mathcal{R}$.
  2) $\mathcal{R}$ entschlüsselt C_i, indem er $M_i = \mathtt{Dec}(K_i, C_i)$ berechnet.

Die Forschung auf dem Gebiet von OT ist sehr vielfältig und schnelllebig. Viele Varianten von OT-Protokollen und sogenannten Erweiterungen existieren, die die Sicherheit oder die Effizienz zu erhöhen versuchen. Für mehr Details sei auf [Ash+17] verwiesen.

7.2 Anwendungsszenario und Anforderungen

Das folgende Szenario entspricht dem Kontext des in Kapitel 5 entwickelten Mechanismus zur Erkennung und Abwehr von Insiderbedrohungen. Daran angelehnt werden die Anforderungen noch einmal zusammengefasst, die umgesetzt werden müssen, damit der vorgeschlagene Mechanismus einerseits dem Ziel der Insiderbedrohungserkennung und -prävention gerecht wird und andererseits den Datenschutz der betroffenen Nutzer respektiert und ermöglicht. Die Anforderungen wurden im Detail bereits in Abschnitt 6.2 hergeleitet und erläutert.

5. In der Veröffentlichung von Chou und Orlandi [CO15] schlagen die Autoren das Voranstellen der Werte s und r als Salt der Schlüsselableitungsfunktion vor. Damit wird den Autoren zufolge ein Zufallsorakel approximiert, was sicherstellt, dass das Orakel lokal für eine Protokollsitzung ist. Das wiederum soll gegen sogenannte Malleability-Attacks (deut. etwa Umformungsangriffe) helfen. Der Einfachheit halber wird dieses Detail hier weggelassen.

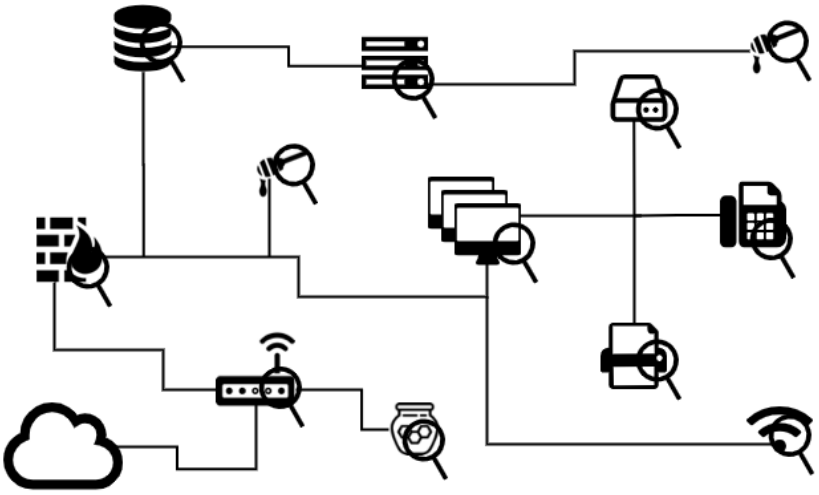


Abbildung 7.1: Das Veracity-Prinzip im Unternehmensnetzwerk

7.2.1 Veracity im Unternehmenskontext

Szenario 7.1

Ein Arbeitgeber hat ein verteiltes System zur Erkennung von Sicherheitsvorfällen in seinem Unternehmen umgesetzt, welches aus mehreren Sensoren besteht, die über die gesamte IT- und physische Infrastruktur verteilt sind. Ziel ist es, Nutzeraktivitäten von mehreren Standpunkten aus überwachen zu können und unerlaubtes Eindringen, unerlaubtes Herausschleusen von Informationen sowie Anomalien im Verhalten der Arbeitnehmer erkennen und verhindern zu können. Alle Sensor- beziehungsweise Ereignisdaten werden an einer zentralen Stelle zusammengefasst, was die Möglichkeit eröffnet, die Daten umfassend zu korrelieren und zu analysieren. Die anfallenden Ereignisdaten sollen für möglicherweise auftretende zeitlich nachgelagerte Untersuchungen im Falle von Sicherheitsvorfällen und als potenzielle Beweisquelle im Falle von sich gegebenenfalls anschließenden Ermittlungen für einen bestimmten Zeitraum archiviert werden.

Dieser als *Veracity* [Gol12] bezeichnete Ansatz, der in Abbildung 7.1 illustriert ist, begegnet einerseits der Gefahr durch U-Insiderbedrohungsakteure, da Aktivitäten, die vorher unerkennbar waren, nun nachvollziehbar werden. Andererseits begegnet der Ansatz auch der Gefahr durch C-K-P-Insiderangreifer, die weitreichende Zutritts-, Zugangs- und Zugriffsrechte sowie detailliertes Insiderwissen haben und somit ihr Verhalten entsprechend der vorhandenen Sicherheitsmechanismen anpassen und Überwachungsmechanismen beziehungsweise Überwachungsdaten manipulieren können, um ihre Angriffsaktivitäten zu verbergen. Die umfassende Überwachung und Aufzeichnung von Aktivitäten in Form von Ereignisdaten von mehreren Standpunkten aus erschwert es den Angreifern, ihre Spuren zu verwischen, da sie nun die Überwachungsdaten an vielen verschiedenen Orten in einer konsistenten Art und Weise manipulieren müssten. Die Manipulation von Daten an einem Standpunkt erzeugt aber durch den von Gollmann [Gol12] formulierten Veracity-Ansatz neue Spuren an anderer Stelle, sodass in der Gesamtheit der Überwachungsdaten nachweisbare Inkonsistenzen auftreten, die es erlauben, die Wahrhaftigkeit (engl. veracity) der vorliegenden Informationen zu verifizieren.

7.2.2 Charakterisierung der zu entwickelnden Datenschutztechnik

Die Natur und der Umfang der gesammelten Ereignisdaten haben schwere Implikationen für die Privatsphäre der Arbeitnehmer, die auch am Arbeitsplatz ein Recht auf Datenschutz haben (vgl. Abschnitt 6.1). Die eingehende Untersuchung von kombinierten und korrelierten Daten erlaubt Ableitungen von persönlichen Informationen, Gewohnheiten, Präferenzen und Arbeitsleistungen [AD08; Sol17]. Die Einschränkung des Zugriffs auf diese Daten nur für Sicherheitspersonal und den Arbeitgeber entfernt diese Implikationen für die Privatsphäre nicht. Um die Datenschutzgefahren zu reduzieren, ist die Technik der Pseudonymisierung eine angemessene datenschutzerhöhende Maßnahme. Auf der einen Seite erlaubt sie die Verkettbarkeit der Ereignisdaten, die sich unabhängig vom Ereignisstandort auf ein Individuum beziehen, und die gegebenenfalls miteinander korreliert werden müssen, um nützliche Einblicke in die Aktivitäten von Nutzern zu bekommen und Anomalien feststellen zu können. Auf der anderen Seite erlaubt Pseudonymisierung die Separierung und Geheimhaltung von realen Identitäten und QIDs. Darüber hinaus erlaubt sie die Re-Identifizierung von Subjekten im Falle von tatsächlich erkannten Sicherheitsvorfällen und anschließenden Investigationen.

Aus dem vorliegenden Szenario 7.1 werden die Sensordaten von Nutzergeräten und Servern in Form von Auditierungsnachrichten exemplarisch für das folgende Pseudonymisierungsverfahren zugrunde gelegt, wie sie auch in Kapitel 5 für die Erkennungs- und Abwehrtechnik von Insiderbedrohungen verwendet werden. Drei Beispiele für solche Ereignisnachrichten sehen wie folgt aus:

```
1 8408: {node: mw-host, timestamp: 1537903944.0, types: {USER_AUTH: {acct:
    mw, addr: 172.16.215.198, auid: unset, exe: /usr/sbin/sshd, hostname:
    172.16.215.198, op: PAM:authentication, pid: 2150, res: success,
    terminal: ssh, uid: root}}}
2 10251: {node: mw-host, timestamp: 1538019655.0, types: {CWD: {cwd:
    /home/mw}, PATH: [{inode: 1091237, item: 0, mode: file,644, name:
    /etc/ld.so.cache, nametype: NORMAL}], PROCTITLE: {proctitle: openssl
    enc -aes-256-cbc -a -salt -in /home/mw/dev/102.txt -out
    /home/mw/dev/102.txt.lol -}, SYSCALL: {a0: 0xffffff9c, a1:
    0x7fbc28a98428, a2: O_RDONLY|O_CLOEXEC, a3: 0x0, auid: mw, comm:
    openssl, exe: /usr/bin/openssl, exit: 3, gid: mw, items: 1, pid:
    1545, ppid: 1536, success: yes, syscall: openat, tty: pts0, uid: mw}}}
3 31317: {node: mw-host, timestamp: 1547121809.485, types: {PROCTITLE:
    {proctitle: /home/mw/Downloads/WTEpZSFwgb -ipc.fd=3 scan}, SOCKADDR:
    {addr: 192.168.225.69, family: inet, port: 82}, SOCKETCALL: {a0:
    0x9d, a1: 0x19d85988, a2: 0x10, nargs: 3}, SYSCALL: {a0: 0x19302860,
    a1: 0x0, a2: 0x0, arch: i386, auid: mw, comm: WTEpZSFwgb, exe:
    /home/mw/Downloads/WTEpZSFwgb, exit: -115(EINPROGRESS), gid: mw,
    items: 0, pid: 1750, ppid: 1734, success: yes, syscall: connect, tty:
    pts0, uid: mw}}}
```

Die hervorgehobenen Datenfelder werden dabei als QIDs behandelt und durch Pseudonyme ersetzt, wobei vor dem Hintergrund der in Abschnitt 6.2.2.3 ausgeführten Erkenntnisse bezüglich QIDs mit dieser Auswahl kein Anspruch auf Vollständigkeit erhoben wird.

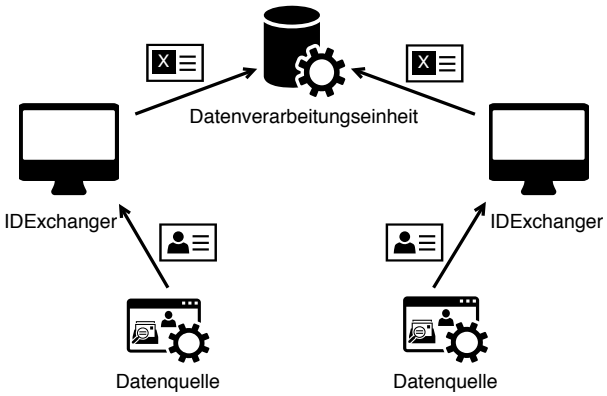
7.2.3 Funktionale Anforderungen

Neben dem Entfernen von Datenfeldern, die für die Insiderbedrohungserkennung und -prävention nicht benötigt werden, wird Pseudonymisierung als Technik der De-Identifizierung eingesetzt, wo immer QIDs vorliegen. Die Pseudonymisierung ermöglicht eine Verkettbarkeit von Ereignissen und muss daher global konsistent erfolgen, damit unterschiedliche Sensordaten den tatsächlich agierenden Personen zugeordnet werden können. Weiterhin soll die Pseudonymisierung umkehrbar sein, sodass in besonderen Fällen eine Re-Identifizierung und damit eine Attribution möglich ist. Die Entscheidung, wann eine Re-Identifizierung durchgeführt werden darf, soll zum Schutz vor unrechtmäßiger Verarbeitung der Daten zuverlässig auf mehrere unabhängige Interessenvertreter und Entscheidungsträger verteilt und technisch durchgesetzt werden. Ein weiterer Schritt zur Datenminimierung sowie zur Speicherbegrenzung ist die Limitierung der Verkettbarkeit auf einen bestimmten Zeitraum oder auf ein bestimmtes Budget. Letztlich sollen keine weiteren Informationen durch das Pseudonymisierungsverfahren selbst anfallen, die potenziell personenbeziehbar sind. Für eine detailliertere Betrachtung dieser Anforderungen allgemein im Kontext der datenschutzfreundlichen Erkennung und Abwehr von Insiderbedrohungen sei auf Abschnitt 6.2 verwiesen.

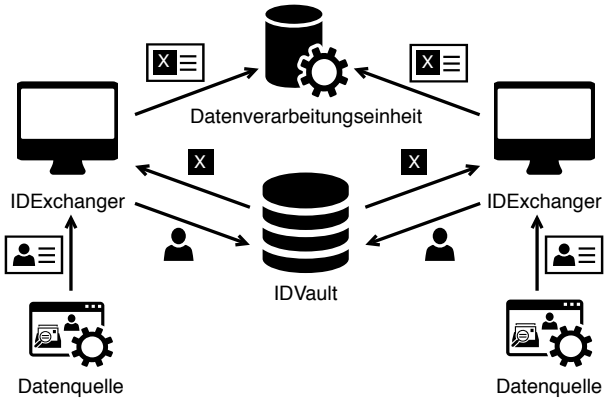
Eine letzte Anforderung hat in ihrer praktischen Umsetzung weitreichende Folgen. Aufgrund der geringen Anzahl an unterschiedlichen möglichen QIDs sollen die Pseudonyme nicht nur allein in Abhängigkeit der jeweiligen QIDs abgeleitet, sondern mit weiterer Entropie angereichert werden. Diese Anforderung zusammen mit der benötigten globalen Konsistenz der Pseudonyme resultiert entweder in einer deterministischen Ableitung von Pseudonymen in Verbindung mit einem geheimen Schlüssel, der allerdings jeder Pseudonym-erzeugenden Stelle bekannt sein muss. Hierbei kann auf eine zentrale Pseudonymisierungsinstanz verzichtet werden. Oder es wird eine zentrale Instanz benötigt, die die Pseudonyme zufällig erzeugen und auf bereits existierende zufällige Pseudonyme für auftretende QIDs hin befragt werden kann. Beide Umsetzung resultieren in unterschiedlichen System- und Bedrohungsmodellen (s. Abschnitt 7.3 und Abschnitt 7.4).

7.2.4 Außerhalb der Betrachtungen

Einige weitere Anforderungen sind für die Korrektheit und Sicherheit einer Pseudonymisierung sowie die Nützlichkeit einer Technik zur Bedrohungserkennung und -prävention von großer Bedeutung oder stellen zumindest einen interessanten Anwendungsfall dar. Sie werden allerdings im Zuge dieser Dissertation nicht weiter betrachtet. Im Speziellen ist das die Authentizität in Bezug auf den Inhalt und den Ursprung der Ereignisnachrichten. Alle beteiligten Komponenten müssen sicherstellen, dass die verarbeiteten Daten authentisch sind. Weitere Anforderungen könnten die Unfälschbarkeit und Integrität der Pseudonymzuordnung, die Anwendung von Gruppenpseudonymen beziehungsweise identische Pseudonyme für eine Menge von QIDs oder die Etablierung einer limitierten Verkettbarkeit entsprechend eines Sitzungskonzepts sein. Sofern diese Anforderungen wichtig sind, müssen sie separat zu den hier vorgestellten Konzepten umgesetzt und durchgesetzt werden.



(a) ohne zentrale Pseudonymisierungsinstanz



(b) mit zentraler Pseudonymisierungsinstanz

Abbildung 7.2: Systemmodelle (a) und (b) des Pseudonymisierungsverfahrens

7.3 Systemmodelle des Pseudonymisierungsverfahrens

Die identifizierten und diskutierten Anforderungen resultieren in den nachfolgend beschriebenen Systemmodellen (a) und (b) des entwickelten Pseudonymisierungsverfahrens. Gemeinsam haben beide Modelle, dass eine Datenquelle Ereignisnachrichten emittiert, die potenziell personenbezogene Daten enthalten. Ein sogenannter *IDExchanger*, der dieser Datenquelle zugeordnet ist, hat die Aufgabe, auftretende QIDs in den Ereignisnachrichten durch Pseudonyme zu ersetzen. Wie diese Pseudonymisierung erfolgt, unterscheidet sich darin, ob eine zentrale Pseudonymisierungsinstanz beteiligt ist oder nicht, und wird nachfolgend getrennt beleuchtet. Die pseudonymisierte Ereignisnachricht wird letztendlich vom IDExchanger zur Datenverarbeitungseinheit gesendet, die für die eigentliche Erkennung von Bedrohungen und Anomalien zuständig ist. Bei der Datenverarbeitungseinheit kann es sich beispielsweise um ein System handeln, das die in Kapitel 5 entwickelte Erkennung und Abwehr von Insiderbedrohungen realisiert oder das die pseudonymisierten Daten anderweitig analysiert oder archiviert. Auf sie wird in diesem Kapitel nicht weiter eingegangen. Die beiden Systemmodelle sind in Abbildung 7.2 veranschaulicht.

Die Integrität und Vertraulichkeit aller Kommunikationen wird durch geeignete Mechanismen geschützt, aber hier nicht näher betrachtet. Der Pseudonymisierungsprozess ist im Allgemeinen transparent für die Datenquellen und die Datenverarbeitungseinheit. Sofern realisierbar, ist es sinnvoll, den IDExchanger direkt in die Datenquelle zu integrieren, damit die Pseudonymisierung so früh wie möglich im Zyklus der Ereignisnachrichten umgesetzt wird.

7.3.1 Systemmodell (a) ohne zentrale Pseudonymisierungsinstanz

Für die eigentliche Pseudonymisierung extrahiert der IDExchanger jeden QID einzeln und ersetzt diesen mit einem Pseudonym, das nach einer vordefinierten Berechnungsvorschrift aus dem QID abgeleitet wird. Diese Berechnungsvorschrift ist allen IDExchangern bekannt, um globale Konsistenz der Pseudonyme zu erreichen. Weiterhin teilen sich alle IDExchanger ein gemeinsames Geheimnis, das als zusätzliche Entropie in die Berechnung der Pseudonyme mit einfließt. Dadurch wird sichergestellt, dass die Pseudonyme nicht nur von den QIDs abhängen. Eine eventuell erforderliche Re-Identifizierung wird durch die Umkehrung der Berechnungsvorschrift erreicht.

7.3.2 Systemmodell (b) mit zentraler Pseudonymisierungsinstanz

Mit einer zentralen Pseudonymisierungsinstanz müssen die Pseudonyme nicht von den IDExchangern deterministisch berechnet, sondern können zentral zufällig erzeugt werden. Dafür extrahiert der IDExchanger jeden QID einzeln, fordert jeweils ein Pseudonym vom sogenannten *IDVault* an und ersetzt den QID mit dem vom IDVault erhaltenen Pseudonym. Der IDVault erhält eine Pseudonymanfrage von einem IDExchanger und kann dann mit zwei Situationen konfrontiert sein. Erstens, es existiert bereits ein Pseudonym für den angefragten QID in seiner Pseudonymzuordnung (vgl. Abschnitt 7.1.1), das somit als Antwort auf die Pseudonymanfrage an den IDExchanger zurückgesendet werden kann. Und zweitens, es existiert noch kein Pseudonym für diesen QID in der Pseudonymzuordnung. In diesem Fall wird ein neues zufälliges Pseudonym generiert und vom IDVault zusammen mit dem korrespondierenden QID in einer Pseudonymzuordnungstabelle abgespeichert. Das Pseudonym wird anschließend an den anfragenden IDExchanger ausgeliefert.

Mit der Anforderung von echt zufälligen Pseudonymen bei gleichzeitiger Aufrechterhaltung der globalen Konsistenz aller Pseudonyme wird eine solche zentralen Instanz benötigt. Die Pseudonymzuordnungstabelle dient dabei nicht nur der globalen Konsistenz, sondern auch der Re-Identifizierung, sofern dies später einmal notwendig wird. Für den Pseudonymisierungsprozess selbst ist eine Kommunikation nur zwischen den IDExchangern und dem IDVault nötig. Die IDExchanger müssen untereinander nicht kommunizieren.[6]

7.4 Bedrohungsmodell des Pseudonymisierungsverfahrens

Mit dem im Rahmen dieser Dissertation entwickelten Pseudonymisierungsverfahren wird nicht der Anspruch eines beweisbaren Datenschutzes gegenüber starken externen oder internen Angreifern erhoben. Tatsächlich verbleiben verschiedene Bedrohungen für den Datenschutz der betroffenen Personen, die direkt die umgesetzten Datenschutzmechanismen untergraben, hauptsächlich aufgrund ihrer fundamentalen Natur. Für beide Systemmodelle kann

- eine bösartige Datenquelle den Pseudonymisierungsprozess für alle lokal verarbeiteten Ereignisnachrichten aushebeln,

6. Auch wenn die IDExchanger alle identische Funktionalität sowie identische Interaktion mit dem IDVault aufweisen, können sie dennoch nicht als eine logische Komponente betrachtet werden, da sie kein Wissen außerhalb des Bereichs ihrer Datenquellen erhalten sollen.

- ein bösartiger IDExchanger seine eigene lokale Pseudonymzuordnungstabelle mit allen lokal verarbeiteten QIDs und zugehörigen Pseudonymen erstellen sowie
- ein bösartiger IDExchanger seine eigenen (zufälligen oder nicht) Pseudonyme verwenden.

Im Systemmodell (b) mit zentraler Pseudonymisierungsinstanz kommt dazu, dass ein bösartiger IDVault das Nachschlagen von bereits vorhandenen QIDs und zugehörigen Pseudonymen sowie die zufällige Generierung von Pseudonymen manipulieren kann.

Abgesehen von organisatorischen Vorgaben und Maßnahmen gibt es keine Schutzmechanismen, die im vorliegenden Pseudonymisierungsverfahren umgesetzt werden könnten, um sich vor diesen Bedrohungen zu schützen. Eine leichte Minderung gegen diese fundamentalen Datenschutzbedrohungen kann höchstens durch eine strikte Trennung aller Komponenten und eine Verhinderung der unerlaubten Zusammenarbeit erreicht werden, da dadurch der Zugriff jeder Komponente auf die lokalen Daten beschränkt und eine globale Bedrohung sensibler Daten verhindert wird.

Dennoch müssen diese Komponenten, vom Gesichtspunkt eines Bedrohungsmodells her, als vertrauenswürdig angesehen und eine Kollaboration verboten werden. Darüber hinaus weisen die in Erkenntnis 3 in Abschnitt 6.2.2.3 erläuterten Forschungsarbeiten darauf hin, dass die Erreichung von absolutem Datenschutz mittels Pseudonymisierung grundsätzlich nicht möglich ist. Vor dem Hintergrund dieser grundsätzlichen Beschränktheit von Pseudonymisierung wird von einer binären Unterscheidung zwischen absolutem Datenschutz auf der einen Seite und absolut keinem Datenschutz auf der anderen abgesehen (vgl. [Swe97; Ohm09, Abschnitt II.C.2]). Im vorgeschlagenen Verfahren wird Datenschutz als ein steigendes oder fallendes informelles Kontinuum gesehen. Mit diesem Verständnis des Begriffs *Datenschutz* erschwert die Minimierung von auftretenden identifizierenden und quasi-identifizierenden Merkmalen die angesprochenen Datenschutzbedrohungen und erhöht die Privatsphäre der betroffenen Individuen. Umgekehrt stellt die Existenz derartiger QIDs inklusive anfallender Metadaten durch einen Pseudonymisierungsprozess selbst eine Bedrohung für den Datenschutz dar, denn sie vereinfachen eine Re-Identifizierung von Individuen. Der Fokus liegt daher auf der strikten Minimierung von QIDs sowie von Metadaten, die durch den Pseudonymisierungsprozess selbst anfallen, wie etwa Wiederverwendungsmuster von Pseudonymen. Es wird versucht, nur diejenigen Informationen beizubehalten, die von der Datenverarbeitungseinheit zur Aufrechterhaltung ihrer Funktionalität sowie für den Pseudonymisierungsprozess zwingend benötigt werden. Die übrigen Quellen potenzieller sensibler Informationen werden so akkurat wie möglich eliminiert. Das Resultat ist kein vollständiger, aber ein erhöhter Datenschutz. Im Einzelnen wird für beide Systemmodelle die Minimierung der folgenden (Meta-)Informationen erbracht:

- Die Kenntnis über den Zusammenhang zwischen QIDs und zugehörigen Pseudonymen[7] erlaubt die Durchführung einer Re-Identifizierung. Die Pseudonymisierungskomponenten (IDExchanger und gegebenenfalls IDVault), die genau über diese Kenntnisse verfügen, müssen allerdings an einem solchen Vorgang nicht beteiligt, sondern können davon ausgeschlossen werden. Dafür werden kryptographische Mechanismen eingesetzt, die eine solche Re-Identifizierung nur speziell berechtigten Personen unter speziellen Umständen ermöglicht (s. Abschnitte 7.5.1.2 und 7.5.2.3).
- Die Ansammlung der pseudonymisierten Ereignisnachrichten erlaubt ganz allgemein die Herstellung von individuellen Profilen über Pseudonyme, die mit der Zeit an Genauigkeit

7. Im Systemmodell (a) ohne zentrale Pseudonymisierungsinstanz ist das die Berechnungsvorschrift zur Ableitung von Pseudonymen. Im Systemmodell (b) mit zentraler Pseudonymisierungsinstanz ist das die Pseudonymzuordnungstabelle.

zunehmen, da alle Ereignisnachrichten, die dem gleichen QID zugehörig sind, miteinander über das entsprechende Pseudonym verkettbar sind. Diese Profile, wenn auch nicht direkt verknüpfbar mit der Identität eines Individuums, könnten einerseits sensible Informationen preisgeben und andererseits illegalerweise de-pseudonymisiert werden, was beides mit der Zeit eine steigende Erfolgswahrscheinlichkeit aufweist. Es wird versucht diese Bedrohung abzuschwächen, indem die Verkettbarkeit verschiedener Ereignisnachrichten begrenzt wird (s. Abschnitte 7.5.1.3 und 7.5.2.4).

Das Systemmodell (b) mit zentraler Pseudonymisierungsinstanz bringt zwar den Vorteil von echt zufälligen Pseudonymen. Es entstehen dadurch allerdings auch mehrere Datenschutzbedrohungen, die im vorliegenden Pseudonymisierungskonzept adressiert werden müssen:

- Der IDVault hat eine globale Sicht auf alle Klartext-QIDs, die im gesamten System vorliegen. Bedrohungen, die damit zusammenhängen, werden durch den Schutz der Vertraulichkeit aller QIDs gegenüber dem IDVault adressiert (s. Abschnitt 7.5.2.1).
- Der IDVault kann Nutzungsmuster aus Pseudonymanfragen als eine Art der Metainformation ableiten, die mit der Zeit die Möglichkeit eröffnet, zusätzliche sensible Informationen abzuleiten. Das entwickelte Verfahren begegnet dieser Bedrohung einerseits mit der Limitierung der Verkettbarkeit einzelner Ereignisnachrichten (s. Abschnitt 7.5.2.4) sowie andererseits mit der Unterbindung von Informationen für den IDVault, welche Einträge der Pseudonymzuordnungstabelle einem abgefragten Eintrag entsprechen (s. Abschnitt 7.5.2.2).

7.5 Datenschutzfreundliche Ereignispseudonymisierung mit begrenzter Verkettbarkeit

Das im Rahmen dieser Dissertation entwickelte Pseudonymisierungsverfahren wurde mit dem Akronym *PEEPLL* getauft, das für *Privacy Enhanced Event Pseudonymisation with Limited Linkability* steht. Zunächst folgt eine detaillierte Betrachtung von PEEPLL im Systemmodell (a) ohne zentrale Pseudonymisierungsinstanz aus Abschnitt 7.3.1. Damit kann eine einfache Erhöhung des Datenschutzes bei einer Insiderbedrohungserkennung und -prävention, wie sie etwa in Kapitel 5 entwickelt wurde, realisiert werden. Die sich daraus ergebende Folge von lokal deterministischen Pseudonymen wird anschließend detaillierten Betrachtungen zu PEEPLL mit echt zufälligen Pseudonymen und damit dem Systemmodell (b) mit zentraler Pseudonymisierungsinstanz aus Abschnitt 7.3.2 gegenübergestellt. Die Datenschutzbedrohungen erweitern sich dadurch allerdings auf die zentrale Pseudonymisierungsinstanz, was zu komplexeren Lösungsansätzen in PEEPLL führt.

Der jeweilige Schutz vor den identifizierten Datenschutzbedrohungen, wie sie bereits in Abschnitt 7.4 aufgezeigt wurden, werden dabei als *Pseudonymisierungs-Schutzziele* in Bezug auf den Entwurf und die Implementierung des vorgestellten Pseudonymisierungsverfahrens definiert. Jedes Schutzziel intendiert die Reduzierung der Auswirkungen, die die Technik zur Erkennung und Abwehr von Insiderbedrohungen auf den Datenschutz der betroffenen Personen hat. Die dabei auftretende übergreifende Herausforderung ist die Aufrechterhaltung der Effektivität dieser Erkennungs- und Präventionstechnik durch Aufrechterhaltung der Verkettbarkeit bestimmter Ereignisnachrichten.

7.5.1 PEEPLL mit lokal deterministischen Pseudonymen

Entsprechend der Ausführungen im Systemmodell (a) ohne zentrale Pseudonymisierungsinstanz in Abschnitt 7.3.1 sowie der Anforderungen in Abschnitt 7.2.3 haben die IDExchanger die Aufgabe, ohne gegenseitige Absprache auftretende QIDs mit global konsistenten Pseudonymen zu ersetzen. Darüber hinaus werden die folgenden Pseudonymisierungs-Schutzziele verfolgt und technisch durchgesetzt.

7.5.1.1 QID-Vertraulichkeit

Definition 7.1 (QID-Vertraulichkeit)**.** Der QID, der durch einen IDExchanger mit einem Pseudonym ersetzt werden soll, ist nur diesem IDExchanger bekannt. Insbesondere die Datenverarbeitungseinheit erfährt aus einem Pseudonym keine Informationen über den zugrunde liegenden QID.

In den nachfolgenden Abschnitten wird zwischen einer *schwachen* und einer *starken QID-Vertraulichkeit* differenziert.

Definition 7.2 (schwache QID-Vertraulichkeit)**.** Die Vertraulichkeit des pseudonymisierten QIDs hängt allein von der Geheimhaltung des QIDs ab.

Das bedeutet, dass aus den veröffentlichten oder weitergegebenen Informationen eines IDExchangers im Zuge einer QID-Pseudonymisierung der pseudonymisierte QID allein durch Erraten oder vollständiges Durchprobieren aller möglicher QIDs gefunden werden kann. *Schwache QID-Vertraulichkeit* kann beispielsweise erreicht werden, indem jeder QID mit dessen Hashwert pseudonymisiert wird. Mit der Kenntnis eines Pseudonyms kann allerdings ein Bruteforce-Angriff durchgeführt werden, indem alle möglichen QIDs gehasht und mit dem bekannten Pseudonym verglichen werden. Die Anzahl aller möglichen QIDs ist in der Regel sehr gering, sodass ein solcher Angriff schnell zum Ziel führt, wie Marx u. a. [Mar+18] zeigen konnten (vgl. auch Abschnitt 7.1.2). Durch Hinzunahme von zusätzlicher Entropie kann dieser Angriff erheblich erschwert werden, was zu einer *starken QID-Vertraulichkeit* führt.

Definition 7.3 (starke QID-Vertraulichkeit)**.** Die Vertraulichkeit des pseudonymisierten QIDs hängt nicht allein von der Geheimhaltung des QIDs, sondern darüber hinaus von der Geheimhaltung eines geheimen Schlüssels ab.

Das bedeutet, dass aus den veröffentlichten oder weitergegebenen Informationen eines IDExchangers im Zuge einer QID-Pseudonymisierung der pseudonymisierte QID nicht allein durch Erraten oder vollständiges Durchprobieren aller möglichen QIDs gefunden werden kann. PEEPLL setzt zur Erreichung der starken QID-Vertraulichkeit HMACs ein, indem alle IDExchanger zu Anfang des Pseudonymisierungsverfahrens von einer vertrauenswürdigen dritten Stelle (engl. trusted third party, TTP) mit einem gemeinsamen geheimen Schlüssel d von ausreichender Länge ausgestattet werden.[8] Mit gegebenem d und einem *QID* errechnet der IDExchanger das

8. Für die Erbringung des Pseudonymisierungs-Schutzziels *Re-Identifizierung nach dem Mehr-Augen-Prinzip* kommt noch eine weitere Aufgabe für die TTP hinzu, die in Abschnitt 7.5.1.2 erläutert wird. Danach wird sie nicht mehr benötigt.

Pseudonym $P_{QID} = \texttt{Mac}(d, QID)$, das mithilfe der Mac-Funktion eines HMACs errechnet wird (vgl. Abschnitt 7.1.2). Darüber hinaus verwenden alle IDExchanger für die Berechnung des Pseudonyms den gleichen geheimen Schlüssel d, sodass P_{QID} über das gesamte System konsistent ist und so *Globale Konsistenz* aufrechterhalten wird.

7.5.1.2 Re-Identifizierung nach dem Mehr-Augen-Prinzip

Definition 7.4 (Re-Identifizierung nach dem Mehr-Augen-Prinzip)**.** Eine Re-Identifizierung ist derart beschränkt, dass sie nur durchgeführt werden kann, wenn an der Pseudonymisierung unbeteiligten Parteien nach dem Mehr-Augen-Prinzip kooperieren.

PEEPLL setzt eine erweiterte Variante des ursprünglich von Shamir [Sha79] veröffentlichten Secret Sharings ein, um das benötigte verteilte Vertrauen bei der Re-Identifizierung zu erreichen (vgl. Abschnitt 7.1.4). Das Schwellwertschema von Desmedt und Frankel [DF90] bringt den Vorteil, dass die einmal verteilten Geheimnisteile bei einer Zusammenführung für eine Re-Identifizierung nicht offengelegt und somit weiterverwendet werden können. Ein einmal re-identifiziertes Pseudonym erlaubt natürlich die Verknüpfung aller mit diesem Pseudonym de-identifizierten Ereignisnachrichten mit dem zugehörigen QID. Andere Pseudonyme jedoch, die mit denselben Geheimnisteilen geschützt sind, bleiben davon unberührt.

Zu Anfang des Pseudonymisierungsverfahrens wird daher einmalig ein ElGamal-basiertes Schwellwertschema von einer TTP instanziiert. Die zugehörigen Geheimnisteile werden an geeignete Entscheidungsträger und Interessenvertreter der Betroffenen verteilt. Der Verschlüsselungsschlüssel $pk = (p, g, z)$ wird zusammen mit dem in Abschnitt 7.5.1.1 erläuterten geteilten geheimen Schlüssel d an alle IDExchanger verteilt. Danach wird die TTP nicht mehr benötigt.

Für jeden *QID*, den ein IDExchanger bearbeitet, berechnet dieser ein deterministisches Pseudonym, indem *QID* zusammen mit d mittels pk verschlüsselt wird. Die benötigten Informationen zur Re-Identifizierung sind somit im Pseudonym enthalten. Da kein IDExchanger Kenntnis von einem der Geheimnisteile des Schwellwertschemas hat, kann also auch kein IDExchanger an einer partiellen Entschlüsselung zum Zweck einer Re-Identifizierung beteiligt werden.[9] Im Detail berechnet sich das Pseudonym P_{QID} zu einem *QID* folgendermaßen:

1. Mithilfe einer Funktion $f : \{0,1\}^n \mapsto \mathbb{Z}_p$ wird aus $MAC_{QID} = \texttt{Mac}(d, QID)$ ein Pseudonymgeheimnis $k \in \mathbb{Z}_p$ errechnet: $f(MAC_{QID}) = k$.[10] Die Berechnung des MAC_{QID} zusammen mit dem gemeinsamen Geheimnis d sorgt dafür, dass k nicht nur von *QID* abhängt, was sonst Schwächen für Bruteforce-Angriffe eröffnen würde. Dadurch wird eine starke QID-Vertraulichkeit erreicht (vgl. Definition 7.3).

2. Mithilfe einer geeigneten Schlüsselableitungsfunktion $\texttt{kdf} : \mathbb{Z}_p \mapsto \{0,1\}^n$ wird aus dem mit dem ElGamal-Verschlüsselungsschlüssel z kombinierten Pseudonymgeheimnis $z^k \in \mathbb{Z}_p$ ein sicherer Schlüssel für ein symmetrisches Verschlüsselungsverfahren Enc abgeleitet: $\texttt{kdf}(z^k) = K$.

9. Ein bösartiger IDExchanger kann natürlich immer eine eigene Tabelle der lokal bearbeiteten *QIDs* und der zugehörigen Pseudonyme anfertigen und somit diese *QIDs* re-identifizieren. Allerdings ist er dabei auf seinen lokal begrenzten Kompetenzbereich beschränkt. Für weitere Hinweise dazu wird auf die Ausführungen in Abschnitt 7.4 verwiesen.
10. Dieses Element würde normalerweise im ElGamal-Kryptosystem zufällig gewählt werden. Der daraus resultierende Indeterminismus würde allerdings der notwendigen globalen Konsistenz der Pseudonyme entgegenlaufen.

3. Das Pseudonym P_{QID} besteht dann aus einem Tupel (c_1, c_2), wobei c_1 das chiffrierte Pseudonymgeheimnis g^k und c_2 den Schlüsseltext eines symmetrischen Verschlüsselungsverfahrens Enc angewandt auf den Sitzungsschlüssel K und den QID enthält:

$$P_{QID} = (c_1, c_2) = (g^k, \mathsf{Enc}(K, QID)).$$

7.5.1.3 Limitierte Verkettbarkeit

Definition 7.5 (Limitierte Verkettbarkeit)**.** Die Verkettbarkeit von Ereignisnachrichten mit Bezug zur gleichen QID ist begrenzt auf eine definierte und limitierte Periode. Anhand der Informationen in zwei Ereignisnachrichten aus unterschiedlichen Perioden ist nicht entscheidbar, ob ein tatsächlicher Bezug zueinander besteht.

Die Begrenzung der Verkettbarkeit ist ein Kompromiss zwischen der Anforderung der *Datenminimierung* und der *Verkettbarkeit*. Dieser Kompromiss muss durch die Anpassung von Variablen zur Konfiguration der Limitierung optimiert werden, was einen hochgradig anwendungsspezifischen Prozess darstellt. Technische Mechanismen zum Erreichen der limitierten Verkettbarkeit werden in PEEPLL beim zugrunde liegenden Systemmodell (a) ohne zentrale Pseudonymisierungsinstanz durch die Begrenzung einer Zeitperiode realisiert, in der die Wiederverwendung von Pseudonymen möglich ist.

Diese *zeitliche Limitierung der Verkettbarkeit* von Pseudonymen ist realisiert durch globale Epochen, an deren Beginn alle Pseudonyme gewechselt werden. Dieser Pseudonymwechsel wird herbeigeführt durch den Wechsel eines epochenspezifischen Merkmals t_i, das derart in die Pseudonymberechnung eingeht, dass keine zwei Pseudonyme aus unterschiedlichen Epochen für den gleichen QID verkettbar sind. Das epochenspezifische Merkmal t_i wird deterministisch aus dem gemeinsamen Geheimnis d, welches zwischen allen IDExchangern geteilt ist, abgeleitet, etwa mithilfe eines geeigneten Pseudozufallszahlengenerators. Entsprechend der Ausführungen zu Punkt 1 in Abschnitt 7.5.1.2 wird dann t_i anstelle von d zur Berechnung des $MAC_{QID} = \mathsf{Mac}(t_i, QID)$ verwendet.

Eine wichtige Voraussetzung bei einem Pseudonymisierungsverfahren mit verteilten Pseudonymisierungskomponenten ist ein zuverlässiger Weg, die Epochengrenzen zwischen den IDExchangern zu synchronisieren. Andernfalls könnten neue Epochen zu unterschiedlichen Zeiten gewechselt werden, was die Anforderung der globalen Konsistenz zumindest für die kurze Zeitdifferenz verletzt.

7.5.1.4 Zusammenführung

Zusammenfassend lassen sich die einzelnen Schritte im vorliegenden Systemmodell (a) ohne zentrale Pseudonymisierungsinstanz zum Schutz der *QID-Vertraulichkeit*, der *Re-Identifizierung nach dem Mehr-Augen-Prinzip* sowie der *Limitierten Verkettbarkeit* wie folgt beschreiben:

1. Eine TTP erzeugt den geheimen Schlüssel $a \in \mathbb{Z}_p$ und den öffentlichen Schlüssel $pk = (p, g, z = g^a)$ eines ElGamal-basierten Schwellwertschemas. Der geheime Schlüssel wird in eine ausreichende Anzahl von Geheimnisteilen aufgeteilt und an verschiedene unabhängige Interessenvertreter und Entscheidungsträger ausgegeben. Der öffentliche Schlüssel wird zusammen mit einem gemeinsamen Geheimnis d unter allen IDExchangern verteilt. Danach wird die TTP nicht mehr benötigt.

2. Aus dem gemeinsamen Geheimnis d erzeugen alle IDExchanger ein gemeinsames Epochengeheimnis t_i, dass für eine definierte Zeitspanne verwendet wird und anschließend anhand einer vorgegebenen Regel durch ein anderes gemeinsames Epochengeheimnis t_{i+1} ersetzt wird. Denkbar ist hierfür etwa ein Pseudozufallszahlengenerator mit d als Startwert oder die Konkatenation des aktuellen Zeitstempels in gewünschter Granularität mit d.

3. Für jeden QID, der von einem IDExchanger pseudonymisiert werden soll, werden die folgenden Schritte durchgeführt:

   a) Aus dem QID und dem gemeinsamen Epochengeheimnis t_i erzeugt der IDExchanger zunächst $MAC_{QID} = \mathtt{Mac}(t_i, QID)$ und daraus ein Pseudonymgeheimnis $k = f(MAC_{QID})$ sowie mithilfe einer festgelegten Schlüsselableitungsfunktion `kdf` und pk einen symmetrischen Schlüssel $K = \mathtt{kdf}(z^k)$ und errechnet daraus ein deterministisches Pseudonym

   $$P_{QID} = (c_1, c_2) = (g^k, \mathtt{Enc}(K, QID)).$$

   Das Pseudonym enthält damit alle notwendigen Informationen zur Re-Identifizierung.

   b) Der IDExchanger ersetzt den QID mit dem epochenspezifischen Pseudonym P_{QID}.

7.5.2 PEEPLL mit echt zufälligen Pseudonymen

Entsprechend der Ausführungen in Abschnitt 7.3.2 zum Systemmodell (b) mit zentraler Pseudonymisierungsinstanz, die als IDVault bezeichnet wird, sowie den Anforderungen in Abschnitt 7.2.3 haben die IDExchanger die Aufgabe, ohne gegenseitige Absprache auftretende QIDs mit global konsistenten Pseudonymen zu ersetzen, die sie beim IDVault anfragen. Der IDVault wiederum erzeugt für jeden noch nicht bearbeiteten QID ein echt zufälliges Pseudonym und speichert die Zuordnung zwischen QID und zugehörigem Pseudonym in einer Pseudonymzuordnungstabelle. Aus dieser werden die Pseudonyme für QIDs entnommen, die bereits vom IDVault bearbeitet wurden. Die Pseudonymzuordnungstabelle dient ebenfalls einer potenziellen Re-Identifizierung. Darüber hinaus werden die folgenden Pseudonymisierungs-Schutzziele verfolgt und technisch durchgesetzt, wobei die drei bereits in Abschnitt 7.5.1 bearbeiteten Pseudonymisierungs-Schutzziele *QID-Vertraulichkeit*, *Re-Identifizierung nach dem Mehr-Augen-Prinzip* sowie *Limitierte Verkettbarkeit* an das Systemmodell sowie sich daraus neu ergebende Bedrohungen für den Datenschutz angepasst werden.

7.5.2.1 QID-Vertraulichkeit

Gemäß Definition 7.1 soll niemand außer dem IDExchanger Informationen über den Klartext-QID erlangen, der von diesem IDExchanger pseudonymisiert werden soll. Insbesondere soll auch der IDVault keine Einblicke darüber erlangen, für welche QIDs er Pseudonyme erzeugt und diese für spätere Pseudonymanfragen abspeichert.

Auch im vorliegenden Systemmodell (b) mit zentraler Pseudonymisierungsinstanz werden HMACs eingesetzt, um starke QID-Vertraulichkeit zu erreichen (vgl. Definition 7.3), indem

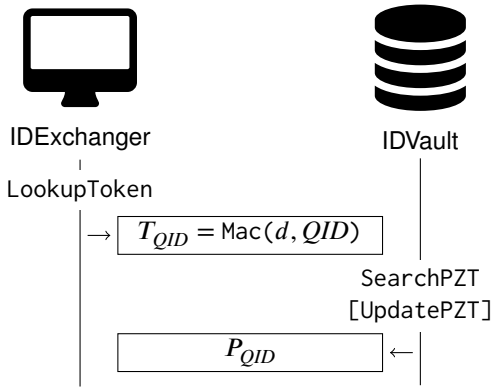


Abbildung 7.3: Die Interaktionen zwischen einem IDExchanger und dem IDVault zur Erreichung der *QID-Vertraulichkeit*. Die Algorithmen `LookupToken`, `SearchPZT` und `UpdatePZT` realisieren die im Text beschriebene Funktionalität.

alle IDExchanger zu Anfang des Pseudonymisierungsverfahrens von einer TTP mit einem gemeinsamen geheimen Schlüssel d von ausreichender Länge ausgestattet werden.[11]

Mit gegebenem d und einem *QID*, der pseudonymisiert werden soll, konvertiert der IDExchanger den Klartext-*QID* in ein Nachschlagewert (engl. lookup token) $T_{QID} = \texttt{Mac}(d, QID)$, der mithilfe der `Mac`-Funktion eines HMACs errechnet wird (vgl. Abschnitt 7.1.2). Dieser Nachschlagewert wird später zur Erkennung eines möglicherweise existierenden zugehörigen Pseudonyms P_{QID} verwendet, ohne dass dafür Informationen über *QID* selbst vorliegen müssen. Darüber hinaus verwenden alle IDExchanger für die Berechnung des Nachschlagewertes den gleichen geheimen Schlüssel d, sodass T_{QID} über das gesamte System konsistent ist und demnach *Globale Konsistenz* aufrechterhalten wird.

Der IDExchanger sendet in einer Pseudonymanfrage T_{QID} an den IDVault, der die globale Pseudonymzuordnungstabelle (*PZT*) besitzt. *PZT* besteht aus einer Menge von Nachschlagewert-Pseudonym-Paaren $PZT = \{(T_{QID_1}, P_{QID_1}), \ldots, (T_{QID_n}, P_{QID_n})\}$. Darin sucht der IDVault nach einem übereinstimmenden Eintrag (T_{QID}, P_{QID}) passend zum Nachschlagewert. Wird ein solcher Eintrag in *PZT* nicht gefunden, generiert der IDVault ein neues zufälliges Pseudonym P_{QID}, speichert es zusammen mit dem Nachschlagewert in *PZT* ab und sendet eine Pseudonymantwort mit P_{QID} an den anfragenden IDExchanger. Der IDExchanger ersetzt *QID* mit dem erhaltenen Pseudonym und fährt fort. Der Ablauf ist in Abbildung 7.3 illustriert.

Der gemeinsame Schlüssel d, der allen IDExchangern aber nicht dem IDVault bekannt ist, fügt dem Nachschlagewert T_{QID} Entropie hinzu, sodass dieser nicht nur von *QID* abhängt. Andernfalls könnte der IDVault durch einen Bruteforce-Angriff aller möglichen *QIDs* den eigentlichen *QID* aus T_{QID} herausfinden (vgl. Abschnitt 7.1.2). Ein solcher Bruteforce-Angriff ist dennoch möglich, sofern IDExchanger in den Besitz von fremden *PZT*-Einträgen kommen, die nichts mit dem aktuell in Bearbeitung befindlichen *QID* zu tun haben. Dieser Umstand kann eintreten, sofern Mechanismen zur Erhaltung der *Unbeobachtbarkeit des passenden Pseudonyms* (s. Abschnitt 7.5.2.2) umgesetzt werden. Da alle IDExchanger den geheimen Schlüssel d kennen, fällt die zusätzliche Entropie bei einem Bruteforce-Angriff durch einen IDExchanger weg. Dieses Pro-

11. Für die Erbringung des Pseudonymisierungs-Schutzziels *Re-Identifizierung nach dem Mehr-Augen-Prinzip* kommt noch eine weitere Aufgabe für die TTP hinzu, die in Abschnitt 7.5.2.3 erläutert wird. Die TTP wird anschließend nicht mehr benötigt.

blem wird als *schwache QID-Vertraulichkeit* (vgl. Definition 7.2 in Abschnitt 7.5.1.1) bezeichnet. Eine Lösung für dieses Problem wird im folgenden Abschnitt 7.5.2.2 eingeführt.

7.5.2.2 Unbeobachtbarkeit des passenden Pseudonyms

Definition 7.6 (Unbeobachtbarkeit des passenden Pseudonyms)**.** Das zu einer Pseudonymanfrage passende Pseudonym in der Pseudonymzuordnungstabelle ist nur dem anfragenden IDExchanger und nicht dem IDVault, der die Pseudonymzuordnungstabelle verwaltet, bekannt.

Nachfolgend wird zwischen einer *partiellen* und einer *vollständigen Unbeobachtbarkeit des passenden Pseudonyms* unterschieden.

Definition 7.7 (partielle Unbeobachtbarkeit des passenden Pseudonyms)**.** Die Unbeobachtbarkeit des passenden Pseudonyms gilt nicht für alle Pseudonymanfragen eines IDExchangers an den IDVault.

Das bedeutet, dass der IDVault unter bestimmten Bedingungen, auf die im Verlauf dieses Abschnitts eingegangen wird, erfährt, welches Pseudonym von einem IDExchanger für eine QID-Pseudonymisierung verwendet wird.

Definition 7.8 (vollständige Unbeobachtbarkeit des passenden Pseudonyms)**.** Die Unbeobachtbarkeit des passenden Pseudonyms gilt für alle Pseudonymanfragen eines IDExchangers an den IDVault.

Wenn zusätzlich zur Erreichung von *QID-Vertraulichkeit* aus Abschnitt 7.5.2.1 auch *Unbeobachtbarkeit des passenden Pseudonyms* erreicht werden soll,[12] dann ist die einfachste Lösung die Auslieferung der gesamten Pseudonymzuordnungstabelle an jeden IDExchanger, der eine Pseudonymanfrage stellt. Dadurch würde der IDVault keine Informationen darüber erlangen, welche Einträge für die IDExchanger von Interesse sind. Abgesehen von einem hohen Datenübertragungsbedarf würde dieser Ansatz allerdings drei Probleme verursachen:

1) *Datenschutzproblem*: Die IDExchanger erfahren alle existierenden Einträge der Pseudonymzuordnungstabelle, was dem Prinzip der *Datenminimierung* sowie der Limitierung des Kompetenzbereichs der IDExchanger auf die lokal bearbeiteten Daten entgegenläuft. PEEPLL balanciert diesen Konflikt aus und spart zusätzlich Bandbreite, indem nur eine kleine Teilmenge der Pseudonymzuordnungstabelle an einen IDExchanger ausgeliefert wird, während zur gleichen Zeit sichergestellt wird, dass die Anzahl der ausgelieferten Einträge größer als Eins ist. Zum Einsatz kommen dabei *Secure Indexes*, die für den vorliegenden Anwendungsfall optimiert wurden.

12. Die nachfolgend erläuterten Techniken lassen sich auch derart verwenden, dass ausschließlich *Unbeobachtbarkeit des passenden Pseudonyms* ohne *QID-Vertraulichkeit* erreicht wird. Das würde eine Erhöhung der Effizienz und eine Reduzierung der Komplexität auf Kosten des Datenschutzes, den *QID-Vertraulichkeit* bietet, bedeuten.

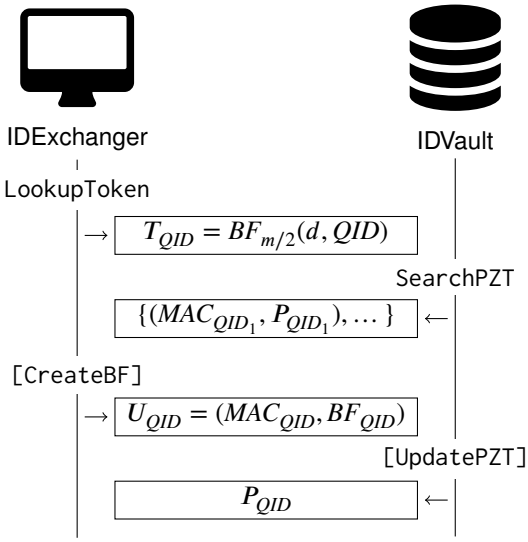


Abbildung 7.4: Die Interaktionen zwischen einem IDExchanger und dem IDVault zur Erreichung der *Unbeobachtbarkeit des passenden Pseudonyms*. Die Algorithmen `LookupToken`, `SearchPZT`, `CreateBF` und `UpdatePZT` realisieren die im Text beschriebene Funktionalität.

2) *Partielle Unbeobachtbarkeit des passenden Pseudonyms*: Das zweite Problem hängt mit dem Fall zusammen, dass ein angefragtes Pseudonym zu einem *QID* auf Seiten des IDVaults noch nicht existiert und somit erst erzeugt werden muss. In diesen Fällen enthält die ausgelieferte Teilmenge der Pseudonymzuordnungstabelle nur irrelevante Einträge, was von einem anfragenden IDExchanger erkannt und gelöst werden muss. Schlussendlich muss der IDExchanger die Erzeugung eines neuen Pseudonyms beim IDVault anfordern, was wiederum die *vollständige Unbeobachtbarkeit des passenden Pseudonyms* vereitelt (vgl. Definition 7.8). Denn dieser Umstand zeigt dem IDVault eindeutig, welches Pseudonym für den IDExchanger von Interesse ist. Eine Lösung ist die *Erzeugung von Dummy-Pseudonymen*.

3) *Schwache QID-Vertraulichkeit*: Der Ansatz der Auslieferung einer kleinen Teilmenge von Einträgen der Pseudonymzuordnungstabelle erlaubt zwar den Schutz der *Unbeobachtbarkeit des passenden Pseudonyms*. Es senkt dabei allerdings den Schutz der *QID-Vertraulichkeit* (vgl. Definition 7.2), da ein IDExchanger fremde Einträge der Pseudonymzuordnungstabelle erhält und diese somit durch die Kenntnis des gemeinsamen Schlüssels d mittels Bruteforce effektiv angreifen kann. Der in PEEPLL umgesetzte Lösungsansatz verwendet ein angepasstes *1-out-of-N OT*-Verfahren.

1) Lösung des Datenschutzproblems

Der nachfolgende Lösungsansatz für das aufgezeigte Datenschutzproblem lehnt sich sehr stark an die *Secure Indexes* von Goh [Goh03] an (vgl. Abschnitt 7.1.6). Mit dem gemeinsamen Geheimnis d erstellt ein IDExchanger einen Bloom-Filter als Nachschlagewert $T_{QID} = BF_{m/2}(d, QID)$ für den *QID*. Der Bloom-Filter wird durch eine Menge von m geheimen Schlüsseln $(d_1, \dots, d_m)$ konstruiert, die vom gemeinsamen Geheimnis d abgeleitet werden. Von diesen m Schlüsseln werden zufällig $\lfloor m/2 \rfloor$ Schlüssel ausgewählt und für jeden d_i aus dieser Schlüsselteilmenge

wird wiederholt eine Pseudozufallsfunktion auf *QID* angewendet, mit der bestimmt wird, welche Bitposition des Bloom-Filters auf den Wert 1 gesetzt wird (vgl. Abschnitt 7.1.5).[13] Der Nachschlagewert T_{QID} wird anschließend zum IDVault gesendet, der die Pseudonymzuordnungstabelle *PZT* verwaltet. *PZT* ist eine Menge von Tripeln jeweils bestehend aus einem Bloom-Filter, einem *MAC* sowie einem zugehörigen Pseudonym für einen bereits bearbeiteten *QID*. Der IDVault antwortet auf die Pseudonymanfrage vom IDExchanger mit einer Teilmenge

$$\{(MAC_{QID_j}, P_{QID_j}) \mid (BF_{QID_j}, MAC_{QID_j}, P_{QID_j}) \in PZT \text{ und } T_{QID} \subset BF_{QID_j}\}$$

aller *MAC*-Pseudonym-Paare aus *PZT*, deren Bloom-Filter eine Obermenge von T_{QID} darstellen. Die erwartete Anzahl an Paaren in dieser Teilmenge wird beeinflusst durch eine künstliche Falsch-Positiv-Rate der Bloom-Filter, die in den Tripeln der Pseudonymzuordnungstabelle gespeichert sind.

Damit ein IDExchanger das korrekte Pseudonym in der erhaltenen Menge an *MAC*-Pseudonym-Paaren erkennen kann, vergleicht er den lokal errechneten $MAC_{QID} = \texttt{Mac}(d, QID)$ mit den erhaltenen *MACs*. Sofern keine Übereinstimmung gefunden wird, muss der IDExchanger die Erzeugung eines neuen Pseudonyms beim IDVault anstoßen. Die Interaktionen sind in den ersten beiden Austauschnachrichten in Abbildung 7.4 illustriert.

2) Vollständige Unbeobachtbarkeit des passenden Pseudonyms

Im Fall einer Pseudonymanfrage für einen *QID* beziehungsweise dessen MAC_{QID}, für den vom IDVault noch kein Pseudonym erzeugt wurde, nimmt der IDExchanger den lokal errechneten MAC_{QID} und erzeugt daraus einen Bloom-Filter nach dem gleichen Prinzip wie bei der Erstellung des Auswahlwertes T_{QID}, allerdings mit dem Unterschied, dass nun anstelle der Schlüsselteilmenge alle m Schlüssel verwendet werden. Zusätzlich wird dem Bloom-Filter eine sogenannte Blendung von b Bits hinzugefügt, indem b zufällig gewählte Bitpositionen im Bloom-Filter auf den Wert 1 gesetzt werden. Die Blendung erreicht und steuert die angesprochene künstliche Falsch-Positiv-Rate, die die Wahrscheinlichkeit beeinflusst, dass mehr als ein Tripel zu einem gegebenen Auswahlwert T_{QID} aus einer Pseudonymanfrage in *PZT* gefunden wird. Das Resultat ist ein Paar $U_{QID} = (BF_{QID}, MAC_{QID})$ des soeben erstellten Bloom-Filters und des *MACs*, die sich für den vorliegenden *QID* ergeben. Dieses Paar wird zum IDVault gesendet, der es zusammen mit einem neu erzeugten zufälligen Pseudonym P_{QID} in *PZT* einfügt und das Pseudonym an den IDExchanger zurücksendet. Die Interaktionen sind in den letzten beiden Austauschnachrichten in Abbildung 7.4 illustriert.

Dieser Vorgang reduziert allerdings den bis hierher etablierten Schutz der *vollständigen Unbeobachtbarkeit des passenden Pseudonyms*. Der in PEEPLL umgesetzte Lösungsansatz für die vollständige Erreichung dieses Pseudonymisierungs-Schutzziels geht allerdings auf die Kosten der Performanz, denn es werden Dummy-Einträge in *PZT* eingeführt. Nach jeder Pseudonymanfrage und der zugehörigen Antwort muss ein IDExchanger die Erzeugung eines neuen Pseudonyms anfordern. Dieses stellt entweder ein Pseudonym für den tatsächlichen *QID* dar, sofern nur irrelevante Einträge vom IDVault zurückgesendet wurden, oder andernfalls ein Pseudonym für einen zufällig erzeugten Dummy-*QID*. Da der IDVault niemals die Klartext-*QIDs* zu

13. Indem nur die Hälfte der Schlüssel für den Nachschlagewert verwendet werden, hier mit $BF_{m/2}$ gekennzeichnet, wird ein Indeterminismus eingeführt, der eine Profilbildung der Nachschlagewerte seitens des IDVaults erschwert, wie auch von Goh [Goh03] diskutiert. Weiterhin wird die Wahrscheinlichkeit für passende Einträge der Pseudonymzuordnungstabelle zu diesem Bloom-Filter, also die Falsch-Positiv-Rate, erhöht.

sehen bekommt, sondern nur zufällig aussehende MAC_{QID}, können die Dummy-*QIDs* auch zufällige Zeichenketten darstellen und sind somit für den IDVault nicht von einem echten MAC_{QID} unterscheidbar.

3) Starke QID-Vertraulichkeit

1-out-of-N OT (vgl. Abschnitt 7.1.7) in Verbindung mit HMACs, wie sie zum Schutz der *QID-Vertraulichkeit* in Abschnitt 7.5.2.1 eingeführt wurden, scheint die natürliche Lösung zum kompletten Schutz der *Unbeobachtbarkeit des passenden Pseudonyms*.[14] Ein Empfänger (IDExchanger) fragt einen Sender (IDVault) nach einem speziellen Datenbankeintrag, der mit einem Zeilenindex erreichbar ist. Der Sender antwortet mit allen Datenbankeinträgen jeweils auf eine spezielle Art und Weise verschlüsselt, sodass der Empfänger nur den Eintrag von ursprünglichem Interesse korrekt entschlüsseln kann. Der Sender erfährt dabei nicht, für welchen Eintrag sich der Empfänger interessiert hat. OT ist allerdings auf den vorliegenden Kontext der Pseudonymisierung mit zentraler Instanz nicht einfach übertragbar. Zum einen benötigt es eine Datenbank fester Länge, deren Einträge über einen Zeilenindex erreichbar sind, der dem IDExchanger bekannt ist. Zum anderen involviert OT massive Berechnungen sowie massive Bandbreite, denn am Ende des OT-Protokolls liefert der Sender die gesamte Datenbank, speziell verschlüsselt, an den Empfänger aus. Dennoch kann eine Anpassung des OT-Verfahrens das Problem der *schwachen QID-Vertraulichkeit* lösen, weshalb es in PEEPLL integriert ist. Die beiden Anpassungen überwinden die angesprochenen Hindernisse von OT:

1. Das erste Hindernis betrifft die fehlenden festen Zeilenindizes, die von einem IDExchanger benötigt werden, um einen spezifischen Eintrag der Pseudonymzuordnungstabelle zu lokalisieren und abzurufen. Eine Lösung ist dabei die Verwendung der *MACs* selbst als Indizes einer Hashtabelle $C = \{MAC_0 \colon C_0, \ldots, MAC_{N-1} \colon C_{N-1}\}$, die die korrespondierenden verschlüsselten Einträge der Pseudonymzuordnungstabelle speichert. Da aber auch das wieder die *MACs* von fremden Einträgen an einen IDExchanger ausliefern würde (vgl. das Problem der schwachen QID-Vertraulichkeit in Punkt 3)), der sie effizient mittels Bruteforce angreifen könnte, wird ein weiterer *MAC* aus den *MACs* zusammen mit dem jeweiligen OT-spezifischen Indexschlüssel K_j für alle Tabelleneinträge $j \in \{0, \ldots, N-1\}$ als OT-Indizes $i_j = \mathtt{Mac}(K_j, MAC_j)$ berechnet. Der IDExchanger, der ein Pseudonym für einen speziellen *QID* anfordert, kann den OT-spezifischen Schlüssel K_{QID} berechnen, der zur Entschlüsselung des angefragten Eintrags aus der Menge der erhaltenen verschlüsselten Einträge $C_0, \ldots, C_{N-1}$ verwendet wird. Dieser OT-spezifische Schlüssel erlaubt dem IDExchanger ebenfalls die Berechnung des korrekten $i_{QID} = \mathtt{Mac}(K_{QID}, MAC_{QID})$ und somit die Identifizierung des angefragten Eintrags in der Hashtabelle der erhaltenen Einträge. Der genaue Ablauf lässt sich anhand von Abbildung 7.5 nachvollziehen.

2. Das zweite Hindernis betrifft den rechnerischen sowie den Datenübertragungsaufwand, die im Allgemeinen durch das OT-Protokoll eingeführt werden. Dieses Hindernis kann zumindest abgeschwächt werden, indem nicht die gesamte Pseudonymzuordnungstabelle als

14. Die Datenschutztechnik von Cooper und Birman [CB95], die auch unter dem Namen *Blind-Message-Service* bekannt ist, bietet zwar ganz ähnliche Eigenschaften, kann allerdings hier nicht angewendet werden. Der Grund liegt darin, dass der IDExchanger andere QIDs kennen müsste, die bereits in der *PZT* vorhanden sind, um sie dann mit dem eigentlichen QID überlagern zu können. Das ist hier zum einen nicht vorgesehen und zum anderen würde es dem Pseudonymisierungs-Schutzziel der *(starken) QID-Vertraulichkeit* entgegenstehen, das an dieser Stelle verfolgt wird.

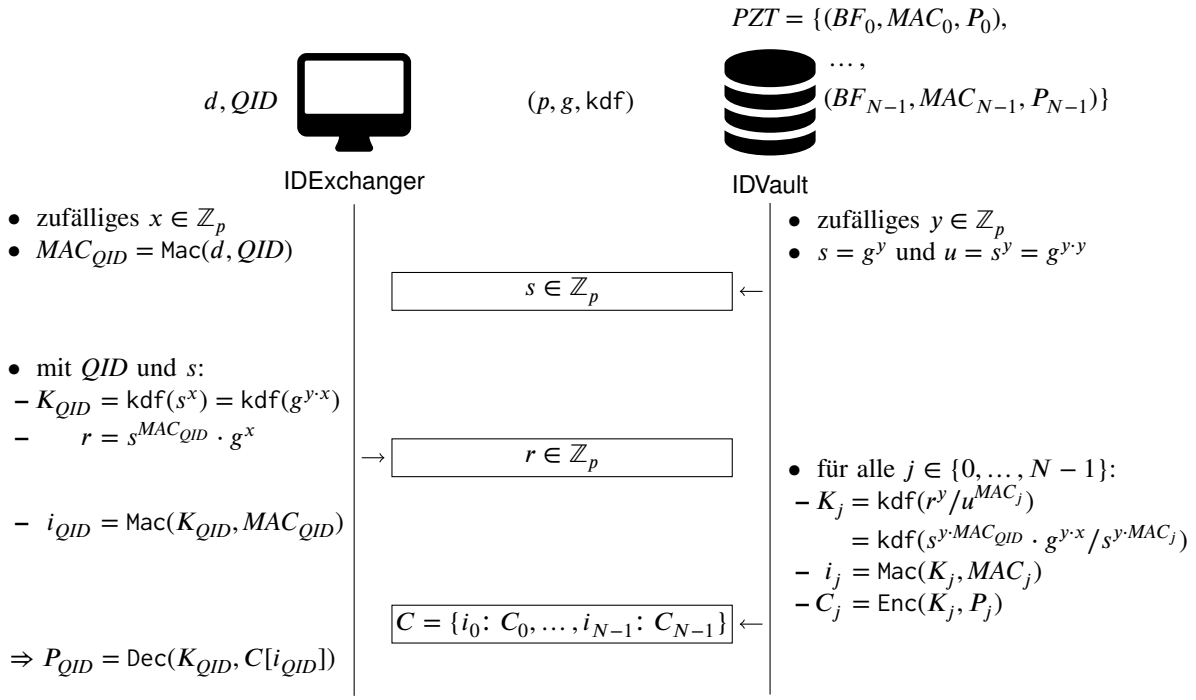


Abbildung 7.5: Die Interaktionen und Berechnungen im Detail zur Lösung der schwache QID-Vertraulichkeit bei gleichzeitiger Unbeobachtbarkeit des passenden Pseudonyms

Eingabe des Senders (IDVaults) verwendet wird, sondern nur eine auf die von einem IDExchanger angefragten Einträge limitierte Teilmenge. Dies stellt einen Kompromiss zwischen erreichbarer Sicherheit des OTs und der Performanz des Pseudonymisierungsverfahrens dar.

7.5.2.3 Re-Identifizierung nach dem Mehr-Augen-Prinzip

Gemäß Definition 7.4 sollen die Informationen, die zu einer potenziellen Re-Identifizierung benötigt werden, nicht von den Pseudonymisierungskomponenten einsehbar sein und auf mehrere unabhängige Interessenvertreter und Entscheidungsträger verteilt werden, sodass eine Kooperation nach dem Mehr-Augen-Prinzip notwendig ist, um eine Re-Identifizierung durchzuführen. Im vorliegenden Systemmodell (b) mit zentraler Pseudonymisierungsinstanz wird dafür äquivalent zum Systemmodell (a) ohne zentrale Pseudonymisierungsinstanz ein ElGamal-basiertes Schwellwertschema eingesetzt, wie in Abschnitt 7.5.1.2 beschrieben, mit dem die Informationen zur Re-Identifizierung auf Seiten der IDExchanger verschlüsselt werden. Im Unterschied dazu müssen allerdings die Verschlüsselungen nicht lokal deterministisch durchgeführt werden, da die Pseudonyme in der Pseudonymzuordnungstabelle mittels eines global konsistenten MAC_{QID_j} als Teil des Auswahlwertes T_{QID_j} identifiziert werden. Die verschlüsselten Informationen zur Re-Identifizierung können somit unabhängig davon berechnet werden.

Zu Anfang des Pseudonymisierungsverfahrens wird einmalig ein ElGamal-basiertes Schwellwertschema von einer TTP instanziiert. Die zugehörigen Geheimnisteile werden an geeignete

Interessenvertreter der Betroffenen und Entscheidungsträger verteilt. Der Verschlüsselungsschlüssel $pk = (p, g, z)$ wird zusammen mit dem gemeinsamen geheimen Schlüssel d an alle IDExchanger verteilt. Danach wird die TTP nicht mehr benötigt.

Für jede Pseudonymerzeugung, die von einem IDExchanger für einen *QID* angestoßen wird, berechnet dieser neben den bisher erläuterten Vorgaben zusätzlich die verschlüsselten Re-Identifizierungsinformationen $R_{QID} = (c_1, c_2)$, indem *QID* mittels pk und d verschlüsselt wird. Im Detail berechnet sich R_{QID} zu einem *QID* folgendermaßen:

1. Der IDExchanger wählt zufällig ein Pseudonymgeheimnis $k \in \mathbb{Z}_p$.
2. Mithilfe einer geeigneten Schlüsselableitungsfunktion $\mathtt{kdf} : \mathbb{Z}_p \mapsto \{0,1\}^n$ wird aus dem mit dem ElGamal-Verschlüsselungsschlüssel z kombinierten Pseudonymgeheimnis $z^k \in \mathbb{Z}_p$ ein sicherer Schlüssel für ein symmetrisches Verschlüsselungsverfahren abgeleitet: $\mathtt{kdf}(z^k) = K$.
3. Die Re-Identifizierungsinformationen R_{QID} bestehen dann aus einem Tupel (c_1, c_2), wobei c_1 das chiffrierte Pseudonymgeheimnis g^k und c_2 den Schlüsseltext eines symmetrischen Verschlüsselungsverfahrens $\mathtt{Enc}$ angewandt auf den Sitzungsschlüssel K und den *QID* enthält:
$$R_{QID} = (c_1, c_2) = (g^k, \mathtt{Enc}(K, QID)).$$

Diese verschlüsselten Re-Identifizierungsinformationen R_{QID} werden anschließend zusammen mit den Informationen zur Pseudonymerzeugung $U_{QID} = (BF_{QID}, MAC_{QID})$ an den IDVault gesendet. Dieser generiert ein neues Pseudonym P_{QID} und speichert das Quadrupel (BF_{QID}, MAC_{QID}, P_{QID}, R_{QID}) in seiner Pseudonymzuordnungstabelle *PZT* für nachfolgende Referenzen.

7.5.2.4 Limitierte Verkettbarkeit

Gemäß Definition 7.5 soll die Verkettbarkeit von Ereignisnachrichten limitiert werden, sodass einzelne Ereignisnachrichten mit Bezug zur gleichen QID nur für eine limitierte Periode miteinander verknüpft werden können. Technische Mechanismen zum Erreichen der limitierten Verkettbarkeit werden in PEEPLL beim zugrunde liegenden Systemmodell (b) mit zentraler Pseudonymisierungsinstanz sowohl durch die Limitierung der Zeitperiode realisiert, in der die Wiederverwendung von Pseudonymen möglich ist, als auch durch die Limitierung der Anzahl der Wiederverwendungen an sich.

Zeitliche Limitierung

Die *Zeitliche Limitierung der Verkettbarkeit* von Pseudonymen wird zusätzlich zum bereits in Abschnitt 7.5.1.3 präsentierten Mechanismus, der epochenspezifische Merkmale t_i auf Seiten der IDExchanger vorsieht, auch von Seiten des IDVaults durchgesetzt. Die Details sehen dafür wie folgt aus.

Die IDExchanger leiten das epochenspezifische Merkmal t_i deterministisch aus dem gemeinsamen Geheimnis d ab, etwa mithilfe eines geeigneten Pseudozufallszahlengenerators. Bei jeder Pseudonymerzeugung, die von einem IDExchanger für einen *QID* angestoßen wird, berechnet dieser dafür nun abgewandelte Informationen $U_{QID} = (BF_{QID}, MAC_{QID_i})$, wobei der MAC_{QID_i} von t_i abhängt: $MAC_{QID_i} = \mathtt{Mac}(t_i, QID)$. Dieser epochenspezifische MAC_{QID_i} muss natürlich auch

an allen anderen Stellen, wo dieser benötigt wird, auf diese Art und Weise berechnet werden. Das ist der Fall beim Durchsuchen der Antwortmenge des IDVaults, nachdem der IDExchanger eine Pseudonymanfrage gestellt hat, sowie bei den 1-out-of-N OT-Berechnungen.

Der große Vorteil beim vorliegenden Systemmodell (b) mit zentraler Pseudonymisierungsinstanz ist der Umstand, dass zusätzlich die zeitliche Limitierung von Seiten des IDVaults unilateral durchgesetzt werden kann, also nicht auf die Unterstützung anderer Komponenten oder auf Anpassungen des Pseudonymisierungsverfahrens angewiesen ist. Dafür entfernt der IDVault jeweils zu den Epochengrenzen alle Bloom-Filter und *MACs* aus seiner Pseudonymzuordnungstabelle. Die verbleibenden Pseudonyme sowie die jeweils zugehörigen Re-Identifizierungsinformationen müssen archiviert werden.

Budget-Limitierung

Durch die Limitierung der Verkettbarkeit von Ereignisnachrichten mithilfe eines Budgets ist es nicht möglich, ein bereits existierendes Pseudonym für einen QID wiederzuverwenden, nachdem die Budgetsumme von vorherigen Wiederverwendungen ein maximales Datenschutzbudget überstiegen hat. Die individuellen Budgets von Wiederverwendungen haben entweder einfach den Wert 1 oder ein Gewicht, welches abhängig vom Kontext und den Auswirkungen von Pseudonymwiederverwendungen auf den Datenschutz von Individuen ist. In seiner einfachsten Form akkumuliert das gesamte Datenschutzbudget die Anzahl der Wiederverwendungen eines Pseudonyms. Ein solcher Ansatz kann durch die Einführung von Nutzungszählern für jedes Pseudonym umgesetzt werden. Im vorliegenden Systemmodell (b) mit zentraler Pseudonymisierungsinstanz kann dieser Ansatz allerdings nur auf Seiten des IDVaults umgesetzt werden. Dieser verwaltet Nutzungszähler für jeden Eintrag in seiner Pseudonymzuordnungstabelle, die bei jeder Verwendung des Eintrags hochgezählt werden.

Da der Mechanismus zum Schutz vor *Unbeobachtbarkeit des passenden Pseudonyms* in Abschnitt 7.5.2.2 dafür sorgt, dass der IDVault nicht genau weiß, für welchen Eintrag sich ein IDExchanger tatsächlich interessiert, müssen die Zähler aller Einträge entsprechend des Budgets von Wiederverwendungen hochgezählt werden. Als Konsequenz stellen die individuellen Budgetzähler, die auf Seiten des IDVaults mit einem Pseudonym assoziiert sind, nur einen unscharfen Wert dar und das Maximalbudget repräsentiert eine obere Schranke für die tatsächlichen Wiederverwendungen von Pseudonymen, die mit hoher Wahrscheinlichkeit nie erreicht wird.

7.5.2.5 Zusammenführung

Zusammen genommen lassen sich die einzelnen Schritte im vorliegenden Systemmodell (b) mit zentraler Pseudonymisierungsinstanz zum Schutz der *QID-Vertraulichkeit*, insbesondere der *starken QID-Vertraulichkeit*, der *Unbeobachtbarkeit des passenden Pseudonyms*, der *Re-Identifizierung nach dem Mehr-Augen-Prinzip* sowie der *Limitierten Verkettbarkeit* wie folgt beschreiben:

1. Eine TTP erzeugt das Geheimnis $a \in \mathbb{Z}_p$ und den öffentlichen Schlüssel $pk = (p, g, z = g^a)$ eines ElGamal-basierten Schwellwertschemas. Der geheime Schlüssel wird in eine ausreichende Anzahl von Geheimnisteilen aufgeteilt und an verschiedene unabhängige Interessenvertreter und Entscheidungsträger ausgegeben. Der öffentliche Schlüssel wird zusammen mit einem gemeinsamen Geheimnis d unter allen IDExchangern verteilt. Danach wird die TTP nicht mehr benötigt.

2. Der IDVault wählt zu Anfang einmalig ein zufälliges $y \in \mathbb{Z}_p$, berechnet $s = g^y$ und $u = s^y$ und sendet s an alle IDExchanger.

3. Aus dem gemeinsamen Geheimnis d erzeugen alle IDExchanger ein gemeinsames Epochengeheimnis t_i, dass für eine definierte Zeitspanne verwendet wird und anschließend anhand einer vorgegebenen Regel durch ein anderes gemeinsames Epochengeheimnis t_{i+1} ersetzt wird. Denkbar ist hierfür etwa ein Pseudozufallszahlengenerator mit d als Startwert oder die Konkatenation des aktuellen Zeitstempels in gewünschter Granularität mit d.

4. Der IDVault löscht an jeder Epochengrenze alle Bloom-Filter, *MACs* und Budget-Zähler aus seiner *PZT*.

5. Für jeden QID, der von einem IDExchanger pseudonymisiert werden soll, werden die folgenden Schritte durchgeführt:

   a) Aus dem QID errechnet der IDExchanger mithilfe von t_i sowie einem zufällig gewählten $x \in \mathbb{Z}_p$ die folgenden Werte:

      - Den epochenspezifischen $MAC_{QID} = \mathtt{Mac}(t_i, QID)$,
      - das OT-Geheimnis $K_{QID} = \mathtt{kdf}(s^x) = \mathtt{kdf}(g^{y \cdot x})$,
      - den OT-Wert des Empfängers $r = s^{MAC_{QID}} \cdot g^x$,
      - einen epochenspezifischen Bloom-Filter $T_{QID} = BF_{m/2}(t_i, QID)$ als Nachschlagewert sowie
      - den OT-Index $i_{QID} = \mathtt{Mac}(K_{QID}, MAC_{QID})$

      und sendet r und T_{QID} an den IDVault.

   b) Der IDVault erzeugt sich die Teilmenge

      $$\{(MAC_j, P_j) \mid (BF_j, MAC_j, P_j) \in PZT \text{ und } T_{QID} \subset BF_j\}$$

      aller *MAC*-Pseudonym-Paare aus *PZT*, deren Bloom-Filter eine Obermenge von T_{QID} darstellen. Für alle Einträge in dieser Teilmenge verringert der IDVault den Budget-Zähler und entfernt den zugehörigen Bloom-Filter sowie den *MAC* aus seiner *PZT*, sofern der Budget-Zähler die Null erreicht hat. Er antwortet auf die Pseudonymanfrage vom IDExchanger mit einer indizierten Menge der verschlüsselten Pseudonyme aus der erzeugten Teilmenge

      $$C = \{i_j \colon C_j \mid i_j = \mathtt{Mac}(K_j, MAC_j), C_j = \mathtt{Enc}(K_j, P_j), K_j = \mathtt{kdf}(r^y / u^{MAC_j})\}$$

   c) Der IDExchanger prüft, ob sein errechneter OT-Index i_{QID} Teil der Antwortliste vom IDVault ist und wählt bei positivem Ergebnis anhand des OT-Indexes i_{QID} das gewünschte verschlüsselte Pseudonym aus der Antwortliste vom IDVault aus und entschlüsselt dieses mit dem OT-Geheimnis K_{QID}: $P_{QID} = \mathtt{Dec}(K_{QID}, C[i_{QID}])$. An dieser Stelle hat der IDExchanger das gewünschte Pseudonym.

   d) Ist der vorige Punkt c) erfolgreich, erzeugt er einen Zufallswert *Nonce* und setzt für die nachfolgenden Punkte $MAC_{QID} = \mathtt{Mac}(t_i, Nonce)$.

   e) Anschließend wählt der IDExchanger ein zufälliges Geheimnis $k \in \mathbb{Z}_p$ und sendet eine Pseudonymerzeugungsanfrage mit den folgenden Werten an den IDVault:

      - Den epochenspezifischen MAC_{QID},

- den epochenspezifischen Bloom-Filter BF_{QID},[15]
- die Re-Identifizierungsinformationen $R_{QID} = (c_1, c_2) = (g^k, \mathsf{Enc}(\mathsf{kdf}(z^k), QID))$.

f) Der IDVault erzeugt ein neues zufälliges Pseudonym P_{QID}, sendet dieses an den IDExchanger und speichert das Tupel $(BF_{QID}, MAC_{QID}, P_{QID}, R_{QID}, BudgetMax - 1)$ in seiner *PZT* ab.

g) Der IDExchanger ersetzt den *QID* mit dem gewünschten epochenspezifischen Pseudonym P_{QID}.

7.6 Evaluation

In diesem Abschnitt wird das entwickelte und im Detail erläuterte Pseudonymisierungsverfahren PEEPLL für die Erhöhung des Datenschutzes bei der Analyse von Insideraktivitäten evaluiert.

7.6.1 Datensatz

Für die Erzeugung des Evaluationsdatensatzes wurde das *Linux Audit System* konfiguriert und eingesetzt, welches bereits in Abschnitt 5.3 beschrieben und dort als Grundlage für die entwickelte Erkennungs- und Abwehrtechnik von Insiderbedrohungen angepasst wurde. Vier Beispiel-Ereignisnachrichten des Linux Audit Systems sehen wie folgt aus.

```
1 node=mw-host type=SYSCALL msg=audit(1547121809.485:31317) :
    arch=i386 syscall=socketcall(connect) success=no exit=-115 a0=connect
    a1=19302860 a2=0 a3=0 items=0 ppid=1734 pid=1750 auid=mw uid=mw
    gid=mw tty=pts0 ses=2 comm="WTEpZSFwgb"
    exe="/home/mw/Downloads/WTEpZSFwgb"
2 node=mw-host type=SOCKETCALL msg=audit(1547121809.485:31317) :
    nargs=3 a0=9d a1=19d85988 a2=10
3 node=mw-host type=SOCKADDR msg=audit(1547121809.485:31317) :
    saddr={ fam=inet laddr=142.190.225.69 lport=82 }
4 node=mw-host type=PROCTITLE msg=audit(1547121809.485:31317) :
    proctitle="/home/mw/Downloads/WTEpZSFwgb -ipc.fd=3 scan"
```

Der Vorteil an Audit-Ereignisnachrichten für die Evaluation der entwickelten Datenschutztechnik liegt in den strukturiert vorliegenden Daten, die anhand von eindeutigen Datenfeldern innerhalb der Ereignisnachrichten identifiziert und interpretiert werden können.[16] Somit kann genau festgelegt werden, bei welchen Datenfeldern es sich potenziell um QIDs handelt, die pseudonymisiert werden müssen. Für die Evaluation wurde eine solche Liste an Datenfeldern festgelegt, die anhand von Tabelle 7.1 nachvollzogen werden kann.

Der für die Evaluation erzeugte Datensatz enthält 43 504 Ereignisnachrichten und ist ca. 9,8 MB groß. Nach einem Zwischenverarbeitungsschritt, der im folgenden Abschnitt beschrieben wird, müssen 162 168 Datenfelder pseudonymisiert werden, wobei davon nur 9305 Datenfelder unterschiedliche Werte aufweisen.

15. Hierbei handelt es sich nicht um den gleichen Bloom-Filter, der vom IDExchanger für T_{QID} erzeugt wurde. Dieser kann allerdings für den hier benötigten Bloom-Filter gegebenenfalls verwendet und entsprechend der Ausführungen in Abschnitt 7.5.2.2 um weitere Bitpositionen sowie um eine Blendung erweitert werden, sofern der IDExchanger nicht Punkt d) ausführen musste.
16. Eine Liste mit den meisten Datenfeldern sowie deren Bedeutung findet sich unter https://github.com/linux-audit/audit-documentation/blob/master/specs/fields/field-dictionary.csv.

Tabelle 7.1: Datenfelder von Ereignisnachrichten des Linux Audit Systems, die pseudonymisiert werden

Datenfelder	Bedeutung	Pseudonymisierungsgrund
node, addr, hostname	Hostname oder IP-Adresse	Direkter oder indirekter Rückschluss auf einen oder mehrere Nutzer
timestamp	Zeitpunkt des Ereignisses	In Verbindung mit weiteren Zeitpunkten lassen sich Aktivitätsprofile erstellen
acct, uid, gid, auid	Nutzer- beziehungsweise Gruppenaccount	Direkte Zuordnung von Ereignissen zu einem oder mehreren speziellen Nutzern
exe, comm	Name eines ausgeführten Programms	In Verbindung mit weiteren Programmnamen lassen sich Aktivitätsprofile erstellen. Die Art eines ausgeführten Programms lässt Rückschlüsse auf einen oder mehrere Nutzer zu
proctitle	Programmname mit Kommandozeilenparamtern	Potenziell sensible Informationen, wie zum Beispiel Passwörter
name, cwd	Dateiname oder -pfad	Potenziell sensible Informationen, wie zum Beispiel Personennamen
unit	Systemdienstname	Potenziell sensible Informationen, wie zum Beispiel Programmnamen
a0, a1, a2, a3	Systemaufruf-Funktionsargumente von `execve()`	Die Funktionsargumente des Systemaufrufs `execve()` werden zu Programmnamen und Kommandozeilenparameter übersetzt. Hier wird bezüglich der Pseudonymisierung eine Fallunterscheidung zu anderen Systemaufrufen vorgenommen

7.6.2 Implementierung und Evaluationsumgebung

Für die Pseudonymisierung der QIDs in Audit-Ereignisnachrichten wurde zunächst ein Parser in Python3 mit 590 Zeilen Code und drei Objektklassen programmiert, der die Datenfelder jeder Ereignisnachricht erfasst sowie alle Ereignisnachrichten zusammenfasst, die zu ein und demselben Ereignis gehören. Als Veranschaulichung dienen die vier Beispiel-Ereignisnachrichten aus Abschnitt 7.6.1. Sie beziehen sich auf dasselbe Ereignis und tragen daher die gleiche Knoteninformation *(mw-host)*, den gleichen Identifier *(31317)* sowie den gleichen Zeitstempel *(1547121809.485)*. Mit einem Zwischenverarbeitungsschritt werden derartige Ereignisnachrichten zu einer konsolidierten Ereignisnachricht zusammengefasst und redundante Datenfelder entfernt. Dadurch werden unnötige Pseudonymisierungen verhindert. Es handelt sich bei diesem Parser um denselben Parser, der bereits in Abschnitt 5.3.4 beschrieben wurde und in der dort vorgestellten Erkennungs- und Abwehrtechnik von Insiderbedrohungen zum Einsatz kommt.

Der Parser wurde weiterhin mit der in diesem Kapitel entwickelten Funktionalität eines IDExchangers ausgestattet, sodass alle Datenfelder aus der Liste der potenziellen QID-Datenfelder (vgl. Tabelle 7.1) pseudonymisiert werden können. Auch der IDExchanger sowie der IDVault wurde in Python3 mit 413 Zeilen Code und 9 Objektklassen implementiert. Die Implementierung eines *ElGamal-basierten Schwellwertschemas*, das für die Re-Identifizierung nach dem Mehr-Augen-Prinzip benötigt wird, wurde im Rahmen einer Masterarbeit durchgeführt [Pet18] und auf Github veröffentlicht.[17] Die Berechnungen des implementierten ElGamal-basierten Schwellwertschemas sowie alle Berechnungen des 1-out-of-N OT-Verfahrens zur Erhaltung der starken QID-Vertraulichkeit wurden für die Evaluation auf die Verwendung von elliptischen Kurven (vgl. Abschnitt 7.1.3.2) hin geändert beziehungsweise softwaretechnisch umgesetzt. Für die Verwendung der *Secure Indexes* zur Erreichung der Unbeobachtbarkeit des passenden Pseudonyms wurde auf eine existierende Implementierung zurückgegriffen,[18] die für die vorliegenden Anwendungsfälle zielgerichtet angepasst wurde. Darüber hinaus wurde die Implementierung des IDExchangers um eine Erfassung von Statistiken erweitert, die nach der vollständigen Bearbeitung eines Datensatzes zur Auswertung ausgegeben werden.

Als Hardware kommt ein MacBook Pro mit 16 GB Arbeitsspeicher und einem 3,1 GHz Intel Core i7 Prozessor zum Einsatz, von dem für das Parsen und die Pseudonymisierung allerdings nur einer der vier Kerne verwendet wird. Bei der Laufzeitumgebung handelt es sich um Python in der Version 3.7.7. Der IDVault wird bei der Evaluation des Systemmodells (b) mit zentraler Pseudonymisierungsinstanz zur besseren Vergleichbarkeit der Ergebnisse auf derselben Hardware gestartet, um Rechnernetzverzögerungen und Paketlaufzeiten in den Statistiken vernachlässigen zu können.

7.6.3 Vorgehensweise

Ein Testdurchlauf besteht darin, den IDExchanger und gegebenenfalls den IDVault zu starten. Der IDExchanger wartet anschließend auf Audit-Ereignisnachrichten, die er über den Standard-Input entgegennimmt. Der IDVault wiederum wartet auf eintreffende Pseudonymanfragen. Anschließend wird jede Audit-Ereignisnachricht aus dem erzeugten Datensatz einzeln und ohne Zeitversatz, wie er etwa anhand des Zeitstempels der Ereignisnachrichten möglich und realistisch wäre, an den IDExchanger gereicht, der sie zunächst parst und mit weiteren zugehörigen

17. Abrufbar unter https://github.com/tompetersen/threshold-crypto.
18. Abrufbar unter https://github.com/cburkert/zidx.

Tabelle 7.2: Die Pseudonymisierungs-Schutzziele, die mit den sieben definierten Testfällen jeweils abgedeckt werden

Testfall	QC		MPU		ReID	LL		Systemmodell
	wQC	sQC	pMPU	fMPU		tLL	bLL	
T0								–
T1		✔			✔	✔		(a)
T2.1		✔			✔	✔	✔	(b)
T2.2	✔		✔		✔	✔	✔	(b)
T2.3		✔	✔		✔	✔	✔	(b)
T2.4	✔			✔	✔	✔	✔	(b)
T2.5		✔		✔	✔	✔	✔	(b)

Abkürzungen: **QC** - QID-Vertraulichkeit (engl. QID confidentiality); **wQC** - schwache (engl. weak) QID-Vertraulichkeit; **sQC** - starke (engl. strong) QID-Vertraulichkeit; **ReID** - Re-Identifizierung; **LL** - limitierte Verkettbarkeit (engl. limited linkability); **tLL** - zeitlich (engl. temporal) limitierte Verkettbarkeit; **bLL** - Budget- (engl. budgetary) limitierte Verkettbarkeit; **MPU** - Unbeobachtbarkeit des passenden Pseudonyms (engl. matching pseudonym unobservability); **pMPU** - partielle (engl. partial) Unbeobachtbarkeit des passenden Pseudonyms; **fMPU** - vollständige (engl. full) Unbeobachtbarkeit des passenden Pseudonyms.

Ereignisnachrichten zusammenfasst. Sowohl während des Parsens als auch beim Finalisieren einer konsolidierten Nachricht ersetzt der IDExchanger die QID-Datenfelder mit entsprechenden Pseudonymen. Nach Bearbeitung der letzten Ereignisnachricht des Datensatzes stoppt der IDExchanger und gibt die gesammelten Statistikdaten aus.

Insgesamt wurden sieben Testfälle mit jeweils unterschiedlichen Konfigurationen von PEEPLL definiert, die zusammen mit den jeweils abgedeckten Pseudonymisierungs-Schutzzielen in Tabelle 7.2 zusammengefasst sind:

- **T0** Keine Pseudonymisierung,
- **T1** PEEPLL mit lokal deterministischen Pseudonymen, Threshold-Verschlüsselung und globalen Epochen (vgl. Abschnitt 7.5.1),
- **T2** PEEPLL mit echt zufälligen Pseudonymen, Threshold-Verschlüsselung, globalen Epochen, Budget-Zählern (vgl. Abschnitt 7.5.2) und
  - **T2.1** MACs anstelle von QIDs als Nachschlagewerte,
  - **T2.2** Bloom-Filtern anstelle von QIDs als Nachschlagewerte,
  - **T2.3** Bloom-Filtern anstelle von QIDs als Nachschlagewerte und Oblivious Transfer,
  - **T2.4** Bloom-Filtern anstelle von QIDs als Nachschlagewerte und Dummy-Pseudonymen,
  - **T2.5** Bloom-Filtern anstelle von QIDs als Nachschlagewerte, Oblivious Transfer und Dummy-Pseudonymen.

Die globalen Epochen wurden für diese Evaluation auf eine definierte Zeitspanne von 60 Sekunden festgelegt.

7.6.4 Auswertungen

Die Auswertung der beschriebenen Testfälle ist vor dem Hintergrund einer prototypischen Implementierung zu betrachten. Es wurde kein gesteigerter Fokus auf eine Performanzoptimierung

Tabelle 7.3: Vergleich der notwendigen Berechnungsschritte bei der vollständigen Pseudonymisierung eines Datensatzes in den Testfällen **T1 – T2.5**

	T1	**T2.1**	**T2.2**	**T2.3**	**T2.4**	**T2.5**
HMAC-Berechnung	1·D	1·D	1·D	2·D + m	(1+(D-P))·D	(2+(D-P))·D + m
EC-Punktoperationen	2·D	2·D	2·P	5·D + 3·P + m + 2	2·D	8·D + m + 2
symm. Ver-/Entschl.	1·D	1·D	1·P	1·D + 1·P + m	1·D	2·D + m
Tabellenabfrage	–	1·D	2·D	2·D	2·D	2·D
Bloom-Filter-Erzeugung	–	–	1·D + 1·P	1·D + 1·P	2·D	2·D
Bloom-Filter-Test	–	–	D	D	D	D

Abkürzungen: **D** - Anz. aller Pseudonymisierungsdurchgänge (hier: 162 168); **P** - Anz. aller unterscheidbaren Pseudonyme (hier: 9305); **m** - Summe aller jeweils vom IDVault gefundenen passenden *PZT*-Einträge bei einem Pseudonymisierungsdurchgang (variabel für jeden Testfall und besonders hoch beim Einsatz von Dummy-Pseudonym-Erzeugungen).

gelegt, sodass die Testfälle hauptsächlich für eine untereinander vergleichende Einschätzung der Performanz-Overhead-Dimensionen herangezogen werden können. Insbesondere können mit dieser Evaluation keine Aussagen über den Einsatz von PEEPLL unter realen Bedingungen, etwa bei der Echtzeit-Pseudonymisierung des vorliegenden Datensatzes, getätigt werden. Das liegt zum einen an zu groben Zeitstempeln der Ereignisnachrichten in dem Datensatz, wodurch etwa 70% der Ereignisnachrichten laut Zeitstempeln den exakt gleichen Zeitpunkt von mehreren weiteren Ereignisnachrichten aufweisen und somit durch den Pseudonymisierungsoverhead verworfen werden müssten. Zum anderen wären spezielle Optimierungen und eine geeignetere Wahl einer hardwarenahen Programmiersprache angebracht, was in dieser Dissertation nicht geleistet werden konnte und für anschließende Arbeiten offenbleibt.

Basis-Overhead

Der Testfall **T0** spiegelt die reine Verarbeitung aller Ereignisnachrichten des vorliegenden Datensatzes durch den Parser wieder. Dabei wird keine Pseudonymisierung durchgeführt, sondern jeder QID wird vom IDExchanger mit sich selbst ersetzt. Eine derartige Verarbeitung des Datensatzes dauerte in der beschriebenen Evaluationsumgebung 4,1275 Sekunden und zeigt somit den grundsätzlichen Performanz-Overhead.

Kalkulatorische Abschätzungen und gemessene Statistiken

Die notwendigen Berechnungsschritte für eine vollständige Pseudonymisierung eines Datensatzes lassen sich einerseits aus den in Abschnitt 7.5.1 und Abschnitt 7.5.2 beschriebenen Vorgängen ableiten, sie sind allerdings auch andererseits von einigen implementierungsspezifischen Designentscheidungen abhängig. So werden zum Beispiel die Bloom-Filter-Tests bei einem Pseudonymisierungsdurchgang in den Testfällen **T2.2 – T2.5** nicht, wie eigentlich notwendig, mit allen zu diesem Zeitpunkt in der *PZT* vorhanden Einträgen einzeln durchgeführt, sondern aufgrund einer speziell gewählten Datenstruktur der *PZT* mit einer einfachen Abfrage der Datenstruktur realisiert. Dadurch sinkt der Aufwand pro Pseudonymisierungsdurchgang von der Größe der *PZT* zu diesem Zeitpunkt auf Eins.

In Tabelle 7.3 werden die notwendigen Berechnungsschritte für eine vollständige Pseudonymisierung eines Datensatzes in den einzelnen Testfällen gegenübergestellt. Daraus wird ersichtlich, in welchem Umfang der weitere Performanz-Overhead der Testfälle im Vergleich zu **T0** ansteigt.

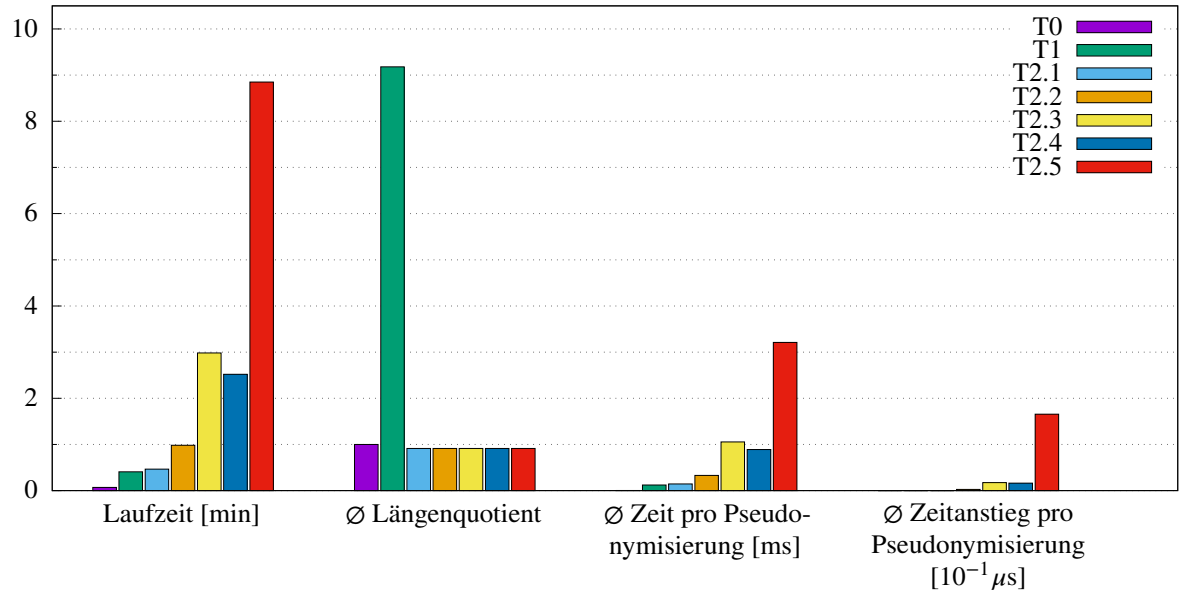


Abbildung 7.6: Vergleichende Statistiken der Testfälle **T0** – **T2.5** in der beschriebenen Evaluationsumgebung

Ein besonders hoher Anstieg ist in den Testfällen **T2.3** und **T2.5** zu erwarten, da aufgrund der Oblivious Transfer-Berechnungen für eine starke QID-Vertraulichkeit eine wesentlich höhere Anzahl an HMAC-Berechnungen, EC-Punktmultiplikationen sowie symmetrischen Ver- und Entschlüsselungen durchgeführt werden müssen.

In Abbildung 7.6 sind die bei der Durchführung der Testfälle in der beschriebenen Evaluationsumgebung tatsächlich gemessenen Statistiken zusammengefasst. Sie werden nachfolgend detaillierter erläutert.

Testfall T1

Mit der Durchführung des Testfalls **T1** wird eine Pseudonymisierung mit lokal deterministischen Pseudonymen entsprechend des Systemmodells (a) ohne zentrale Pseudonymisierungsinstanz durchgeführt (vgl. Abschnitt 7.5.1). Pro QID-Pseudonymisierungsdurchgang wird dabei eine HMAC-Berechnung, zwei Punktmultiplikationen auf einer elliptischen Kurve sowie eine symmetrische Verschlüsselung durchgeführt. Die Punktmultiplikationen und die Verschlüsselung sind für die Erzeugung der Re-Identifizierungsinformationen notwendig, die in diesem Testfall gleichzeitig das Pseudonym darstellen. Die Gesamtlaufzeit bei der Verarbeitung des vorliegenden Datensatzes betrug 24,3609 Sekunden, wobei jeder einzelne Pseudonymisierungsdurchgang im Mittel 0,1218 Millisekunden dauerte. Da jeder Pseudonymisierungsdurchgang unabhängig voneinander die jeweils gleichen Berechnungsschritte durchführt, bleibt diese Dauer pro Pseudonymisierung konstant. Ein Vergleich der Länge des QIDs mit der Länge des zugehörigen Pseudonyms zeigt auf, dass im Mittel jedes Pseudonym 9,18x länger als der zugehörige QID ist. Verglichen mit allen anderen Testfällen, bei denen eine Pseudonymisierung durchgeführt wird, ist der Längenquotient der einzige Performanznachteil dieser Konfiguration. Die PEEPLL-Konfiguration dieses Testfalls ist zusammenfassend in der vorliegenden Evaluationsumgebung für Szenarien geeignet, in denen die Ereignisnachrichten einen Mindestzeitabstand von 1,355 Millisekunden haben, was der maximal gemessenen Pseudonymisierungsdauer entspricht.

Alle weiteren Testfälle werden im Systemmodell (b) mit zentraler Pseudonymisierungsinstanz durchgeführt und garantieren somit zufällig erzeugte Pseudonyme (vgl. Abschnitt 7.5.2). Dadurch konnte auch die Länge der Pseudonyme vom IDVault global auf acht Zeichen beschränkt werden, was beim vorliegenden Datensatz zu einem mittleren Längenquotienten von 0,9125 im Vergleich mit der Länge der QIDs führte.

Testfall T2.1

Testfall **T2.1** hat im Unterschied zu **T1** pro QID-Pseudonymisierungsdurchgang eine zusätzliche Tabellenabfrage seitens des IDVaults, der bei einer Pseudonymanfrage die *PZT* konsolidieren muss. Mit einer geeigneten Datenstruktur für die *PZT* müssen nicht alle vorhandenen Einträge durchsucht werden, sondern kann der QID mit einer einzigen Tabellenabfrage auf Vorhandensein geprüft werden. Der Performanz-Overhead liegt demnach erwartungsgemäß sehr nahe dem von Testfall **T1** bei einer Gesamtlaufzeit von 27,9672 Sekunden und einer mittleren Dauer eines einzelnen Pseudonymisierungsdurchgangs von 0,1446 Millisekunden. Auch hier ist kein Anstieg dieser Pseudonymisierungsdauer zu verzeichnen. Die PEEPLL-Konfiguration dieses Testfalls ist zusammenfassend in der vorliegenden Evaluationsumgebung für Szenarien geeignet, in denen die Ereignisnachrichten einen Mindestzeitabstand von 0,85 Millisekunden haben, was der maximal gemessenen Pseudonymisierungsdauer entspricht.

Testfall T2.2

Im Testfall **T2.2** werden erstmals Bloom-Filter eingesetzt, um eine partielle beziehungsweise später in **T2.4** und **T2.5** eine vollständige Unbeobachtbarkeit des passenden Pseudonyms zu erreichen (vgl. Abschnitt 7.5.2.2). Einerseits verlagert sich dadurch die Information, ob ein neues Pseudonym erzeugt werden muss, vom IDVault zum IDExchanger, sodass die EC-Punktmultiplikationen und die symmetrischen Verschlüsselungen für die Berechnung der Re-Identifizierungsinformationen vom IDExchanger nicht mehr bei jedem QID-Pseudonymisierungsdurchgang, sondern nur noch bei neu zu erzeugenden Pseudonymen durchgeführt werden müssen. Andererseits erhöht sich die notwendige Anzahl an Tabellenabfragen, da der IDVault den gegebenen Bloom-Filter in einer Pseudonymanfrage mit den Einträgen in der *PZT* abgleichen muss. Weiterhin ergibt sich ein Performanz-Overhead durch die Erzeugung der Bloom-Filter sowie durch die angesprochenen Bloom-Filter-Tests. Wie bereits bei der kalkulatorischen Abschätzung erläutert, wurde hierbei allerdings eine besonders geeignete Datenstruktur für die *PZT* gewählt, sodass sich die Tabellenabfragen beim Bloom-Filter-Test sowie die Anzahl der Bloom-Filter-Tests pro Pseudonymisierungsdurchgang konstant und nicht in Abhängigkeit der Größe der *PZT* verhält. In den Statistiken vom vorliegenden Testfall machen sich diese Unterschiede in einer leicht höheren Gesamtlaufzeit von 58,8685 Sekunden und einer ebenfalls leicht höheren mittleren Dauer eines einzelnen Pseudonymisierungsdurchgangs von 3,3081 Millisekunden bemerkbar. Darüber hinaus ist ein sehr geringer mittlerer Zeitanstieg von 2,6447 Nanosekunden pro Pseudonymisierungsdurchgang zu verzeichnen. Die Reduzierung der EC-Punktmultiplikationen und der symmetrischen Verschlüsselungen gleichen laut der Statistiken den Bloom-Filter-Overhead also teilweise aus. Die PEEPLL-Konfiguration dieses Testfalls ist zusammenfassend in der vorliegenden Evaluationsumgebung für Szenarien geeignet, in denen die Ereignisnachrichten einen Mindestzeitabstand von 4,641 Millisekunden haben, was der maximal gemessenen Pseudonymisierungsdauer entspricht.

Testfall T2.3

Die Besonderheit von Testfall **T2.3** im Unterschied zu **T2.2** ist die Verwendung des in Abschnitt 7.5.2.2 beschriebenen 1-out-of-N OT-Mechanismus zur Erreichung der starken QID-Vertraulichkeit, wobei N nicht die gesamte Anzahl an Einträgen der *PZT* umfasst, sondern wie in **T2.2** durch Bloom-Filter auf eine Teilmenge reduziert wird. Insgesamt erhöht sich dadurch die Anzahlen der HMAC-Berechnungen, der EC-Punktmultiplikationen sowie der symmetrischen Ver- und Entschlüsselungen signifikant (vgl. Tabelle 7.3). Das zeigt sich auch anhand der Statistiken. In Abbildung 7.7 ist der Verlauf der Dauer jedes einzelnen Pseudonymisierungsdurchgangs visualisiert, der bei anwachsender Größe der *PZT* und damit verbundener wachsenden Anzahl passender *PZT*-Einträge bei einer Pseudonymanfrage sichtlich ansteigt. Jeder dieser passenden *PZT*-Einträge muss entsprechend des 1-out-of-N OT-Mechanismus speziell bearbeitet und verschlüsselt werden. Der mittlere Zeitanstieg von 17,374 Nanosekunden pro Pseudonymisierungsdurchgang ist demnach nicht nur wie in Testfall **T2.2** auf den Bloom-Filter-Overhead zurückzuführen, sondern spiegelt unter anderem noch diesen erhöhten Berechnungsaufwand wider. Die Gesamtlaufzeit erhöhte sich dadurch auf 178,8634 Sekunden, bei einer mittleren Dauer eines einzelnen Pseudonymisierungsdurchgangs von 1,0539 Millisekunden. Die PEEPLL-Konfiguration dieses Testfalls ist zusammenfassend in der vorliegenden Evaluationsumgebung für Szenarien geeignet, in denen die Ereignisnachrichten einen Mindestzeitabstand von 37,574 Millisekunden haben, was der maximal gemessenen Pseudonymisierungsdauer entspricht.

Testfall T2.4

Im Vergleich zu Testfall **T2.2** wird in **T2.4** bei jedem einzelnen Pseudonymisierungsdurchgang ein neues Pseudonym erzeugt, um vollständige Unbeobachtbarkeit des passenden Pseudonyms zu erreichen (vgl. Abschnitt 7.5.2.2). Dadurch werden bei jedem Pseudonymisierungsdurchgang Berechnungsschritte durchgeführt, die in **T2.2** nur bei einem neu zu erzeugendem Pseudonym durchgeführt werden mussten. Diese Differenzen[19] sind allerdings wesentlich geringer als die Anzahl der symmetrischen Ver- und Entschlüsselungen im Testfall **T2.3**. Dennoch zeigen die Statistiken fast gleiche Werte zu Testfall **T2.3**. Die Gesamtlaufzeit bei der Pseudonymisierung des vorliegenden Datensatzes betrug 151,14 Sekunden und somit eine mittlere Dauer eines einzelnen Pseudonymisierungsdurchgangs von 8,8966 Millisekunden. Der mittlere Zeitanstieg von 16,0544 Nanosekunden pro Pseudonymisierungsdurchgang zeigt auf, dass sich die teilweise höheren Anzahlen der notwendigen Berechnungen mit den teilweise niedrigeren im Vergleich zu Testfall **T2.3** in der vorliegenden Evaluierungsumgebung aufwiegen. Die PEEPLL-Konfiguration dieses Testfalls ist zusammenfassend in der vorliegenden Evaluationsumgebung für Szenarien geeignet, in denen die Ereignisnachrichten einen Mindestzeitabstand von 86,274 Millisekunden haben, was der maximal gemessenen Pseudonymisierungsdauer entspricht.

Testfall T2.5

Im letzten Testfall **T2.5** sind alle Datenschutzmechanismen von PEEPLL aktiviert und somit addieren sich die erhöhten Berechnungsanzahlen von **T2.3** und **T2.4** im Vergleich zu **T2.2**. Dieser Performanz-Overhead ist erwartungsgemäß auch in den Statistiken zu erkennen und schlägt sich

19. Zu den Differenzen gehört zum Beispiel die Anzahl der symmetrischen Verschlüsselungen im Zuge der Erzeugung der Re-Identifizierungsinformationen, die nun bei 2x der Anzahl aller Pseudonymisierungsdurchgänge (hier: 162 168) liegt. Vorher lagen sie bei 2x der Anzahl aller neu erzeugten Pseudonyme (hier: 9305).

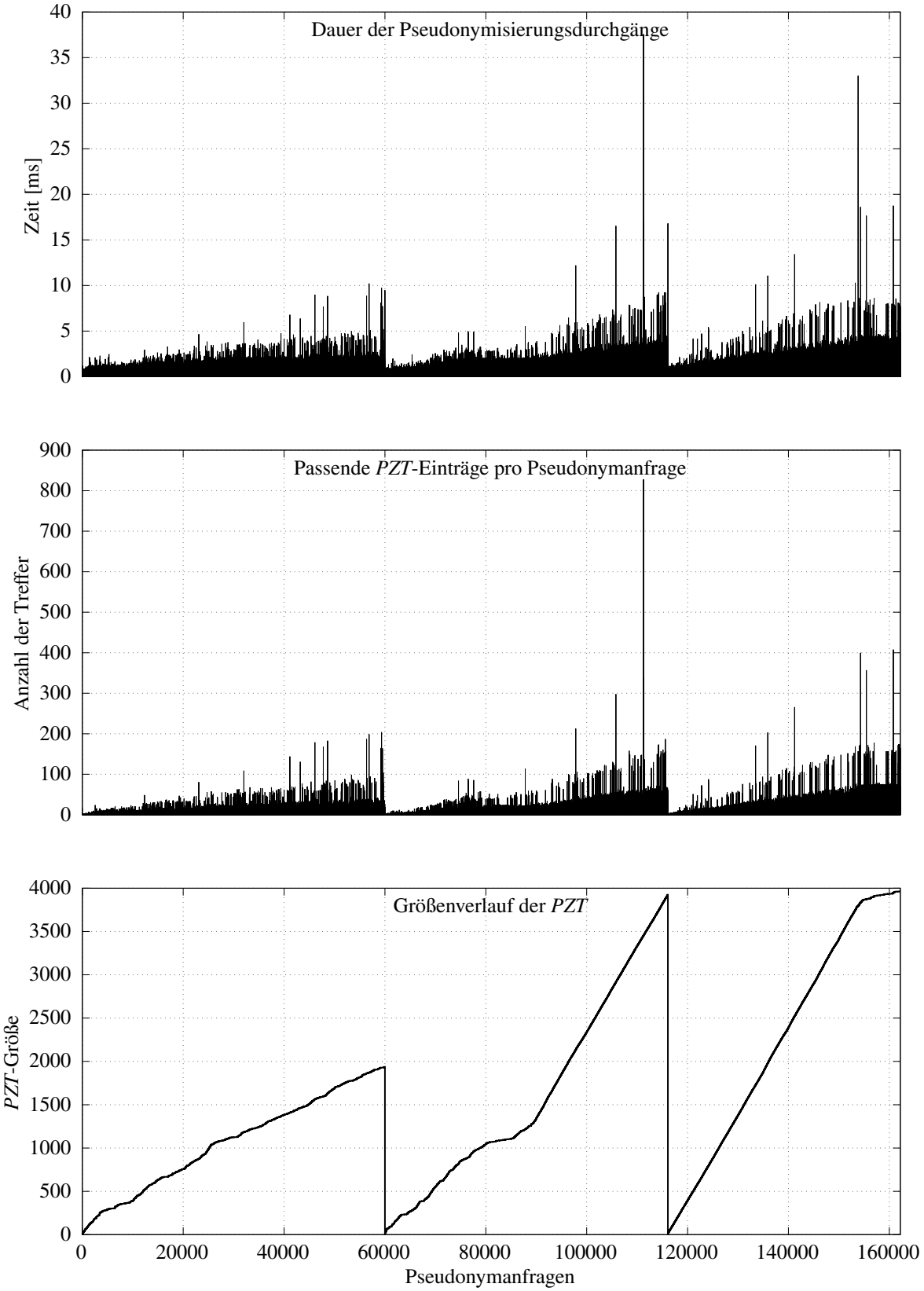


Abbildung 7.7: Verlauf der *PZT*-Größe und damit verbundene ansteigende Zahl der passenden *PZT*-Einträge sowie ansteigende Dauer der Pseudonymisierungsdurchgänge beim Durchlauf von Testfall **T2.3**

in einer Gesamtlaufzeit von 530,9465 Sekunden, einer mittleren Dauer eines einzelnen Pseudonymisierungsdurchgangs von 3,2099 Millisekunden sowie eines mittleren Zeitanstiegs von 165,431 Nanosekunden pro Pseudonymisierungsdurchgang nieder. Die PEEPLL-Konfiguration dieses Testfalls ist zusammenfassend in der vorliegenden Evaluationsumgebung für Szenarien geeignet, in denen die Ereignisnachrichten einen Mindestzeitabstand von 170,965 Millisekunden haben, was der maximal gemessenen Pseudonymisierungsdauer entspricht.

7.7 Erweiterungsmöglichkeiten

In diesem Abschnitt werden weitere Ideen präsentiert, die in zukünftigen Arbeiten eingehender untersucht und umgesetzt werden könnten.

Datenschutzfreundliche Ereignispseudonymisierung ohne TTP In beiden Systemmodellen wird jeweils zu Beginn des Pseudonymisierungsprozesses eine TTP benötigt, die das Schlüsselpaar eines ElGamal-basierten Schwellwertschemas erzeugt sowie den öffentlichen Schlüssel pk dieses Schlüsselpaares zusammen mit einem gemeinsamen Geheimnis d unter den IDExchangern verteilt. Diese TTP kann mit geeigneten Verfahren umgangen werden. Dafür ist zunächst der Austausch des gemeinsamen Geheimnisses d unter der Menge der IDExchanger ohne eine TTP notwendig. Existierende Protokolle für diese Aufgabe bauen auf der Arbeit von Steiner, Tsudik und Waidner [STW96] auf, worin der ursprüngliche Diffie-Hellman-Schlüsselaustausch [DH76] auf Gruppenkommunikationen erweitert wird.

Danach muss die Erzeugung des Schlüsselpaares eines ElGamal-basierten Schwellwertschemas sowie die Verteilung der Geheimnisteile zur Rekonstruktion des geheimen ElGamal-Schlüssels ohne eine TTP durchgeführt werden. Daran beteiligt werden dürfen nur die Geheimnisträger, die die Pseudonymisierungskomponenten (also die IDExchanger und den IDVault) ausschließen. Existierende Protokolle bauen auf den Arbeiten von Pedersen [Ped91] und Gennaro u. a. [Gen+99] auf.

Budget-Limitierung auf Seiten der IDExchanger Die *Limitierte Verkettbarkeit* anhand eines Budgets wird nur im Systemmodell (b) mit zentraler Pseudonymisierungsinstanz von Seiten des IDVault umgesetzt. Eine datenschutzfreundliche Lösung für die IDExchanger wäre für beide Systemmodelle von Vorteil. Einerseits ist das Konzept der Budget-Limitierung im Systemmodell (a) ohne zentrale Pseudonymisierungsinstanz bisher überhaupt nicht möglich. Andererseits könnte im Systemmodell (b) mit zentraler Pseudonymisierungsinstanz ein verteiltes Vertrauen äquivalent zur zeitlichen Limitierung der Verkettbarkeit etabliert werden. Dadurch muss nur wenigstens eine der beiden Pseudonymisierungsseiten, die IDExchanger oder der IDVault, korrekt handeln, um diese Art der Limitierung effektiv durchzusetzen. Allerdings dürften dadurch keine anderen Pseudonymisierungs-Schutzziele bedroht und keine der Anforderungen verletzt werden.

Ununterscheidbarkeit der Wiederverwendung Im Systemmodell (b) mit zentraler Pseudonymisierungsinstanz können von IDExchangern subtile Metainformationen über die Lebensdauer von Pseudonymen in Erfahrung gebracht werden. Entsprechend der Antwort eines IDVaults

auf eine Pseudonymanfrage kann der IDExchanger entscheiden, ob ein Pseudonym bereits existierte oder erst noch neu erzeugt werden muss und demnach in der aktuellen Epoche bisher von keinem anderen IDExchanger angefragt wurde. Ein entsprechendes Schutzziel vor dieser Datenschutzbedrohung kann als *Ununterscheidbarkeit der Wiederverwendung* bezeichnet werden.

Insbesondere die in PEEPLL umgesetzten Mechanismen zum Schutz der *Unbeobachtbarkeit des passenden Pseudonyms* in Abschnitt 7.5.2.2 verhindern ein solches Pseudonymisierungs-Schutzziel, da ein IDExchanger die Fälle unterscheiden können muss, ob ein Pseudonym bereits vorhanden ist oder neu vom IDVault erzeugt werden muss. Dieses Problem könnte durch weitere Untersuchungen und neue Ideen möglicherweise gelöst werden. Besondere Herausforderungen dabei sind möglichst keine weiteren Dummy-Pseudonymerzeugungen, die von einem IDExchanger angestoßen werden müssen und die Pseudonymzuordnungstabelle fluten würden, sowie ein möglichst geringer Berechnungsaufwand.

Performanzoptimierung und Echtzeit-Evaluation unter realen Bedingungen Für die prototypische Umsetzung von PEEPLL wurde kein besonderer Fokus auf eine Performanzoptimierung gelegt, sodass sich die in Abschnitt 7.6 vorgenommene Evaluation auf eine Einschätzung und einen Vergleich der Performanz-Overhead-Dimensionen in verschiedenen PEEPLL-Konfigurationen beschränkt. In weiteren Arbeiten kann PEEPLL in einer hardwarenahen Programmiersprache softwaretechnisch umgesetzt und in einer definierten Evaluationsumgebung zum Beispiel Audit-Ereignisnachrichten in Echtzeit pseudonymisieren. Dabei kann der zu verzeichnende Nachrichtenverlust erfasst und mit geeigneten Puffern abgefedert werden.

7.8 Fazit

Das im Rahmen dieser Dissertation entwickelte und in diesem Kapitel beschriebene Pseudonymisierungsverfahren namens PEEPLL wurde mit dem Ziel entwickelt, den Datenschutz für jene Insider zu erhöhen, die Sicherheitsmaßnahmen zur Erkennung und Abwehr von Insiderbedrohungen ausgesetzt sind. Dabei wurde zur Komplementierung des Forschungsbeitrags B4 aus Abschnitt 1.3 sowohl der vorliegende Rechts- und Interessenskonflikt (vgl. Abschnitt 6.1) gelöst als auch die gegenläufigen Anforderungen an derartige Sicherheitsmaßnahmen (vgl. Abschnitt 6.2) ausgeglichen.

Der Kontext wurde dabei in Abschnitt 7.2.1 allgemein auf die weitreichende Erfassung und Analyse von Insideraktivitäten in Unternehmen gesetzt. Konkret wurde für die praktische Umsetzung von PEEPLL die in Kapitel 5 entwickelte Erkennungs- und Abwehrtechnik von Insiderbedrohungen zugrunde gelegt (vgl. Abschnitt 7.2.2). Anhand dessen wird deutlich, dass die für die Sicherheitsmaßnahme erfassten Daten, die sich über mehrere Ereignisnachrichten erstrecken, nur in ihrer Verknüpfung für Analysezwecke hilfreich sind. Dennoch kann der direkt oder indirekt vorhandene Personenbezug aus den Ereignisnachrichten entfernt werden, solange eine Verkettbarkeit der Ereignisdaten erhalten und eine Re-Identifizierung möglich bleibt. Eine Pseudonymisierung der Datenfelder einzelner Ereignisnachrichten ist somit ein probates Mittel für den Insiderdatenschutz. Diese Pseudonymisierung muss aufgrund von potenziell verteilten Datenquellen, aus denen die Ereignisnachrichten hervorgehen, global konsistent und damit gegebenenfalls koordiniert ablaufen.

Für diese letzte Anforderung wurden zwei mögliche Lösungswege identifiziert und als dezentrales beziehungsweise zentrales Systemmodell des neuen Pseudonymisierungsverfahrens in Abschnitt 7.3 festgehalten. Im dezentralisierten Ansatz erfolgt die Pseudonymisierung durch mehrere IDExchanger für jede Datenquelle gesondert und unabhängig voneinander. Die globale Konsistenz erfolgt durch eine festgelegte Berechnungsvorschrift, mit der alle IDExchanger ihre vorliegenden QIDs deterministisch pseudonymisieren. Gleiche QIDs werden dadurch auf gleiche Pseudonyme abgebildet. Im zentralisierten Ansatz wird die Pseudonymisierung ebenfalls durch IDExchanger für jede Datenquelle gesondert vorgenommen, allerdings werden die Pseudonyme von einem zentralen IDVault angefragt. Der IDVault kann dadurch echt zufällige Pseudonyme für auftretende QIDs erzeugen und diese für weitere Anfragen und damit für eine globale Konsistenz abspeichern.

Ausgehend von der praktischen Umsetzung dieser beiden Systemmodelle, wurden in Abschnitt 7.4 Fälle identifiziert, die dem angestrebten Insiderdatenschutz durch die De-Identifizierungstechnik der Pseudonymisierung entgegenstehen. Dazu zählen die Re-Identifizierungsinformationen, die mit Kenntnis der Berechnungsvorschrift im dezentralisierten Systemmodell beziehungsweise mit Kenntnis der Pseudonymzuordnungstabelle, die im zentralisierten Systemmodell beim IDVault vorliegt, uneingeschränkt verwendet und damit der technisch durchgesetzte Datenschutz aufgehoben werden kann. Darüber hinaus erzeugt das Pseudonymisierungsverfahren an sich weitere subtile Informationen, die selbst ohne Kenntnis über die Verbindung zwischen Pseudonymen und QIDs gewisse Rückschlüsse über Personenbezüge erleichtern. Darunter fallen etwa Auftrittsmuster einzelner Pseudonyme, die von einem IDVault im zentralisierten Systemmodell durch seine globale Sicht auf alle vorhandenen Pseudonyme erfasst und unerlaubt analysiert werden können.

Diese und weitere Datenschutzbedrohungen resultieren in der Definition von Pseudonymisierungs-Schutzzielen, die das Ziel verfolgen, unnötige Informationen beziehungsweise Metainformationen zu eliminieren oder zumindest zu reduzieren. Die notwendigen Erweiterungen eines klassischen Pseudonymisierungsverfahrens wurden für das dezentralisierte Systemmodell in Abschnitt 7.5.1 und für das zentralisierte Systemmodell in Abschnitt 7.5.2 im Detail beschrieben. Zum Einsatz kommen dabei die erwähnten kryptographischen Funktionen, wie etwa schlüsselbasierte Hashfunktionen und HMACs, Secure Indexes oder ein ElGamal-basiertes Schwellwertschema. Die Pseudonymisierungs-Schutzziele lauten *QID-Vertraulichkeit*, *Re-Identifizierung nach dem Mehr-Augen-Prinzip*, *Limitierte Verkettbarkeit* sowie *Unbeobachtbarkeit des passenden Pseudonyms*, wobei letzteres nur im zentralisierten Systemmodell von Bedeutung ist und verhindert, dass der IDVault Kenntnis über die von den IDExchangern angeforderten Pseudonyme erhält.

Mit der Erhöhung des Datenschutzes durch Erreichen der besagten Pseudonymisierungs-Schutzziele wird der Aufwand für die Bearbeitung von Ereignisnachrichten signifikant angehoben. Dieser negative Einfluss auf die Performanz bei der Datenerhebung von Insideraktivitäten ergibt sich unmittelbar durch den gesteigerten Einsatz kryptographischer Funktionen. Wie hoch dieser Performanzverlust in etwa ausfällt und vor allem wie stark die Aktivierung einzelner Pseudonymisierungs-Schutzziele darauf Einfluss nimmt, wurde in Abschnitt 7.6 evaluiert. Mit einem vorbereiteten Testdatensatz an aufgezeichneten Ereignisnachrichten wurde die softwaretechnische Umsetzung von PEEPLL in sieben verschiedenen Testfällen ausgeführt. Dabei wurden Rechnernetz-assoziierte Details wie etwa Paketverluste und -laufzeiten vernachlässigt, um eine bessere Vergleichbarkeit der reinen Funktionalitäten auch zwischen den beiden Systemmodellen zu erhalten. Die gesammelten Statistiken wurden anhand einer zuvor aufgestellten kalkulato-

rischen Abschätzung auf Plausibilität geprüft sowie untereinander verglichen. Die Ergebnisse zeigen, dass der Einsatz von Secure Indexes zum Erreichen der *Unbeobachtbarkeit des passenden Pseudonyms* in Verbindung mit weiteren kryptographischen Funktionen den größten Performanzverlust verursacht. Dies beruht darauf, dass die kryptographischen Operationen nicht nur auf ein Antwortelement des IDVaults, sondern auf eine Menge von Antwortelementen angewendet werden müssen. Der Performanzverlust weist zudem einen hohen Anstieg im zeitlichen Verlauf der Pseudonymisierung auf, sofern die Erzeugung von Dummy-Pseudonymen umgesetzt wird. Anhand dieser statistischen Auswertung kann abgewogen werden, welche Konfiguration von Pseudonymisierungs-Schutzzielen in PEEPLL für einen konkreten Einsatzzweck gewünscht und vertretbar sind.

8 Schlussfolgerungen und Ausblick

Insider besitzen eine herausgehobene Stellung, die es ihnen erlaubt, Aktivitäten durchzuführen, die für Outsider unmöglich oder zumindest weniger einfach möglich sind. Daraus ergibt sich ein besonderes Bedrohungspotenzial für die Domäne von Insidern, denn diese Aktivitäten können bewusst oder unbewusst auch gegen die Domäne beziehungsweise gegen andere Insider der Domäne gerichtet werden. In dieser Dissertation wurde der Fragestellung nachgegangen, wie sich dieses Bedrohungspotenzial genauer beschreiben lässt und welche Bedrohungseigenschaften identifiziert werden können, die gegebenenfalls gezieltere und koordinierte Forschung und Entwicklung von Gegenmaßnahmen erlauben. Dabei wurde aufgezeigt, welche Fehlkonzeptionen und fehlende Grundlagen auf dem Gebiet der Insiderthematik vorliegen und dadurch Fortschritte bei der Bekämpfung von Insiderbedrohungen behindert werden. Daran anschließend wurde eine neue, technische Erkennungs- und Abwehrmaßnahme entworfen und umgesetzt, die dem Problem der Insiderbedrohungen in einem abgesteckten Teilgebiet adäquat begegnet. Vor dem Hintergrund einer *Dual-Use-Technologie* wurde weiterhin offengelegt, welche Bedrohung für Insider von einer solchen Sicherheitsmaßnahme ausgeht. Diese Bedrohung schlägt sich in einem Missbrauchspotenzial für eine unrechtmäßige Überwachung und einen unrechtmäßigen Eingriff in die Privatsphäre von Insidern nieder. Um auch dieser Problematik zu begegnen, wurde eine Datenschutztechnik umgesetzt und verbessert, welche das Missbrauchspotenzial reduziert und dabei gleichzeitig das Erkennungs- und Abwehrpotenzial von Insiderbedrohungen aufrechterhält.

In diesem Abschlusskapitel werden zunächst in Abschnitt 8.1 die eingangs gestellten Forschungsfragen noch einmal aufgegriffen und ihre Beantwortung anhand der geleisteten Forschungsbeiträge überprüft. In Abschnitt 8.2 werden weitere gewonnene Erkenntnisse der Dissertation zusammengefasst und Schlussfolgerungen daraus abgeleitet. Zuletzt erfolgen Schlussbemerkungen und ein Ausblick in Abschnitt 8.3.

8.1 Ergebnisse und Überprüfung der Forschungsfragen

Im Rahmen dieser Dissertation wurden vier Forschungsfragen formuliert und bearbeitet. In diesem Abschnitt wird zusammenfassend aufgezeigt, welche Beiträge die Dissertation zur Beantwortung der Forschungsfragen leisten konnte. Nähere Details finden sich in den jeweiligen Fazit-Abschnitten 2.7, 3.7, 4.11, 5.9, 6.4 und 7.8 der einzelnen Kapitel.

Forschungsfrage 1: Insiderdefinitionen

Welche Charakteristiken definieren einen Insider und welche Insidertypen lassen sich daraus ableiten?

In Kapitel 2 wurden die fünf Charakteristiken *Credentials*, *Knowledge*, *Privileges*, *Trust* sowie *Uncertainty* identifiziert, die eine Person von einem Outsider zu einem Insider machen (vgl. Abschnitt 2.3). Im Unterschied zu Charakteristiken aus den Insiderdefinitionen existierender Literatur (vgl. etwa [And80; Pat03; HP11] sowie die Abschnitte 2.1.1 und 2.2.5) zeichnen sie sich

dadurch aus, dass sie von einer Domäne selbst bereitgestellt werden müssen und demnach einen gewissen Insidergrad beschreiben. Jede dieser Insidercharakteristiken definiert einen eigenen speziellen Basis-Insidertyp und kann in unterschiedlichen Ausprägungen sowie in Kombination mit anderen Insidercharakteristiken vorliegen.

Wie hängen diese Insidertypen miteinander zusammen beziehungsweise wie lassen sie sich voneinander abgrenzen?

Aus den Insidertypen und deren Kombination wurde eine Insidertaxonomie hergeleitet, die von den fünf Basis-Insidertypen bis hin zu einem kombinierten Insidertyp reicht, der alle Insidercharakteristiken in sich vereint (vgl. Abschnitt 2.6). Mit der Taxonomie konnte der Zusammenhang beziehungsweise die Abgrenzung der einzelnen Insidertypen aufgezeigt werden. Sie bietet somit wichtige, einheitliche und strukturgebende Grundlagen für die Forschung und Entwicklung auf dem Gebiet der Insiderthematik. Diese Grundlagen gehen über die vorgeschlagenen Insidertaxonomien in der existierenden Literatur hinaus [And80; Neu99; Tug00; Woo00; MF01; CR05; Pfl+10], die sich auf unterschiedliche Teilmengen dieser Insidertypen beziehen, nur für einen spezifischen Kontext anwendbar sind oder Eigenschaften von Insidern mit Eigenschaften von Bedrohungen vermischen.

Mit welchen Mitteln lassen sich Insidertypen einerseits aus Insiderszenarien und andererseits aus Forschungsbeiträgen und Methodenbeschreibungen herleiten, um beide Seiten entsprechend des inhaltlichen Fokus zusammenzuführen?

Mit der Beschreibung von Insidermodellen in Abschnitt 2.5 und der Anwendung einer qualitativen Inhaltsanalyse, die in Abschnitt 2.5.2 formal beschrieben wurde, wird die Herausarbeitung von Insidertypen aus vorliegenden Insiderszenarien spezifiziert. Daraus ergibt sich die Möglichkeit, Forschungsarbeiten zu identifizieren, die mit ihren Insiderdefinitionen den gleichen Insidertyp fokussieren. Diese Methodik bildet zusammen mit der Insidertaxonomie und den identifizierten Insidercharakteristiken eine neue und universell anwendbare Insiderontologie. Mit ihrer Hilfe wurde die Systematisierung von existierenden Forschungsarbeiten in Abschnitt 2.5.3 sowie in Kapitel 4 bereits begonnen. Darüber hinaus wird die Koordination zukünftiger Forschungs- und Entwicklungsarbeiten ermöglicht.

Forschungsfrage 2: Insiderbedrohungen

Welche Bedrohungen für und durch Insider existieren und von welchem Insidertyp geht welche konkrete Bedrohung aus?

In Kapitel 3 wurde aufgezeigt, dass die Interaktionen zwischen Insidern, Outsidern und der Domäne mehrseitige Bedrohungen hervorrufen können, die jeweils unterschiedlich zu behandeln sind. Der Fokus wurde für diese Dissertation in Abschnitt 3.2 auf die Bedrohungen für Domänen durch Insider gelegt. Querbezüge zu den anderen Bedrohungen wurden allerdings aufgezeigt und ebenfalls beleuchtet, da es sich teilweise um die gleichen Bedrohungen nur aus einer anderen Perspektive handelt. Die fokussierten Insiderbedrohungen wurden in Abschnitt 3.2.1 tiefgehend aufgeschlüsselt und neu charakterisiert. Dabei wurden drei unterschiedliche Klassen von Insiderbedrohungen herausgearbeitet, die *Unbefugte Weitergabe des Insidergrades*, die *Unbefugte Eskalation des Insidergrades* und die *Unbefugte Verhinderung des Insidergrades*. An dieser Stelle kam die explizite Trennung zwischen der Charakterisierung eines Insiders in Kapitel 2 und der einer Insiderbedrohung zum Tragen und erlaubte letzteres zunächst unabhängig von konkret vorliegenden Insidergraden. In existierenden Forschungsarbeiten wurde diese Trennung nicht

vorgenommen [And80; Woo00; MF01; Pat03; MT04; Bis+09]. Dadurch blieben bisher zwei Erkenntnisse verborgen, die in dieser Dissertation aufgedeckt und verwendet beziehungsweise näher untersucht wurden. Zum einen ist das die notwendige Verwendung von Insidergraden bei einer Bedrohungsaktion, damit tatsächlich von einer Insiderbedrohung gesprochen werden kann (vgl. Definition 3.1 in Abschnitt 3.2.1). Zum anderen ist das der Einfluss von vorliegenden und angewendeten Insidergraden auf Insiderbedrohungen. Dieser Einfluss wurde in den Abschnitten 3.2.2 bis 3.2.5 getrennt für alle in Kapitel 2 identifizierten Insidercharakteristiken auf alle drei identifizierten Klassen von Insiderbedrohungen diskutiert und in einem kurzen Abriss daraus mögliche und passgenaue Sicherheitsmechanismen abgeleitet. Darüber hinaus wurde explizit aufgezeigt, welche Insidercharakteristik für welche Insiderbedrohung notwendige Voraussetzung ist und damit die Frage beantwortet, von welchem Insidertyp welche konkrete Bedrohung ausgeht. Zusätzlich wurden diese Ergebnisse für eine bessere Übertragbarkeit in die Praxis in den Abschnitten 3.2.2.6, 3.2.3.3 und 3.2.4.4 den vom BSI definierten elementaren Gefährdungen zugeordnet.

Inwieweit adressieren existierende Sicherheitsmechanismen einzelne Aspekte dieser Insiderbedrohungen und lassen sich somit den identifizierten Insidertypen beziehungsweise deren Bedrohungen zuordnen?

Mit den entwickelten Grundlagen und Forschungsergebnissen aus den Kapiteln 2 und 3 wurde in Kapitel 4 eine Einordnung von neun grundlegenden Erkennungs- und Abwehrmechanismen von Insiderbedrohungen angefertigt. Diese umfassen zum Beispiel die *Plausibilität mehrerer Datenquellen*, welche die Bedrohung einer U-Eskalation durch K-P-Insider anvisiert (vgl. Abschnitt 4.5 und Tabelle 4.1 in Abschnitt 4.10). Die betrachteten Mechanismen stellen keine abschließende Liste dar, aber beinhalten viele Mechanismen, die in der Forschung und Entwicklung im Kontext von Insiderbedrohungen zum Einsatz kommen. Dabei wurde aufgezeigt, dass es einen Unterschied macht, welcher Insidergrad als Voraussetzung für einen Mechanismus vorliegen muss und welche Insidertypen mit diesem Mechanismus anvisiert werden. Die Ergebnisse erlauben eine Einordnung der Mechanismen in die Insidertaxonomie und dadurch Erkenntnisse darüber, welche Sicherheitsmechanismen für die Erkennung und Abwehr von welchen Insiderbedrohungen beziehungsweise Insidertypen zum Einsatz kommen können. Darüber hinaus wurden wissenschaftliche Arbeiten zu den einzelnen Mechanismen aufgeführt, sodass ersichtlich wird, dass beispielsweise Maybury u. a. [May+05] sowie Bowen u. a. [Bow+10] den angesprochenen Erkennungs- und Abwehrmechanismus *Plausibilität mehrerer Datenquellen* in ihren Arbeiten verwenden und somit einen Bezug zum bereits erwähnten anvisierten Insidertyp K-P-Insider beziehungsweise zur anvisierten Insiderbedrohung der U-Eskalation haben.

Forschungsfrage 3: Insiderüberwachung

Wie lassen sich Insideraktivitäten möglichst detailliert erfassen und auf Insiderbedrohungen hin auswerten?

In Kapitel 5 wurde eine neue Technik zur Erkennung und Abwehr von Insiderbedrohungen in einer IT-Umgebung entworfen und softwaretechnisch umgesetzt. Diese Technik macht sich die Tatsache zunutze, dass Aktivitäten von Insidern an Rechnern als Interaktionen zwischen Prozessen, Dateisystemobjekten und Sockets abgebildet werden. Diese Interaktionen werden mithilfe von Systemaufrufen von Betriebssystemen gesteuert und erlauben einen unverfälschten Blick auf die Ziele einer durchgeführten Aktivität. Die Systemaufrufe wurden mithilfe eines

bereits existierenden Auditierungsmechanismus im Linux-Betriebssystem erfasst und aufgezeichnet (vgl. Abschnitt 5.3). Bisherige Arbeiten konzentrieren sich entweder auf die Erfassung und Analyse von Systemaufrufsequenzen [For+96; HFS98; WFP99; ELS01; ARG14; MRA17] oder die Häufigkeiten von Systemaufrufen [LV02; CH14]. Der in dieser Dissertation entwickelte Ansatz erweitert diese Arbeiten und setzt eine Erfassung aller verfügbaren Informationen eines Systemaufrufs mithilfe eines neu entwickelten Werkzeugs namens `audisp-hostmon` um. Dabei wird unter anderem erfasst, welcher Prozess einen Systemaufruf durchführt, um welchen Systemaufruf es sich handelt und welche Systemaufrufparameter dabei übergeben wurden, welche Dateisystemobjekte beziehungsweise Sockets davon beeinflusst wurden und ob der Systemaufruf erfolgreich durchgeführt oder mit einer Fehlermeldung abgebrochen wurde. Das entwickelte Werkzeug erlaubt die Erfassung, Aufbereitung und Filterung der Systemaufrufdaten in Echtzeit und stellt somit die Grundlage für weiterführende Analysetechniken dar. Diese Grundlage ermöglicht eine frühzeitige Erkennung von bedrohlichen Insideraktivitäten und eine anschließende Unterbindung von weiteren Aktivitäten.

Mit welchen Techniken lässt sich die dabei anfallende Datenmenge sowohl sinnvoll reduzieren, ohne dabei wichtige Aktivitätsdetails zu verlieren, als auch automatisiert auswerten, ohne dabei an Genauigkeit zu verlieren?

Die anfallende Datenmenge bei der detaillierten Erfassung von Insideraktivitäten anhand der Systemaufrufinformationen kann weder manuell sinnvoll ausgewertet noch von einem Erkennungs- und Abwehrmechanismus direkt und adäquat verarbeitet werden. Aus diesem Grund wurde in Abschnitt 5.4 eine Überführung der Daten in SysGraphen entwickelt, die eine Repräsentation der Insideraktivitäten darstellen. Sie kodieren nicht nur die Interaktionen zwischen den Prozessen, Dateisystemobjekten und Sockets, sondern auch die chronologische Abfolge dieser Interaktionen. Weiterhin wurde in Abschnitt 5.5 basierend auf der Häufigkeit vorkommender Netzwerkmotive eine Methode vorgestellt, die eine Art Fingerabdruck spezieller Teil-SysGraphen berechnet. Dieser als SysGraph-Signatur benannte Fingerabdruck wurde derart konzipiert, dass er für ähnliche SysGraphen und somit für ähnliche Insideraktivitäten möglichst nur geringe Abweichungen aufweist. Dieses Vorgehen, diese Signaturen und vor allem die zugrunde liegenden Informationen aus Systemaufrufen unterscheiden sich von existierenden Arbeiten, die ebenfalls Techniken zur graphenbasierten Bedrohungserkennung vorstellen [CH00; NC03; EGH10; GSB17], dabei allerdings auf Ereignisinformationen aus der Anwendungsebene oder nicht-technischen Bereichen zurückgreifen, beispielsweise aus der E-Mail-Kommunikation oder den Freundschaftsbeziehungen in sozialen Netzwerken. Mithilfe von geeigneten Ähnlichkeits- und Distanzmetriken zwischen SysGraph-Signaturen wurde in Abschnitt 5.6 die neue Erkennung und Abwehr von Insiderbedrohungen komplettiert. Die dabei verwendeten Metriken, die auf der Kosinusähnlichkeit beziehungsweise auf der Kosinus- und Euklidischen Distanz basieren, sind ersetzbar, sodass der hier entwickelte Mechanismus in den generischen Ansatz von Henderson u. a. [Hen+10] integriert werden könnte, um weitere Metriken sowie verschiedene Analyseverfahren dynamisch anwenden zu können. In Abschnitt 5.7 wurde die neue Erkennungs- und Abwehrtechnik anhand von realen Daten in einem definierten Insiderbedrohungsszenario evaluiert. Die Ergebnisse zeigen eine Erkennungsrate von 99,97% bei einem Ähnlichkeitsschwellwert in Höhe von 98,6% zu einer bekannten ähnlichen Insiderangriffsaktivität und eine Erkennungsrate von 100% bei einem Ähnlichkeitsschwellwert in Höhe von 80%.

Forschungsfrage 4: Insiderdatenschutz

Welche Rechte, Pflichten und Anforderungen haben sowohl Insider als auch die Domäne eines Insiders als Gegenpart in Bezug auf den Datenschutz?

In Kapitel 6 wurden rechtliche Rahmenbedingungen in Deutschland sowie die gegenläufigen Anforderungen beim Einsatz von Maßnahmen zur Erkennung und Abwehr von Insiderbedrohungen jeweils aus Sicht des Arbeitgebers als Domäne und aus Sicht der betroffenen Arbeitnehmer als Insider herausgearbeitet. Der rechtliche Rahmen leitet sich für die Seite der betroffenen Arbeitnehmer aus dem Grundrecht auf Privatsphäre und informationelle Selbstbestimmung ab, das aus den Art. 7 und 8 der EU-GRCh sowie dem Art. 2 Abs. 1 in Verbindung mit Art. 2 Abs. 1 aus dem deutschen GG hervorgeht. Es wurde in Abschnitt 6.1.1 aufgezeigt, dass es sich beim Einsatz von Erkennungs- und Abwehrmaßnahmen um eine Verarbeitung personenbezogener Daten handelt, die in den sachlichen Anwendungsbereich der EU-DSGVO fällt, sofern keine bereichsspezifischen Datenschutzregelungen existieren. Weiterhin wurde herausgearbeitet, dass der damit vorhandene Schutzrahmen auch für den Beschäftigungskontext und damit für Situationen gilt, in denen Arbeitgeber Sicherheitsmaßnahmen zur Erkennung und Abwehr von Insiderbedrohungen einsetzen und damit Schaden für ihr Unternehmen abwenden wollen. Umgekehrt ist ein solcher Einsatz für Arbeitgeber rechtlich durchaus erlaubt und auch durch die DSGVO nicht prinzipiell verboten. Mögliche und für den rechtlichen Rahmen eines Arbeitgebers wichtige Erlaubnistatbestände wurden in Abschnitt 6.1.2 beleuchtet. Dazu gehören gegebenenfalls vorliegende anderweitige rechtliche Verpflichtungen, beispielsweise eine Sorgfalts- und Legalitätspflicht, sowie die Wahrung berechtigter Interessen, die sich auch für die Seite der Arbeitgeber aus den Grundrechten ableiten. Diese beinhalten das Recht auf Berufs- und unternehmerische Freiheit (Art. 15 und 16 der EU-GRCh sowie Art. 12 des deutschen GG), dessen Verletzung durch die Erkennung und Abwehr von Insiderbedrohungen verhindert werden kann. Anhand der in Abschnitt 6.2 identifizierten Anforderungen wurde aufgezeigt, dass eine Domäne beim Einsatz von Erkennungs- und Abwehrtechniken das Ziel verfolgt, unterschiedliche Aktivitätsereignisse aufzuzeichnen, miteinander verketten und realen Personen zuordnen zu können. Demgegenüber stehen die Datenschutzprinzipien, die aus Sicht der betroffenen Insider rechtlich geboten sind und technisch durchgesetzt werden müssen. Betrachtet wurden dabei die Zweckbindung, die Datenminimierung, die De-Identifizierung sowie die Speicherbegrenzung.

Wie lassen sich wichtige Prinzipien des Datenschutzes auch im Kontext der Insiderüberwachung praktisch realisieren, sodass eine Insiderüberwachung nicht für andere Zwecke missbraucht werden kann, ohne dabei wichtige Maßnahmen der Domäne für den Schutz vor Insiderbedrohungen unbrauchbar zu machen?

Die praktische Umsetzung der aufgezeigten wichtigen Prinzipien des Datenschutzes wurde in Abschnitt 6.2.2 vorbereitet und in Kapitel 7 mit der Entwicklung eines neuen Pseudonymisierungsverfahrens namens PEEPLL durchgeführt. Im Vergleich mit existierenden Arbeiten, die sich auf den Datenschutz beim Einsatz von IDSs ausschließlich mit Teilaspekten der genannten Prinzipien beschäftigen [SFR97; BK99; LPS04; Loc+05], wurde in dieser Dissertation die umfassende Realisierung aller Prinzipien und Anforderungen verfolgt. Als Basis-Datenschutztechnik wurde die Pseudonymisierung verwendet, da sie allen mehrseitigen Anforderungen gerecht wird. Insbesondere die Verkettbarkeit von Aktivitätsereignissen bleibt erhalten und im Fall einer Bedrohungserkennung kann die dadurch erreichte De-Identifizierung umgekehrt und eine Attribution bestimmter Aktivitäten zu Akteuren durchgeführt werden. Die Speicherbegrenzung wurde indirekt durch eine Limitierung der Verkettbarkeit adressiert. Die Zweckbindung wurde ähnlich zu den Arbeiten von Biskup und Flegel [BF00a; BF00b] mithilfe eines kryptographischen

Schwellwertschemas technisch durchgesetzt, sodass eine Re-Identifizierung nur noch stattfinden kann, wenn mehrere unabhängige Akteure nach dem Vorbild des Mehr-Augen-Prinzips kooperieren. Die Datenminimierung wurde einerseits dadurch umgesetzt, dass irrelevante Daten von vornherein gefiltert und nicht weiterverarbeitet werden. Andererseits wurden weitere subtilere Informationsquellen, die durch die praktische Umsetzung des neuen Pseudonymisierungsverfahrens entstehen, identifiziert und nach Möglichkeit reduziert beziehungsweise verhindert.

Welche zusätzlichen Maßnahmen können umgesetzt werden, um Schwächen von Datenschutztechniken zum Schutz der Insider vor unrechtmäßiger Überwachung bei der praktischen Realisierung zu beheben?

Aus den im vorigen Absatz angesprochenen subtilen Informationsquellen, die erst durch die praktische Umsetzung des Pseudonymisierungsverfahrens entstehen, wurden in Abschnitt 7.5 die Pseudonymisierungs-Schutzziele *QID-Vertraulichkeit* sowie *Unbeobachtbarkeit des passenden Pseudonyms* hergeleitet. Mithilfe der kryptographischen Funktionen HMAC und 1-out-of-N OT in Verbindung mit dem Einsatz von Bloom-Filtern wurde aufgezeigt, wie diese Schutzziele praktisch durchgesetzt und in das neue Pseudonymisierungsverfahren integriert werden können. Dadurch konnte eine Erhöhung des Datenschutzes erreicht werden, wenngleich die Evaluation von PEEPLL einen messbaren Performanzverlust bei der Verarbeitung von Ereignisdaten im Zuge der Erkennung und Abwehr von Insiderbedrohungen verursacht. Diesem Performanzverlust wurde entgegengewirkt, indem verschiedene Pseudonymisierungs-Schutzziele in PEEPLL konfigurierbar sind und somit dynamisch an die Umstände eines konkreten Einsatzszenarios angepasst werden können.

8.2 Weitere Erkenntnisse und Schlussfolgerungen

Eine große Fehlkonzeption, die bei der Behandlung von Insiderbedrohungen sowohl in der wissenschaftlichen Forschung als auch in der unternehmerischen Praxis gemacht wird, zeigt sich darin, dass der Insiderbegriff bereits von vornherein negativ belastet ist. Das wird zum Beispiel anhand des Begriffs *Insiderhandel* deutlich, der synonym für einen illegalen Handel mit Insiderwissen verwendet wird [Mah95]. Tatsächlich müssten bei genauerer Betrachtung jeglicher erlaubter und unerlaubter Austausch von Informationen, Waren oder Dienstleistungen als Insiderhandel verstanden werden. Der illegale Charakter ist jedoch anderweitig begründet und kann nicht nur an einem vorliegenden Insidergrad festgemacht werden. Gleiches gilt für *Insideraktivitäten*, die allgegenwärtig sind und das Funktionieren einer Domäne aufrechterhalten, allerdings häufig als Bedrohungs- oder Angriffsaktivitäten aufgefasst werden. Aus dieser Fehlkonzeption entsteht die gängige Praxis, die Beschreibung eines Insiders mit der einer Insiderbedrohung zu vermischen. Diese subtile Vereinfachung hat große, nachteilige Auswirkungen auf die Bekämpfung von Insiderbedrohungen. Sie verdeckt das Potenzial, dass aus einer getrennten, detaillierten Betrachtung von Insidergrad-gebenden Charakteristiken (Forschungsbeitrag B1) sowie von tatsächlichen Bedrohungseigenschaften der Insiderbedrohungen (Forschungsbeitrag B2) hervorgeht. Weitere Erkenntnisse bezüglich Fehlkonzeptionen in der Insiderthematik wurden in Abschnitt 2.2 zusammengetragen. Sie sollten in zukünftigen Forschungsarbeiten auf dem Gebiet der Insiderthematik beachtet und vermieden werden.

Durch die detaillierte Benennung und Einordnung von Bedrohungen in Verbindung mit Insidern in Kapitel 3 wird deutlich, dass einerseits mehrseitige Bedrohungen vorliegen, die durch Insider aber auch für Insider entstehen (vgl. Abschnitt 3.1). Andererseits handelt es sich bei

einer Bedrohung durch Insider nur dann um eine tatsächliche Insiderbedrohung, wenn ein vorliegender Insidergrad bei der Bedrohungsaktion zum Einsatz kommen (vgl. Abschnitt 3.2.1). Diese strikte Abtrennung von Insiderbedrohungen erlaubt eine Klassifizierung dieser Bedrohungen unabhängig von konkret vorliegenden Insidercharakteristiken und darüber hinaus die Herausarbeitung, welche konkreten Auswirkungen ein vorliegender Insidergrad auf diese Klassen von Insiderbedrohungen hat (Forschungsbeitrag B2). Ebenso erlaubt sie bei der Forschung und Entwicklung von Sicherheitsmaßnahmen in zukünftigen Arbeiten den expliziten Fokus auf spezielle Klassen von Insiderbedrohungen sowie auf die angesprochenen Auswirkungen einzelner Insidercharakteristiken auf Insiderbedrohungen.

In Abschnitt 3.2.5.5 wurde erarbeitet, welchen aus Sicht einer Domäne negativen Einfluss die Insidercharakteristik *Uncertainty* auf den Erfolg, die Verdecktheit und die Auswirkung einer Bedrohungsaktion hat. Dabei wurde ersichtlich, dass diese Charakteristik gleichzeitig für die meisten legitimen Aktivitäten von Insidern nicht unbedingt benötigt wird. Darüber hinaus hat die Systematisierung von Insiderdefinitionen aus existierenden Forschungsarbeiten in Abschnitt 2.5.3 aufgezeigt, dass die *Uncertainty* in dieser Literatur zur Insiderthematik unterrepräsentiert ist. Dort ist eher ein Fokus auf *Credentials* und *Privileges* zu verzeichnen. Dementsprechend wurde bei der Entwicklung einer neuen Erkennungs- und Abwehrtechnik von Insiderbedrohungen in Kapitel 5 (Forschungsbeitrag B3) der Fokus auf die Reduzierung der *Uncertainty* gelegt, wenngleich noch unklar ist, ob sich der erwähnte Fokus in der Literatur nur auf die Insiderdefinitionen von Forschungsarbeiten beschränkt oder auch auf die verwendeten grundlegenden Sicherheitsmaßnahmen übertragbar ist. Hier besteht noch Forschungsbedarf.

Durch die Herausarbeitung der Bezüge von grundlegenden Sicherheitsmechanismen zu speziellen Insidertypen und -bedrohungen in Kapitel 4 (Forschungsbeitrag B2) wird ersichtlich, dass der Fokus auf einen anvisierten Insidertypen in einer Forschungsarbeit nicht anhand einer von den Autoren gewählten Insiderdefinition erkennbar ist. Die in Abschnitt 2.6 aufgeführte Arbeit von Spitzner [Spi03b] zeigt diese Diskrepanz zwischen der gewählten Insiderdefinition und dem tatsächlich durch die Sicherheitsmaßnahme anvisierten Insidertyp exemplarisch. Sofern also eine systematische Einordnung von Forschungsarbeiten in die aus dieser Dissertation hervorgehende Insidertaxonomie erfolgen soll, muss dies anhand der Zuordnung des verwendeten grundlegenden Sicherheitsmechanismus zu den Insidertypen geschehen.

Die Erfassung von Insideraktivitäten an einem Rechner auf Systemaufrufebene entsprechend der Arbeiten in Kapitel 5 (Forschungsbeitrag B3) erlaubt die Aufdeckung tatsächlicher Ressourceninteraktionen und Informationsflüssen, ohne dabei von Verschleierungstechniken behindert zu werden. Dieses Vorgehen hat demnach großes Potenzial, wenn es darum geht, sich bei der Erkennung und Abwehr von Insiderbedrohungen auf die Ziele von Insideraktivitäten zu konzentrieren und nicht auf die Wege, mit denen diese Ziele erreicht werden können. Dennoch zeigen die Ergebnisse der Evaluation in Abschnitt 5.7 auf, dass selbst bei gleichem Ziel, in dem Fall der unerlaubten Verschlüsselung von Dateien auf einem Netzlaufwerk, messbare Unterschiede in den in Abschnitt 5.5 entwickelten SysGraph-Signaturen und damit auch in den Systemaufrufen vorliegen können. Neun Kryptotrojaner-Proben konnten in der Trainingsphase auf drei unterschiedliche Cluster aufgeteilt werden, obwohl alle Proben das gleiche Ziel verfolgen. Darüber hinaus wiesen zwei der insgesamt 6199 Kryptotrojaner-Proben in der Testphase einen derart großen Unterschied auf, dass sie keinem der drei Cluster zugeordnet werden konnten. Es bleibt somit offen, herauszufinden, welche Aktivitätsschritte diese Unterschiede hervorrufen und ob es andere Metriken oder gänzlich andere Signaturen benötigt, die robuster gegenüber diesen Unterschieden arbeiten können. Darüber hinaus heben die SysGraph-Signaturen gleichförmige

und häufige Ressourceninteraktionen hervor. Vereinzelte und unscheinbare Aktivitäten bleiben darin allerdings eher verborgen. Die Effektivität einer Erkennung und Abwehr von fortgeschrittenen und andauernden Bedrohungen (engl. advanced persistent threats) basierend auf diesen SysGraph-Signaturen ist demnach höchst fragwürdig.

8.3 Schlussbemerkungen und Ausblick

Insiderbedrohungen sind Bedrohungen per Definition. Sie existieren, weil Insidergrade für die Aufgabenerfüllung von einer Domäne bereitgestellt werden müssen und dabei gleichzeitig für bewusste oder unbewusste Bedrohungsaktionen gegen die Domäne oder andere Insider verwendet werden können. Daher ist es nicht verwunderlich, dass Insiderbedrohungen so alt sind, wie die Menschheitsgeschichte selbst. In der jüngeren Vergangenheit wurden immer wieder Insiderangriffe bekannt [Zet11; Kan14; Ege17; Tho17; Pie18; Tho18; Won19; Scr20], die aufzeigen, dass das Problem von Insiderbedrohungen nach wie vor ein ungelöstes Problem ist. Vor diesem Hintergrund ist es allerdings überraschend, dass ein tiefgehendes Verständnis der Insiderthematik sowie eine umfassende Systematisierung und Bündelung von Forschungs- und Entwicklungsarbeiten bisher nicht erreicht werden konnte.

Die Erhebung und Veröffentlichung von Statistiken, die auf die Präsenz und Schwere von Insiderbedrohungen aufmerksam machen [Pon+19; Ver19], sind wichtige Bausteine bei der Bekämpfung von Insiderbedrohungen. Gleiches gilt für die Erfassung und Analyse von realen Fällen von Insiderangriffen sowie deren Umstände, wie es etwa durch das *CERT National Insider Threat Center*[1] an der Carnegie Mellon University vorangetrieben wird. Daraus hervor gehen wichtige Ratgeber und Seminare, die gelernte Lektionen und bewährte Praktiken zusammenfassen und der Öffentlichkeit zur Verfügung stellen. Doch durch die zugrunde liegende Einzelfall-Basis lassen sich zukünftige Insiderangriffe allzu leicht an die Ratgeber anpassen. Das Resultat kommt einer Symptombekämpfung gleich und läuft auf eine fortwährende Überarbeitung und Anpassung an aktuelle Entwicklungen von Insiderbedrohungen hinaus. Grundlegende Erkenntnisse und Wissensschöpfungen bleiben dadurch aus, die dabei helfen könnten, die Wurzeln des Problems offenzulegen und nachhaltig zu dessen Lösung beizutragen.

Die Forschung und Entwicklung auf dem Gebiet der Erkennungs- und Abwehrmaßnahmen von Insiderbedrohungen ist als sehr aktiv einzuschätzen. Allerdings müssen diese Arbeiten in Zukunft anhand einer eindeutigen und umfassenden Insiderontologie inklusive einheitlicher, passgenauer und in allen Bereichen anwendbarer Terminologie strukturiert werden. Zusätzlich wird es nötig sein, die bereits existierenden Arbeiten anhand formaler Methoden dieser Ontologie zuzuordnen. Die Forschungsergebnisse und -erkenntnisse dieser Dissertation verfolgen das Ziel, dabei grundlegende Beiträge zu leisten, hinterlassen allerdings weiterhin offene Fragen und möglicherweise strittige Punkte, die eine Weiterentwicklung und Verbesserung benötigen.

Die Bedrohungen für Insider selbst, wenn sie Sicherheitsmaßnahmen zur Erkennung und Abwehr von Insiderbedrohungen ausgesetzt sind, wurden in der Forschung auf dem Gebiet der Insiderthematik bisher wenig bis gar nicht adressiert und bearbeitet. Die Entwicklungen der letzten Jahre auf dem Gebiet des Datenschutz-Rechts mit der Einführung der EU-DSGVO stärkt die Seite der betroffenen Insider, allerdings bleibt der Dual-Use-Charakter von Erkennungs- und Abwehrtechniken und ein unentdeckter Missbrauch durch eine Domäne davon unberührt. Umso wichtiger ist eine durch diese Dissertation aufgezeigte und angestoßene Entwicklung

1. Die Webseite des CERT National Insider Threat Centers ist unter https://www.cert.org/insider-threat/ erreichbar.

von technisch durchsetzbaren Kontroll- und Datenschutzmechanismen, die den mehrseitigen Interessen und Pflichten aller Beteiligten gerecht werden.

A Anhang

A.1 Aus der Dissertation hervorgegangene Vorveröffentlichungen

2016 E. Zimmer, J. Lindemann, D. Herrmann und H. Federrath. *Catching Inside Attackers: Balancing Forensic Detectability and Privacy of Employees*. In: *Open Problems in Network Security - IFIP WG 11.4 International Workshop, iNetSec 2015, Zurich, Switzerland, October 29, 2015, Revised Selected Papers*. Hrsg. von J. Camenisch und D. Kesdogan. Bd. 9591. Lecture Notes in Computer Science. Springer, 2016, S. 43–55.

2020 E. Zimmer, C. Burkert, T. Petersen und H. Federrath. *PEEPLL: Privacy-Enhanced Event Pseudonymisation with Limited Linkability*. In: *Proceedings of the 35th Annual ACM Symposium on Applied Computing*. SAC '20. Brno, Czech Republic: Association for Computing Machinery, 2020, S. 1308–1311.

A.2 Insidermodelle und Insidertypen existierender Insiderdefinitionen

Nachfolgend aufgeführt sind die 83 analysierten Insiderdefinitionen, die aus 47 unterschiedlichen wissenschaftlichen Publikationen extrahiert wurden, sowie deren Abgeleitete Insidermodelle und Insidertypen.

A.2.1 Anderson [And80]

Definition: „[T]he term "external penetration" is not confined to the usual case of an outsider attempting to access to a computer resource in an organization of which he is not a part. The term is meant to convey, in addition to the previous case, the notion of an employee of the organization who has physical access to the building housing the computer system but who is not an authorized computer user."

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: negligible → P-Outsider

Insider Type: C_A-Insider

Domain: A computer system

Insider Characteristics:

- Credentials: no → C-Outsider

Insider Type: C-Outsider

Definition: „He can be any category of individual; either an external penetrator[1] who has succeeded in penetrating the installation access controls, [...]“

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: low – high → P-Insider

Insider Type: C_A-P-Insider

Domain: A computer system

Insider Characteristics:

- Credentials: no → C-Outsider
- Privileges: low – high → P-Insider

Insider Type: P-Insider

Definition: „[...] or an employee without full access to a computer system, [...]“

Domain: The organisation, a computer system

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: low – middle → P-Insider

Insider Type: C_A-P-Insider

1. Which means either a C_A-Insider-P-Outsider or a C-Outsider. See page 229.

Definition: „[...] or possibly an employee with full access to a computer system who wishes to exploit another legitimate users identification and password that he may have obtained.“

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider[2]
- Privileges: low – high → P-Insider

Insider Type: C_A-P-Insider

Domain: A computer system

Insider Characteristics:

- Credentials: stolen → C_M-Insider
- Privileges: low – high → P-Insider

Insider Type: C_M-P-Insider

Definition: „The legitimate user as a threat to information resources is a case of misfeasance in that it involves the misuse of authorized access both to the system and to its data.“

Domain: The organisation

Insider Characteristics:

- Privileges: low – high → P-Insider

Insider Type: P-Insider

Domain: A computer system

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: low – high → P-Insider

Insider Type: C_A-P-Insider

2. From the point of view of the target, the stolen username and password are not representing the Credentials. They represent the access to resources. The employee still identifies to the target as him/herself.

Definition: „The assumption regarding clandestine users is that the user has or can seize supervisory control of the machine and as such can either operate below the level at which audit trail data is taken or can use privileges or system primitives to evade audit trail data being recorded for him.“

Domain: The organisation, a computer system

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: high → P-Insider
- Uncertainty: high → U-Insider

Insider Type: C_A-P-U-Insider

A.2.2 Loch, Carr und Warkentin [LCW92]

Definition: „[A] threat can be internal to the organization as the result of employee action [...].“

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider

Insider Type: C_A-Insider

A.2.3 Committee on Information Systems Trustworthiness [Com98]

Definition: „A person with legitimate physical access to computer equipment.“

Domain: The organisation hosting the computer equipment

Insider Characteristics:

- Credentials: legit → C_A-Insider[3]
- Privileges: low – high → P-Insider

Insider Type: C_A-P-Insider

3. The context of this definition makes it clear, that a membership / authentication is assumed.

Definition: „A person with some sort of organizational status that causes members of the organization to view requests or demands as being authorized.“

Domain: The organisation

Insider Characteristics:

- Trust: low – high → T-Insider

Insider Type: T-Insider

Definition: „A person with some level of privilege or authority with regard to the computer system.“

Domain: The organisation

Insider Characteristics:

- Privileges: low – high → P-Insider

Insider Type: P-Insider

A.2.4 Neumann [Neu99]

Definition: „We assume that relative to a particular computational framework, insiders are users who have been authenticated to operate within that framework; where necessary, we qualify that to include reference to the authorized privileges that are associated with a particular authentication.“

Domain: The computational framework

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: low – high ∨ negligible → P-Outsider ∨ P-Insider

Insider Type: C_A-P-Insider ∨ C_A-Insider

A.2.5 Lundin und Jonsson [LJ00]

Definition: „A masquerader can be defined as a person, either external or internal, who uses an account on the system for which he is not authorized.“

Domain: A system

Insider Characteristics:

- Credentials: stolen → C_M-Insider

Insider Type: C_M-Insider

Definition: „An insider is a legitimate user who misuses the system.“

Domain: A system

Insider Characteristics:

- Credentials: legit → C_A-Insider

Insider Type: C_A-Insider

A.2.6 Strunk u. a. [Str+00]

Definition: „A successful intruder can obtain the rights and identity of a legitimate user or administrator. With these rights, it is possible to disrupt a system by accessing, modifying, or destroying critical data.“

Domain: The organisation

Insider Characteristics:

- Credentials: stolen → C_M-Insider
- Privileges: low – high → P-Insider

Insider Type: C_M-P-Insider

A.2.7 Einwechter [Ein02]

Definition: „Insider Attacks are an unusual type of threat. Unlike external attacks, the intruder is someone who has been entrusted with authorized access to the network. In fact, the attacker requires access in order to fulfil their obligations to the victim organization. Furthermore, they often have a substantial amount of knowledge about the network architecture, including where their targeted files or systems are located.“

Domain: The network

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Knowledge: middle → K-Insider

Insider Type: C_A-K-Insider

A.2.8 Schultz [Sch02]

Definition: „[Insiders are] those who are authorized to use computers and networks.“

Domain: The organisation hosting the computers and networks

Insider Characteristics:

- Privileges: low – high → P-Insider

Insider Type: P-Insider

Domain: The computers and networks

Insider Characteristics:

- Credentials: stolen – legit → C-Insider

Insider Type: C-Insider

A.2.9 Nguyen, Reiher und Kuenning [NRK03]

Definition: „Insiders represent the greatest threat to computer security because they understand their organization's business and how their computer systems work. They have both the confidentiality and access to perform these attacks. [...] The insiders also represent the greatest challenge to securing the company network because they are authorized a level of access to the file system and granted a degree of trust. “

Domain: The organisation

Insider Characteristics:

- Knowledge: low – middle → K-Insider
- Privileges: low – high → P-Insider
- Trust: low – high → T-Insider

Insider Type: K-P-T-Insider

A.2.10 Patzakis [Pat03]

Definition: „[M]any incident response teams fail to recognize and prepare for security compro- mises perpetrated by insiders. [...] As reflected by recent surveys, many incidents [...] are the work of rogue employees [...]“

Domain: Within the perimeter

Insider Characteristics:

- Credentials: legit → C_A-Insider

Insider Type: C_A-Insider

Definition: „[...] and other trusted individuals.“

Domain: Within the perimeter

Insider Characteristics:

- Trust: low – high → T-Insider

Insider Type: T-Insider

A.2.11 Spitzner [Spi03a]

Definition: „[S]omeone who is technically skilled, highly motivated, and has access to extensive resources. For example, this threat may be an employee working for a large corporation, but in reality they are employed by a competitor to engage in corporate espionage. A second example is highly skilled, disgruntled employee motivated to cause a great deal of damage before they are fired. A third example could be a spy working for a foreign country. Regardless of who the insider is, we are dealing with a highly dangerous threat, one that is extremely difficult to detect. They have access to critical information; they know the structure of the organization."

Domain: The organisation

Insider Characteristics:

- Credentials: no $\vee$ legit $\rightarrow$ C-Outsider $\vee$ C_A-Insider
- Knowledge: middle $\rightarrow$ K-Insider
- Privileges: high $\rightarrow$ P-Insider

Insider Type: C_A-K-P-Insider $\vee$ K-P-Insider

A.2.12 Brackney und Anderson [BA04]

Definition: „An already trusted person with access to sensitive information and information systems."

Domain: The organisation

Insider Characteristics:

- Privileges: middle $\rightarrow$ P-Insider
- Trust: low – high $\rightarrow$ T-Insider

Insider Type: P-T-Insider

Domain: The information system

Insider Characteristics:

- Credentials: stolen – legit $\rightarrow$ C-Insider
- Trust: low – high $\rightarrow$ T-Insider

Insider Type: C-T-Insider

Definition: „Someone with access [to information systems or services], [...]"

Domain: Information systems or services

Insider Characteristics:

- Credentials: stolen – legit → C-Insider

Insider Type: C-Insider

Definition: „[...] privileges [of information systems or services], [...]“

Domain: Information systems or services

Insider Characteristics:

- Privileges: low – high → P-Insider

Insider Type: P-Insider

Definition: „[...] or knowledge of information systems or services.“

Domain: Information systems or services

Insider Characteristics:

- Knowledge: low – high → K-Insider

Insider Type: K-Insider

A.2.13 Jha u. a. [Jha+04]

Definition: „[…] the masquerade-detection problem, which is defined as determining the identity of the user that generated a given execution trace.“

Domain: Anything

Insider Characteristics:

- Credentials: stolen → C_M-Insider

Insider Type: C_M-Insider

A.2.14 Maxion und Townsend [MT04]

Definition: „There are many ways for a masquerader to gain access to legitimate user accounts, e.g., through a purloined password or a hacker's break in. [...]"

Domain: The organisation

Insider Characteristics:

- Credentials: stolen → C_M-Insider

Insider Type: C_M-Insider

Definition: „The term [masquerader] may also be extended to encompass abuse of legitimate privileges – the case in which a user "masquerades" as himself; such a person is sometimes termed an "insider," especially when the person is an accepted member of the organization sponsoring the target system. "

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: low – high → P-Insider

Insider Type: C_A-P-Insider

A.2.15 Aleman-Meza u. a. [Ale+05]

Definition: „Insider Threat refers to the potential malevolent actions by employees within an organization, a specific type of which relates to legitimate access of documents."

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: low – high → P-Insider

Insider Type: C_A-P-Insider $\vee$ C_A-Insider

A.2.16 Bishop [Bis05]

Definition: „An insider with respect to rules R is a user who may take action that would violate some set R of rules in the security policy were the user not trusted. The insider is trusted to take the action only when appropriate, as determined by the insider's discretion."

Domain: The system enforcing the rules R

Insider Characteristics:

- Credentials: stolen – legit → C-Insider
- Privileges: low – high → P-Insider
- Trust: middle → T-Insider

Insider Type: C-P-T-Insider

A.2.17 Butts, Mills und Baldwin [BMB05]

Definition: „Any individual who has been granted any level of trust in an information system."

Domain: The information system

Insider Characteristics:

- Trust: low – high ∨ negligible → T-Insider ∨ T-Outsider

Insider Type: T-Insider ∨ T-Outsider

A.2.18 Chinchani u. a. [Chi+05]

Definition: „We assume that every legitimate user is an insider. [...] Insiders are in a unique position with the privileges entrusted to them and the knowledge about their com- putational environment [...]. They are a part of an organization and bound by the organization policy [...]."

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Knowledge: low – high → K-Insider
- Privileges: low – high → P-Insider
- Uncertainty: low → U-Insider

Insider Type: C_A-K-P-U-Insider

Definition: „External attackers can become insiders too by compromising an internal system and learning about the computers in the neighborhood.“

Domain: The organisation

Insider Characteristics:

- Credentials: stolen → C_M-Insider
- Knowledge: low – middle → K-Insider

Insider Type: C_M-K-Insider

A.2.19 Cole und Ring [CR05]

Definition: „Outside affiliates are non-trusted outsiders who use open access to gain access to an organization’s resources [and have no legitimate reason to access the building].“

Domain: The organisation

Insider Characteristics:

- Credentials: no → C-Outsider
- Privileges: low → P-Insider
- Trust: negligible → T-Outsider

Insider Type: P-Insider

Definition: „An insider affiliate is a spouse, friend, or even client of an employee who uses the employee’s credentials to gain access [and has no legitimate reason to access the building].“

Domain: The organisation

Insider Characteristics:

- Credentials: stolen → C_M-Insider

Insider Type: C_M-Insider

Definition: „Insider associates are people who have limited authorized access [and have legitimate reason to access the building]. [. . .] Limited access usually takes the form of having physical access to the facility but not access to the network.“

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: negligible → P-Outsider

Insider Type: C_A-Insider

Domain: The network

Insider Characteristics:

- Credentials: no → C-Outsider

Insider Type: C-Outsider

Definition: „A pure insider is an employee with all the rights and access associated with being employed by the company. Typically, they have keys or a badge to get access to the facility, a logon to get access to the network, and can walk around the building unescorted.“

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: low – high → P-Insider

Insider Type: C_A-P-Insider

Domain: The network

Insider Characteristics:

- Credentials: legit → C_A-Insider

Insider Type: C_A-Insider

Definition: „Elevated pure insider is an [pure] insider who has additional privileged access. This usually includes system administrators who have root or administrator access on the network.“

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: high → P-Insider

Insider Type: C_A-P-Insider

Domain: The network

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: high → P-Insider

Insider Type: C_A-P-Insider

A.2.20 Liu u. a. [Liu+05]

Definition: „The “insider thred” involves the actions of a trusted and privileged user who is inappropriately accessing or disseminating sensitive information or otherwise compromising information systems.“

Domain: Information systems

Insider Characteristics:

- Privileges: low – high → P-Insider
- Trust: low – high → T-Insider

Insider Type: P-T-Insider

Definition: „Insider threat refers to cases in which users who have legitimate access to a system abuse their privileges for purposes not related to their authorized use. “

Domain: A system

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: low – high → P-Insider

Insider Type: C_A-P-Insider

A.2.21 Maybury u. a. [May+05]

Definition: „An insider as anyone in an organization with approved access, privilege, or knowledge of information systems, information services, and missions.“

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: low – high → P-Insider
- Knowledge: low – high → K-Insider

Insider Type: P-Insider ∨ K-Insider ∨ C_A-Insider

A.2.22 Ning und Sun [NS05]

Definition: „Insiders have full control of some [network] nodes.“

Domain: The network

Insider Characteristics:

- Privileges: middle → P-Insider

Insider Type: P-Insider

A.2.23 Magklaras, Furnell und Brooke [MFB06]

Definition: „An insider is a person that has been legitimately given the capability of accessing one or many components of the IT infrastructure, by interacting with one or more authentication mechanisms (plain text password, PKI, biometric or smart card token). [...] It also means that an insider is less likely to get caught by implemented security measures because of the level of trust that she enjoys.“

Domain: The IT infrastructure

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: low – high → P-Insider
- Uncertainty: middle → U-Insider

Insider Type: C_A-P-U-Insider

A.2.24 Shirey [Shi07]

Definition: „An insider [is] an entity that is authorized to access system resources but uses them in a way not approved by the party that granted the authorization."

Domain: A system

Insider Characteristics:

- Privileges: low – high → P-Insider

Insider Type: P-Insider

Definition: „A user (usually a person) that accesses a system from a position that is inside the system's security perimeter."

Domain: A system

Insider Characteristics:

- Credentials: stolen – legit → C-Insider
- Privileges: low – high → P-Insider

Insider Type: P-Insider ∨ C-Insider

A.2.25 Greitzer u. a. [Gre+08]

Definition: „The insider is an individual currently or at one time authorized to access an organization's information system, data, or network; such authorization implies a degree of trust in the individual."

Domain: The organisation

Insider Characteristics:

- Privileges: low – high → P-Insider
- Trust: low – high → T-Insider

Insider Type: P-T-Insider

Domain: An organisation's Information system, data, or network

Insider Characteristics:

- Credentials: stolen – legit → C-Insider
- Trust: low – high → T-Insider

Insider Type: C-T-Insider

A.2.26 Kamra, Terzi und Bertino [KTB08]

Definition: „[S]ubjects that are legitimate users of the system, and thus may have access rights to data and resources. "

Domain: A system

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: low – high ∨ negligible → P-Outsider ∨ P-Insider

Insider Type: C_A-P-Insider ∨ C_A-Insider

A.2.27 Pfleeger [Pfl08]

Definition: „An insider can be an employee, student, or other member of a host institution that operates a computer system to which the insider has legitimate access [...]"

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: low – high → P-Insider

Insider Type: C_A-P-Insider

Domain: A computer system

Insider Characteristics:

- Credentials: legit → C_A-Insider

Insider Type: C_A-Insider

Definition: „[... An insider can be] an associate, contractor, business partner, supplier, computer maintenance technician, guest, or someone else who has a formal or informal business relationship with the institution [...]“

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider

Insider Type: C_A-Insider

Definition: „[... An insider can be] anyone authorized to perform certain activities, for example a bank's customer who uses the bank's system to access his or her account [...]“

Domain: A system

Insider Characteristics:

- Privileges: low – high → P-Insider

Insider Type: P-Insider

Definition: „[... An insider can be] anyone properly identified and authenticated to the system including, perhaps, someone masquerading as a legitimate insider, or someone to whom an insider has given access (for example by sharing a password) [...]“

Domain: A system

Insider Characteristics:

- Credentials: stolen – legit → C-Insider

Insider Type: C-Insider

Definition: „[... An insider can be] someone duped or coerced by an outsider to perform actions on the outsider's behalf [...]“

Domain: Anything

Insider Characteristics:

- Nothing mentioned

Insider Type: Outsider

Definition: „[... An insider can be] a former insider, now using previously conferred access credentials not revoked when the insider status ended or using access credentials secretly created while an insider to give access later.“

Domain: The former organisation

Insider Characteristics:

- Credentials: stolen → C_M-Insider

Insider Type: C_M-Insider

A.2.28 Bishop u. a. [Bis+09]

Definition: „[A] security policy is inherently represented by the access control rules employed by an organization. So, an insider is defined with regard to two primitive actions: 1. violation of a security policy using legitimate access, [...]“

Domain: The organisation

Insider Characteristics:

- Privileges: low – high → P-Insider

Insider Type: P-Insider

Definition: „[A] security policy is inherently represented by the access control rules employed by an organization. So, an insider is defined with regard to two primitive actions: [...] 2. violation of an access control policy by obtaining unauthorized access.“

Domain: The organisation

Insider Characteristics:

- Privileges: low – high → P-Insider

Insider Type: P-Insider

Definition: „[P]revious definitions gave rules or descriptions intended to allow the reader to determine who is an insider, resulting in a binary distinction: an entity is either an insider or not an insider. We argue that a non-binary approach is required, to indicate degrees of “insiderness,” and that the access control rules for an organization can be used to develop these degrees. We define someone as an insider with respect to access to some data or resource X.“

Domain: The organisation

Insider Characteristics:

- Privileges: low – high → P-Insider

Insider Type: P-Insider

A.2.29 Doss und Tejay [DT09]

Definition: „The term insider can be defined as any user that either currently or at one time was authorized to access an organization’s information systems. [...] External attacks can become or considered insiders through the proxy of a current insider. [...] Insiders have greater privileges and knowledge of their organization than external attackers. Insiders have the trust of their organization [. . .]. The insider typically knows the location of the assets or targets. There is little to no need for the insider to perform reconnaissance activities [... and] Insiders may also know where security controls are located and how they are used.“

Domain: The organisation

Insider Characteristics:

- Knowledge: low – high → K-Insider
- Privileges: low – high → P-Insider
- Trust: low – high → T-Insider
- Credentials: no → C-Outsider[4]

Insider Type: K-P-T-Insider

Domain: Information system, data, or network

Insider Characteristics:

- Credentials: stolen – legit → C-Insider
- Knowledge: low – high → K-Insider
- Privileges: low – high → P-Insider
- Trust: low – high → T-Insider

Insider Type: C-K-P-T-Insider

A.2.30 Pfleeger und Stolfo [PS09]

Definition: „[P]eople with legitimate access who behave in ways that put our data, our systems, our organizations, and even our businesses' viability at risk."

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider[5]
- Privileges: low – high → P-Insider[6]

Insider Type: P-Insider ∨ C_A-Insider

Definition: „[P]eople and systems who receive short- or long-term access to an organization's systems."

Domain: The organisation

Insider Characteristics:

- Privileges: low – high → P-Insider

4. „at one time was"
5. Assuming the legitimate access is meant for the target itself and not resources of the target.
6. Assuming the legitimate access is meant for resources of the target and not the target itself.

Insider Type: P-Insider

Domain: A system

Insider Characteristics:

- Credentials: stolen – legit → C-Insider

Insider Type: C-Insider

A.2.31 Probst u. a. [Pro+10a]

Definition: „An insider is a person that has been legitimately empowered with the right to access, represent, or decide about one or more assets of the organization's structure."

Domain: The organisation

Insider Characteristics:

- Privileges: low – high → P-Insider

Insider Type: P-Insider

A.2.32 Bishop u. a. [Bis+10]

Definition: „Let P_L and P_H be representations of a policy at different levels of the Unifying Policy Hierarchy. If a subject has a different set of rights in P_L than it has in P_H, it is called an insider. An insider attack occurs when an insider employs any rights that exist in P_L and that do not exist in P_H."

Domain: A system

Insider Characteristics:

- Privileges: low – high → P-Insider

Insider Type: P-Insider

A.2.33 Bowen u. a. [Bow+10]

Definition: „We define insider threats by [...] Masqueraders (attackers who impersonate another inside user) [... and who are] presumed to have less knowledge of a system than the victim user [...].“

Domain: Anything

Insider Characteristics:

- Credentials: stolen → C_M-Insider
- Knowledge: low – middle ∨ negligible → K-Insider ∨ K-Outsider

Insider Type: C_M-Insider ∨ C_M-K-Insider

Definition: „We define insider threats by [...] Traitors (an inside attacker using their own legitimate credentials) who each have varying levels of knowledge.“

Domain: Anything

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Knowledge: low – high ∨ negligible → K-Insider ∨ K-Outsider

Insider Type: C_A-K-Insider ∨ C_A-Insider

A.2.34 Greitzer und Frincke [GF10]

Definition: „The “insider” is an individual currently or at one time authorized to access an organization’s information system, data, or network; such authorization implies a degree of trust in the individual. “

Domain: The organisation

Insider Characteristics:

- Privileges: low – high → P-Insider
- Trust: low – high → T-Insider

Insider Type: P-T-Insider

Domain: Information system, data, or network

Insider Characteristics:

- Credentials: no ∨ legit → C-Outsider ∨ C_A-Insider
- Trust: low – high → T-Insider
- Knowledge: middle → K-Insider

Insider Type: K-T-Insider ∨ C_A-T-Insider

A.2.35 Kandias u. a. [Kan+10]

Definition: „[A]n insider is a human entity that has/had access to the information system of an organization and does not comply with the security policy of the organization.“

Domain: The organisation

Insider Characteristics:

- Privileges: low – high → P-Insider

Insider Type: P-Insider

Domain: The information system

Insider Characteristics:

- Credentials: stolen – legit → C-Insider

Insider Type: C-Insider

Domain: The organisation; the information system

Insider Characteristics:

- Credentials: no → C-Outsider[7]
- Knowledge: low – high → K-Insider

Insider Type: K-Insider

A.2.36 Mathew u. a. [Mat+10]

Definition: „An insider is a database subject who has personal knowledge of information stored in one or more fields marked confidential.“

7. „that had access“

Domain: The database

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Knowledge: low – middle → K-Insider

Insider Type: C_A-K-Insider

A.2.37 Pfleeger u. a. [Pfl+10]

Definition: „A person with legitimate access to an organization's computers and networks."

Domain: The organisation

Insider Characteristics:

- Privileges: low – high → P-Insider

Insider Type: P-Insider

Domain: Information system, data, or network

Insider Characteristics:

- Credentials: legit → C_A-Insider

Insider Type: C_A-Insider

A.2.38 Yaseen und Panda [YP10]

Definition: „It is defined as the threat that is caused by a malicious insider who has authorized access privileges and knowledge of the computer systems of an organization, and is inspired to antagonistically influence the organization."

Domain: The organisation

Insider Characteristics:

- Knowledge: low – high → K-Insider
- Privileges: low – high → P-Insider

Insider Type: K-P-Insider

Domain: The computer systems

Insider Characteristics:

- Credentials: stolen – legit → C-Insider
- Knowledge: low – high → K-Insider

Insider Type: C-K-Insider

A.2.39 Hunker und Probst [HP11]

Definition: „[A]n insider is defined as an individual with privileged access to an IT system.“

Domain: The information system

Insider Characteristics:

- Privileges: low – high → P-Insider

Insider Type: P-Insider

Definition: „We would observe that in practice – at least to the extent that we are able to observe real incidents – the problem of real interest is the “real real insider”; an individual deeply embedded in an organization, highly trusted, and in a position to do great damage if so inclined (e.g., a high level executive, or a systems administrator).“

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: high → P-Insider
- Trust: high → T-Insider

Insider Type: C_A-P-T-Insider

A.2.40 Fuchs und Pernul [FP12b]

Definition: „Insider threats represent the historical cause of the majority of incidents with the authorized, non-technical employee being the typical potential threat to Information Security “

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: low – high → P-Insider

Insider Type: C_A-P-Insider

Definition: „[I]ndividuals that have a trusted relationship with organizations - namely (former) employees, contractors, or consultants - represent a typical potential threat for Information Security. Those authorized non-technical insiders directly interact with an organization's Information Systems, have some type of authority on those systems, and know about security processes and how to circumvent them."

Domain: The organisation

Insider Characteristics:

- Credentials: no ∨ legit → C-Outsider ∨ C_A-Insider
- Knowledge: low – high → K-Insider
- Privileges: low – high → P-Insider
- Trust: low – high → T-Insider
- Uncertainty: low – high → U-Insider

Insider Type: K-P-T-U-Insider ∨ C_A-K-P-T-U-Insider

A.2.41 Kissel [Kis13]

Definition: „An entity with authorized access (i.e., within the security domain) that has the po- tential to harm an information system or enterprise through destruction, disclosure, modification of data, and/or denial of service."

Domain: Security domain (e.g., an information system or enterprise)

Insider Characteristics:

- Credentials: stolen – legit → C-Insider

Insider Type: C-Insider

A.2.42 Vickers [US13]

Definition: „Anyone who has authorized access to [Department of Defence (DoD)] resources by virtue of employment, volunteer activities, or contractual relationship with DoD.“

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Privileges: low – high → P-Insider

Insider Type: C_A-P-Insider

Definition: „A person with authorized access, who uses that access, wittingly or unwittingly, to harm national security interests or national security through unauthorized disclosure, data modification, espionage, terrorism, or kinetic actions resulting in loss or degradation of resources or capabilities.“

Domain: Anything

Insider Characteristics:

- Privileges: low – high → P-Insider

Insider Type: P-Insider

Definition: „A person, known or suspected, who uses their authorized access to DoD facilities, personnel, systems, equipment, information, or infrastructure to damage and disrupt operations, compromise DoD information, or commit espionage on behalf of an [Foreign Intelligence Entity].“

Domain: DoD facilities, personnel, systems, equipment, information, or infrastructure

Insider Characteristics:

- Credentials: stolen – legit → C-Insider

Insider Type: C-Insider

A.2.43 Kaplan u. a. [Kap+15]

Definition: „The errant insider is unaware of security protocols, ignores cybersecurity policies or simply commits errors resulting in sensitive data being compromised or networks becoming vulnerable.“

Domain: Anything

Insider Characteristics:

- Knowledge: negligible → K-Outsider

Insider Type: K-Outsider

Definition: „The hijacked insider has their credentials compromised by someone externally, giving the outsider the same level of access as the insider.“

Domain: Anything

Insider Characteristics:

- Credentials: stolen → C_M-Insider
- Privileges: low – high → P-Insider

Insider Type: C_M-P-Insider

Definition: „Finally, the malevolent insider is willing to steal or compromise data for personal gain.“

Domain: Anything

Insider Characteristics:

- Nothing mentioned

Insider Type: Outsider

A.2.44 McGough u. a. [McG+15]

Definition: „An insider threat can be defined as [...]: the intent to inflict harm by one who has special knowledge or access to confidential information. “

Domain: Anything

Insider Characteristics:

- Knowledge: low – high → K-Insider
- Privileges: low – high → P-Insider

Insider Type: P-Insider ∨ K-Insider

A.2.45 Maasberg, Warren und Beebe [MWB15]

Definition: „Insiders have unique access to information systems. [...] Compared to an outsider, insiders typically have some level of access, authorization, and/or advanced organizational knowledge.“

Domain: The organisation

Insider Characteristics:

- Privileges: low – high → P-Insider
- Credentials: stolen – legit → C-Insider
- Knowledge: low – high ∨ high → K-Insider

Insider Type: P-Insider ∨ K-Insider ∨ C-Insider ∨ C-K-P-Insider

Definition: „The insider threat occurs when trusted members of an organization “behave in ways that put our data, our systems, our organizations, and even our businesses’ viability at risk”. “

Domain: The organisation

Insider Characteristics:

- Credentials: legit → C_A-Insider
- Trust: low – high → T-Insider

Insider Type: C_A-T-Insider

A.2.46 Shulman, Cherny und Dulce [SCD17]

Definition: „Embodiments of the invention provide for methods, systems, and apparatuses using reverse honey tokens to enable a “compromised insider” production honeypot and thereby detect if an intruder has compromised a client end station in an attempt to gain unauthorized access to enterprise data of one or more servers.“

Domain: The organisation

Insider Characteristics:

- Credentials: no → C-Outsider
- Privileges: low – high → P-Insider

Insider Type: P-Insider

A.2.47 Khandelwal [Kha17]

Definition: „FIO (foreign intelligence officers) actors [who can steal documents].“

Domain: Anything

Insider Characteristics:

- Privileges: low – high → P-Insider

Insider Type: P-Insider

A.3 Formale Methodik zur Analyse von Bedrohungsszenarien und -definitionen

In Ergänzung zu Abschnitt 2.5.2 wird hier die formale Methodik zur Analyse von Insiderszenarien und -definitionen auf diejenigen Charakteristiken erweitert, die eine genaue Beschreibung beziehungsweise Modellierung des Bedrohungspotenzials eines Bedrohungsagenten erlauben. Bei den Charakteristiken handelt es sich um *Abilities*, *Behaviour*, *Compliance*, *Intention* und *Resources*, die in Abschnitt 3.2.1.1 eingeführt wurden.

Zur Erinnerung: Die zum Einsatz kommende *strukturierende Inhaltsanalyse* besteht den Autoren Mayring und Fenzl [MF14, Kap. 38.3.3] zufolge aus zwei Schritten. Im ersten Schritt werden theoriegeleitet-deduktiv festgestellte Kategorien benannt und beschrieben. Im zweiten Schritt werden Textteile anhand eines Kodierleitfadens diesen Kategorien zugeordnet. Die bereits vorhandene Kategorie der *Insidercharakteristiken* aus Schritt 1 wird nun um die Kategorie der *Bedrohungscharakteristiken* ergänzt. Die konkret genannten Bedrohungscharakteristiken stellen damit die Subkategorien dar:

- ***Abilities* (A)**,
- ***Behaviour* (B)**,
- ***Compliance* (C)**,
- ***Intention* (I)** und
- ***Resources* (R)**.

Die Definitionen der Subkategorien findet sich in Abschnitt 3.2.1.1 und eine numerische Ordnung innerhalb der Subkategorien ist in Tabelle A.1 zusammengefasst. Der Kodierleitfaden für Schritt 2 wird nun entsprechend der in Tabelle A.2 beschriebenen Spezifikationen erweitert.

Tabelle A.1: Nummerische Ordnung der Charakteristikausprägungen der in Abschnitt 3.2.1.1 aufgeführten Bedrohungscharakteristiken

	Abilities (A)	**Behaviour (B)**	**Compliance (C)**	**Intention (I)**	**Resources (R)**
-1	incompetent	omitted	–	–	blank
0	not stated	not stated	not stated	not stated	not stated
1	proficient	passive	–	intentionally good-natured	little
1,5	proficient – skilled	–	–	non-malicious	little – limited
2	skilled	active	non-compliant	accidental	limited
2,5	proficient – expert	–	–	intentionally or unintentionally	little – unlimited
3	expert	adaptive	compliant	intentionally malicious	unlimited

Tabelle A.2: Kodierleitfaden für die Zuordnung von Textteilen verschiedener Bedrohungsbeschreibungen zu den jeweiligen Ordnungen der Bedrohungscharakteristiken

Ordnung	Subkategorie	Kodierregel	Ankerbeispiel
-1	**A**, **B**, **R**	Die Subkategorie wird als nicht-vorhanden beschrieben.	A(-1): „A script kiddie [...] is someone who lacks programming knowledge and uses existing software to launch an attack. Often a script kiddie will use these programs without even knowing how they work or what they do“ [Put18]; B(-1): Ein Sicherheitsdienstmitarbeiter lässt seine obligatorische Schließrunde ausfallen und bemerkt dadurch nicht, dass ein Mitarbeiter einen Keil in eine Außentür gelegt hat, um nach Arbeitsschluss ungestört wertvolle Güter entwenden und verkaufen zu können.
0	**A**, **B**, **C**, **I**, **R**	Die Subkategorie wird nicht genannt und es lässt sich auch weder aus der Kodiereinheit noch aus der Kontexteinheit explizit darauf schließen.	A(0), B(0), C(0), I(0), R(0): „An insider [attacker] can be an employee, student, or other member of a host institution that operates a computer system to which the insider has legitimate access [...]“ [Pfl08].
1	**A**, **R**	Die Bedrohungsbeschreibung deutet auf ein eindeutig niedriges Level der Subkategorie hin.	A(1): „Insider [attackers] represent the historical cause of the majority of incidents with the authorized, non-technical employee being the typical potential threat to Information Security“ [FP12b].
	B	Die beschriebene Person interagiert nicht aktiv mit der vorliegenden Domäne, sondern betrachtet nur passiv den Zustand der Domäne sowie die möglicherweise vorhandenen Ein- und Ausgaben.	B(1): Ein Sicherheitsadministrator hat die Aufgabe, bereits aufgezeichnete Computerereignisse durchzusehen und auf ungewöhnliche Vorkommnisse hin auszuwerten. Er druckt sich dabei Teile der Daten und Metadaten auf seinem privaten Drucker aus, aus denen sich sensible Informationen ableiten lassen und verkauft sie an Konkurrenten der Domäne.

	I	Die Absicht der beschriebenen Person wird als bewusst gutartig beziehungsweise gutmütig beschrieben.	I(1): Um das Leben eines Patienten in einer Notfallsituation zu retten, greift ein Krankenpfleger unbefugt auf die Akte des Patienten zu und bringt somit die lebensrettende Medikation zur Stabilisierung in Erfahrung.
1,5	**A**, **I**, **R**	Die Beschreibung der Subkategorie deutet auf eine Ordnung zwischen 1 und 2 hin.	R(1,5): Bei einem Bruteforce-Angriff auf gehashte Passwörter benötigt ein Angreifer entweder genügend Rechenleistung oder einfach nur eine zufällig passende Wörterliste, um schnell zum Ziel zu kommen.
2	**A**, **R**	Die Bedrohungsbeschreibung deutet auf ein eindeutig mittleres Level der Subkategorie hin.	A(2): „[Insider attackers] represent the greatest threat to computer security because they understand their organization's business and how their computer systems work. They have both the confidentiality and access to perform these attacks“ [NRK03].
	B	Die beschriebene Person interagiert aktiv mit der vorliegenden Domäne und trägt möglicherweise eigene Eingaben an die Domäne heran.	B(2): Ein Sicherheitsadministrator hat die Aufgabe, relevante Computerereignisse aufzuzeichnen und auf ungewöhnliche Vorkommnisse hin auszuwerten. Er konfiguriert die Aufzeichnungsregeln und konzentriert sich dabei speziell auf Ereignisse, die Daten und Metadaten liefern, aus den sensible Informationen abgeleitet werden können. Er druckt sich dabei Teile der Daten und Metadaten aus und verkauft sie an Konkurrenten der Domäne.
	C	Die beschriebene Person agiert auf unerlaubte Art und Weise beziehungsweise hält sich nicht an Vorgaben und Spezifikationen.	C(2): Ein Krankenpfleger verwendet die ihm bekannten Zugangsdaten eines Arztes, um Einsicht in das medizinische Patientendatensystem seiner Einrichtung zu erhalten (vgl. Beispiel 2.3 in Abschnitt 2.5.1).

	I	Die Absicht der beschriebenen Person wird als nicht-vorhanden beziehungsweise unbewusst oder versehentlich beschrieben.	I(2): „A person with authorized access, who uses that access [...] unwittingly, to harm national security interests [...]“ [US13].
2,5	**A**, **I**, **R**	Die Beschreibung der Subkategorie deutet auf eine Ordnung zwischen 1 und 3 hin.	I(2,5): „A person with authorized access, who uses that access, wittingly or unwittingly, to harm national security interests [...]“ [US13]; A(2,5): „An entity [...] that has the potential to harm an information system or enterprise through destruction, disclosure, modification of data, and/or denial of service“ [Kis13].
3	**A**, **R**	Die Bedrohungsbeschreibung deutet auf ein eindeutig hohes Level der Subkategorie hin.	A(3), R(3): „[S]omeone who is technically skilled, highly motivated, and has access to extensive resources. For example, this threat may be an employee working for a large corporation, but in reality they are employed by a competitor to engage in corporate espionage. A second example is highly skilled, disgruntled employee motivated to cause a great deal of damage before they are fired. A third example could be a spy working for a foreign country“ [Spi03a].
	B	Die beschriebene Person interagiert adaptiv mit der vorliegenden Domäne und trägt eigene Eingaben an die Domäne heran, die sie unter Einbezug vergangener Beobachtungen des Domänenzustandes verändert.	B(3): Ein Sicherheitsadministrator hat die Aufgabe, Computerereignisse aufzuzeichnen und auf ungewöhnliche Vorkommnisse hin auszuwerten. Er konfiguriert die Aufzeichnungsregeln und konzentriert sich dabei speziell auf Ereignisse, die Daten und Metadaten liefern, aus den sensible Informationen abgeleitet und an Konkurrenten der Domäne verkauft werden können. Er passt seine Konfigurationen an, nachdem er feststellt, dass die bisher aufgezeichneten Ereignisse keine für seinen Plan brauchbaren Informationen liefern.

C	Die beschriebene Person agiert auf erlaubte Art und Weise beziehungsweise hält sich an Vorgaben und Spezifikationen.	C(3): Ein Streifenpolizist macht eine Halterabfrage eines korrekt geparkten Autos, um an die Kontaktdaten der attraktiven jungen Halterin zu gelangen (vgl. Beispiel 2.2 in Abschnitt 2.5.1).
I	Die Absicht der beschriebenen Person wird als bewusst bösartig beziehungsweise böswillig beschrieben.	I(3): „It is defined as the threat that is caused by a malicious insider [...]“ [YP10].

Literaturverzeichnis

[AD08] S. Amann und Deutscher Depeschendienst. *Stasi-Methoden beim Discounter. Lidl ließ Mitarbeiter systematisch bespitzeln*. Online News. Spiegel, 26. März 2008. URL: https://www.spiegel.de/wirtschaft/stasi-methoden-beim-discounter-lidl-liess-mitarbeiter-systematisch-bespitzeln-a-543431.html (besucht am 24. 07. 2020) (siehe S. 2, 149, 166, 186).

[AD15] F. Armknecht und A. Dewald. *Privacy-Preserving Email Forensics*. In: *Digital Investigation* 14 (2015). The Proceedings of the Fifteenth Annual DFRWS Conference, S. 127–136 (siehe S. 172).

[Aft15] S. Aftergood. *Soviet Spy Ronald W. Pelton to be Released from Prison*. Online News. Federation of American Scientists, 23. Nov. 2015. URL: https://fas.org/blogs/secrecy/2015/11/pelton-release/ (besucht am 24. 07. 2020) (siehe S. 54).

[Ale+05] B. Aleman-Meza u. a. *An Ontological Approach to the Document Access Problem of Insider Threat*. In: *Intelligence and Security Informatics*. Hrsg. von P. Kantor u. a. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, S. 486–491 (siehe S. 38, 86, 94, 239).

[And80] J. P. Anderson. *Computer Security Threat Monitoring and Surveillance*. Techn. Ber. Fort Washington, PA: James P. Anderson Co., 1980. URL: http://csrc.nist.gov/publications/history/ande80.pdf (besucht am 24. 07. 2020) (siehe S. 18–20, 22, 24, 27, 28, 37, 38, 70, 219–221, 229).

[ARG14] N. Assem, T. Rachidi und M. T. E. Graini. *Intrusion detection using bayesian classifier for arbitrarily long system call sequences*. In: *IADIS International Journal on Computer Science and Information Systems* 9.1 (2014), S. 71–81 (siehe S. 107, 222).

[Ash+17] G. Asharov u. a. *More Efficient Oblivious Transfer Extensions*. In: *Journal of Cryptology* 30.3 (1. Juli 2017), S. 805–858 (siehe S. 184).

[BA04] R. Brackney und R. H. Anderson, Hrsg. *Understanding the Insider Threat. Proceedings of a March 2004 Workshop*. Santa Monica, CA: RAND Corporation, 2004 (siehe S. 18, 19, 38, 237).

[BE01] R. A. Botha und J. H. P. Eloff. *Separation of duties for access control enforcement in workflow environments*. In: *IBM Systems Journal* 40.3 (2001), S. 666–682 (siehe S. 70).

[BF00a] J. Biskup und U. Flegel. *Threshold-based Identity Recovery for Privacy Enhanced Applications*. In: *Proceedings of the 7th ACM Conference on Computer and Communications Security*. CCS '00. Athens, Greece: ACM, 2000, S. 71–79 (siehe S. 172, 223).

[BF00b] J. Biskup und U. Flegel. *Transaction-Based Pseudonyms in Audit Data for Privacy Respecting Intrusion Detection*. In: *Recent Advances in Intrusion Detection*. Hrsg. von H. Debar, L. Mé und S. F. Wu. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, S. 28–48 (siehe S. 172, 223).

[Bis+08] M. Bishop u. a. *We Have Met the Enemy and He is Us*. In: *Proceedings of the 2008 Workshop on New Security Paradigms*. NSPW '08. New York, NY, USA: ACM, 2008, S. 1–12 (siehe S. 43).

[Bis+09] M. Bishop u. a. *Case Studies of an Insider Framework*. In: *42nd Hawaii International Conference on System Sciences*. HICSS '09. Jan. 2009, S. 1–10 (siehe S. 19, 22, 24, 38, 221, 248).

[Bis+10] M. Bishop u. a. *A Risk Management Approach to the "Insider Threat"*. In: *Insider Threats in Cyber Security*. Hrsg. von C. W. Probst u. a. Bd. 49. Advances in Information Security. Springer US, 2010, S. 115–137 (siehe S. 19, 23, 38, 251).

[Bis05] M. Bishop. *Position: "Insider" is Relative*. In: *Proceedings of the 2005 Workshop on New Security Paradigms*. Hrsg. von C. F. Hempelmann und V. Raskin. NSPW '05. New York, NY, USA: ACM, 2005, S. 77–78 (siehe S. 19, 38, 240).

[BK99] R. Büschkes und D. Kesdogan. *Privacy Enhanced Intrusion Detection*. In: *Proceedings of the Conference on Multilateral Security for Global Communication*. München: Addison-Wesley-Longman, 1999, S. 187–207 (siehe S. 171, 223).

[Blo70] B. H. Bloom. *Space/Time Trade-offs in Hash Coding with Allowable Errors*. In: *Communications of the ACM* 13.7 (Juli 1970), S. 422–426 (siehe S. 171, 182).

[BMB05] J. W. Butts, R. F. Mills und R. O. Baldwin. *Developing an Insider Threat Model Using Functional Decomposition*. In: *Computer Network Security: Proceedings of the third International Workshop on Mathematical Methods, Models, and Architectures for Computer Network Security*. Hrsg. von V. Gorodetsky, I. Kotenko und V. Skormin. MMM-ACNS 2005. Berlin, Heidelberg: Springer, 2005, S. 412–417 (siehe S. 18, 19, 35–38, 240).

[Bor14] J. Bort. *How The Hackers Broke Into Sony And Why It Could Happen To Any Company*. Online News. Business Insider, 19. Dez. 2014. URL: https://www.businessinsider.com/how-the-hackers-broke-into-sony-2014-12 (besucht am 24. 07. 2020) (siehe S. 2).

[Bow+10] B. M. Bowen u. a. *Monitoring Technologies for Mitigating Insider Threats*. In: *Insider Threats in Cyber Security*. Hrsg. von C. W. Probst u. a. Bd. 49. Advances in Information Security. Springer US, 2010, S. 197–217 (siehe S. 28, 38, 88, 89, 221, 251).

[BSS15] K. Berlin, D. Slater und J. Saxe. *Malicious Behavior Detection Using Windows Audit Logs*. In: *Proceedings of the 8th ACM Workshop on Artificial Intelligence and Security*. AISec '15. Denver, Colorado, USA: Association for Computing Machinery, 2015, S. 35–44 (siehe S. 108).

[Bun19] Bundesamt für Sicherheit in der Informationstechnik. *IT-Grundschutz-Kompendium*. 2. Edition. Bundesanzeiger-Verlag, Feb. 2019 (siehe S. 11, 12, 84, 85).

[BWJ08] C. Bettini, X. S. Wang und S. Jajodia. *How Anonymous Is k-Anonymous? Look at Your Quasi-ID*. In: *Secure Data Management*. Hrsg. von W. Jonker und M. Petković. Springer Berlin Heidelberg, 2008, S. 1–15 (siehe S. 169).

[CB95] D. A. Cooper und K. P. Birman. *Preserving privacy in a network of mobile computers*. In: *Proceedings 1995 IEEE Symposium on Security and Privacy*. 1995, S. 26–38 (siehe S. 200).

[CBK09] V. Chandola, A. Banerjee und V. Kumar. *Anomaly Detection: A Survey*. In: *ACM Compututing Surveys (CSUR)* 41.3 (Juli 2009), 15:1–15:58 (siehe S. 90, 91, 129, 132).

[CBR03] W. R. Cheswick, S. M. Bellovin und A. D. Rubin. *Firewalls and Internet Security: Repelling the Wily Hacker*. 2. Aufl. USA: Addison-Wesley Longman Publishing Co., Inc., 2003 (siehe S. 76).

[CF14] M. Carvalho und R. Ford. *Moving-Target Defenses for Computer Networks*. In: *IEEE Security Privacy* 12.2 (März 2014), S. 73–76 (siehe S. 77).

[CH00] D. J. Cook und L. B. Holder. *Graph-based data mining*. In: *IEEE Intelligent Systems and Their Applications* 15.2 (2000), S. 32–41 (siehe S. 105, 222).

[CH14] G. Creech und J. Hu. *A semantic approach to host-based intrusion detection systems using contiguousand discontiguous system call patterns*. In: *IEEE Transactions on Computers* 63.4 (2014), S. 807–819 (siehe S. 107, 222).

[Che+06] J. Chen u. a. *NeMoFinder: Dissecting Genome-wide Protein-protein Interactions with Meso-scale Network Motifs*. In: *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '06. Philadelphia, PA, USA: ACM, 2006, S. 106–115 (siehe S. 104).

[Chi+05] R. Chinchani u. a. *Towards a theory of insider threat assessment*. In: *International Conference on Dependable Systems and Networks*. DSN '05. Juni 2005, S. 108–117 (siehe S. 38, 87, 240).

[CL11] D. D. Clark und S. Landau. *Untangling Attribution*. In: *Harvard National Security Journal* 2.2 (2011), S. 323–352 (siehe S. 166).

[CLY17] L. Cheng, F. Liu und D. D. Yao. *Enterprise data breach: causes, challenges, prevention, and future directions*. In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 7.5 (2017) (siehe S. 128).

[CO15] T. Chou und C. Orlandi. *The Simplest Protocol for Oblivious Transfer*. In: *Progress in Cryptology – LATINCRYPT 2015*. Hrsg. von K. Lauter und F. Rodríguez-Henríquez. Cham: Springer International Publishing, 2015, S. 40–58 (siehe S. 183, 184).

[Com17] Common Criteria Development Board. *Common Criteria for Information Technology Security Evaluation*. Version 3.1 Revision 5. Apr. 2017. URL: https://www.commoncriteriaportal.org (besucht am 24. 07. 2020) (siehe S. 108).

[Com98] Committee on Information Systems Trustworthiness. *Trust in cyberspace*. Hrsg. von F. B. Schneider. Washington, DC, USA: National Academy Press, 1998 (siehe S. 19, 21, 22, 37, 38, 232).

[Coo71] S. A. Cook. *The Complexity of Theorem-proving Procedures*. In: *Proceedings of the Third Annual ACM Symposium on Theory of Computing*. STOC '71. Shaker Heights, Ohio, USA: ACM, 1971, S. 151–158 (siehe S. 104).

[Cor04] J. Corbet. *The lightweight auditing framework*. Linux Weekly News, LWN, 7. Apr. 2004. URL: https://lwn.net/Articles/79326/ (besucht am 24. 07. 2020) (siehe S. 108).

[CR05] E. Cole und S. Ring. *Insider threat: Protecting the enterprise from sabotage, spying, and theft*. Elsevier, 2005 (siehe S. 20, 36, 38, 220, 241).

[Cra16] L. Cranor. *Time to rethink mandatory password changes*. Blog Post. Federal Trade Commission, 2. März 2016. URL: https://www.ftc.gov/news-events/blogs/techftc/2016/03/time-rethink-mandatory-password-changes (besucht am 24. 07. 2020) (siehe S. 67).

[CRT17] C. Culnane, B. I. P. Rubinstein und V. Teague. *Health Data in an Open World*. In: *CoRR* abs/1712.05627 (2017) (siehe S. 169).

[CW00] C. Clases und T. Wehner. *Vertrauen*. Essay. Lexikon der Psychologie. Spectrum Akademischer Verlag, 2000. URL: https://www.spektrum.de/lexikon/psychologie/vertrauen/16374 (besucht am 24. 07. 2020) (siehe S. 77).

[Däu17] W. Däubler. *Gläserne Belegschaften. Das Handbuch zum Beschäftigtendatenschutz*. 7. Aufl. Bund Verlag, 2017 (siehe S. 150, 152, 153, 161, 163, 164).

[Des14a] G. Destuynder. *Audisp-cef. CEF plugin for audisp (Linux Audit)*. Jan. 2014. URL: https://github.com/gdestuynder/audisp-cef (besucht am 24. 07. 2020) (siehe S. 113).

[Des14b] G. Destuynder. *Audisp-json*. Jan. 2014. URL: https://github.com/gdestuynder/audisp-json (besucht am 24. 07. 2020) (siehe S. 113).

[DF90] Y. Desmedt und Y. Frankel. *Threshold cryptosystems*. In: *Advances in Cryptology - CRYPTO '89 Proceedings*. Hrsg. von G. Brassard. Bd. 435. Lecture Notes in Computer Science. New York, NY: Springer, 1990, S. 307–315 (siehe S. 172, 181, 193).

[DH76] W. Diffie und M. Hellman. *New directions in cryptography*. In: *Information Theory, IEEE Transactions on* 22.6 (Nov. 1976), S. 644–654 (siehe S. 178, 214).

[Die17] R. Diestel. *Graph Theory*. 5th Edition. Bd. 173. Graduate Texts in Mathemtatics. Berlin/Heidelberg, Germany: Springer-Verlag, 2017 (siehe S. 102, 115).

[Dou+16] M. Douriez u. a. *Anonymizing NYC Taxi Data: Does It Matter?* In: *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*. Okt. 2016, S. 140–148 (siehe S. 169).

[DSGVO16] European Parliament and Council of the European Union. *Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)*. In: *Official Journal of the European Union* L 119/1 (59 4. Mai 2016), S. 1–88 (siehe S. 153, 176).

[DT09] G. Doss und G. Tejay. *Developing Insider Attack Detection Model: A Grounded Approach*. In: *2009 IEEE International Conference on Intelligence and Security Informatics*. Juni 2009, S. 107–112 (siehe S. 38, 249).

[Dwo06] C. Dwork. *Differential Privacy*. In: *33rd International Colloquium on Automata, Languages and Programming, part II (ICALP 2006)*. Bd. 4052. Lecture Notes in Computer Science. Springer Verlag, Juli 2006, S. 1–12 (siehe S. 168).

[ED17] S. Eckert und A. Dewes. *Dark Data*. Presentation at DEFCON 25. English slides available. 20. Aug. 2017. URL: https://media.defcon.org/DEF%20CON%2025/DEF%20CON%2025%20presentations/DEF%20CON%2025%20-%20Eckert-and-Dewes-Dark-Data-UPDATED.pdf (besucht am 24. 07. 2020) (siehe S. 169).

[Ege17] B. Egelko. *Spy conviction upheld for Bay Area businessman*. SFGate Online News, 5. Mai 2017. URL: https://www.sfgate.com/business/article/Spy-conviction-upheld-for-Bay-Area-businessman-11125307.php (besucht am 24. 07. 2020) (siehe S. 226).

[EGH10] W. Eberle, J. Graves und L. Holder. *Insider Threat Detection Using a Graph-Based Approach*. In: *Journal of Applied Security Research* 6.1 (2010), S. 32–81 (siehe S. 105, 222).

[EH07] W. Eberle und L. Holder. *Anomaly detection in data represented as graphs*. In: *Intelligent Data Analysis* 11.6 (2007), S. 663–689 (siehe S. 105).

[Ein02] N. Einwechter. *Preventing and Detecting Insider Attacks Using IDS*. In: *SecurityFocus* (März 2002) (siehe S. 18, 19, 29, 38, 234).

[Elg85] T. Elgamal. *A public key cryptosystem and a signature scheme based on discrete logarithms*. In: *IEEE Transactions on Information Theory* 31.4 (Juli 1985), S. 469–472 (siehe S. 178).

[ELS01] E. Eskin, W. Lee und S. J. Stolfo. *Modeling system calls for intrusion detection with dynamic window sizes*. In: *Proceedings DARPA Information Survivability Conference and Exposition II. DISCEX'01*. Bd. 1. Juni 2001, S. 165–175 (siehe S. 107, 222).

[Eve+11] B. S. Everitt u. a. *Cluster Analysis*. 5. Aufl. Wiley Series in Probability and Statistics. Wiley, 2011 (siehe S. 139).

[FG15] H. Fleischer und W. Goette. *Münchener Kommentar zum Gesetz betreffend die Gesellschaften mit beschränkter Haftung – GmbHG*. 3. Auflage. München: C.H. Beck, 2015 (siehe S. 159).

[FG19] R. Faith und S. Grubb. *Linux Audit*. Red Hat, 2019. URL: https://people.redhat.com/sgrubb/audit/ (besucht am 24. 07. 2020) (siehe S. 108).

[FGO18] M. Franzen, I. Gallner und H. Oetker. *Kommentar zum europäischen Arbeitsrecht*. 2. Auflage. München: C.H. Beck, 2018 (siehe S. 150, 154, 155, 157, 158, 160).

[For+19] R. A. Ford u. a. *Privacy protection during insider threat monitoring*. US Patent App. 10/318,729. 11. Juni 2019. URL: https://patents.google.com/patent/US10318729B2/en (besucht am 24. 07. 2020) (siehe S. 170).

[For+96] S. Forrest u. a. *A sense of self for Unix processes*. In: *Proceedings of the 1996 IEEE Symposium on Security and Privacy*. Mai 1996, S. 120–128 (siehe S. 106, 222).

[FP00] H. Federrath und A. Pfitzmann. *Gliederung und Systematisierung von Schutzzielen in IT-Systemen*. In: *Datenschutz und Datensicherheit: DuD* 24.12 (2000), S. 704–710. Vieweg (siehe S. 11).

[FP12a] H. Federrath und A. Pfitzmann. *Datenschutz und Datensicherheit*. In: *Taschenbuch der Informatik*. Hrsg. von U. Schneider und D. Werner. 7th Edition. Munich: Fachbuchverlag Leipzig im Carl Hanser Verlag, 2012, S. 481–503 (siehe S. 14, 49).

[FP12b] L. Fuchs und G. Pernul. *Minimizing insider misuse through secure Identity Management*. In: *Security and Communication Networks* 5.8 (2012), S. 847–862 (siehe S. 38, 85, 255, 262).

[FP97] H. Federrath und A. Pfitzmann. *Bausteine zur Realisierung mehrseitiger Sicherheit*. In: *Mehrseitige Sicherheit in der Kommunikationstechnik, Verfahren, Komponenten*. Hrsg. von G. Müller und A. Pfitzmann. Addison-Wesley-Longman, 1997, S. 83–104 (siehe S. 11, 13, 61).

[Fun11] C. J. Fung. *Collaborative Intrusion Detection Networks and Insider Attacks*. In: *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications* 2.1 (2011), S. 63–74 (siehe S. 170).

[Gau12] P. Gauravaram. *Security Analysis of salt||password Hashes*. In: *Proceedings of the International Conference on Advanced Computer Science Applications and Technologies*. ACSAT '12. Nov. 2012, S. 25–30 (siehe S. 70, 76).

[Gen+99] R. Gennaro u. a. *Secure Distributed Key Generation for Discrete-Log Based Cryptosystems*. In: *Advances in Cryptology — EUROCRYPT '99*. Hrsg. von J. Stern. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, S. 295–310 (siehe S. 214).

[GF10] F. L. Greitzer und D. A. Frincke. *Combining traditional cyber security audit data with psychosocial data: towards predictive modeling for insider threat mitigation*. In: *Insider Threats in Cyber Security*. Hrsg. von C. W. Probst u. a. Bd. 49. Advances in Information Security. Springer US, 2010, S. 85–113 (siehe S. 38, 252).

[GGF17] P. A. Grassi, M. Garcia und J. Fenton. *NIST Special Publication 800-63-3 - Digital Identity Guidelines*. Rev. 3. United States Department of Commerce: National Insititue of Standards und Technology, Juni 2017. URL: https://doi.org/10.6028/NIST.SP.800-63-3 (besucht am 24. 07. 2020) (siehe S. 54).

[GK07] J. A. Grochow und M. Kellis. *Network Motif Discovery Using Subgraph Enumeration and Symmetry-Breaking*. In: *Research in Computational Molecular Biology*. Hrsg. von T. Speed und H. Huang. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, S. 92–106 (siehe S. 104).

[Goh03] E.-J. Goh. *Secure Indexes*. Cryptology ePrint Archive, Report 2003/216. Last revised 16 Mar 2004. 2003. URL: https://eprint.iacr.org/2003/216 (besucht am 24. 07. 2020) (siehe S. 183, 198, 199).

[Gol12] D. Gollmann. *Veracity, Plausibility, and Reputation*. In: *Information Security Theory and Practice. Security, Privacy and Trust in Computing Systems and Ambient Intelligent Ecosystems*. Hrsg. von I. Askoxylakis, H. Pöhls und J. Posegga. Bd. 7322. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, S. 20–28 (siehe S. 61, 71, 87, 185).

[Gra+17] P. A. Grassi u. a. *NIST Special Publication 800-63B - Digital Identity Guidelines. Authentication and Lifecycle Management*. United States Department of Commerce: National Insititue of Standards und Technology, Juni 2017. URL: https://doi.org/10.6028/NIST.SP.800-63b (besucht am 24. 07. 2020) (siehe S. 67, 76).

[Gre+08] F. L. Greitzer u. a. *Combating the Insider Cyber Threat*. In: *IEEE Security Privacy* 6.1 (Jan. 2008), S. 61–64 (siehe S. 38, 84, 245).

[Gre+14] F. L. Greitzer u. a. *Analysis of Unintentional Insider Threats Deriving from Social Engineering Exploits*. In: *2014 IEEE Security and Privacy Workshops*. San Jose, CA, USA, Mai 2014, S. 236–250 (siehe S. 53).

[Gru09] T. Gruber. *Ontology*. In: *Encyclopedia of Database Systems*. Hrsg. von L. Liu und M. T. Özsu. Boston, MA: Springer US, 2009, S. 1963–1965 (siehe S. 15).

[Gru11] S. Grubb. *Native Host Intrusion Detection with RHEL6 and the Audit Subsystem*. Presentation at the 7th Red Hat Summit. Boston, US, März 2011. URL: https://people.redhat.com/sgrubb/audit/audit_ids_2011.pdf (besucht am 24. 07. 2020) (siehe S. 109).

[GSB17] A. Gamachchi, L. Sun und S. Boztas. *Graph Based Framework for Malicious Insider Threat Detection*. In: *50th Hawaii International Conference on System Sciences, HICSS 2017, Hilton Waikoloa Village, Hawaii, USA, January 4-7, 2017*. Hrsg. von T. Bui. ScholarSpace / AIS Electronic Library (AISeL), 2017, S. 1–10 (siehe S. 90, 91, 106, 222).

[GTD11] S. Gürses, C. Troncoso und C. Diaz. *Engineering Privacy by Design*. In: *Conference on Computers, Privacy & Data protection, CPDP*. Brussels, Belgium, 2011 (siehe S. 167).

[GZ09] F. Giunchiglia und I. Zaihrayeu. *Lightweight Ontologies*. In: *Encyclopedia of Database Systems*. Hrsg. von L. Liu und M. T. Özsu. Boston, MA: Springer US, 2009, S. 1613–1619 (siehe S. 15).

[Hau15] B. Hauer. *Data and Information Leakage Prevention Within the Scope of Information Security*. In: *IEEE Access* 3 (2015), S. 2554–2565 (siehe S. 128).

[HCA11] E. M. Hutchins, M. J. Cloppert und R. M. Amin. *Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains*. In: *Leading Issues in Information Warfare & Security Research* 1.1 (2011). Hrsg. von J. Ryan, S. 80–106. Academic Conferences Limited (siehe S. 68, 73, 81, 87).

[Hen+10] K. Henderson u. a. *Metric Forensics: A Multi-Level Approach for Mining Volatile Graphs*. In: *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '10. Washington, DC, USA: Association for Computing Machinery, 2010, S. 163–172 (siehe S. 105, 222).

[Her17] A. Hern. *'Anonymous' browsing data can be easily exposed, researchers reveal*. Online News. The Guardian, 1. Aug. 2017. URL: https://www.theguardian.com/technology/2017/aug/01/data-browsing-habits-brokers (besucht am 24. 07. 2020) (siehe S. 169).

[HFS98] S. A. Hofmeyr, S. Forrest und A. Somayaji. *Intrusion detection using sequences of system calls*. In: *Journal of computer security* 6.3 (1998), S. 151–180 (siehe S. 106, 222).

[HLS16] M. Holt, B. Lang und S. G. Sutton. *Potential employees' ethical perceptions of active monitoring: The dark side of data analytics*. In: *Journal of Information Systems* 31.2 (2016), S. 107–124. American Accounting Association (siehe S. 166).

[Hoe14] J.-H. Hoepman. *Privacy Design Strategies*. In: *ICT Systems Security and Privacy Protection*. Hrsg. von N. Cuppens-Boulahia u. a. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, S. 446–459 (siehe S. 167).

[Hol17] M. Holland. *Vom Drucker verraten: NSA-Dokument enttarnt Whistleblowerin*. Heise Online News, 6. Juni 2017. URL: https://www.heise.de/newsticker/meldung/Vom-Drucker-verraten-NSA-Dokument-enttarnt-Whistleblowerin-3734692.html (besucht am 24. 07. 2020) (siehe S. 88).

[HP11] J. Hunker und C. W. Probst. *Insiders and insider threats – An Overview of Definitions and Mitigation Techniques*. In: *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications* 2.1 (2011), S. 4–27 (siehe S. 8, 13, 19–23, 28, 38, 53, 219, 255).

[HPK11] J. Han, J. Pei und M. Kamber. *Data Mining: Concepts and Techniques*. 3rd. The Morgan Kaufmann Series in Data Management Systems. San Francisco, CA, USA: Elsevier Science, 2011 (siehe S. 130, 132).

[Hub16] R. Huber. *Syscall Auditing at Scale*. Blog Post. Code available at https://github.com/slackhq/go-audit. Slack Engineering, 21. Nov. 2016. URL: https://slack.engineering/syscall-auditing-at-scale-e6a3ca8ac1b8 (besucht am 24. 07. 2020) (siehe S. 113, 142).

[Hun17] T. Hunt. *Passwords Evolved: Authentication Guidance for the Modern Era*. Blog Post. 26. Juli 2017. URL: https://www.troyhunt.com/passwords-evolved-authentication-guidance-for-the-modern-era/ (besucht am 24. 07. 2020) (siehe S. 67).

[HWF19] S. Haas, F. Wilkens und M. Fischer. *Efficient Attack Correlation and Identification of Attack Scenarios based on Network-Motifs*. In: *arXiv preprint arXiv:1905.06685* (2019) (siehe S. 106).

[ISO13] International Organization for Standardization. *ISO/IEC 27002 – Information technology – Security techniques – Code of practice for information security controls*. Okt. 2013. URL: https://www.iso.org/standard/54533.html (besucht am 24. 07. 2020) (siehe S. 84, 85).

[IW08] V. M. Igure und R. D. Williams. *Taxonomies of attacks and vulnerabilities in computer systems*. In: *IEEE Communications Surveys & Tutorials* 10.1 (Jan. 2008), S. 6–19 (siehe S. 15).

[Jac05] B. Jacobs. *Select before you Collect*. In: *Ars Aequi* 54 (2005). English slides available at www.cs.ru.nl/B.Jacobs/TALKS/govcert05.pdf, S. 1006–1009 (siehe S. 167).

[Jan06] D. Jansen. *Einführung in die Netzwerkanalyse: Grundlagen, Methoden, Forschungsbeispiele*. Springer-Verlag, 2006 (siehe S. 102).

[Jha+04] S. Jha u. a. *A Filtering Approach to Anomaly and Masquerade Detection*. Techn. Ber. Citeseer, 2004. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.92.2211 (besucht am 24. 07. 2020) (siehe S. 38, 238).

[JLN04] H. Jin, J. Lotspiech und S. Nusser. *Traitor Tracing for Prerecorded and Recordable Media*. In: *Proceedings of the 4th ACM Workshop on Digital Rights Management*. DRM '04. Washington DC, USA: ACM, 2004, S. 83–90 (siehe S. 88).

[Jun+14] H. Juneja u. a. *Linux Audit heka plugin (Go)*. Code available at https://github.com/mozilla/audit-go. Mozilla Winter of Security, Aug. 2014. URL: https://wiki.mozilla.org/Security/Mentorships/MWoS/2014/Linux_Audit_heka_plugin_(Go) (besucht am 24. 07. 2020) (siehe S. 113, 142).

[Kam09] G. Kammerer-Galahn. *Compliance – Herausforderung für Unternehmensleiter und deren Rechtsberater. Pflichten, Haftungsrisiken und Versicherungsschutz*. In: *Anwaltsblatt Jahrgang 59* 2 (2009), S. 77–83 (siehe S. 158, 159).

[Kan+10] M. Kandias u. a. *An Insider Threat Prediction Model*. In: *Trust, Privacy and Security in Digital Business*. Hrsg. von S. Katsikas, J. Lopez und M. Soriano. Bd. 6264. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2010, S. 26–37 (siehe S. 38, 90, 91, 253).

[Kan14] C. Kanaracus. *IT pro gets 4 years in prison for sabotaging ex-employer's system*. Computerworld Online News, 21. Mai 2014. URL: https://www.computerworld.com/article/2489761/it-pro-gets-4-years-in-prison-for-sabotaging-ex-employer-s-system.html (besucht am 24. 07. 2020) (siehe S. 226).

[Kap+15] J. M. Kaplan u. a. *Beyond Cybersecurity: Protecting Your Digital Business*. John Wiley & Sons, 2015 (siehe S. 19, 28, 38, 257).

[Kas+04] N. Kashtan u. a. *Efficient sampling algorithm for estimating subgraph concentrations and detecting network motifs*. In: *Bioinformatics* 20.11 (März 2004), S. 1746–1758 (siehe S. 103).

[Kas+09] Z. R. M. Kashani u. a. *Kavosh: a new algorithm for finding network motifs*. In: *BMC Bioinformatics* 10.1 (Okt. 2009), S. 318–330 (siehe S. 103, 104).

[Kas18] Kaspersky Lab. *Anzahl der von Verschlüsselungsschädlingen betroffenen Kaspersky-Anwender weltweit in den Jahren 2015/2016 bis 2017/2018*. 27. Juni 2018. URL: https://de.statista.com/statistik/daten/studie/720941/umfrage/anzahl-der-von-verschluesselungsschaedlingen-betroffenen-anwender-weltweit/ (besucht am 24. 07. 2020) (siehe S. 136).

[KBC97] H. Krawczyk, M. Bellare und R. Canetti. *RFC2104 - HMAC: Keyed-Hashing for Message Authentication*. Feb. 1997. URL: https://tools.ietf.org/html/rfc2104 (besucht am 24. 07. 2020) (siehe S. 177).

[Kha17] S. Khandelwal. *Source Code for CIA's Tool to Track Whistleblowers Leaked by Wikileaks*. The Hacker News, 28. Apr. 2017. URL: https://thehackernews.com/2017/04/wikiLeaks-scribbles-cia-whistleblower.html (besucht am 24. 07. 2020) (siehe S. 38, 260).

[Kis13] R. Kissel. *NISTIR 7298 - Glossary of Key Information Security Terms*. Rev. 2. United States Department of Commerce: National Insititue of Standards und Technology, Mai 2013. URL: https://doi.org/10.6028/NIST.IR.7298r2 (besucht am 24. 07. 2020) (siehe S. 10, 38, 256, 264).

[KL14] J. Katz und Y. Lindell. *Introduction to Modern Cryptography*. 2nd. Chapman & Hall/-CRC Press, 2014 (siehe S. 14, 171, 177–179, 183).

[Kop+08] R. Koppel u. a. *Workarounds to Barcode Medication Administration Systems: Their Occurrences, Causes, and Threats to Patient Safety*. In: *Journal of the American Medical Informatics Association* 15.4 (Juli 2008), S. 408–423 (siehe S. 34).

[KRA15] O. Koucham, T. Rachidi und N. Assem. *Host intrusion detection using system call argument-based clustering combined with Bayesian classification*. In: *2015 SAI Intelligent Systems Conference (IntelliSys)*. 2015, S. 1010–1016 (siehe S. 108).

[Kre19] S. Krempl. *Datenethik-Kommission: Verbot von De-Anonymisierung und Profilbildung*. Heise Online News, 23. Okt. 2019. URL: https://www.heise.de/newsticker/meldung/Datenethik-Kommission-Verbot-von-De-Anonymisierung-und-Profilbildung-4566788.html (besucht am 24. 07. 2020) (siehe S. 2).

[Kro16] N. Krohm. *Abschied vom Schriftformgebot der Einwilligung-Lösungsvorschläge und künftige Anforderungen*. In: *ZD (Zeitschrift für Datenschutz)* (2016), S. 368–373 (siehe S. 157).

[KS19] D.-K. Kipker und D. Scholz. *Das IT-Sicherheitsgesetz 2.0. Neue Rahmenbedingungen für die Cybersicherheit in Deutschland*. In: *MultiMedia und Recht, MMR* (Juli 2019), S. 431–435 (siehe S. 159).

[KST12] J. Köbler, U. Schöning und J. Torán. *The Graph Isomorphism Problem. Its Structural Complexity*. Progress in Theoretical Computer Science. Birkhäuser Boston, 2012 (siehe S. 104).

[KTB08] A. Kamra, E. Terzi und E. Bertino. *Detecting Anomalous Access Patterns in Relational Databases*. In: *The VLDB Journal* 17.5 (Aug. 2008), S. 1063–1077 (siehe S. 38, 246).

[KVG13] M. Kandias, N. Virvilis und D. Gritzalis. *The Insider Threat in Cloud Computing*. In: *Critical Information Infrastructure Security*. Hrsg. von S. Bologna u. a. Berlin, Heidelberg: Springer, 2013, S. 93–103 (siehe S. 85).

[Lag95] J. L. Lagrange. *Leçons élémentaires sur les mathématiques*. In: *Ecole Normale* (1795) (siehe S. 180).

[LCW92] K. D. Loch, H. H. Carr und M. E. Warkentin. *Threats to Information Systems: Today's Reality, Yesterday's Understanding*. In: *MIS Quarterly* 16.2 (1992), S. 173–186. Management Information Systems Research Center, University of Minnesota (siehe S. 38, 232).

[LDM10] G. Loukides, J. C. Denny und B. Malin. *The disclosure of diagnosis codes can breach research participants' privacy*. In: *Journal of the American Medical Informatics Association* 17.3 (Mai 2010), S. 322–327 (siehe S. 169).

[LF19] K. Leswing und J. Forkin. *Former Apple lawyer in charge of preventing insider trading is indicted on insider trading charges*. CNBC Online News, 24. Okt. 2019. URL: https://www.cnbc.com/2019/10/24/apple-lawyer-indicted-for-insider-trading.html (besucht am 24. 07. 2020) (siehe S. 2, 17).

[Liu+05] A. Liu u. a. *A comparison of system call feature representations for insider threat detection*. In: *Proceedings of the 6th Annual IEEE SMC Information Assurance Workshop*. Juni 2005, S. 340–347 (siehe S. 38, 243).

[Liu+09] Y. Liu u. a. *SIDD: A Framework for Detecting Sensitive Data Exfiltration by an Insider Attack*. In: *42nd Hawaii International Conference on System Sciences*. Jan. 2009, S. 1–10 (siehe S. 90, 91).

[LJ00] E. Lundin und E. Jonsson. *Anomaly-based intrusion detection: privacy concerns and other problems*. In: *Computer Networks* 34.4 (2000), S. 623–640 (siehe S. 28, 36, 38, 233).

[LM14] M. Levine und R. Manning. *Prosoziales Verhalten*. In: *Sozialpsychologie*. Hrsg. von K. Jonas, W. Stroebe und M. Hewstone. 6. Auflage. Springer Berlin Heidelberg, 2014, S. 357–400 (siehe S. 72).

[Loc+05] M. E. Locasto u. a. *Towards collaborative security and P2P intrusion detection*. In: *Proceedings from the Sixth Annual IEEE SMC Information Assurance Workshop*. West Point, NY, USA: IEEE, Juni 2005, S. 333–339 (siehe S. 171, 223).

[LP16] A. Lavrenovs und K. Podins. *Privacy violations in Riga open data public transport system*. In: *2016 IEEE 4th Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE)*. Nov. 2016, S. 1–6 (siehe S. 169).

[LPS04] P. Lincoln, P. A. Porras und V. Shmatikov. *Privacy-Preserving Sharing and Correlation of Security Alerts.* In: *Proceedings of the 13th USENIX Security Symposium*. Hrsg. von M. Blaze. San Diego, CA: USENIX Association, 2004, S. 239–254 (siehe S. 171, 223).

[LTZ08] F. T. Liu, K. M. Ting und Z.-H. Zhou. *Isolation forest*. In: *2008 Eighth IEEE International Conference on Data Mining*. IEEE. Dez. 2008, S. 413–422 (siehe S. 106).

[LV02] Y. Liao und V. R. Vemuri. *Using text categorization techniques for intrusion detection*. In: *USENIX Security Symposium*. Bd. 12. 2002, S. 51–59 (siehe S. 107, 222).

[Mah95] A. Mahler. *Lücke im Gesetz*. Hrsg. von R. Augstein. DER SPIEGEL 34/1995. 21. Aug. 1995. URL: https://www.spiegel.de/spiegel/print/d-9207824.html (besucht am 24. 07. 2020) (siehe S. 224).

[Mal16] S. T. Malone. *Using an Expanded Cyber Kill Chain Model to Increase Attack Resiliency*. In: *Black Hat Conference* (2016) (siehe S. 74).

[Mar+18] M. Marx u. a. *Hashing of personally identifiable information is not sufficient*. In: *Sicherheit 2018, Beiträge der 9. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI), 25.-27.4.2018, Konstanz.* Hrsg. von H. Langweg u. a. Bd. P-281. LNI. Gesellschaft für Informatik e.V., Apr. 2018, S. 55–68 (siehe S. 177, 192).

[Mat+10] S. Mathew u. a. *A Data-Centric Approach to Insider Attack Detection in Database Systems*. In: *Recent Advances in Intrusion Detection: 13th International Symposium, RAID 2010, Ottawa, Ontario, Canada, September 15-17, 2010. Proceedings*. Hrsg. von S. Jha, R. Sommer und C. Kreibich. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, S. 382–401 (siehe S. 29, 36, 38, 90, 91, 253).

[May+05] M. Maybury u. a. *Analysis and detection of malicious insiders*. Techn. Ber. DTIC Document, 2005. URL: https://www.researchgate.net/publication/228617716_Analysis_and_detection_of_malicious_insiders (besucht am 24. 07. 2020) (siehe S. 38, 88, 221, 244).

[MC17] F. Murtagh und P. Contreras. *Algorithms for hierarchical clustering: an overview, II*. In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 7.6 (2017), e1219 (siehe S. 139).

[McG+15] A. S. McGough u. a. *Detecting Insider Threats Using Ben-ware: Beneficial Intelligent Software for Identifying Anomalous Human Behaviour*. In: *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications* 6.4 (Dez. 2015), S. 3–46 (siehe S. 38, 90, 91, 258).

[McK81] B. D. McKay. *Practical Graph Isomorphism*. Department of Computer Science, Vanderbilt University Tennessee, USA, 1981 (siehe S. 104).

[MF01] G. Magklaras und S. Furnell. *Insider Threat Prediction Tool: Evaluating the probability of IT misuse*. In: *Computers & Security* 21.1 (2001), S. 62–73 (siehe S. 19, 24, 220, 221).

[MF14] P. Mayring und T. Fenzl. *Qualitative Inhaltsanalyse*. In: *Handbuch Methoden der empirischen Sozialforschung*. Hrsg. von N. Baur und J. Blasius. Wiesbaden: Springer Fachmedien Wiesbaden, 2014, S. 543–556 (siehe S. 34, 35, 261).

[MFB06] G. Magklaras, S. Furnell und P. Brooke. *Towards an Insider Threat Prediction Specification Language*. In: *Information Management & Computer Security* 14.4 (2006), S. 361–381 (siehe S. 38, 244).

[Mil+02] R. Milo u. a. *Network Motifs: Simple Building Blocks of Complex Networks*. In: *Science* 298.5594 (2002), S. 824–827 (siehe S. 102, 103).

[Mog07] R. Mogull. *Understanding and Selecting a Data Loss Prevention Solution*. Techn. Ber. SANS Institute, 2007. URL: https://securosis.com/assets/library/reports/DLP-Whitepaper.pdf (besucht am 24. 07. 2020) (siehe S. 90).

[Mon+13] Y.-A. de Montjoye u. a. *Unique in the crowd: The privacy bounds of human mobility*. In: *Scientific reports* 3 (2013), S. 1376. Nature Publishing Group (siehe S. 169).

[Mon+15] Y.-A. de Montjoye u. a. *Unique in the shopping mall: On the reidentifiability of credit card metadata*. In: *Science* 347.6221 (2015), S. 536–539. American Association for the Advancement of Science (siehe S. 169).

[MP97] G. Müller und A. Pfitzmann, Hrsg. *Mehrseitige Sicherheit in der Kommunikationstechnik, Verfahren, Komponenten*. Addison-Wesley-Longman, 1997.

[MPH13] D. A. Mundie, S. Perl und C. L. Huth. *Toward an Ontology for Insider Threat Research: Varieties of Insider Threat Definitions*. In: *2013 Third Workshop on Socio-Technical Aspects in Security and Trust*. Juni 2013, S. 26–36 (siehe S. 20).

[MRA17] T. Mouttaqi, T. Rachidi und N. Assem. *Re-evaluation of combined Markov-Bayes models for host intrusion detection on the ADFA dataset*. In: *2017 Intelligent Systems Conference (IntelliSys)*. 2017, S. 1044–1052 (siehe S. 107, 222).

[MS07] M. A. Maloof und G. D. Stephens. *ELICIT: A System for Detecting Insiders Who Violate Need-to-Know*. In: *Recent Advances in Intrusion Detection*. Hrsg. von C. Kruegel, R. Lippmann und A. Clark. Bd. 4637. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2007, S. 146–166 (siehe S. 86, 94).

[MSA03] S. Maslov, K. Sneppen und U. Alon. *Correlation profiles and motifs in complex networks*. In: *Handbook of graphs and networks: from the genome to the internet*. Hrsg. von S. Bornholdt und H. G. Schuster. John Wiley & Sons, 2003, S. 168–198 (siehe S. 102, 103).

[MT04] R. Maxion und T. Townsend. *Masquerade detection augmented with error analysis*. In: *Transactions on Reliability* 53.1 (März 2004), S. 124–147 (siehe S. 24, 38, 221, 239).

[MU05] M. Mitzenmacher und E. Upfal. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, 2005 (siehe S. 183).

[MW02] R. Metschke und R. Wellbrock. *Datenschutz in Wissenschaft und Forschung*. 3. Aufl. Berliner Beauftragter für Datenschutz und Akteneinsicht, Dez. 2002 (siehe S. 152).

[MWB15] M. Maasberg, J. Warren und N. Beebe. *The Dark Side of the Insider: Detecting the Insider Threat through Examination of Dark Triad Personality Traits*. In: *48th Hawaii International Conference on System Sciences*. HICSS '15. Jan. 2015, S. 3518–3526 (siehe S. 13, 38, 40, 259).

[NC03] C. C. Noble und D. J. Cook. *Graph-Based Anomaly Detection*. In: *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '03. Washington, D.C.: Association for Computing Machinery, 2003, S. 631–636 (siehe S. 105, 222).

[Neu99] P. Neumann. *The Challenges of Insider Misuse*. In: *Workshop on Preventing, Detecting, and Responding to Malicious Insider Misuse* (1999). SRI Computer Science Laboratory. RAND Corporation (siehe S. 18, 19, 21, 22, 38, 220, 233).

[NF14] A. Narayanan und E. W. Felten. *No silver bullet: De-identification still doesn't work*. White Paper. 2014. URL: http://www.randomwalker.info/publications/no-silver-bullet-de-identification.pdf (besucht am 24. 07. 2020) (siehe S. 169).

[Now11] T. Nowey. *Konzeption eines Systems zur überbetrieblichen Sammlung und Nutzung von quantitativen Daten über Informationssicherheitsvorfälle*. Wiesbaden: Vieweg+Teubner Verlag|Springer Fachmedien, 2011 (siehe S. 10).

[NP01] M. Naor und B. Pinkas. *Efficient Oblivious Transfer Protocols*. In: *Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms*. SODA '01. Washington, D.C., USA: Society for Industrial und Applied Mathematics, 2001, S. 448–457 (siehe S. 183).

[NRK03] N. Nguyen, P. Reiher und G. H. Kuenning. *Detecting insider threats by monitoring system call activity*. In: *IEEE Systems, Man and Cybernetics Society Information Assurance Workshop*. United States Military Academy, West Point, Juni 2003, S. 45–52 (siehe S. 38, 107, 235, 263).

[NS05] P. Ning und K. Sun. *How to Misuse AODV: A Case Study of Insider Attacks against Mobile Ad-hoc Routing Protocols*. In: *Ad Hoc Networks* 3.6 (2005), S. 795–819 (siehe S. 36, 38, 244).

[Ohm09] P. Ohm. *Broken promises of privacy: Responding to the surprising failure of anonymization*. In: *UCLA Law Review* 57.6 (2009), S. 1701–1778. HeinOnline (siehe S. 167, 169, 190).

[Pat03] J. Patzakis. *New incident response best practices: Patch and proceed is no longer acceptable incident response*. In: *Guidance Software, Pasadena, CA, Tech. Rep* (2003) (siehe S. 18, 24, 38, 219, 221, 236).

[Ped91] T. P. Pedersen. *A Threshold Cryptosystem without a Trusted Party*. In: *Advances in Cryptology — EUROCRYPT '91*. Hrsg. von D. W. Davies. Berlin, Heidelberg: Springer Berlin Heidelberg, 1991, S. 522–526 (siehe S. 214).

[Pet18] T. Petersen. *Datenschutzfreundliche Speicherung unternehmensinterner Überwachungsdaten mittels Pseudonymisierung und kryptographischer Schwellwertschemata*. Masterarbeit. Universität Hamburg, Fachbereich für Informatik, Arbeitsbereich Sicherheit in verteilten Systemen, März 2018 (siehe S. 207).

[Pfi06] A. Pfitzmann. *Multilateral Security: Enabling Technologies and Their Evaluation*. In: *Emerging Trends in Information and Communication Security*. Hrsg. von G. Müller. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, S. 1–13 (siehe S. 11, 78, 166).

[Pfl+10] S. Pfleeger u. a. *Insiders Behaving Badly: Addressing Bad Actors and Their Actions*. In: *IEEE Transactions on Information Forensics and Security* 5.1 (März 2010), S. 169–179 (siehe S. 20, 21, 38, 53, 220, 254).

[Pfl08] C. P. Pfleeger. *Reflections on the Insider Threat*. In: *Insider Attack and Cyber Security. Beyond the Hacker*. Hrsg. von S. J. Stolfo u. a. Advances in Information Security. Boston, MA: Springer US, 2008, S. 5–16 (siehe S. 18, 21, 36–38, 170, 246, 262).

[PG11] P. Papadimitriou und H. Garcia-Molina. *Data Leakage Detection*. In: *IEEE Transactions on Knowledge and Data Engineering* 23.1 (Jan. 2011), S. 51–63 (siehe S. 88).

[PH10] A. Pfitzmann und M. Hansen. *A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management*. 2010. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.204.9762 (besucht am 24. 07. 2020) (siehe S. 165, 168).

[Pie18] B. Pierson. *Ex-IBM employee from China gets five years prison for stealing code*. Reuters Online News, 19. Jan. 2018. URL: https://uk.reuters.com/article/us-ibm-crime-china/ex-ibm-employee-from-china-gets-five-years-prison-for-stealing-code-idUKKBN1F82P9 (besucht am 24. 07. 2020) (siehe S. 226).

[Pon+19] Ponemon Institute LLC u. a. *The Cost of Cybercrime. Ninth Annual Cost of Cybercrime Study*. Accenture, 2019. URL: https://www.accenture.com/_acnmedia/pdf-96/accenture-2019-cost-of-cybercrime-study-final.pdf (besucht am 24. 07. 2020) (siehe S. 43, 226).

[Pro+10a] C. W. Probst u. a. *Aspects of Insider Threats*. In: *Insider Threats in Cyber Security*. Hrsg. von C. W. Probst u. a. Bd. 49. Advances in Information Security. Springer US, 2010, S. 1–15 (siehe S. 19, 22, 38, 43, 251).

[Pro+10b] C. W. Probst u. a., Hrsg. *Insider Threats in Cyber Security*. Bd. 49. Advances in Information Security. Springer US, 2010.

[PS09] S. L. Pfleeger und S. J. Stolfo. *Addressing the Insider Threat*. In: *IEEE Security Privacy* 7.6 (Nov. 2009), S. 10–13 (siehe S. 37, 38, 250).

[PTF16] J. Polonetsky, O. Tene und K. Finch. *Shades of Gray: Seeing the Full Spectrum of Practical Data De-Intentification*. In: *Santa Clara Law Review* 56.3 (2016), S. 593–630. HeinOnline (siehe S. 167).

[Put18] P. Putman. *Script Kiddie: Unskilled Amateur or Dangerous Hackers?* United States Cybersecurity Magazine, 14. Sep. 2018. URL: https://www.uscybersecurity.net/script-kiddie/ (besucht am 24. 07. 2020) (siehe S. 262).

[PV98] P. Poerting und W. Vahlenkamp. *Internal strategies against corruption: Guidelines for preventing and combating corruption in police authorities*. In: *Crime, Law and Social Change* 29.2 (1. März 1998), S. 225–249. Kluwer Academic Publishers (siehe S. 70).

[Red10] Red Hat Customer Content Services. *Security Guide. A Guide to Securing Red Hat Enterprise Linux 6*. In: *Product Documentation for Red Hat Enterprise Linux 6*. Red Hat, Nov. 2010 (siehe S. 109).

[Red19] Red Hat Customer Content Services. *Security Hardening. Securing Red Hat Enterprise Linux 8*. In: *Product Documentation for Red Hat Enterprise Linux 8*. Red Hat, Mai 2019 (siehe S. 108, 109).

[Ree15] T. Reed. *Process and socket auditing with osquery*. Code available at https://github.com/osquery/osquery. osquery Foundation, 16. Aug. 2015. URL: https://osquery.readthedocs.io/en/stable/deployment/process-auditing/ (besucht am 24. 07. 2020) (siehe S. 113).

[RHM19] L. Rocher, J. M. Hendrickx und Y.-A. de Montjoye. *Estimating the success of re-identifications in incomplete datasets using generative models*. In: *Nature Communications* 10.1 (23. Juli 2019), S. 3069. Nature Publishing Group (siehe S. 169).

[Rie19] K. Riesenhuber. *Beck'scher Online-Kommentar Datenschutzrecht*. Hrsg. von H. A. Wolff und S. Brink. 29. Edition. München: C.H. Beck, 2019 (siehe S. 162).

[Rot10] M. A. Rothstein. *Is Deidentification Sufficient to Protect Health Privacy in Research?* In: *The American Journal of Bioethics* 10.9 (2010), S. 3–11. Taylor & Francis (siehe S. 169).

[Row04] R. Rowlingson. *A ten step process for forensic readiness*. In: *International Journal of Digital Evidence* 2.3 (2004), S. 1–28 (siehe S. 72, 165).

[RPM97] K. Rannenberg, A. Pfitzmann und G. Müller. *Sicherheit, insbesondere mehrseitige IT-Sicherheit*. In: *Mehrseitige Sicherheit in der Kommunikationstechnik, Verfahren, Komponenten*. Hrsg. von G. Müller und A. Pfitzmann. Addison-Wesley-Longman, 1997, S. 21–29 (siehe S. 13, 43, 78).

[RS10] P. Ribeiro und F. Silva. *G-Tries: An Efficient Data Structure for Discovering Network Motifs*. In: *Proceedings of the 2010 ACM Symposium on Applied Computing*. SAC '10. Sierre, Switzerland: ACM, 2010, S. 1559–1566 (siehe S. 104, 123).

[SCD17] A. Shulman, M. Cherny und S. Dulce. *Compromised insider honey pots using reverse honey tokens*. US Patent 9,667,651. Imperva Inc. 30. Mai 2017 (siehe S. 38, 89, 259).

[Sch+01] M. Schonlau u. a. *Computer Intrusion: Detecting Masquerades*. In: *Statistical Science* 16.1 (2001), S. 58–74. Institute of Mathematical Statistics (siehe S. 22).

[Sch02] E. E. Schultz. *A framework for understanding and predicting insider attacks*. In: *Computers & Security* 21.6 (2002), S. 526–531 (siehe S. 38, 235).

[Sch16] B. Schneier. *Frequent Password Changes Is a Bad Security Idea*. Blog Post. Schneier on Security, Aug. 2016. URL: https://www.schneier.com/blog/archives/2016/08/frequent_passwo.html (besucht am 24. 07. 2020) (siehe S. 67).

[Scr20] A. Scroxton. *Cryptocurrency scammers attack Twitter in insider breach*. ComputerWeekly Online News, 16. Juli 2020. URL: https://www.computerweekly.com/news/252486219/Cryptocurrency-scammers-attack-Twitter-in-insider-breach (besucht am 24. 07. 2020) (siehe S. 226).

[SER12] A. Shabtai, Y. Elovici und L. Rokach. *A Survey of Data Leakage Detection and Prevention Solutions*. Springer Publishing Company, Incorporated, 2012 (siehe S. 55, 90, 167).

[SFR97] M. Sobirey, S. Fischer-Hübner und K. Rannenberg. *Pseudonymous Audit for Privacy Enhanced Intrusion Detection*. In: *Information Security in Research and Business*. IFIP — The International Federation for Information Processing. Boston, MA: Springer, 1997, S. 151–163 (siehe S. 171, 223).

[SGG18] A. Silberschatz, G. Gagne und P. B. Galvin. *Operating System Concepts*. Wiley, 2018 (siehe S. 98).

[Sha79] A. Shamir. *How to Share a Secret*. In: *Commun. ACM* 22.11 (Nov. 1979), S. 612–613 (siehe S. 172, 180, 181, 193).

[She+02] S. S. Shen-Orr u. a. *Network motifs in the transcriptional regulation network of Escherichia coli*. In: *Nature Genetics* 31.1 (Mai 2002), S. 64–68 (siehe S. 103).

[Shi07] R. W. Shirey. *RFC4949 – Internet Security Glossary (Version 2)*. Network Working Group, Aug. 2007. URL: https://tools.ietf.org/html/rfc4949 (besucht am 24. 07. 2020) (siehe S. 10, 19, 22, 38, 245).

[SHS08] M. B. Salem, S. Hershkop und S. J. Stolfo. *A Survey of Insider Attack Detection Research*. In: *Insider Attack and Cyber Security. Beyond the Hacker*. Hrsg. von S. J. Stolfo u. a. Advances in Information Security. Boston, MA: Springer US, 2008, S. 69–90 (siehe S. 170).

[Shu+16] X. Shu u. a. *Fast Detection of Transformed Data Leaks*. In: *IEEE Transactions on Information Forensics and Security* 11.3 (März 2016), S. 528–542 (siehe S. 90).

[Sid14] J. Siddle. *I Know Where You Were Last Summer: London's public bike data is telling everyone where you've been*. Blog Post. 10. Apr. 2014. URL: https://vartree.blogspot.com/2014/04/i-know-where-you-were-last-summer.html (besucht am 24. 07. 2020) (siehe S. 169).

[Sil+12] G. Silowash u. a. *Common Sense Guide to Mitigating Insider Threats*. Techn. Ber. CMU/SEI-2012-TR-012. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2012. URL: http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=34017 (besucht am 24. 07. 2020) (siehe S. 43, 84, 85, 91).

[SM11] F. M. Sibai und D. A. Menascé. *Defeating the insider threat via autonomic network capabilities*. In: *Third International Conference on Communication Systems and Networks (COMSNETS 2011)*. Jan. 2011, S. 1–10 (siehe S. 91).

[SM15] S. Shah und B. M. Mehtre. *An overview of vulnerability assessment and penetration testing techniques*. In: *Journal of Computer Virology and Hacking Techniques* 11.1 (1. Feb. 2015), S. 27–49 (siehe S. 87).

[Sol17] O. Solon. *Big Brother isn't just watching: workplace surveillance can track your every move*. Online News. The Guardian, 6. Nov. 2017. URL: https://www.theguardian.com/world/2017/nov/06/workplace-surveillance-big-brother-technology (besucht am 24. 07. 2020) (siehe S. 2, 17, 166, 186).

[Spi03a] L. Spitzner. *Honeypots: Catching the insider threat*. In: *Proceedings of the 19th Computer Security Applications Conference*. IEEE Computer Society, Dez. 2003, S. 170–179 (siehe S. 38, 40, 89, 236, 264).

[Spi03b] L. Spitzner. *Honeytokens: The Other Honeypot*. SecurityFocus, 17. Juli 2003. URL: https://community.broadcom.com/symantecenterprise/communities/community-home/librarydocuments/viewdocument?DocumentKey=74450cf5-2f11-48c5-8d92-4687f5978988 (besucht am 24. 07. 2020) (siehe S. 89, 225).

[SS05] F. Schreiber und H. Schwöbbermeyer. *Frequency Concepts and Pattern Detection for the Analysis of Motifs in Networks*. In: *Transactions on Computational Systems Biology* 3 (2005), S. 89–104 (siehe S. 104).

[SS10] S. Sinclair und S. W. Smith. *What's wrong with access control in the real world?* In: *IEEE Security & Privacy* 8.4 (2010), S. 74–77 (siehe S. 84, 85).

[Sto+08] S. J. Stolfo u. a., Hrsg. *Insider Attack and Cyber Security. Beyond the Hacker*. Advances in Information Security. Boston, MA: Springer US, 2008.

[Str+00] J. D. Strunk u. a. *Self-securing Storage: Protecting Data in Compromised System*. In: *Proceedings of the 4th Conference on Symposium on Operating System Design & Implementation*. Bd. 4. OSDI '00. San Diego, California: USENIX Association, 2000 (siehe S. 38, 234).

[STW96] M. Steiner, G. Tsudik und M. Waidner. *Diffie-Hellman Key Distribution Extended to Group Communication*. In: *Proceedings of the 3rd ACM conference on Computer and communications security*. New Delhi, India: ACM Press, 14. März 1996, S. 31–37 (siehe S. 214).

[Swe02] L. Sweeney. *Achieving k-anonymity privacy protection using generalization and suppression*. In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 10.05 (2002), S. 571–588. World Scientific (siehe S. 167, 168).

[Swe97] L. Sweeney. *Weaving Technology and Policy Together to Maintain Confidentiality*. In: *The Journal of Law, Medicine & Ethics* 25.2-3 (1997). PMID: 11066504, S. 98–110 (siehe S. 167, 169, 190).

[Tho17] I. Thomson. *Engineer gets 18 months in the clink for looting ex-bosses' FTP server*. The Register Online News, 8. Aug. 2017. URL: https://www.theregister.com/2017/08/08/us_engineer_gets_18mo_ftp_access/ (besucht am 24. 07. 2020) (siehe S. 226).

[Tho18] I. Thomson. *Rogue IT admin goes off the rails, shuts down Canadian train switches*. The Register Online News, 14. Feb. 2018. URL: https://www.theregister.com/2018/02/14/rogue_it_admin_canadian_railway_switches/ (besucht am 24. 07. 2020) (siehe S. 226).

[Thü+14] G. Thüsing u. a. *Beschäftigtendatenschutz und Compliance. Effektive Compliance im Spannungsfeld von BDSGP, Persönlichkeitsschutz und betrieblicher Mitbestimmung*. 2. Auflage. München: C.H. Beck, 2014 (siehe S. 159, 164).

[Tis+16] M. Tischer u. a. *Users Really Do Plug in USB Drives They Find*. In: *2016 IEEE Symposium on Security and Privacy (SP)*. Mai 2016, S. 306–319 (siehe S. 84).

[Tug00] T. Tuglular. *A preliminary Structural Approach to Insider Computer Misuse Incidents*. In: *1st European Anti-Malware Conference*. EICAR '00. 2000 (siehe S. 19, 220).

[US13] M. J. Vickers. *Department of Defense Instruction: Countering Espionage, International Terrorism, and the Counterintelligence (CI) Insider Threat*. Retrieved from Defense Technical Information Center (DTIC). Okt. 2013. URL: https://www.hsdl.org/?view&did=745624 (besucht am 24. 07. 2020) (siehe S. 38, 256, 264).

[Ver19] Verizon. *2019 Data Breach Investigations Report*. Mai 2019. URL: https://enterprise.verizon.com/resources/reports/2019-data-breach-investigations-report.pdf (besucht am 24. 07. 2020) (siehe S. 43, 226).

[Vir20] VirusTotal. *Academic Malware Samples*. 4. März 2020. URL: https://www.virustotal.com/ (besucht am 24. 07. 2020) (siehe S. 137).

[VK83] V. L. Voydock und S. T. Kent. *Security Mechanisms in High-Level Network Protocols*. In: *ACM Computing Surveys (CSUR)* 15.2 (1983), S. 135–171 (siehe S. 11).

[VT19] D. Volz und B. Tau. *FBI's Use of Surveillance Database Violated Americans' Privacy Rights, Court Found*. Online News. The Wall Street Journal, 8. Okt. 2019. URL: https://www.wsj.com/articles/fbis-use-of-foreign-surveillance-tool-violated-americans-privacy-rights-court-found-11570559882 (besucht am 24. 07. 2020) (siehe S. 33).

[Wei19] E.-M. Weiß. *FBI-Agenten missbrauchen zehntausendfach Datenbank*. Heise Online News, 9. Okt. 2019. URL: https://www.heise.de/newsticker/meldung/FBI-Agenten-missbrauchen-zehntausendfach-Datenbank-4549939.html (besucht am 24. 07. 2020) (siehe S. 1, 33).

[Wer05] S. Wernicke. *A Faster Algorithm for Detecting Network Motifs*. In: *Algorithms in Bioinformatics*. Hrsg. von R. Casadio und G. Myers. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, S. 165–177 (siehe S. 103, 104, 123).

[WFP99] C. Warrender, S. Forrest und B. Pearlmutter. *Detecting intrusions using system calls: alternative data models*. In: *Proceedings of the 1999 IEEE Symposium on Security and Privacy*. Mai 1999, S. 133–145 (siehe S. 107, 222).

[Woh00] P. Wohlmacher. *Digital Certificates: A Survey of Revocation Methods*. In: *Proceedings of the 2000 ACM Workshops on Multimedia*. MULTIMEDIA '00. Los Angeles, California, USA: Association for Computing Machinery, 2000, S. 111–114 (siehe S. 67).

[Won19] J. C. Wong. *Former Google self-driving car engineer charged with theft of trade secrets*. The Guardian Online News, 27. Aug. 2019. URL: https://www.theguardian.com/technology/2019/aug/27/anthony-levandowski-google-trade-secrets-theft (besucht am 24. 07. 2020) (siehe S. 226).

[Woo00] B. Wood. *An Insider Threat Model for Adversary Simulation*. In: *Research on Mitigating the Insider Threat to Information Systems – #2. Proceedings of a Workshop Held August, 2000*. Hrsg. von R. H. Anderson u. a. Santa Monica, CA: RAND Corporation, 2000, S. 41–48 (siehe S. 18, 19, 22, 24, 220, 221).

[XW05] R. Xu und D. Wunsch. *Survey of Clustering Algorithms*. In: *IEEE Transactions on Neural Networks* 16.3 (Mai 2005), S. 645–678 (siehe S. 130).

[YP10] Q. Yaseen und B. Panda. *Predicting and Preventing Insider Threat in Relational Database Systems*. In: *Information Security Theory and Practices. Proceedings of the 4th IFIP WG 11.2 International Workshop on Security and Privacy of Pervasive Systems and Smart Devices*. Hrsg. von P. Samarati u. a. WISTP '10. Berlin, Heidelberg: Springer, Apr. 2010, S. 368–383 (siehe S. 38, 254, 265).

[Yu+14] Z. Yu u. a. *Detection of Insider Attacks in Cloud Based e-Healthcare Environment*. In: *International Conference on Information Technology*. Dez. 2014, S. 195–200 (siehe S. 88).

[ZDO14] R. Zhuang, S. A. DeLoach und X. Ou. *Towards a Theory of Moving Target Defense*. In: *Proceedings of the First ACM Workshop on Moving Target Defense - MTD '14* (2014), S. 31–40 (siehe S. 77).

[Zet11] K. Zetter. *Bradley Manning Charged With 22 New Counts, Including Capital Offense*. Wired Online News, 3. Feb. 2011. URL: https://www.wired.com/2011/03/bradley-manning-more-charge/ (besucht am 24. 07. 2020) (siehe S. 226).

[Zha+07] Q. Zhang u. a. *Aggregate Query Answering on Anonymized Tables*. In: *2007 IEEE 23rd International Conference on Data Engineering*. Apr. 2007, S. 116–125 (siehe S. 168).

[Zim+16] E. Zimmer u. a. *Catching Inside Attackers: Balancing Forensic Detectability and Privacy of Employees*. In: *Open Problems in Network Security - IFIP WG 11.4 International Workshop, iNetSec 2015, Zurich, Switzerland, October 29, 2015, Revised Selected Papers*. Hrsg. von J. Camenisch und D. Kesdogan. Bd. 9591. Lecture Notes in Computer Science. Springer, 2016, S. 43–55 (siehe S. 8, 149, 170, 175, 229).

[Zim+20] E. Zimmer u. a. *PEEPLL: Privacy-Enhanced Event Pseudonymisation with Limited Linkability*. In: *Proceedings of the 35th Annual ACM Symposium on Applied Computing*. SAC '20. Brno, Czech Republic: Association for Computing Machinery, 2020, S. 1308–1311 (siehe S. 149, 175, 229).

[ZMR10] Y. Zhang, F. Monrose und M. K. Reiter. *The Security of Modern Password Expiration: An Algorithmic Framework and Empirical Analysis*. In: *Proceedings of the 17th ACM Conference on Computer and Communications Security*. CCS '10. Chicago, Illinois, USA: ACM, 2010, S. 176–186 (siehe S. 67).

[ZZ18] X. Zhou und R. Zafarani. *Fake News: A Survey of Research, Detection Methods, and Opportunities*. Dez. 2018. URL: https://arxiv.org/abs/1812.00315 (besucht am 24. 07. 2020) (siehe S. 64).

Eidesstattliche Erklärung

Ich erkläre, dass ich die vorliegende Arbeit selbstständig, unter Angabe aller Zitate und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe.

Dipl.-Inf. Ephraim Zimmer
Hamburg, den 31. Juli 2020