UNIVERSITÄT BONN

Bedrohung durch Identitätsdatendiebstahl

Datenerhebung, Analyse und Mitigation

Dissertation

ausgearbeitet von

Timo Malderle

geboren in Lüdenscheid

zur Erlangung des akademischen Grades
Doctor Rerum Naturalium (Dr. rer. nat.)

vorgelegt an der
Rheinischen Friedrich-Wilhelms-Universität Bonn
Mathematisch-Naturwissenschaftliche Fakultät

im Promotionsfach
Informatik

Bonn, Oktober 2020

Angefertigt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät
der Rheinischen Friedrich-Wilhelms-Universität Bonn

Erster Gutachter:	Prof. Dr. Michael Meier Universität Bonn
Zweiter Gutachter:	Prof. Dr. Matthew Smith Universität Bonn
Tag der Promotion:	17. Februar 2021
Erscheinungsjahr:	2021

Danksagung

Die vorliegende Arbeit ist im Rahmen meiner Tätigkeit an der Universität Bonn im Fachbereich der IT-Sicherheit entstanden. Die Idee für diese Arbeit entwickelte ich während meiner Mitarbeit an dem BMBF-finanzierten Forschungsprojekt *Effektive Information nach digitalem Identitätsdiebstahl (EIDI)*. Insgesamt hat die Forschung für meine Dissertation dreieinhalb Jahre gedauert, wobei meine beruflichen und privaten Forschungsaktivitäten voneinander profitieren konnten.

Meinem Doktorvater Prof. Dr. Michael Meier danke ich für die eingeräumten Freiheiten und die hervorragende Betreuung bei der Erstellung meiner gesamten Arbeit. Der rege fachliche Austausch mit ihm sowie seine konstruktive Unterstützung haben maßgeblich zu dieser Arbeit beigetragen. Des Weiteren danke ich Prof. Dr. Matthew Smith sowie der weiteren Prüfungskommission für die Begutachtung meiner Arbeit.

Für die wertvollen Ideen, die hilfreichen Fachgespräche und die wissenschaftliche Unterstützung danke ich ganz herzlich Dr. Matthias Wübbeling, Dr. Felix Boes, Pascua Theus und Saffija Kasem-Madani. Vielen Dank für die zahlreichen Diskussionen und die freundschaftliche Zusammenarbeit bei der Erstellung von gemeinsamen Publikationen. Felix Wiedemann, Tilmann Haak und Ingo Chao von der New Work SE danke ich für die fachliche Unterstützung und die Bereitstellung der statistischen Daten für meine Arbeit. Des Weiteren danke ich allen Kollegen des Forschungsprojektes EIDI für den konstruktiven Austausch und das Vermitteln von umfangreichem Wissen im Bereich Datenschutz, Psychologie und Recht.

Ich danke all den Studierenden, die mich durch ihre Abschlussarbeiten oder das Implementieren von Werkzeugen unterstützt haben. Ganz besonders danke ich Sven Knauer und Gina Muuss für die Mitarbeit bei der Erstellung wissenschaftlicher Publikationen. Sophie Jenke danke ich für das intensive Lektorat meiner Arbeit.

Meine Wertschätzung gilt insbesondere meiner Verlobten Joëlle Lang für ihre ständige Motivation und riesige Unterstützung bei der Anfertigung meiner Publikationen sowie bei dem Schreiben dieser Arbeit. Auch danke ich ihr von ganzem Herzen für das gefühlt tausendfache Korrekturlesen dieser Arbeit.

Rösrath, Oktober 2020

Timo Malderle

Kurzfassung

Immer häufiger werden in großem Stil Identitätsdaten gestohlen, die anschließend für kriminelle Aktivitäten missbraucht werden. Es existiert jedoch kein Ansatz, um eine breite Masse an betroffenen Personen automatisiert zu warnen. In der Literatur sind Konzepte zu finden, die es sicherheitsaffinen Personen ermöglichen, sich über die eigene Betroffenheit von Identitätsdatendiebstahl zu informieren. Jedoch sind weder in der Literatur datenschutzkonforme Verfahren zu finden, die den Ansatz verfolgen, einen Großteil der Benutzer zu schützen, noch sind solche Systeme im Einsatz. In dieser Arbeit wird ein Verfahren entwickelt, um datenschutzkonform gestohlene Identitätsdaten zu verarbeiten und anschließend möglichst viele betroffene Benutzer zu warnen. Dazu werden gestohlene Identitätsdaten gesammelt und automatisiert ausgewertet, um mit einem neuen technischen Warnkonzept bereits während der Erstellung dieser Arbeit mehr als 1.388.665 Benutzer zu warnen. Es ist zu erwarten, dass mit dem in dieser Arbeit vorgestellten Konzept noch deutlich mehr betroffene Personen gewarnt und auch geschützt werden können.

Inhaltsverzeichnis

1 Einleitung

Der steigende Grad der Vernetzung von Wirtschaft sowie Privatpersonen sorgt dafür, dass Abläufe vereinfacht oder neue Arten von Dienstleistungen geschaffen werden. Endanwender können eine Vielzahl von verfügbaren Onlinediensten nutzen und bei einer riesigen Menge an Online-Shops Waren bestellen. Hinzu kommen Onlinedienste, deren Verwendung für Benutzer unumgänglich ist. Beispielsweise legen viele Telefonanbieter, Versicherungen und öffentliche Versorgungsunternehmen den eigenen Kunden bei Vertragsschluss ein Benutzerkonto an, um hierüber beispielsweise Rechnungen bereitzustellen. Im Laufe der Zeit sammeln viele Benutzer eine unübersichtliche Menge an Benutzerkonten. Je nach Art eines Onlinedienstes hinterlegen Benutzer eine ganze Reihe von personenbezogenen Identitätsdaten bei dem jeweiligen Dienst wie Name, Anschrift oder Kreditkartennummern.

Kriminelle erkennen in diesen Daten ein finanzielles Potenzial für die persönliche Bereicherung, weshalb regelmäßig versucht wird, diese Daten unrechtmäßig zu entwenden und zu missbrauchen. Dazu greifen Kriminelle verschiedene Onlinedienste an, indem sie beispielsweise versuchen über Sicherheitslücken in Systemen der Onlinedienste an die Identitätsdaten der Benutzer zu gelangen. Gelingt ein Angriff, kommt es zu einem Identitätsdaten-Breach. Die erbeuteten Daten werden als Identitätsdaten-Leak bezeichnet und können im Anschluss in den verschiedensten Szenarien missbraucht werden. Missbrauchsszenarien reichen von der Mitnutzung des abonnierten Musik-Streaming-Dienstes bis hin zum vollständigen Identitätsdiebstahl, bei dem die Identität des Opfers genutzt wird, um weitreichenden Betrug zu begehen.

Zum Schutz vor unbefugtem Zugriff verwenden Onlinedienste verschiedene Authentifikationsverfahren. Dabei ist die Passwort-basierte Authentifikation die geläufigste Methode zur Absicherung von Benutzerkonten bei Onlinediensten. Die Anzahl an zu verwaltenden Passwörtern stellt viele Benutzer vor eine Herausforderung, weswegen gleiche Passwörter bei verschiedenen Onlinediensten verwendet werden. Durch die Mehrfachverwendung von Passwörtern wird das Problem des Identitätsdiebstahls verstärkt. Kommt es bei einem Onlinedienst zu einem Identitätsdaten-Breach, sind auch die Benutzerkonten bei anderen Diensten betroffen, wenn Benutzer dort die gleichen Zugangsdaten verwenden. Hierdurch gelingen pro Jahr mehrere tausend Identitätsdaten-Breaches, deren resultierende Identitätsdaten-Leaks über Jahre hinweg im Internet verbreitet werden.

Warn-Dienste wie *have i been pwned* [50] ermöglichen sicherheitsaffinen Benutzern die eigenen Identitätsdaten dahingehend zu überprüfen, ob sie in einem Identitätsdaten-Leak enthalten sind. Problematisch an einer solchen Art von Dienst ist, dass nur ein geringer Anteil von Benutzern solche Warn-Dienste kennt und auch tatsächlich nutzt. Über diese Art von Diensten hinaus gibt es kein Konzept, um die breite Masse von Benutzern zu schützen. Das Forschungsprojekt *Effektive Information nach digitalem Identitätsdiebstahl (EIDI)* [116] hat dieses Problem erkannt und sich zum Ziel gesetzt, ein effektives Warn-System zu entwickeln, mit dem deutlich mehr Betroffene geschützt werden können. Die vorliegende Arbeit ist im Kontext dieses Forschungsprojektes entstanden und fokussiert sich auf die technische Konzeption notwendiger Systeme.

Ziel dieser Arbeit ist, ein technisches Konzept für einen zentralen Frühwarndienst zu erarbeiten, der öffentliche Identitätsdaten-Leaks sammelt, automatisiert verarbeitet und mit den Ergebnissen betroffene Personen warnt. Für diesen zentralen Frühwarndienst sollen drei Komponenten entwickelt werden, die in dieser Arbeit getrennt voneinander präsentiert werden. Die erste Komponente ist ein Konzept zum Sammeln von öffentlich verfügbaren Identitätsdaten-Leaks. Basierend auf den Erfahrungen von umfangreichen manuellen Recherchen wird ein Vorgehen erarbeitet, um geeignete Quellen für Identitätsdaten-Leaks zu identifizieren. Dabei werden auch Eigenschaften von Leaks und deren Unterschiede vorgestellt. Die zweite Kom-

ponente ist ein Konzept für einen Parser, der Identitätsdaten-Leaks automatisiert analysiert, um dann die in den Leak-Daten enthaltenen Identitätsdatensätze zu extrahieren. Um die Funktionalität des Parsers nachzuweisen, wird der Parser in einer Evaluation anschließend genauer untersucht. Die dritte Komponente ist ein Übertragungsprotokoll, mit dem ein Frühwarndienst entdeckte Identitätsdaten-Leaks mit Onlinediensten teilen kann.

Der wissenschaftliche Beitrag dieser Arbeit ist ein evaluiertes Konzept eines Frühwarnsystems für Identitätsdaten-Leaks. Das Konzept basiert auf einem Vorgehen zum Sammeln von Identitätsdaten, damit ein Frühwarndienst möglichst zeitnah an neue Identitätsdaten-Leaks gelangt. Zur Verarbeitung enthält das Konzept ein Verfahren zum automatisierten Extrahieren von Identitätsdaten aus Leak-Dateien. Damit die Analyseergebnisse auch die Betroffenen erreichen, wird ein Konzept zur Warnung und eine technische Realisierung präsentiert.

Die Arbeit gliedert sich in folgende Kapitel: In Kapitel 2 wird zunächst zum Thema hingeführt, der genaue Kontext vorgestellt und die Notwendigkeit für ein solches Frühwarnsystem abgeleitet. Dazu wird zuerst auf das Prinzip der Passwort-basierten Authentifikation eingegangen, um anschließend genauer den aktuellen Umgang mit Passwörtern bei Benutzern darzustellen. Benutzerkonten werden in der Regel mittels Passwort-basierter Authentifikation geschützt, um einen Zugriff durch Dritte zu verhindern. Regelmäßig kommt es jedoch vor, dass Identitätsdaten gestohlen werden. Deshalb werden im Anschluss verschiedene Angriffsvektoren vorgestellt, die von Angreifern für den Diebstahl von Identitätsdaten eingesetzt werden (siehe Abschnitt 2.3). Basierend auf den dargestellten Erkenntnissen lässt sich auf die Notwendigkeit einer Abhilfe schließen (siehe Abschnitt 2.4). Hieraus werden die Forschungsfragen für diese Arbeit abgeleitet (siehe Abschnitt 2.5). Zum Abschluss des Kapitels werden die im Kontext dieser Arbeit entstandenen wissenschaftlichen Publikationen aufgelistet, um die wissenschaftliche Qualität der Teilergebnisse aufzuzeigen (siehe Abschnitt 2.6).

In Kapitel 3 werden die in der Literatur vorhandenen Vorarbeiten dargestellt. Darauffolgend wird in Kapitel 4 ein Vorgehen zum Sammeln von Identitätsdaten-Leaks vorgestellt. Dabei werden charakteristische Eigenschaften von Identitätsdaten-

Leaks und deren Verbreitung untersucht. Anhand dieser Eigenschaften werden Verfahren zum Sammeln von Identitätsdaten-Leaks entworfen. Für die gesammelten Identitätsdaten-Leaks wird in Kapitel 5 ein Parser entwickelt, welcher die Daten automatisiert verarbeiten kann. Um mit den verarbeiteten Identitätsdaten-Leaks auch Personen schützen zu können, wird in Kapitel 6 ein technisches Konzept für ein Frühwarnsystem vorgestellt. In Kapitel 7 werden die gesamten Systeme in einer gemeinsamen Evaluation in einem realen Warnszenario getestet. Abschließend werden die wichtigsten Beiträge dieser Arbeit in Kapitel 8 zusammengefasst.

2 Online-Identitäten und ihre Sicherheit

Im letzten Jahrzehnt hat die Informations- und Kommunikationstechnologie sich in vielen Bereichen der Gesellschaft verstärkt etabliert. Gerade durch das rasante Wachstum der Anzahl verfügbarer Onlinedienste werden viele Bereiche der sozialen Interaktion nachhaltig verändert. Darüber hinaus ist eine Veränderung im Konsumverhalten zu beobachten. Durch die Verfügbarkeit von Online-Shops und der Möglichkeit von Express-Lieferungen werden eine zunehmende Anzahl an Einkäufen vom stationären Handel auf den E-Commerce-Bereich verlagert. Gemein haben viele dieser Dienste, dass deren Benutzer sich ein Benutzerkonto anlegen müssen, um eine Dienstleistung nutzen oder aber Ware bestellen zu können. Beim Registrierungsvorgang müssen Benutzer einen Identifikator, beispielsweise eine E-Mail-Adresse, und ein selbst gewähltes Passwort angeben. Durch die Vielfalt an verfügbaren Onlinediensten und Online-Shops sammeln die meisten Benutzer im Laufe der Zeit eine schwer überschaubare Menge von Benutzerkonten an. Hierdurch verlieren viele Benutzer den Überblick und verwenden unsichere Passwörter oder gute Passwörter mehrfach.

Die Problematik resultierend aus der Überforderung der Benutzer beim Passwort-Management soll in diesem Kapitel genauer dargestellt und quantifiziert werden. Dabei sollen die beschriebenen Sachverhalte zum Thema dieser Arbeit hinführen. Dazu werden zunächst in Abschnitt 2.1 die Grundlagen der Passwort-Authentifikation dargestellt. Anschließend wird in Abschnitt 2.2 das Benutzerverhalten im Umgang mit Passwörtern genauer beschrieben. Passwörter werden ver-

wendet, um unbefugten Zugriff durch Dritte auf Benutzerkonten zu verhindern. Da es Kriminellen trotzdem regelmäßig gelingt, werden in Abschnitt 2.3 die Ursachen für Identitätsdatendiebstahl genauer erläutert. Um dem dargestellten Bedrohungspotenzial entgegenzuwirken, werden in Abschnitt 2.4 erste Folgerungen und Ideen zur Abhilfe vorgestellt. Hieraus lassen sich die Forschungsfragen ableiten, die in dieser Arbeit beantwortet werden sollen. Eine Darstellung der Forschungsfragen ist in Abschnitt 2.5 zu finden. Abschließend sind in diesem Kapitel die Publikationen aufgelistet, welche im Kontext dieser Arbeit entstanden sind (siehe Abschnitt 2.6).

2.1 Passwortbasierte Authentifikation

In den meisten Staaten dieser Welt erhalten Neugeborene die eigene amtliche Identität mit der Geburtsurkunde direkt nach der Geburt bescheinigt. Eine Geburtsurkunde enthält Merkmale wie Name, Geburtsdatum, Eltern und Geburtsort, damit das Baby möglichst eindeutig identifiziert und von anderen Personen unterschieden werden kann. Im Laufe eines Lebens werden zusätzlich andere Merkmale ausgestellt, die eine Person eindeutig identifizieren sollen. Solche nachträglich ausgestellten Merkmale sind beispielsweise die Personalausweisnummer, Steueridentifikationsnummer und Sozialversicherungsnummer. Auch wirtschaftliche Unternehmen stellen eigene identifizierende Merkmale aus: Krankenkassenmitgliedsnummer, Telefonnummer, IBAN und viele weitere. Der Grund für diese Vielfalt an Merkmalen ist, dass solche Institutionen aus Praktikabilitätsgründen eine eigene Personendatenbank pflegen und die darin enthaltenen Personen eindeutig identifizieren wollen. Bei der Nutzung des Internets ist dies sehr ähnlich. Beispielsweise stellt ein E-Mail-Provider einem Benutzer eine eindeutige E-Mail-Adresse aus, damit Nachrichten, die für diesen Benutzer bestimmt sind, auch tatsächlich bei diesem ankommen. Eine E-Mail-Adresse kann im Nachgang nicht nur für den Versand von E-Mails genutzt werden, sondern auch um sich mit dieser E-Mail-Adresse als Identifikationsmerkmal bei anderen Diensten zu registrieren. Diese anderen Dienste pflegen wiederum eigene Benutzerdatenbanken, um die eigenen Benutzer voneinander unterscheiden zu können und zu jedem Benutzer die für den Dienst notwendigen Informationen

zu besitzen. Solche digitalen Identitäten werden für verschiedene Zwecke benötigt wie zum Beispiel:

- Erreichbarkeit
- Kommunikation
- Abrechnung
- Diensterbringung

Die meisten Dienste bieten Benutzern einen Benutzerbereich mit personalisierten Inhalten an. Abhängig vom jeweiligen Dienst gibt es zumindest einen Profilbereich, welcher es dem Benutzer ermöglicht, das eigene Benutzerprofil zu bearbeiten. Diese personalisierten Bereiche müssen vor unberechtigtem Zugriff geschützt werden. Zum Schutz solcher Benutzerkonten hat sich das textbasierte Passwort etabliert. Durch ein Passwort wird das Konzept *Authentifikation durch Wissen* genutzt. Ein Benutzer bestätigt bei einem Anmeldeprozess, dass er der Benutzer ist, für den er sich ausgibt, indem er nachweist, dass er ein Geheimnis in Form des Passworts kennt.

Damit ein Dienst überprüfen kann, dass ein Passwort das Richtige ist, muss der Dienst eine Repräsentation des Passworts abspeichern und dieses mit jeder Eingabe im Anmeldeprozess vergleichen. Die Speicherung des Passworts im Klartext birgt viele Sicherheitsrisiken. Ein Sicherheitsrisiko ist, dass jeder Angreifer mit Zugriff auf die Benutzerdatenbank sich als ein beliebiger Benutzer ausgeben kann. Obwohl die Speicherung der Passwörter als Klartext in der Benutzerdatenbank schon lange nicht mehr Stand der Technik ist, kommt dies jedoch regelmäßig vor [96, 4, 14].

Um nicht den Klartext eines Passworts in der Benutzerdatenbank abspeichern zu müssen, können sogenannte Einwegfunktionen genutzt werden. Eine Einwegfunktion ist eine mathematische Funktion, die einen Eingabe-String einer beliebigen Länge verarbeitet und als Ausgabe einen String mit einer festen Länge produziert [97]. Jedoch gibt es keine kryptografische Funktion, mit der eine Eingabe-Zeichenkette aus einer Ergebnis-Zeichenkette berechnet werden kann [97]. Eine weitere Eigenschaft einer effektiven Einwegfunktion ist, dass sie möglichst kollisionsfrei arbeitet [97]. Das heißt, dass zwei verschiedene Eingabe-Zeichenketten nicht die gleiche

Ergebnis-Zeichenkette ergeben dürfen [97]. In der Regel wird zur Realisierung einer Einwegfunktion ein Hash-Verfahren verwendet.

Solche Hash-Verfahren werden unter anderem von Onlinediensten genutzt, um die Benutzerpasswörter nicht im Klartext abspeichern zu müssen. Aufgrund der Unumkehrbarkeit eignen sich diese Verfahren dazu, um Repräsentationen von Passwörtern als Hash-Wert in einer Benutzerdatenbank abzuspeichern. Bei jedem Anmeldeprozess wird die Passworteingabe mit dem genutzten Hash-Verfahren verarbeitet und anschließend mit dem Eintrag in der Benutzerdatenbank verglichen. Da ein Hash-Wert nicht auf den Eingabewert zurückgerechnet werden kann und keine Klartextpasswörter gespeichert werden, kann jemand, der Zugriff auf die Benutzerdatenbank erlangt, sich nicht direkt als einer der gespeicherten Benutzer ausgeben.

Jedoch könnte ein Angreifer eine allgemeine Liste mit den häufigsten Passwörtern nutzen, um die Hash-Werte für die häufigsten Passwörter vorzuberechnen. Ein Vergleich von den vorberechneten Hash-Werten mit den Hash-Werten aus der Benutzerdatenbank wird für eine hohe Anzahl an Einträgen Aufschluss über die dahinterstehenden Klartextpasswörter geben [97]. Um diese Art von *Wörterbuchangriffen* zu verhindern, kann ein *Salt* verwendet werden [97]. Ein Salt ist ein zufällig gewählter String. Dieser Salt wird in einer definierten Form mit dem Klartextpasswort konkateniert und anschließend wird dieses Ergebnis als Eingabe für die Hash-Funktion genutzt [97]. Zusätzlich wird dieser zufällig gewählte Salt als Klartext im Eintrag des jeweiligen Benutzers in der Benutzerdatenbank gespeichert. Wird bei einem Anmeldeversuch nun ein Passwort eingegeben, wird zunächst der für den Benutzer entsprechende Salt aus der Datenbank herausgesucht, dieser mit dem Passwort konkateniert und danach mit dem Hash-Verfahren der Hash-Wert berechnet. Ein Vergleich mit dem abgespeicherten Hash zeigt nun, ob der Anmeldeversuch legitim ist. Wörterbuchangriffe werden durch die Verwendung von Salts deutlich erschwert, da für jeden möglichen Salt ein eigenes Wörterbuch vorberechnet werden müsste.

Es existieren verschiedene Hash-Verfahren, die auf kryptografischer Ebene unterschiedlich aufgebaut sind, jedoch genau das zuvor dargestellte Konzept von Einwegfunktionen nutzen. Für die Speicherung von Passwörtern eignet sich nach

heutigem Stand der Technik jedoch nicht jedes existierende Hash-Verfahren. Ältere Hash-Verfahren wie MD5 weisen häufig Schwächen in der kryptografischen Konzeption auf, sodass mit vergleichsweise geringem Aufwand Kollisionen berechnet werden können [104]. Eine Hash-Kollision bedeutet im Passwortkontext, dass zwei verschiedene Passwörter denselben Hash-Wert ergeben. Dieses Problem ist jedoch für die Passwortsicherheit nur in geringem Umfang ausschlaggebend. Deutlich problematischer ist, dass die Berechnung von Hash-Werten mit älteren Hash-Verfahren auf heutiger Hardware in einer hohen Geschwindigkeit realisierbar ist, da zur Berechnung nur ein geringer Rechenaufwand benötigt wird. So lassen sich auch *Brute-Force-Angriffe* (Erklärung siehe Unterabschnitt 2.3.1) gegen Hash-Werte, in die ein Salt eingebunden ist, mit überschaubarem Aufwand realisieren [122]. Aktuellere Hash-Verfahren wie beispielsweise Argon2[1] besitzen den Vorteil, dass deren Berechnung deutlich rechenintensiver ist und dadurch *Brute-Force-Angriffe* erschwert werden. Problematisch ist jedoch, dass Softwareentwickler zum Schutz von Passwörtern bereits veraltete Hash-Verfahren wie MD5 verwenden [79]. Dadurch kann es bei aktuellen Softwaresystemen zu umfassenden Sicherheitsvorfällen kommen, die durch Verwendung aktueller Sicherheitsstandards hätten verhindert werden können.

Ein Identitätsdaten-Leak aus dem Jahr 2015 enthält 32 Millionen Hash-Werte von Passwörtern [129], die mit einem Hash-Verfahren namens Bcrypt inklusive Salt berechnet wurden. Bcrypt ist ein aktueller Hash-Algorithmus, dessen Berechnung eines Hash-Wertes ebenfalls rechenintensiv ist. Ein Forscher hat diese Hash-Werte aus dem Leak mittels *Brute-Force* versucht auf den Klartext zurückzuführen [89]. Das Ergebnis ist, dass mit handelsüblicher Hardware für das Berechnen von Kryptowährung innerhalb von 5 Tagen und 3 Stunden genau 4.007 Klartextpasswörter berechnet werden konnten [89]. Wäre statt Bcrypt ein Hash-Verfahren verwendet worden, welches deutlich geringere Ressourcen zur Berechnung benötigt wie beispielsweise MD5, dann wären in der gleichen Zeit deutlich mehr Klartextpasswörter aus den Hash-Werten ermittelt worden.

[1] Argon2: https://github.com/p-h-c/phc-winner-argon2.

Um ein Benutzerkonto unter anderem vor diesem Problem besser abzusichern, bieten viele Dienste an, ein Benutzerkonto mit einem zusätzlichen Faktor in Ergänzung zum Passwort abzusichern. Dieser zweite Faktor kann eines der folgenden Konzepte darstellen: *Authentifikation durch Wissen*, *Authentifikation durch Besitz* oder *Authentifikation durch Biometrie* [37]. Ein häufig genutztes Verfahren ist die Verwendung von generierten Einmalpasswörtern, sogenannte *One-Time-Password-Tokens* [17]. Bei diesem Verfahren wird ein Authentifikations-Token genutzt. Dieser Token kann ein Hardware-Token in Form eines physischen Geräts sein oder aber in Form eines Software-Tokens, beispielsweise als App auf einem Smartphone. Ein solcher Token teilt mit dem Web-Dienst ein *Shared-Secret*, auf dessen Grundlage und in Verbindung mit dem aktuellen Zeitstempel ein *One-Time-Password-Token* berechnet wird [17]. Dieser Token wird alle 30 bis 60 Sekunden neu berechnet, sodass jeder Token nur ein gewisses Zeitfenster gültig ist [17]. Ein solches Verfahren liefert einen höheren Schutz für ein Benutzerkonto nach einem Identitätsdatendiebstahl, da einem Angreifer zum Missbrauch der Identität noch der Token als zweiter Faktor für die Authentifikation fehlt. Dies gilt jedoch nur solange, wie das für die Berechnung des Tokens zugrunde liegende *Shared-Secret* geheim gehalten wird.

Auch wenn dieses Verfahren die Sicherheit von Benutzern nennenswert erhöht, wird es nicht umfangreich eingesetzt. Eine Studie aus 2015 zeigt, dass nur 6,5 % der Benutzerkonten aus einer Stichprobe mit 101.047 Google-Konten durch einen zweiten Faktor geschützt sind [88]. Im Jahr 2016 nutzten nur 1 % aller Benutzer beim Dienst *Dropbox*[2] einen zweiten Faktor zur Authentifikation [46]. Auch im Jahr 2018 nutzten weniger als 10 % aller Google-Benutzer einen zweiten Faktor [76].

Aus diesen Zahlen wird deutlich, dass Benutzer nicht immer die sicherste Möglichkeit zur Authentifikation nutzen. Das liegt vermutlich unter anderem an der Gewohnheit der Benutzer, dem gesteigerten Aufwand und der damit einhergehenden Komplexitätserhöhung sowie der Ungewissheit, was passiert, wenn der zweite Faktor verloren geht.

[2] Dropbox: https://dropbox.com.

Um diese Annahme zu verdeutlichen, wird im nächsten Abschnitt die Passwortnutzung aus Benutzerperspektive genauer analysiert. Im Anschluss wird darauf eingegangen, welchen Einfluss das Benutzerverhalten auf Identitätsdatendiebstahl hat. Andere Merkmale wie zum Beispiel Biometrie werden nur sehr selten zur Authentifikation genutzt und daher nicht weiter betrachtet.

2.2 Passwortverwendung aus Benutzerperspektive

Im Zuge der Digitalisierung sammeln sich bei jedem Internetnutzer eine umfangreiche Menge an Benutzerkonten an. Die durchschnittliche Anzahl an Benutzerkonten pro Benutzer schwankt in der Literatur, scheint jedoch auch noch nicht in quantitativen Studien umfassend erforscht zu sein. In einer Studie aus dem Jahr 2014 mit 27 Probanden wurden per Interview im Durchschnitt 27 Benutzerkonten pro Person ermittelt [110, 109]. In einer Studie aus dem Jahr 2016 wurden von den gleichen Forschern 348 Probanden mit einem Online-Fragebogen befragt. Diese Studie hat ergeben, dass im Durchschnitt jeder Nutzer 25,2 Benutzerkonten besitzt [109]. In einer weiteren Studie aus dem Jahr 2017 besitzen Benutzer 26,3 aktiv genutzte Online-Konten [86]. Jedoch scheint diese Anzahl aus 2017 recht gering, wenn sie mit den Ergebnissen der Studien aus 2014 und 2016 verglichen wird, da über die Jahre keine größere Veränderung zu erkennen ist. Eine Studie aus dem Jahr 2007 zeigt, dass dort schon im Durchschnitt 25 Konten pro Benutzer vorhanden sind [34]. Vermutlich beziehen sich die genannten Zahlen alle auf aktiv von den Probanden genutzte Konten, da bei allen Studien die Probanden nach der Anzahl an Benutzerkonten gefragt wurden und selten genutzte Konten vergessen worden sein könnten. Es lässt sich vermuten, dass die gesamte Anzahl an Benutzerkonten inklusive selten oder gar nicht mehr genutzter Dienste heute deutlich höher liegt.

Eine Unternehmensstudie zeigt, dass im Jahr 2020 jeder Internetnutzer im Durchschnitt 207 Benutzerkonten besitzt [63]. Diese Zahl wird von einer Studie eines Herstellers eines Passwortmanagers bekräftigt. Dort wurde eine Auswertung der von den eigenen Benutzern im Passwortmanager gespeicherten Passwörtern angefertigt. Im Durchschnitt besaß jeder Benutzer dieses Passwortmanagers im Jahr 2017 insgesamt 191 Konten bei Onlinediensten [41]. Ein wissenschaftlich belastbarer

Durchschnittswert müsste in einer zukünftigen Arbeit erhoben werden, jedoch wird deutlich, dass Benutzer eine umfassende Sammlung von Benutzerkonten bei verschiedenen Onlinediensten besitzen können. Für den Großteil solcher Onlinedienste bedarf es eines Passworts für die Anmeldung, welches sich der Benutzer als Geheimnis in irgendeiner Form merken muss. In einer Erhebung aus dem Jahr 2019 mit 1.045 deutschen Internetnutzern wurde herausgefunden, dass 37 % der Probanden sich ihre Passwörter merken [125]. Weitere 28 % notieren ihre Passwörter auf Zetteln [125]. Nur 10 % nutzen einen Passwortmanager und weitere 9 % speichern ihre Passwörter im Browser [125].

Bei vielen Benutzerkonten benötigt ein Benutzer viele Passwörter. 51 bis 59 % aller Nutzer benutzen deshalb mehrfach das gleiche Passwort und zusätzlich auch leicht veränderte Versionen davon [86, 120, 125]. Des Weiteren wird gezeigt, dass ein durchschnittlicher Nutzer 79 % seiner Passwörter mehrfach, entweder exakt (67 % der Passwörter) oder abgewandelt (63 % der Passwörter) wiederverwendet [86]. Von Benutzern nur leicht veränderte Passwörter lassen sich jedoch mit geringem Aufwand recht zuverlässig mittels maschinellen Lernens erraten [120].

Eine Untersuchung aus dem Jahr 2016 zeigt, dass das Durchschnittspasswort 8,98 Zeichen lang ist [124]. Dabei bestehen die meisten Passwörter aus reinen alphanumerischen Zeichenketten, nur 14 % aller Passwörter enthalten Symbole [124]. Auch wird belegt, dass starke Passwörter häufig wiederverwendet werden, genauso wie Passwörter, welche häufig eingegeben werden [124].

Problematisch ist auch, dass die Wiederverwendung eines Passworts bei mehreren Diensten dazu führt, dass die Sicherheit der entsprechenden Benutzerkonten bei jedem einzelnen Dienst gefährdet wird. Der Benutzer überträgt bei der Authentifikation sein Klartextpasswort mittels Transportverschlüsselung an den Dienst, welcher das Klartextpasswort dann zur Authentifikation weiterverarbeitet. Was ein Dienst aber genau mit dem übermittelten Klartextpasswort macht, ist aus Benutzersicht nicht nachvollziehbar. Ein fahrlässiger Dienst könnte die Passwörter beispielsweise im Klartext abspeichern oder ein maliziöser Dienst diese Informationen sogar missbrauchen. Verwendet ein Benutzer somit das gleiche Passwort bei unterschiedlichen Diensten, so besitzt jeder dieser Dienste die theoretische Möglichkeit sich beim jeweils anderen Dienst zu authentifizieren.

Ein Grund für die hohe Menge an unsicheren Passwörtern könnte sein, dass sich viele Irrtümer über die Passwortgestaltung und Passworthandhabung in den Köpfen der Benutzer manifestiert haben [74]. Solche Irrtümer sind beispielsweise, dass durch das Hinzufügen eines Sonderzeichens ein unsicheres Passwort sicher wird oder Passwörter bei häufig genutzten Diensten wiederverwendet werden dürfen [74]. Ein weiteres Problem ist, dass vielen Benutzern die Kritikalität ihres E-Mail-Kontos nicht bewusst ist [74, 117]. Bei vielen Onlinediensten wird das jeweilige Benutzerkonto mit der E-Mail-Adresse des Benutzers verknüpft. Vergisst ein Benutzer sein Passwort für einen Onlinedienst, dann kann er bei vielen Diensten die *Passwort-vergessen-Funktion* nutzen. In der Regel funktioniert dies so, dass der Benutzer per E-Mail einen Link zugesendet bekommt, mit dem er dann ein neues Passwort setzen kann. Hat ein Angreifer Zugriff auf das E-Mail-Konto eines Benutzers, kann der Angreifer bei anderen Diensten die *Passwort-vergessen-Funktion* verwenden, um einen Zugang zu weiteren Diensten des Benutzers zu bekommen.

Die dargestellten Problematiken führen dazu, dass ein umfangreicher Anteil aller Benutzerkonten von vielen Onlinediensten unzureichend geschützt ist. In einer Invivo-Studie aus 2019 mit 670.000 Probanden konnte nachgewiesen werden, dass mindestens 1,5 % aller genutzten Login-Daten gestohlen und in Identitätsdaten-Leaks veröffentlicht wurden [114]. Während dieser Studie nutzten 47,3 % aller Probanden mindestens eine Kombination aus E-Mail-Adresse und Passwort, die in den Leak-Daten enthalten war [114]. Somit wurde in dieser Studie gezeigt, dass von den 670.000 Personen jeder Zweite durch Identitätsdatendiebstahl betroffen ist. In einer vorangegangenen Studie wird gezeigt, dass 6,7 % aller Googlemail-E-Mail-Adressen aus mehreren großen Identitätsdaten-Leaks ein für Google gültiges Passwort enthielten [113]. Diese Werte zeigen auf, wie umfangreich die Problematik des Identitätsdatendiebstahls ist. Dazu kommt, dass die betroffenen Benutzer häufig den Diebstahl nicht bemerken und das kompromittierte Benutzerkonto ohne Maßnahmen weiterverwenden. Wang u. a. weisen nach, dass 70 % aller Benutzer ein gestohlenes Passwort mindestens ein Jahr lang nach einem Identitätsdatendiebstahl weiterhin nutzen [120]. Wenn zusätzlich beachtet wird, dass viele Benutzer ein Passwort mehrfach verwenden, steigt das Bedrohungspotenzial deutlich an.

2.3 Identitätsdatendiebstahl

Unzureichend abgesicherte Benutzerkonten bei Onlinediensten besitzen für Kriminelle eine gewisse Attraktivität, da sie diese Benutzerkonten für ihre kriminellen Aktivitäten teilweise mit geringem Aufwand missbrauchen können. Um ein Benutzerkonto dafür nutzen zu können, benötigt ein Krimineller in der Regel zunächst eine Nutzerkennung mit dazugehörigem Passwort. Diese Zugangsdaten können über primäre (Kapitel 2.3.1) oder sekundäre Angriffsvektoren (Kapitel 2.3.2) beschafft werden. Im Folgenden soll eine Übersicht über mögliche Vorgehen zur Beschaffung von gültigen Zugangsdaten gegeben werden. Es wird dabei auf die geläufigsten Verfahren eingegangen — der Anspruch auf Vollständigkeit wird jedoch nicht erhoben.

2.3.1 Primäre Angriffsvektoren

Die **primären Angriffsvektoren** sind solche, die sich ohne vorherige Angriffe umsetzen lassen. Diese sind in fünf Arten eingeteilt und in Abbildung 1 dargestellt:

- **Malware**: Schadsoftware wird verwendet, um die auf den infizierten Geräten gespeicherten oder eingegebenen Passwörter zu entwenden [113]. Damit Schadsoftware auf den Rechnern der Opfer installiert wird, werden beispielsweise infizierte E-Mail-Anhänge oder ein kostenlos zum Download angebotener und infizierter PDF-Reader genutzt.
- **Phishing**: Durch die Konfrontation der Opfer mit gefälschten E-Mails oder Websites versucht ein Angreifer Zugangsdaten von Opfern zu erhalten [113]. Häufig werden Inhalte von originalen Websites, wie Namen, Schriftarten und Logos genutzt, um die Absicht der Entwendung von Zugangsdaten zu verschleiern. Das Ziel ist, dass ein Opfer eine Phishing-E-Mail für eine Nachricht vom dargestellten Absender hält und anschließend beispielsweise auf einer gefälschten Website seine Zugangsdaten eingibt [17].
- **Data-Breaches**: Data-Breaches sind in diesem Kontext umfangreiche Sammlungen von vielen Benutzerdaten, die häufig aus den Benutzerdatenbanken von Dienstanbietern stammen [113]. Die von Onlinediensten eingesetzten

Systeme können durch Fehlkonfigurationen oder Sicherheitslücken in der eingesetzten Software angreifbar sein. Angreifer nutzen diese Schwachstellen, um Benutzerdaten aus den Benutzerdatenbanken zu entwenden.

- **Brute-Force**: Probiert ein Angreifer alle möglichen Passwörter für eine Nutzerkennung bei einem Onlinedienst aus, findet der Angreifer in der Theorie irgendwann das richtige Passwort. Jedoch schützen sich Dienste in der Regel vor solchen Angriffen, indem sie beispielsweise eine bestimmte Anzahl an Anfragen von einer IP-Adresse in einem definierten Zeitraum zulassen. Werden zu viele Passwörter ausprobiert, werden Anfragen von der spezifischen IP-Adresse nicht mehr beantwortet oder das betroffene Benutzerkonto gesperrt. Ist eine API nicht abgesichert, limitiert nur die Netzanbindung zum Server des Dienstbetreibers und die Performance der eingesetzten Systeme die benötigte Zeit für einen erfolgreichen Brute-Force-Angriff, da auf diese Weise ein hoher Durchsatz an Anmeldeversuchen möglich ist.
- **Use-After-Free** [43]: Viele E-Mail-Provider vergeben E-Mail-Aliasse neu, wenn das zum Alias gehörende E-Mail-Konto nicht mehr existiert. Einige kostenlose E-Mail-Provider löschen E-Mail-Postfächer, wenn diese über einen gewissen Zeitraum nicht genutzt werden, was zur Freigabe des genutzten Alias führt [43]. Ein Angreifer kann sich nach der Freigabe eines zuvor registrierten Alias ein eigenes Konto mit diesem registrieren. Da er dann die Kontrolle über den Alias hat, kann er anschließend ausprobieren, bei welchen Diensten der vorherige Benutzer registriert war und dort die *Passwort-vergessen-Funktion* nutzen, um Zugriff auf weitere Benutzerkonten zu bekommen [43].

In einer Studie mit einer Laufzeit von einem Jahr konnten 788.000 Opfer von herkömmlichen Keyloggern[3] identifiziert werden [113]. Darüber hinaus wurden 12,4 Millionen mögliche Opfer von sogenannten Phishing-Kits und 1,9 Milliarden Opfer von Identitätsdaten-Breaches ermittelt [113]. Phishing ist ein häufig genutzter Angriffsvektor. Eine Ursache dafür ist vermutlich die Existenz von *Phishing-Kits*, die es Angreifern ohne umfassende technische Kenntnisse ermöglicht, eigene Phishing-Kampagnen aufzusetzen [87]. Verwunderlich ist, dass die Entwickler dieser Kits selbst die Daten entwenden und verbreiten [87].

[3] Keylogger sind hard- oder softwarebasierte Systeme, welche den kompletten Eingabestrom des Benutzers aufzeichnen. Aus diesem Strom können im Nachgang beispielsweise Passwörter extrahiert werden.

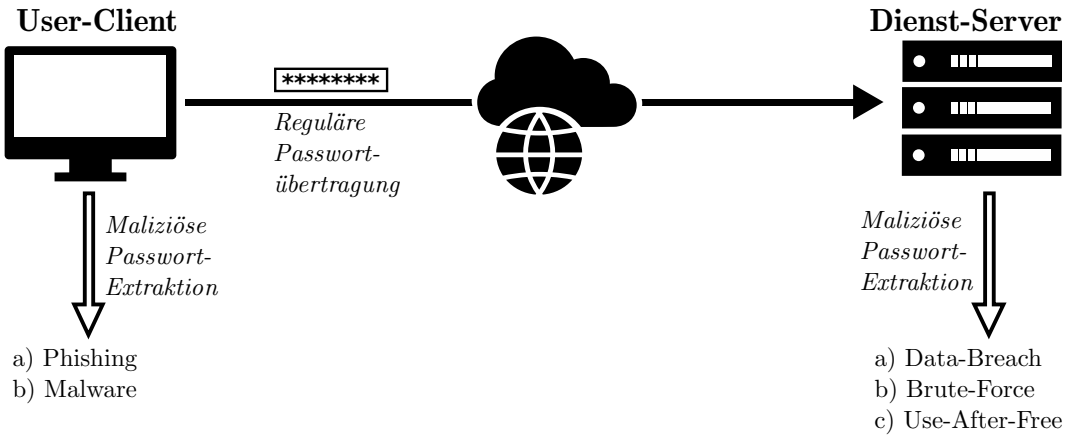


Abbildung 1: *Darstellung der primären Angriffsvektoren.*

2.3.2 Sekundäre Angriffsvektoren

Sekundäre Angriffsvektoren setzen einen erfolgreichen primären Angriff voraus.

- **Credential-Stuffing**: Bei Credential-Stuffing-Angriffen nutzen Angreifer bereits entwendete Passwörter, um Zugang zu weiteren Benutzerkonten zu erlangen. Hierzu gibt es unzählige Varianten. Beispielsweise werden die häufigsten Passwörter zur Anmeldung ausprobiert, um ein Benutzerkonto zu kompromittieren. Eine weitere Möglichkeit wird im Folgenden dargestellt. Wird eine Kombination aus E-Mail-Adresse und Passwort für einen Onlinedienst gestohlen, ist das betroffene Benutzerkonto kompromittiert. Angreifer können diese Anmeldedaten bei weiteren Diensten austesten, um zu überprüfen, ob der Benutzer genau diese Anmeldedaten mehrfach verwendet. Auf diese Weise können weitere Benutzerkonten bei anderen Diensten kompromittiert werden.
- **Passwort-Reset**: Erhält ein Angreifer Zugriff auf ein E-Mail-Konto, ist dieser in der Lage, mittels der *Passwort-vergessen-Funktion* bei anderen Diensten ein neues Passwort zu setzen und somit Zugang auf das entsprechende Benutzerkonto zu erhalten.
- **Hash-Cracking**: Anders als bei den zuvor genannten Angriffen handelt es sich beim Hash-Cracking nicht um einen Online-Angriff, der während des Angriffs mit Systemen eines Dienstes interagieren muss. Bei diesem Offline-Angriff

liegt dem Angreifer beispielsweise eine gestohlene Benutzerdatenbank mit Nutzerkennungen und Passwort-Hash-Werten vor. Um an das zu einem Hash-Wert zugehörige Klartextpasswort zu gelangen, kann der Angreifer versuchen den Hash-Wert zu brechen. Dazu berechnet er die Hash-Werte von möglichen Passwörtern und vergleicht die Ergebnisse mit dem gegebenen Hash-Wert. Für diesen Angriff muss eine große Menge an Hash-Werten berechnet werden. Jedoch wird die benötigte Zeit bis zum Erfolg nur durch das Hash-Verfahren und der eingesetzten Hardware des Angreifers beeinflusst.

Mittels dieser Angriffsvektoren erbeuten Angreifer regelmäßig große Mengen an Zugangsdaten. Oft enthalten die erbeuteten Daten weitere Informationen über die Benutzer wie Anschrift, Telefonnummern und weitere.

2.3.3 Konsequenzen des Identitätsmissbrauchs

Während bei *Identitätsdatendiebstahl* nur Identitätsdaten wie E-Mail-Adresse und Passwort gestohlen werden, werden beim *Identitätsdiebstahl* diese gestohlenen Zugangsdaten missbräuchlich verwendet. Werden Zugangsdaten gestohlen, ohne diese für eine Anmeldung bei dem zugehörigen Dienst zu verwenden, wird von einem *Identitätsdatendiebstahl* gesprochen. Sobald diese Daten für einen Betrug eingesetzt werden, wird von einem *Identitätsdiebstahl* gesprochen.

Da bei einem Identitätsdiebstahl von der vereinbarten Verarbeitung der Benutzerdaten durch einen Dienst abgewichen wird, können folgende Schäden auftreten: „. . . finanzieller Verlust, Rufschädigung oder wirtschaftliche oder gesellschaftliche Nachteile . . . “ [7].

Nach Johansen werden die Schäden durch Identitätsdiebstahl in vier Dimensionen eingeteilt [59]:

- Finanzielle Schäden
- Emotionale Schäden
- Physische Schäden
- Soziale Schäden

Finanzielle Schäden sind die naheliegendsten. Es sind jedoch nicht nur Schäden gemeint, bei denen finanzielle Mittel gestohlen werden. Es können nach einem Identitätsdiebstahl auch finanzielle Mittel benötigt werden, um die eigene Kreditwürdigkeit wiederherzustellen, ein kompromittiertes Bankkonto neu zu eröffnen oder andere administrative Aufgaben zur Schadensminimierung durchzuführen [59]. Alle psychischen Auswirkungen von Identitätsdiebstahl auf die Opfer werden als **emotionale Schäden** bezeichnet. Beispielsweise kann die Schadensbewältigung zu einem Stressempfinden führen oder Ängste bei den Opfern auslösen, die in Depressionen oder Suizidgedanken enden können [59]. Bei den **physischen Schäden** wirkt sich der Identitätsdiebstahl auf die physische Realität durch Veränderungen aus. Beispielsweise kann jemand sein Haus verlieren, wenn die eigene Kreditwürdigkeit negativ durch einen Identitätsdiebstahl beeinflusst wird [59]. **Soziale Schäden** beinhalten negative Veränderungen im sozialen Umfeld des Opfers. Solche Veränderungen könnten persönliche Beziehungen zu anderen Personen wie Freundschaften beschädigen [59].

Gerade die sozialen Schäden sollten nicht unterschätzt werden. Im Jahr 2015 wurden Identitätsdaten von einem Dienst gestohlen und veröffentlicht, die Informationen aus dem höchstpersönlichsten Lebensbereich enthielten. Die Veröffentlichung dieser Daten war vermutlich bei zwei Opfern dieses Identitätsdiebstahls der Grund für einen anschließenden Suizid [93]. Dieses Beispiel soll verdeutlichen, welche gravierenden Auswirkungen von einem solchen Angriff ausgehen können. Auch das Ausmaß von Identitätsdiebstahl ist kein geringes. Insgesamt sind 30 % von 1.025 Befragten aus Deutschland im Jahr 2019 schon einmal Opfer von irgendeiner Form von Identitätsdiebstahl geworden [108]. In einer anderen Studie aus dem Jahr 2016

wurde eine Zahl von 33 % Betroffenen bei einer Stichprobe mit 1.000 Probanden erhoben [90]. Eine weitere Studie aus dem Jahr 2017 liefert einen vergleichbaren Wert [112]. Diese drei Studien liefern somit ähnliche Werte und zeigen, dass ca. jeder Dritte schon einmal Opfer von Identitätsdiebstahl geworden ist. Ganz andere Ergebnisse liefert eine Studie aus dem Jahr 2019 [103]. In dieser Studie wurden in Zusammenarbeit mit einem Meinungsforschungsinstitut Datensätze von 5.000 erwachsenen Amerikanern ausgewählt. Die ausgewählten Datensätze waren repräsentativ für die USA verteilt und enthielten pro Datensatz eine E-Mail-Adresse. Diese E-Mail-Adressen wurden mithilfe des Dienstes *Have I Been Pwned*[4] dahingehend überprüft, ob sie in einem Identitätsdaten-Leak enthalten sind. Das Ergebnis ist, dass die E-Mail-Adressen von fast 83 % der US-Amerikaner bereits abhandengekommen sind. Dabei seien die 83 % als untere Schwelle anzusehen, da der genutzte Dienst nicht alle existierenden Identitätsdaten-Leaks in seiner Datenbank vorhält und das Marktforschungsinstitut nur eine und nicht alle E-Mail-Adressen der Probanden in den Datensätzen gespeichert hat [103]. Obwohl bei 83 % der Probanden Identitätsdaten gestohlen wurden, folgt daraus nicht, dass auch alle gestohlenen Identitätsdaten für einen Identitätsdiebstahl tatsächlich missbräuchlich verwendet wurden. Von den betroffenen Personen aus der Studie von 2016 wurden 29 % finanziell geschädigt, wobei der durchschnittliche Schaden bei 1.366 Euro lag [90]. Der gesamte Schaden für Deutschland lag im Jahr 2017 bei rund 2,6 Milliarden US-Dollar [112].

2.4 Folgerungen & Abhilfe

Aus dem vorangegangenen Abschnitt werden die durch Identitätsdiebstahl verursachten Schäden, das Ausmaß und die damit verbundenen Risiken deutlich. Die mit Identitätsdiebstahl aufgelisteten Schäden sind jedoch nicht nur für Privatpersonen relevant. Auch Unternehmen haben ein Interesse daran, ihre Kunden und Mitarbeiter vor Identitätsdiebstahl zu schützen, um dadurch eigene Imageverluste, Zahlungsausfälle oder Angriffe auf die eigene Infrastruktur zu vermeiden. Der präventive Schutz und die reaktive Mitigation vor Identitätsdiebstahl sind somit

[4] Troy Hunt: Have I Been Pwned, https://haveibeenpwned.com/.

für Privatpersonen sowie für Unternehmen wünschenswert. Deswegen ist das Ziel dieser Arbeit, ein Verfahren zur Warnung von Privatpersonen und Unternehmen zu konzipieren, um bei der Realisierung von reaktiven Maßnahmen zu unterstützen.

Schutzmaßnahmen können durch den Benutzer selbst umgesetzt werden. Die Passwortänderung, die Überprüfung anderer Benutzerkonten auf Unregelmäßigkeiten und die polizeiliche Anzeige sind dabei Möglichkeiten. Es kommt jedoch auf das betroffene Konto an, welche Maßnahmen hilfreich und notwendig sind. Unternehmen können weitere sinnvolle Schutzmaßnahmen ergreifen, die sowohl den Benutzer als auch die Infrastruktur des Unternehmens schützen. Beispielsweise kann das Unternehmen das ganze Benutzerkonto sperren oder spezielle Funktionen deaktivieren, wie das Versenden von Nachrichten oder das Bezahlen mit der hinterlegten Kreditkarte.

2.4.1 Grundidee eines Frühwarnsystems

Die Grundidee für diese Arbeit ist die Konzeption und Entwicklung eines Frühwarnsystems, welches zeitnah betroffene Personen automatisiert über den Identitätsdatendiebstahl informiert. Die technische Konzeption eines solchen Frühwarnsystems ist im Rahmen des Forschungsprojektes *EIDI - Effektive Information nach digitalem Identitätsdiebstahl* entstanden. Dieses Forschungsprojekt wird vom Bundesministerium für Bildung und Forschung mit dem Förderkennzeichen 16KIS0696K finanziert (2017 bis 2020). Teil dieser Arbeit ist die technische und funktionale Entwicklung eines Konzepts für ein solches Warnsystem. Datenschutzrechtliche, juristische oder psychologische Fragestellungen wurden von anderen Projektpartnern bearbeitet. Die Ergebnisse der jeweiligen Projektpartner haben Anforderungen an das Warnsystem hervorgebracht, welche in der Konzeption berücksichtigt werden.

Darstellung des Gesamtkonzepts

Das Frühwarnsystem soll eine möglichst umfangreiche Menge an Opfern von Identitätsdatendiebstahl warnen, damit diese sich vor weiteren Auswirkungen schützen können. Leak-Informationsdienste wie der *HPI-Leakchecker*[5] oder der Dienst *have i*

[5] HPI-Leakchecker: https://sec.hpi.de/ilc/.

been pwned[6] erreichen vermutlich nicht genügend Betroffene, da diese Dienste bekannt sein und zusätzlich auch regelmäßig verwendet werden müssen. Aus diesem Grund soll ein anderer Ansatz zur Warnung verfolgt werden. Es soll ein zentraler Dienst gestaltet werden, welcher Identitätsdaten-Leaks sammelt, diese aufbereitet und anschließend über geeignete Warnkanäle die Opfer informiert. Dieses Vorgehen kann in drei Schritte aufgeteilt werden, welche im Folgenden genauer erläutert werden:

1. **Identitätsdaten-Leaks sammeln:** Ein zentraler Dienst durchsucht das Internet nach öffentlich zugänglichen Identitätsdaten-Leaks. Damit diese Aufgabe effizient umgesetzt werden kann, bedarf es Verfahren, mit denen die im Internet verfügbaren Datenmengen durchsucht und die für den Dienst benötigten Daten herausgefiltert werden können. Werden dabei Identitätsdaten-Leaks identifiziert, werden diese heruntergeladen.
2. **Automatisierte Analyse der Daten:** Die heruntergeladenen Dateien enthalten in der Regel strukturierte Daten von unbekannter Struktur. Die Daten weisen somit zwar eine Struktur auf, jedoch ist diese unbekannt, da diese häufig individuell vom Verfasser des Leaks gestaltet wird. Um eine Warnung der Betroffenen zu ermöglichen, muss der Inhalt genauer analysiert werden. Dazu muss die Datenstruktur erkannt werden, um die enthaltenen Daten wie E-Mail-Adressen und Passwörter zu ermitteln.
3. **Versand von Warnungen:** Der zentrale Warndienst hat im letzten Schritt die Möglichkeit, die betroffenen Personen selbst zu informieren, indem er diese direkt kontaktiert. Dazu müsste der Warndienst Kontakt zu den Betroffenen über geeignete Kommunikationskanäle aufnehmen. Diese Möglichkeit ist sehr beschränkt und abhängig von den im Leak enthaltenen Daten realisierbar. Deswegen werden für die Warnung Kooperationspartner eingebunden, welche bessere Möglichkeiten besitzen, um die eigenen Kunden über vertrauenswürdigere Kommunikationskanäle zu warnen. Als Kooperationspartner bieten sich Onlinedienste an, da diese häufig über weitere Informationen zu deren Benutzern verfügen, welche zur geeigneten Kontaktaufnahme genutzt

[6] have i been pwned: https://haveibeenpwned.com/.

werden können. Zur Warnung sendet der zentrale Warndienst notwendige Informationen aus den aufbereiteten Leak-Daten an die Kooperationspartner. Jeder Kooperationspartner gleicht die enthaltenen Daten mit der eigenen Benutzerdatenbank ab und informiert die betroffenen Benutzer.

2.4.2 Datenschutzrechtliche und juristische Aspekte

Bei einem solchen Forschungsprojekt stellen sich nicht nur Fragen der technischen Realisierbarkeit, sondern auch die rechtliche Zulässigkeit des geplanten Vorhabens aufgrund der verarbeiteten Daten. Aus den gesetzlichen Vorgaben ergeben sich Anforderungen, die bei einer Entwicklung berücksichtigt werden müssen. Die datenschutzrechtlichen und juristischen Aspekte wurden im EIDI-Forschungsprojekt genauer untersucht. Die wichtigsten und für diese Arbeit relevantesten Erkenntnisse werden im Folgenden skizziert.

Art. 14 Abs. 1 DSGVO sieht vor, dass Personen informiert werden müssen, wenn ihre Daten verarbeitet werden, diese jedoch nicht bei der Person selbst erhoben worden sind [118]. Da in diesem Forschungsprojekt Identitätsdaten verarbeitet werden sollen, müssten die Betroffenen direkt über die Verarbeitung in Kenntnis gesetzt werden, weil die Betroffenen die Daten nicht selbst zur Verarbeitung freigegeben haben. Auch wenn die Idee des Projektes ist, die von Identitätsdatendiebstahl betroffenen Personen zu warnen, müsste nach Art. 14 Abs. 1 DSGVO eine Benachrichtigung erfolgen. Technisch als auch organisatorisch ist dies eventuell jedoch nicht realisierbar, da die betroffenen Personen häufig weder eindeutig identifiziert noch auf einem sicheren Kanal informiert werden können. Art. 14 Abs. 5 b DSGVO sieht vor, dass eine Information der Personen nicht notwendig ist, wenn dies unverhältnismäßig oder unmöglich ist [118]. Zusätzlich gibt es in Art. 14 Abs. 5 b DSGVO eine Befreiung von dieser Notwendigkeit für die Wissenschaft [118, 25]. Jedoch soll die Verarbeitung auf nur benötigte Daten nach Art. 5 Abs. 1 c DSGVO beschränkt werden [118]. Das bedeutet, dass nicht benötigte Daten aus Identitätsdaten-Leaks möglichst frühzeitig verworfen werden.

Bereits bei der Konzeption des Systems soll das Paradigma *Privacy by Design* nach Art. 25 DSGVO angewendet werden [118]. Von Beginn der Entwicklung des Frühwarnsystems werden sämtliche Verarbeitungsschritte datenschutzrechtlich durchdacht und das zu entwickelnde System von Grund auf datenschutzkonform konzipiert. Dazu zählt, dass die erhobenen Daten nach Art. 5 Abs. 1 c sowie Art. 89 Abs. 1 DSGVO datenminimierend gespeichert werden müssen [25]. Art. 32 Abs. 1 und Art. 25 Abs. 1, 2 DSGVO sehen dazu verschiedene technische Maßnahmen vor wie Pseudonymisierung und Anonymisierung [25]. Zum Schutz der Rechte und Freiheiten von den betroffenen Personen muss somit bei der Konzeption eines solchen Systems darauf geachtet werden, dass so wenige Daten wie nötig gespeichert werden und diese zu speichernden Daten pseudonymisiert und verschlüsselt werden.

Wenn dieses Warnsystem produktiv betrieben werden soll, muss festgelegt werden, ob dieses System ein normales oder hohes Risiko für natürliche Personen darstellt. Sollte sich durch ein solches Warnsystem ein datenschutzrechtliches hohes Risiko für Personen ergeben, müsste eine Datenschutz-Folgenabschätzung nach Art. 35 Abs. 1 S. 1 DSGVO angefertigt werden [38, 25].

Der deutsche Staat könnte durch das Verfassungsrecht, genauer dem Persönlichkeitsrecht, dazu verpflichtet sein, die digitalen Identitäten von deutschen Staatsbürgern zu schützen [39]. Ob und in welcher Form diese Interpretation von [39] Anwendung findet, ist rechtlich zurzeit ungeklärt.

2.5 Forschungsfragen

Zur technischen Konzeption eines solchen Warnsystems müssen neuartige Verfahren und Systeme entwickelt werden. Aus den zuvor dargestellten Aspekten lassen sich daher folgende Forschungsfragen ableiten:

1. Wie können Identitätsdaten-Leaks gesammelt werden, wie werden diese Daten verbreitet und welche Eigenschaften besitzen sie?
2. Wie können Identitätsdaten-Leaks vollautomatisiert analysiert und normalisiert werden, sodass die Syntax und Semantik des Identitätsdaten-Leaks korrekt erkannt werden?
3. Wie können Betroffene eines Identitätsdatendiebstahls von einem Onlinedienst geeignet geschützt werden?

Diese Forschungsfragen werden in der vorliegenden Arbeit beantwortet.

2.6 Publikationen

Während der Erstellung der vorliegenden Dissertation sind folgende wissenschaftliche Beiträge entstanden:

1. **Timo Malderle**, Matthias Wübbeling, Michael Meier: *Sammlung geleakter Identitätsdaten zur Vorbereitung proaktiver Opfer-Warnung*. In: Paul Drews, Burkhardt Funk, Peter Niemeyer und Lin Xie (Hrsg.), Multikonferenz Wirtschaftsinformatik 2018, Data driven X - Turning Data into Value. Lüneburg, 1381-1393, 2018.
2. **Timo Malderle**, Matthias Wübbeling, Sven Knauer, Michael Meier: *Ein Werkzeug zur automatisierten Analyse von Identitätsdaten-Leaks*. In: Langweg, H., Meier, M., Witt, B. C. & Reinhardt, D. (Hrsg.), SICHERHEIT 2018. Bonn: Gesellschaft für Informatik e.V., 43-54, 2018.
3. **Timo Malderle**, Matthias Wübbeling, Sven Knauer, Arnold Sykosch, Michael Meier: *Gathering and Analyzing Identity Leaks for a proactive Warning of affected*

Users. In Proceedings of the ACM International Conference on Computing Frontiers (CF '18). ACM, New York, NY, USA, 2018.

4. Daniel Gruss, Michael Schwarz, Matthias Wübbeling, Simon Guggi, **Timo Malderle**, Stefan More, Moritz Lipp: *Use-After-FreeMail: Generalizing the Use-After-Free Problem and Applying it to Email Services*. In Proceedings of the 2018 on Asia Conference on Computer and Communications Security (ASIACCS '18). ACM, New York, NY, USA, 297-311. 2018.

5. **Timo Malderle**, Matthias Wübbeling, Sven Knauer and Michael Meier: *Warning of Affected Users About an Identity Leak*. In: Madureira A., Abraham A., Gandhi N., Silva C., Antunes M. (eds) Proceedings of the Tenth International Conference on Soft Computing and Pattern Recognition (SoCPaR 2018). SoCPaR 2018. Advances in Intelligent Systems and Computing, vol 942. Springer, Cham, 2020.

6. **Timo Malderle**, Matthias Wübbeling, Michael Meier: *Effektive Warnung bei Identitätsdiebstahl an Hochschulen*. In: Sicherheit in vernetzten Systemen - 27. DFN-Konferenz. Hrsg. von Albrecht Ude. Hamburg: BOOKS ON DEMAND, 2020.

7. **Timo Malderle**, Sven Knauer, Martin Lang, Matthias Wübbeling, Michael Meier: *Track Down Identity Leaks Using Threat Intelligence*. In: Furnell S., Mori P., Weippl E., Champ O. (eds) ICISSP 2020 - Proceedings of the 6th International Conference on Information Systems Security and Privacy. SCITEPRESS – Science and Technology Publications. Malta, Valetta, 2020.

8. **Timo Malderle**, Felix Boes, Gina Muuss, Matthias Wuebbeling and Michael Meier: *Credential Intelligence Agency - A Threat Intelligence Approach to Mitigate Identity Theft*. In: Steven Furnell, Paolo Mori, Edgar Weippl and Olivier Camp (eds), ICISSP 2020 - Revised Selected Papers, Springer, 2020. (In Submission)

9. Saffija Kasem-Madani, **Timo Malderle**, Felix Boes, Michael Meier: *Privacy-Preserving Warning Management for an Identity Leakage Warning Network*. In: European Interdisciplinary Cybersecurity Conference, 2020. (In Submission)

3 Verwandte Arbeiten

In diesem Kapitel werden der Forschungsbereich und angrenzende Forschungsfelder dieser Arbeit genauer vorgestellt. Diese Arbeit bewegt sich im Forschungsbereich des Identitätsdiebstahls und beschäftigt sich mit reaktiven Schutzmaßnahmen gegen den Missbrauch von Identitätsdaten-Leaks. Um das Missbrauchspotenzial von Identitätsdaten-Leaks besser einschätzen zu können, wird in diesem Kapitel zunächst darauf eingegangen, wie Angreifer beim Missbrauch von Identitätsdaten-Leaks vorgehen und wie sie ihre Angriffe effektiver gestalten. Nachdem die aktuell in der Wissenschaft diskutierten Angriffsmethoden vorgestellt wurden, werden anschließend Verfahren vorgestellt, mit denen sich Identitätsdaten-Breaches als Sicherheitsvorfall erkennen lassen. Daraufhin wird beschrieben, welche Beiträge zum Verarbeiten von Identitätsdaten-Leaks bereits in der Vergangenheit entstanden sind. Aufbauend hierauf werden Dienste vorgestellt, die es Benutzern sowie Unternehmen ermöglichen, einen Leak-Status zu einer Identität abzufragen. Um Benutzer proaktiv zu warnen, wie es in dieser Arbeit beschrieben wird, wird ein geeigneter Kommunikationskanal benötigt. Die wissenschaftlichen Vorarbeiten zur Gestaltung solcher Warnnachrichten werden abschließend in diesem Kapitel vorgestellt.

Ein weiteres für diese Arbeit wichtiges Forschungsgebiet ist die Untersuchung des Benutzerverhaltens im Umgang mit Authentifikations-Methoden. Auf aktuelle Forschungen aus diesem Gebiet ist bereits in Abschnitt 2.2 eingegangen worden. Um eine doppelte Darstellung des Forschungsfeldes zu vermeiden, wird auf eine erneute Diskussion der vorgestellten Arbeiten in diesem Kapitel verzichtet (Verweis auf Abschnitt 2.2).

3.1 Angriffe basierend auf Identitätsdaten-Leaks

In der Literatur werden Methoden entwickelt, welche Credential-Stuffing-Angriffe effizienter gestalten sollen [48, 121, 84, 128, 85]. Pal u. a. entwickeln ein Verfahren, mit dem aus bereits in Identitätsdaten-Leaks enthaltenen Passwörtern weitere valide Passwörter abgeleitet werden können, die noch nicht in einem Leak enthalten sind [84]. Dazu trainieren sie mit bereits in Identitätsdaten-Leaks enthaltenen Klartextpasswörtern ein Modell, welches Vorhersagen für weitere mögliche Passwörter machen kann. Mit diesem Vorgehen gelingt es, in Simulationen 16 % von Passwort-Hash-Werten mit weniger als 1000 Versuchen zu brechen, wobei die Klartextpasswörter der Hash-Werte noch nicht in den Trainingsdaten enthalten sind [84]. Um die immer besser werdenden Credential-Stuffing-Angriffe auf Seiten eines Onlinedienstes zu erkennen, gibt es ebenfalls Untersuchungen. Diese besitzen das Ziel, Credential-Stuffing-Angriffe durch Angreifer von Falscheingaben durch legitime Benutzer zu unterscheiden [123].

Offline-Cracking von Passwort-Hashes wird ebenfalls intensiver untersucht. Die letzten Untersuchungen zeigen, dass immer noch aktuelle Hash-Verfahren wie `PBKDF2` und `BCRYPT` die Benutzerpasswörter unzureichend vor Offline-Cracking-Angriffen schützen [9]. Jedoch werden Ansätze entwickelt, die durch eine fragmentbasierte Verarbeitung von Hash-Funktionen die veralteten Hash-Verfahren *MD5* und *SHA1* sicherer gestalten sollen [11, 10], sodass diese wieder zum Passwort-Hashing eingesetzt werden können.

Alternativ zu Credential-Stuffing-Angriffen und Passwort-Hash-Cracking können mehrere Identitätsdaten-Leaks dazu genutzt werden, um mittels der Aggregation dieser Leaks neue Informationen zu gewinnen. Heen u. a. zeigen in einer Untersuchung von mehreren Identitätsdaten-Leaks, dass bis zu 8,8 % der enthaltenen Identitätsdatensätze mit Datensätzen aus anderen Leaks aggregiert werden können [45]. Beispielsweise wird eine Aggregation auf das Passwort, einen Hash-Wert oder eine E-Mail-Adresse ausgeführt. Die hieraus entstehende Deanonymisierung von Benutzern [45] könnte für weitere Credential-Stuffing-Angriffe oder aber für Social-Engineering-Angriffe genutzt werden.

3.2 Erkennung von Identitätsdaten-Breaches

Im vorherigen Abschnitt ging es um die Verbesserung von Angriffen, die auf Identitätsdaten-Leaks basieren. Um diese Angriffe zu verhindern, muss vorzugsweise der eigentliche Sicherheitsvorfall, bei dem Benutzerdaten entwendet werden, verhindert oder zumindest erkannt werden.

Um geeignete Gegenmaßnahmen für Identitätsdiebstahl zu entwickeln, ist es hilfreich, Genaueres über das Verhalten von Angreifern in Erfahrung zu bringen. Onaolapo u. a. untersuchen, wie genau Angreifer auf veröffentlichte Zugangsdaten reagieren. Ergebnis der Untersuchungen ist, dass Angreifer versuchen, ihren Zugriff auf die kompromittierten Dienste über Server zu senden, die sich in der Nähe der Opfer befinden, um auf diese Art die Systeme zur Betrugserkennung zu umgehen [83]. Kriminelle verwenden zur Verschleierung ihrer Herkunft häufig *Tor-Exit-Nodes*, um auf die Dienste zuzugreifen [83]. Ein weiteres Problem im Bereich Identitätsdiebstahl resultiert aus der Nutzung von Phishing-Websites. Es wird genauer untersucht, was mit Zugangsdaten passiert, wenn Benutzer diese irrtümlich auf Phishing-Websites eingeben. Das Ergebnis ist, dass diese Daten in Echtzeit nicht nur mit einem Angreifer geteilt werden, da häufig durch Angreifer sogenannte Phishing-Kits eingesetzt werden [87]. Weiterhin wird untersucht, wie Identitätsdaten-Leaks verbreitet werden. Es wird festgestellt, dass Identitätsdaten-Leaks über drei verschiedene Arten von Onlinediensten verbreitet werden: Diskussionsforen, Online-Marktplätze und Paste-Pages [65]. Außerdem werden Identitätsdaten-Leaks häufig gewinnbringend verkauft [65].

Um geeignete Maßnahmen einzuleiten, sobald solche Identitätsdaten abhandenkommen, werden in verschiedenen Untersuchungen Erkennungsdienste für solche Vorfälle entwickelt. Um das Abhandenkommen solcher Daten rechtzeitig zu erkennen, besteht die Möglichkeit Test-Benutzerkonten bei verschiedenen Onlinediensten anzulegen [15]. Als Identifikator für die Testkonten werden E-Mail-Adressen genutzt, deren Postfächer von einem Erkennungsdienst betrieben werden [15]. Sollte bei einem externen Dienst eine Kompromittierung der Benutzerdaten geschehen, kommen die Zugangsdaten des Testkontos abhanden. Die Idee für einen solchen Erkennungsdienst ist, dass Angreifer die neu erworbenen Zugangsdaten bei dem

E-Mail-Provider austesten [15]. Sollte eine Anmeldung am vom Erkennungsdienst betriebenen E-Mail-Postfach erfolgen, kann darauf geschlossen werden, dass der entsprechende Onlinedienst die ihm anvertrauten Zugangsdaten verloren hat [15].

Alternativ wird untersucht, wie Identitätsdaten-Breaches auf der Seite von Onlinediensten verhindert werden können. Dazu werden in einer empirischen Erhebung die Ursachen für Identitätsdaten-Breaches aus der Vergangenheit untersucht [94]. Das Ergebnis ist, dass gängige IT-Sicherheitssysteme wie *Intrusion-Detection-Systeme* und *Data-Leakage-Prevention-Systeme* ein hohes Maß an Schutz vor Daten-Leaks bieten [94]. Zusätzlich tragen sogenannte *Data-Breach-Alerting-Services* ebenfalls zum Schutz vor Daten-Leaks bei [94].

3.3 Verarbeiten von Identitätsdaten-Leaks

In dieser Arbeit wird ein Warndienst konzipiert, der Warnungen bezüglich Identitätsdaten-Leaks versenden soll. Um herauszufinden, wessen Identitätsdaten in einem Leak enthalten sind, muss dieser analysiert werden. Die in der Literatur vorgestellten Konzepte und Verfahren für eine Verarbeitung von Identitätsdaten-Leaks werden deshalb in diesem Abschnitt vorgestellt.

Bereits 2015 geben Jaeger u. a. einen ersten Überblick, über die verschiedenen Quellen für Identitätsdaten-Leaks [57]. Auch werden die Eigenschaften von Identitätsdaten-Leaks dargestellt und auf die Gefahren der Mehrfachverwendung von Passwörtern hingewiesen [58]. Aufbauend auf diesen Arbeiten wird ein Konzept für einen Parser von Identitätsdaten-Leaks vorgestellt [42, 56]. Dieses Konzept beinhaltet ein Vorgehen zur automatisierten Verarbeitung von Leak-Dateien, der Extraktion von Identitätsdaten, als auch eine Kontrolle der Leak-Authentizität [56]. Jedoch bleiben einige Umsetzungen von manchen technischen Komponenten unklar. Auch eignet sich dieses Konzept nicht dazu, um Identitätsdaten-Leaks mit häufig auftretenden Eigenschaften zu verarbeiten (die Eigenschaften von Identitätsdaten-Leaks werden später in Abschnitt 5.1 vorgestellt). Eine genauere Evaluation des implementierten Konzepts ist in der Literatur bis heute nicht zu finden. Weitere Forschungsarbeiten zur Verarbeitung von Identitätsdaten-Leaks sind in der Literatur ebenfalls nicht vorhanden.

3.4 Informationsdienste für Identitätsdaten-Leaks

Verschiedene Onlinedienste ermöglichen es Benutzern zu überprüfen, ob ihre Identitätsdaten bereits in einem Identitätsdaten-Leak enthalten sind. Zur Nutzung dieser Dienste trägt ein Benutzer meist seine E-Mail-Adresse in ein Textfeld auf der Website eines Dienstes ein. Der Dienst durchsucht die eigene Leak-Datenbank nach der eingegebenen E-Mail-Adresse. Eine Reihe von bekannten Leak-Informationsdiensten sind Folgende:

- *have i been pwned* [50]
- *HPI-Leakchecker* [44]
- *Uni-Bonn-Leak-Checker* [115]
- *Spy-Cloud* [107]
- *Avast-Hack-Check* [3]

Das Ergebnis von Anfragen erhalten die Benutzer abhängig vom jeweiligen Dienst auf verschiedenen Wegen. Bei *have i been pwned* wird einem Benutzer das Ergebnis direkt im Browser nach Eingabe einer E-Mail-Adresse angezeigt. Hierbei ist jedoch ethisch und datenschutzrechtlich bedenklich, dass jede Person diesen Dienst für E-Mail-Adressen von anderen Personen nutzen kann. Jeder ist somit in der Lage, herauszufinden, welche Person bei welchem Dienst angemeldet ist und ob Zugangsdaten abhandengekommen sind. Welche Dienste ein Benutzer verwendet, kann als Angelegenheit betrachtet werden, die zur Privatsphäre einer jeden Person zählt. Ethisch vertretbarer ist deshalb die Übermittlung der Ergebnisse per E-Mail an die eingegebene E-Mail-Adresse, so wie es die restlichen Leak-Informationsdienste umsetzen. Dadurch sind andere Personen in der Lage fremde E-Mail-Adressen in die genannten Dienste einzugeben, jedoch wird das Ergebnis nur dem vermutlich rechtmäßigen Identitätsinhaber zugestellt.

Des Weiteren werden Dienste für Unternehmen angeboten, welche über eine Schnittstelle eine automatisierte Überprüfung des Leak-Status von Benutzern des Unternehmens ermöglichen. Folgende drei bekannte Dienste bieten derzeit APIs mit der beschriebenen Funktion an: *Enzoic* [22], *Spy-Cloud* [107], *have i been pw-*

Tabelle 1: *Vergleich verschiedener Leak-Informationsdienste.*

	Überprüfung				
	E-Mail	Passwort	E-Mail + Passwort	DSGVO-konform	Warnungs-Abo
HIBP	✓	✓			
HPI-Leakchecker	✓			✓	
Uni-Bonn Leakchecker	✓			✓	
Spycloud	✓	✓			✓
Avast-Hack-Check	✓			✓	
1Password		✓		✓	
Firefox-Monitor	✓			✓	✓
Google PCe			✓	✓	

ned [50]. In Abschnitt 6.8 findet eine genaue Betrachtung dieser Dienste und der dazugehörigen Konzepte statt. An dieser Stelle kann jedoch gesagt werden, dass gerade bei den Diensten *Enzoic* und *Spycloud* die Privatsphäre von Personen nicht bestmöglich geschützt wird. Für Endanwender gibt es ebenfalls Produkte, die über die reine Website der Leak-Informationsdienste hinausgehen. Beispielsweise bietet der Passwortmanager *1Password* eine Funktion, mit der die gespeicherten Passwörter automatisch auf eine Kompromittierung überprüft werden [102]. Des Weiteren ist in den Browser *Firefox* eine Überprüfung von Passwörtern mit dem Produkt *Firefox-Monitor* integriert [78]. Sowohl *1Password* als auch der *Firefox-Monitor* greifen dabei auf die Datenbasis von *have i been pwned* zurück [102, 78]. Google bietet ebenfalls ein ähnliches Produkt für Benutzer an. Das Google-Chrome-Plugin mit dem Namen *Password Checkup extension* ermöglicht ebenfalls eine Überprüfung der eigenen Passwörter auf eine Kompromittierung [40]. Für dieses Plugin betreibt Google jedoch eine eigene Datenbank mit Identitätsdaten-Leaks [91].

Die Funktionen der vorgestellten Leak-Informationsdienste für Benutzer ist in Tabelle 3 dargestellt. Zu sehen ist, dass die meisten genannten Dienste nur eine Überprüfung der E-Mail-Adresse zulassen. Diese Dienste liefern lediglich Informa-

tionen darüber, in welchen Identitätsdaten-Leaks die abgefragte E-Mail-Adresse zu finden ist. Ändert ein betroffener Benutzer sein Passwort, dann bleibt diese Warnung weiterhin erhalten. Weitere Dienste ermöglichen eine Abfrage, ob ein Passwort bereits in einem Identitätsdaten-Leak enthalten ist. Dabei wird lediglich das Passwort ohne E-Mail-Adresse überprüft. Hilfreich ist dies, um beispielsweise bei einer Wahl eines neuen Passworts zu überprüfen, ob dies bereits generell in einem Identitätsdaten-Leak vorhanden ist. Nur das Google-Chrome-Plugin besitzt die Möglichkeit, um den Status einer Kombination aus E-Mail-Adresse und Passwort zu überprüfen. Die Dienste *Spycloud* und *Firefox-Monitor* bieten beide die Möglichkeit für Benutzer, die eigene E-Mail-Adresse zu hinterlegen und bei einer zukünftigen Betroffenheit per E-Mail informiert zu werden. Da *have i been pwned* und *Spycloud* Informationen über dritte Personen herausgeben, wird diese Eigenschaft hier als nicht DSGVO-konform gewertet. Eine konkrete juristische Untersuchung müsste diese Annahme jedoch noch belegen.

3.5 Warn-Nachricht über Identitätsdaten-Leaks

Im Jahr 2012 wurden Identitätsdaten vom Onlinedienst *LinkedIn*[1] gestohlen. Erst im Mai 2016 wurde bekannt gegeben, dass weitere Schritte zum Schutz der betroffenen Benutzer eingeleitet werden [99, 49]. Auch im Mai 2016 wurden die betroffenen Benutzer per E-Mail kontaktiert und gebeten, ihr Passwort zu ändern [49]. In einer Befragung von 249 betroffenen *LinkedIn*-Benutzern (von Juni bis September) wurde herausgefunden, dass lediglich 46 % der Befragten bis zur Befragung ihr Passwort geändert hatten [49]. Diejenigen, die ihr Passwort geändert hatten, änderten dies im Durchschnitt 26,3 Tage nach dem Erhalt der Benachrichtigungs-E-Mail [49]. Diese Ergebnisse führen zu der Schlussfolgerung, dass eine einzelne Benachrichtigung der Benutzer nicht ausreicht, um einen umfänglichen Schutz der Betroffenen zu erreichen. Schließlich ist das Ziel, dass 100 % der kompromittierten Benutzerkonten geschützt sind. Zu dieser Studie muss erwähnt werden, dass keine genauere Betrachtung der Warn-E-Mail durchgeführt wurde. Die geringe Anzahl an geänderten Passwörtern kann auch an einer nicht optimal ausgestalteten Warn-E-Mail liegen.

In einer anderen Arbeit wird die Gestaltung von Warnnachrichten untersucht. Dazu wurden Probanden in mehreren quantitativen und qualitativen Studien befragt [36]. Als Ergebnis dieser Arbeit werden fünf Anforderungen an eine Warnnachricht genannt [36]:

- Eine Warnung soll möglichst genau den Grund für die gestohlenen Passwörter nennen [36].
- Onlinedienste sollen bei betroffenen Konten einen Passwort-Reset durchführen [36].
- Es soll empfohlen werden, dieses Passwort auch bei anderen Diensten zu ändern [36].
- Auch soll die Empfehlung gegeben werden, eine Zwei-Faktor-Authentifizierung und einen Passwort-Manager zu nutzen [36].
- Eine Benachrichtigung soll per E-Mail versendet und über einen direkten Kommunikationskanal wie der Login-Webpage angezeigt werden [36].

[1] LinkedIn: https://de.linkedin.com/.

Diese ermittelten Anforderungen sollen bei der tatsächlichen Ausgestaltung einer Warn-Nachricht unterstützen. Des Weiteren werden getestete Beispieltexte für eine Warnung mittels verschiedener Kommunikationskanäle vorgestellt [36]. Die in der hier vorliegenden Arbeit vorgestellte und im Forschungsprojekt *EIDI* verwendete Warnmeldung (siehe Abschnitt A) basiert nicht auf der Untersuchung von Golla u. a. [36]. Der verwendete Warntext ist in Zusammenarbeit von den Projektmitglieder des Forschungsprojektes entstanden.

4 Sammeln von Identitätsdaten-Leaks

Dieses Kapitel basiert auf den bereits veröffentlichten Arbeiten „Sammlung geleakter Identitätsdaten zur Vorbereitung proaktiver Opfer-Warnung“ [72], „Track Down Identity Leaks Using Threat Intelligence“ [67] und „Credential Intelligence Agency - A Threat Intelligence Approach to Mitigate Identity Theft“ [66].

In den Medien wird regelmäßig über Identitätsdaten-Leaks berichtet. Die häufigste Ursache für einen solchen Leak ist das Finden und Ausnutzen von Schwachstellen in Systemen von Onlinediensten durch Angreifer, sodass zum Beispiel Benutzerdaten illegal kopiert und anschließend missbraucht werden. Welche Tätigkeiten genau mit den Daten nach einem solchen Breach durchgeführt werden, lässt sich nur vermuten. In Abschnitt 4.1 wird deshalb auf mögliche Arten der Verbreitung von Identitätsdaten-Leaks eingegangen. Dazu wird der Begriff der *Datensenke* eingeführt, um anschließend verschiedene Verbreitungsarten vorzustellen (siehe Unterabschnitt 4.1.1). Nachdem diese Einordnung abgeschlossen wird, werden die Eigenschaften von Identitätsdaten-Leaks dargestellt (siehe Unterabschnitt 4.1.2), um ein Verständnis für die Komplexität solcher Leaks zu schaffen.

Angreifer verbreiten die gestohlenen Leak-Daten, um sie anderen Kriminellen zur Verfügung zu stellen, damit das Missbrauchspotenzial der Daten vollständig ausgeschöpft wird und die größtmöglichen Gewinne erbeutet werden. Möchte man die Opfer der Identitätsdiebstähle schützen, dann kann an dieser Stelle angesetzt werden, um die veröffentlichten Daten für eine Warnung der Opfer zu sammeln. Da im Rahmen dieser Arbeit ein technisches Konzept zur Warnung der durch

Identitätsdatendiebstahl betroffenen Personen erstellt werden soll, muss auch ein Vorgehen zur Sammlung solcher Identitätsdaten entwickelt werden. Ein solches Vorgehen wird in Abschnitt 4.2 vorgestellt. Zum Abschluss dieses Kapitels werden die mithilfe dieses Vorgehens gesammelten Daten ausgewertet (siehe 4.3).

4.1 Verbreitung von Identitätsdaten-Leaks

In den vergangenen Jahren kam es teilweise mehrfach im Monat vor, dass Identitätsdaten bei Diensten medienwirksam missbräuchlich dupliziert oder entwendet wurden [75]. Im Zeitraum von 2017 bis 2019 kam es laut McCandless u. a. zu 112 Identitätsdaten-Leaks bei verschiedenen Diensten [75]. Was die Angreifer mit diesen Daten tatsächlich machen, ist ungewiss. Erbeuten Angreifer solche Daten, können sie diese Daten selbst missbrauchen, um einen finanziellen Vorteil zu erlangen [72]. Alternativ können sie die Daten auch verkaufen. Als Käufer kommen andere Kriminelle in Frage, welche die gekauften Daten zum Betrug verwenden. Jedoch kommen auch Datenhehler als Käufer in Frage, welche die Daten weiterverkaufen. Eine weitere häufig vorkommende Möglichkeit zur Verwendung der erbeuteten Daten stellt die freie Veröffentlichung dar, um die Reputation als *Hacker* in der Cyber-Kriminellen-Community zu steigern [72].

In dieser Arbeit werden nur Identitätsdaten-Leaks betrachtet, die veröffentlicht wurden. Die anderen genannten Möglichkeiten der Angreifer zur Verwertung der Identitätsdaten-Leaks werden nicht betrachtet, da aus ethischen Beweggründen keine finanziellen Mittel zum Kauf von Daten aufgewendet werden sollen. Ebenso wenig werden echte Leak-Daten in Foren zum Tausch angeboten.

Sollen Identitätsdaten-Leaks veröffentlicht werden, muss der Datensatz anderen zugänglich gemacht werden. Dazu müssen die Daten an einem Ort abgelegt werden, auf den die Interessenten des Leaks zugreifen können. Ein solcher Ort wird im Folgenden als *Datensenke* [72] bezeichnet. Im Allgemeinen ist eine Datensenke ein Speicherort, in dem beliebige Daten gespeichert werden können. Durch eine *URL* lässt sich eine Datensenke eindeutig darstellen [68]. Die Unterschiede von verschiedenen Typen von Datensenken und Leak-Daten werden in den folgenden Unterkapiteln erläutert.

4.1.1 Kategorien von Datensenken

Datensenken können in ihrer Zugänglichkeit unterschiedlich beschränkt werden. In der Zugänglichkeit lässt sich auch das erste Kriterium von Datensenken festlegen. Kann auf den Inhalt einer Datensenke ohne Zugriffsbeschränkung zugegriffen werden, ist der Inhalt öffentlich zugänglich. Deswegen wird im Folgenden eine solche Datensenke auch als eine *öffentliche Datensenke* bezeichnet [72]. Ist eine Datensenke nur einem bestimmten Personenkreis durch die Absicherung mit technischen Maßnahmen zugänglich, wird im Folgenden von einer *geschlossenen Datensenke* gesprochen. Werden technische Maßnahmen zur Absicherung der Datensenke verwendet, die mit geringem Ressourcenaufwand umgangen werden können, wird im weiteren Verlauf von *halböffentlichen Datensenken* gesprochen.

Wie bereits beschrieben, lassen sich Datensenken über eine eindeutige URL identifizieren, um die in der Senke enthaltenen Daten herunterzuladen. Allerdings muss eine solche URL zunächst identifiziert werden, bevor aus der entsprechenden Datensenke die Leak-Daten heruntergeladen werden können. Solche URLs werden in verschiedenen Quellen aufgelistet: Beispielsweise Paste-Pages, Leak-Annoucement-Pages oder Foren [72, 57].

Ein bekannter Dienst, der zur Verbreitung von Identitätsdaten-Leaks genutzt wird, ist *Pastebin*[1]. *Pastebin* ist eine sogenannte „Paste-Page“, die unter anderem zur Verbreitung von Identitätsdaten-Leaks genutzt wird [57]. Diese Art von Onlinedienst bietet die Möglichkeit beliebige Texte auf den Server des Dienstes zu laden, um diesen über eine URL anderen Personen zugänglich zu machen. Allerdings ermöglichen manche dieser *Paste-Pages* auch, sich die letzten Veröffentlichungen anderer Benutzer anzusehen. Diese Funktion kann durch eine Applikation automatisiert genutzt werden, um jeden neu veröffentlichten Textbeitrag herunterzuladen. Alternativ steht dafür unter anderem bei *Pastebin* eine API zur Verfügung. Da der Dienst jedoch nicht nur für Leak-Daten genutzt wird, muss nachträglich analysiert werden, ob es sich bei den gesammelten Daten um Identitätsdaten handelt. Solche *Paste-Pages* können automatisiert mit einem *Fetcher* durchsucht werden, sodass Daten in Datensenken automatisiert heruntergeladen werden können [72]. Bietet eine Paste-Page eine API, einen RSS-Feed oder eine Website mit festen Strukturen an, lässt sich ein automatisierter Abruf mit geringem Ressourceneinsatz umsetzen.

[1] Pastebin: https://pastebin.com/.

Eine weitere Möglichkeit zum Auffinden von geeigneten Datensenken ist die Nutzung von sogenannten *Leak-Announcement-Pages*, auf der URLs zu Datensenken mit Identitätsdaten-Leaks veröffentlicht werden [57]. Häufig werden URLs zu neuen Identitätsdaten-Leaks auch in speziellen Foren verbreitet [57, 72]. Es gibt Foren, die sich nur mit gestohlenen Daten auseinandersetzen, jedoch gibt es auch Foren, in denen gestohlene Daten nur eine Kategorie von vielen darstellen. In diesen Foren bieten Forenbenutzer Identitätsdaten-Leaks an oder bitten andere Benutzer, einen gesuchten Leak über das Forum zur Verfügung zu stellen. Zur Verteilung der Leak-Daten werden sie in eine geeignete Datensenke geladen. Die URL zu dieser Datensenke wird anschließend in einem Forenbeitrag veröffentlicht.

Eine etwas andere Möglichkeit zum Auffinden von Identitätsdaten-Leaks ist die Nutzung eines *Crawlers* oder einer geeigneten Suchmaschine. Gelegentlich werden diese Daten auch auf Webservern mit aktiviertem *Directory Listing* abgelegt. Das *Directory Listing* listet die in einem Ordner enthaltenen Dateien auf, welche auch Identitätsdaten enthalten können. Mit einem Crawler kann nach solchen Servern gesucht werden.

In der Art, wie sich die genannten Datensenken identifizieren lassen, gibt es zwei grundlegende Unterschiede. Manche dieser Datensenken sind aufgrund der technischen Struktur so geschaffen, dass sie sich mit einer geeigneten Applikation automatisiert auffinden lassen und der in der Datensenke enthaltene Inhalt geladen werden kann. Diese Datensenken werden als *automatisiert identifizierbare Datensenken* bezeichnet [72]. Die Quellen, die für eine automatisierte Analyse zum Auffinden von Datensenken genutzt werden, zeichnen sich durch feste und gleichbleibende Strukturen aus. Dies ermöglicht mit geringen Ressourcen eine Implementierung einer Applikation zum automatisierten Auffinden von Identitätsdaten-Leaks.

Wenn die zu analysierenden Quellen keine ausreichenden Strukturen aufweisen, ist die Automatisierung des Auffindens der Datensenken mit einem zu umfangreichen Aufwand verbunden. In diesem Fall eignet sich ein manueller Ansatz. Diese *manuell zu identifizierenden Datensenken* müssen durch einen Analysten in einem Recherchevorgang identifiziert und der Inhalt geladen werden [72]. Beispielsweise muss in Foren häufig mit natürlicher Sprache interagiert werden. Viele Quellen

wechseln häufig ihre Strukturen oder werden abgeschaltet und an anderer Stelle mit anderen Systemen wieder angeboten. Diese technischen Veränderungen sorgen dafür, dass ein automatisierter Ansatz mit zu umfangreichem Ressourceneinsatz verbunden wäre.

Die Speicherung und Zugänglichkeit zu Datensenken können mit verschiedenen Lösungen umgesetzt werden. *File-Hosting-Provider* sind Onlinedienste, auf deren Server beliebige Dateien hochgeladen werden können, welche im Nachgang über eine Webpage wieder heruntergeladen werden können. Mithilfe eines solchen Dienstes lassen sich beliebige Datenmengen oft anonym mit anderen Personen teilen. Deswegen wird diese Art von Dienst häufig als Datensenke für die Verbreitung von Identitätsdaten-Leaks verwendet [57].

Eine weitere Möglichkeit, die zur Verbreitung von Identitätsdaten-Leaks genutzt wird, ist die Verwendung von dezentralen Netzwerken zum Teilen von Dateien wie *BitTorrent* oder *Usenet* [57]. *BitTorrent* ist ein File-Sharing-Protokoll, mit dem in einem Peer-to-Peer-Netzwerk Dateien geteilt werden können. Auch das *Usenet* wird für die Verbreitung von Identitätsdaten-Leaks genutzt [72]. Alternativ kommen auch reguläre Webserver als Datensenke in Frage. Manche *Leak-Announcement-Pages* und Foren legen die Leak-Dateien nicht in externen Datensenken ab, sondern binden sie in die eigenen Inhalte ein. Weitere Verbreitungswege sind denkbar, im *EIDI-Projekt* wurden jedoch nur die genannten Arten zur Leak-Beschaffung verwendet. Es sei erwähnt, dass manche Dienste nicht über das reguläre *Clear-Web* erreichbar sind, sondern nur über das Anonymisierungs-Netzwerk *Tor*[2].

4.1.2 Veröffentlichung und Inhalt von Identitätsdaten-Leaks

Sind bei einem Unternehmen Benutzerdaten abhandengekommen, ist ein Identitätsdaten-Leak entstanden. Angreifer, welche diese Daten entwendet haben, besitzen mehrere Möglichkeiten zur Verwendung dieser Daten. Sie können die Daten selbst für Straftaten missbräuchlich verwenden oder sie veröffentlichen. Werden die missbräuchlich entwendeten Daten zum ersten Mal veröffentlicht, kann von ei-

[2] The Tor Project: https://www.torproject.org/.

ner *primären Veröffentlichung* gesprochen werden. Vor dieser Veröffentlichung waren die in dem Leak enthaltenen Daten für die Öffentlichkeit unbekannt. So bekommen auch weitere Kriminelle potenziell Zugriff auf zuvor abgesicherte Benutzerkonten, woraus eine signifikante Bedrohung für die betroffenen Benutzer resultiert.

Allerdings kommt es auch vor, dass von einem primär veröffentlichten Leak erstmal nur eine geringe Bedrohung ausgehen kann. Beispielsweise liegt dieser Fall vor, wenn in den veröffentlichten Daten nur ein Identifikator und ein Passwort-Hash pro Datensatz enthalten sind. Kriminelle können durch einen solchen Leak nicht auf die betroffenen Benutzerkonten zugreifen, da ihnen das Klartextpasswort fehlt. Es existiert aber eine ganze Community, die sich mit dem Brechen von Passwort-Hashes auseinandersetzt[3], mit dem Ziel die Wertigkeit von Leak-Daten zu erhöhen. Gelingt der Community einen Teil an Klartextpasswörtern aus den Passwort-Hashes eines Leaks zu rekonstruieren, wird der Leak in einer überarbeiteten Version mit Klartextpasswörtern erneut veröffentlicht. Die Bedrohung der betroffenen Benutzer wird durch diese Überarbeitung erhöht. Zusätzlich können Identitätsdaten-Leaks mit anderen Daten aus sozialen Netzwerken oder weiteren Leaks aggregiert werden, wodurch die Gefährdung für die Betroffenen ansteigt.

Regelmäßig werden viele meist ältere Identitätsdaten-Leaks gemeinsam in einer großen Sammlung erneut veröffentlicht. Diese Sammlungen werden als *Collections* bezeichnet und fassen mehrere Identitätsdaten-Leaks als eine Sammlung zusammen. Es wurde beobachtet, dass solche Sammlungen mehr als 1 Milliarde Datensätze enthalten können. Die bisher größten und medial prominentesten Identitätsdaten-Leak-Sammlungen sind aus dem Jahr 2019 mit den Namen *Collection #1*, *Collection #2*, *Collection #3*, *Collection #4*, *Collection #5* [20]. Häufig enthalten *Collections* nur Benutzernamen oder E-Mail-Adressen und zusätzlich ein Klartextpasswort. Die Kombination von Identifikator und Klartextpasswort wird auch als *Combo* bezeichnet [57]. Bei manchen dieser Sammlungen kommt es vor, dass nicht mehr nachvollzogen werden kann, zu welchen Diensten die enthaltenen Identitätsdaten gehören. Auch

[3] Beispielsweise: GPUHASH.me: https://gpuhash.me/, CrackStation: https://crackstation.net/, OnlineHashCrack: https://www.onlinehashcrack.com/, Hashes.org: https://hashes.org/, FAST LM HASH ONLINE CRACKING: http://rainbowtables.it64.com/ (Alle abgerufen am 15.02.2020).

können in diesen Sammlungen Identitätsdaten enthalten sein, die nicht von einem Dienst stammen, sondern bei den Benutzern mittels Malware entwendet wurden [57]. Beispielsweise kann eine Malware sämtliche Login-Eingaben bei Onlinediensten im Browser auf dem kompromittierten System des Benutzers aufzeichnen und an einen Angreifer versenden.

4.2 Prozess

Das in dieser Arbeit konzipierte Warnsystem zum Schutz vor Identitätsdatendiebstahl benötigt Identitätsdaten-Leaks, um die Personen, deren personenbezogenen Daten in diesen Leaks enthalten sind, warnen zu können. Um ein solches System mit aktuellen Daten befüllen zu können, wird ein Konzept zum Sammeln solcher Daten benötigt.

4.2.1 Konzept zum Sammeln der Identitätsdaten-Leaks

Die in den vorherigen Abschnitten dargestellten Sachverhalte werden in diesem Kapitel in einen Prozess überführt, der für das effiziente Sammeln von Identitätsdaten-Leaks geeignet ist. Der Prozess [72] ist in Abbildung 2 dargestellt und besteht im Wesentlichen aus zwei Hauptkomponenten, die für das eigentliche Sammeln der Identitätsdaten verantwortlich sind.

Einerseits ist dies der *Fetcher*, der automatisiert Identitätsdaten-Leaks sammelt. Der Fetcher besteht aus einzelnen Teilapplikationen, bei der jede Teilapplikation für das Crawlen bestimmter Quellen zuständig ist. Jede dieser Teilapplikationen wurde dafür entwickelt, dass sie eine bestimmte Website analysiert und nach neuen Datensenken durchsucht. Wird eine neue Datensenke gefunden, werden die Daten in einem anschließenden Prozess heruntergeladen und extrahiert.

Andererseits gibt es den *Analysten*, der geeignete Quellen manuell durchsucht, um dort Identitätsdaten-Leaks herunterzuladen. Dazu durchsucht er themenspezifische Foren und Leak-Announcement-Pages. Findet der Analyst einen Hinweis auf einen

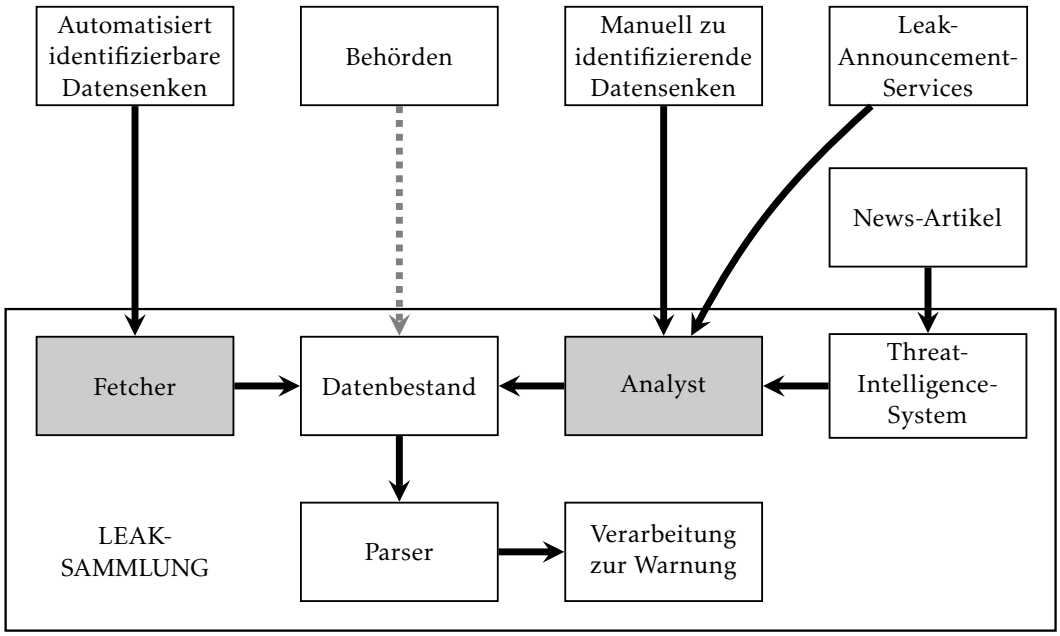


Abbildung 2: *Prozess zum manuellen und automatisierten Sammeln von Identitätsdaten-Leaks (Erweiterung aus [72]).*

relevanten Identitätsdaten-Leak, ruft er die Datensenke ab, um den enthaltenen Leak herunterzuladen.

Die Abbildung 2 zeigt, dass der *Fetcher* die *automatisiert identifizierbaren Datensenken* nach Identitätsdaten-Leaks durchsucht und bei einem erfolgreichen Fund die Daten in den Datenbestand verschiebt. Ähnlich stellt sich dies bei dem *Analysten* dar. Jedoch lassen sich in aller Regel die *manuell zu identifizierenden Datensenken* nicht mit einem einheitlichen Prozess auffinden. Deswegen wird zur effektiven Suche von Identitätsdaten-Leaks ein Experte benötigt, der auf die stark heterogenen Strukturen der zu durchsuchenden Quellen eingehen kann. Zur Recherche von neuen Leaks nutzt der *Analyst* geeignete *Leak-Announcement-Services* wie Foren oder direkt Leak-Announcement-Pages. Besitzt der Analyst Kenntnis darüber, bei welchen Onlinediensten aktuell Identitätsdaten missbräuchlich entwendet wurden, kann er gezielt nach diesem Identitätsdaten-Leak suchen. Das Verfolgen von aktuellen Nachrichten ist jedoch wenig effizient, da der *Analyst* aus einer großen Menge

an Artikeln die für ihn relevanten herausfiltern muss. Deswegen wird in Abschnitt 4.2.2 ein *Threat-Intelligence-System* vorgestellt, welches dem *Analysten* nur relevante Berichterstattung über aktuelle Leaks anzeigt.

Die von dem *Fetcher* und dem *Analysten* gesammelten Daten werden zusammen in einen unverarbeiteten *Datenbestand* überführt, der im nachfolgenden Prozess genauer analysiert und normalisiert wird. Das System zur Analyse der Leak-Daten wird als *Parser* bezeichnet und in Kapitel 5 genauer vorgestellt. Als Ergänzung ist denkbar, dass *Behörden* wie das Bundeskriminalamt ein Interesse daran haben, selbst beschlagnahmte Identitätsdaten-Leaks einem solchen hier beschriebenen Warnsystem zur Verfügung zu stellen, um betroffene Personen zu warnen.

4.2.2 Threat-Intelligence-Service für Identitätsdaten-Leaks

Die im Rahmen dieser Arbeit entstandenen Veröffentlichungen „Track Down Identity Leaks Using Threat Intelligence“ [67] und „Credential Intelligence Agency - A Threat Intelligence Approach to Mitigate Identity Theft“ [66] stellen ein Threat-Intelligence-System vor, welches auf aktuelle Identitätsdaten-Leaks hinweist. Dieses Threat-Intelligence-System basiert darauf, dass journalistische Fachbeiträge aus dem Bereich der IT-Sicherheit häufig zeitnah über neue Identitätsdaten-Leaks berichten. Im Folgenden wird der Kontext als auch das Konzept dieses Systems vorgestellt.

Auf IT-Sicherheit spezialisierte Journalisten berichten häufig zeitnah von den neusten Identitätsdaten-Leaks. Der Diebstahl von Identitätsdaten ist aber nur ein Thema von vielen, über das auf Nachrichtenplattformen mit dem Fokus auf IT-Sicherheit berichtet wird. Liest ein Analyst einen Bericht über einen neuen Vorfall des Identitätsdatendiebstahls, kann er gezielt nach diesem Leak recherchieren. Ein Analyst stößt auch ohne solche Berichte auf neue Identitätsdaten-Leaks, allerdings ist es effektiver, wenn möglichst viele Personen wachsam das Geschehen der Kriminellen verfolgen. Ein Analyst kann sich zu Nutze machen, dass Journalisten genau dieses Geschehen beobachten und darüber berichten. Demnach ist es sinnvoll, der Berichterstattung von möglichst vielen Fachjournalisten zu folgen. Jedoch fällt dabei eine Menge an thematisch irrelevanten Artikeln an, die ein Analyst manuell aussortieren müsste. Aus diesem Grund bedarf es eines Systems, welches aus der

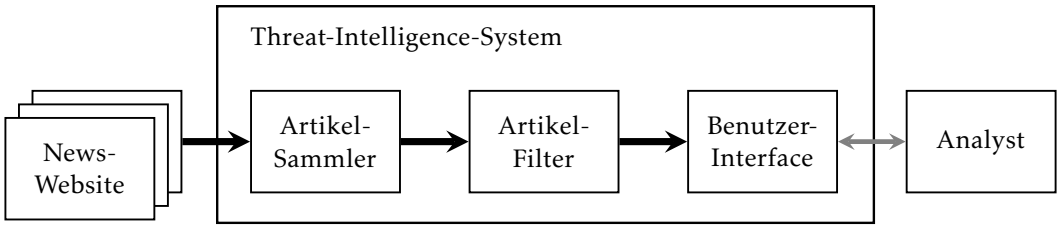


Abbildung 3: *Aufbau des Threat-Intelligence-Systems zur Meldung neuer Identitätsdaten-Leaks.*

Menge der gesamten Berichterstattung nur die für den Analysten relevanten Artikel herausfiltert. Die Grundidee des Systems besteht darin, die Inhalte von englischsprachigen Nachrichten-Websites herunterzuladen und diese mittels maschineller Lernverfahren zu klassifizieren. Ein Klassifikator bewertet dazu, ob ein Nachrichtenartikel über einen aktuellen Identitätsdaten-Leak berichtet. Liegt dieser Fall vor, wird der Artikel dem Analysten angezeigt, sodass er weiß, dass es einen neuen Leak gibt, nach dem er suchen sollte.

Das System unterteilt sich in drei Teilsysteme: *Artikelsammler*, *Artikelfilter* und *Benutzerinterface*. Diese drei Teilsysteme sind in Abbildung 3 dargestellt. Der **Artikelsammler** hat die Aufgabe, alle neuen Artikel von vorgegebenen Quellen herunterzuladen. Als geeignete Quellen wurden 15 englischsprachige Nachrichten-Websites ausgewählt, die vorzugsweise über IT-sicherheitsrelevante Themen berichten. Diese Websites sind: *Comodo*[4], *GBHackers*[5], *HackRead*[6], *Help Net Security*[7], *Infosecurity-Magazine*[8], *Security Gladiators*[9], *Security Week*[10], *Techworm*[11], *The Hacker News*[12],

[4]Comodo: https://blog.comodo.com.
[5]GBHackers: https://gbhackers.com/category/data-breach/.
[6]HackRead: https://www.hackread.com/hacking-news.
[7]Help Net Security: https://www.helpnetsecurity.com/view/news/.
[8]Infosecurity-Magazine: https://www.infosecurity-magazine.com/news/.
[9]Security Gladiators: https://securitygladiators.com/internet-security-news/.
[10]Security Week: https://www.securityweek.com.
[11]Techworm: https://www.techworm.net.
[12]The Hacker News: https://thehackernews.com.

Threat Post[13], *The Guardian*[14], *Information Week*[15], *Naked Security*[16], *Trendmicro*[17], *Cyberdefense Magazine*[18]. Englischsprachige Dienste werden genutzt, da die verfügbaren Werkzeuge zur Textvorverarbeitung für die englische Sprache solider funktionieren und zusätzlich eine größere Auswahl an relevanten Nachrichten-Websites existiert. Zum Herunterladen der Artikel werden zwei unterschiedliche Ansätze eingesetzt. Wenn die jeweilige Website einen RSS-Feed anbietet, wird dieser abonniert. Aus diesem Feed werden die einzelnen Artikel durch den **Artikelsammler** exportiert und jeder dieser Artikel in Titel, Autor, Veröffentlichungsdatum und den eigentlichen Text zerlegt. Vorteilhaft an diesem Ansatz ist, dass sich RSS-Feeds mit geringen Ressourcen automatisiert verarbeiten lassen. Bietet eine Website keinen RSS-Feed an, wird unter Einbindung des Frameworks *Scrapy*[19] ein Website-spezifischer Crawler entwickelt, der die Website automatisiert nach neuen Artikeln durchsucht. Die relevanten Attribute wie Titel, Autor, Datum und der eigentliche Text werden mithilfe des Frameworks *newspaper3k*[20] aus der reinen HTML-Struktur extrahiert, die auf diese Weise heruntergeladen wurde. Im initialen Durchlauf (September 2018) des *Artikelsammlers* wurden insgesamt 52.382 Artikel aus den 15 Nachrichten-Websites extrahiert [67]. In einer Wiederholung des Durchlaufs (Juni 2020) wurden 55.742 Artikel geladen [66]. Der älteste geladene Artikel ist aus dem Jahr 2002 [66].

Die zeitliche Verteilung der Veröffentlichungen der Artikel ist in Abbildung 4 dargestellt. Vom Jahr 2002 bis 2011 steigt die Anzahl der veröffentlichten Artikel an, was daran liegt, dass in diesem Zeitraum die 15 Nachrichten-Websites online gestellt wurden und ihre ersten Artikel veröffentlicht haben. Wird der Zeitraum von 2011 bis heute betrachtet, so werden im Durchschnitt 465 Artikel pro Monat veröffentlicht.

[13] Threat Post: https://threatpost.com/blog/.
[14] The Guardian: https://www.theguardian.com/international/.
[15] Information Week: https://www.informationweek.com/.
[16] Naked Security: https://nakedsecurity.sophos.com/.
[17] Trendmicro: https://www.trendmicro.com/vinfo/us/security/news/.
[18] Cyberdefense Magazine: http://www.cyberdefensemagazine.com/category/news/.
[19] Scrapinghub: *Scrapy*, https://scrapy.org/.
[20] Lucas Ou-Yang: *Newspaper3k: Article scraping & curation*, https://github.com/codelucas/newspaper.

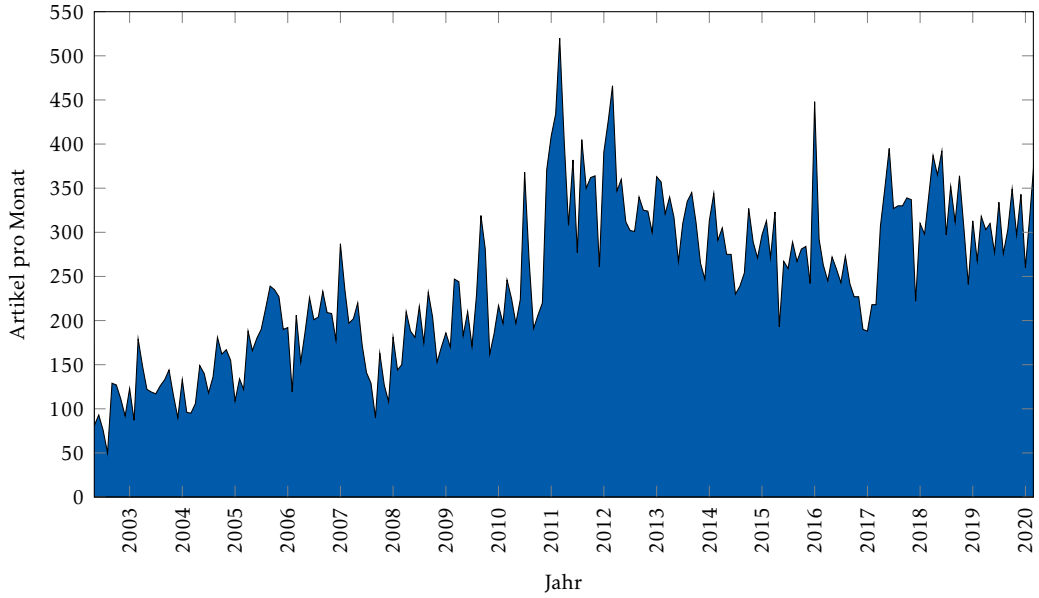


Abbildung 4: *Gesammelte News-Artikel mit dem Artikelsammler - Verteilung nach Veröffentlichungsdatum der Artikel [66].*

Die geladenen Artikel werden in einer Datenbank gespeichert, damit sie im nachfolgenden Schritt von dem *Artikelfilter* genauer analysiert werden können. Der *Artikelfilter* hat die Aufgabe, die für den Analysten relevanten Artikel herauszusuchen. Relevante Artikel sind solche, die über Geschehnisse und Aspekte berichten, die in einem Zusammenhang mit der Thematik des Identitätsdatendiebstahls stehen. Der genaue Prozess [66] ist in Abbildung 5 detailliert dargestellt.

Die Filterung der relevanten Artikel geschieht mittels eines trainierten Klassifikators. Als Klassifikator wird eine *Support Vector Machine (SVM)* genutzt, da diese sich im Bereich des *Natural Language Processsing* als erprobtes Verfahren herausgestellt hat [12]. Um einen solchen Klassifikator zu erstellen, muss zunächst ein Test- und Trainings-Set erstellt werden. Insgesamt wurden 15.217 Artikel manuell klassifiziert [67]. Von diesen 15.217 Artikeln haben 1.996 Artikel über ein Thema mit Zusammenhang zum Identitätsdatendiebstahl berichtet [67].

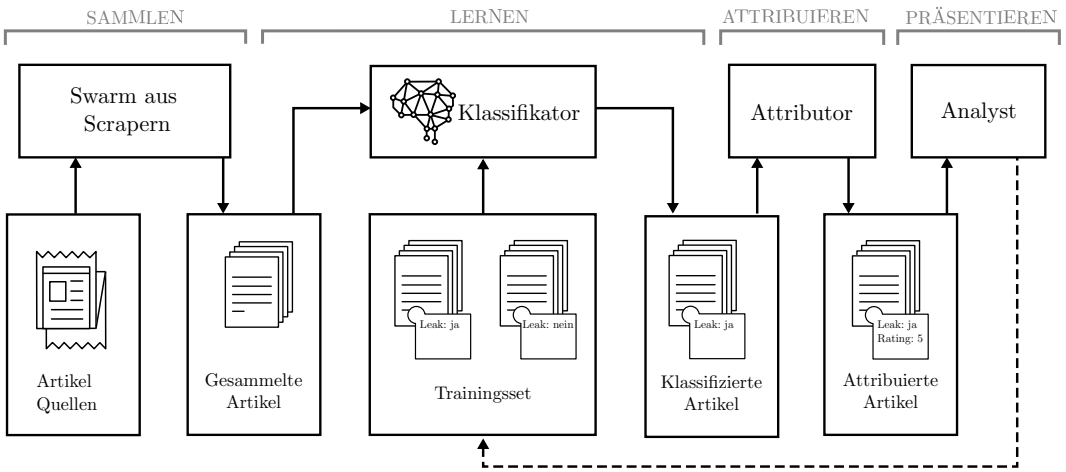


Abbildung 5: *Prozess zum Sammeln, Filtern, Attribuieren und Präsentieren der relevanten Artikel [66].*

Dazu wird zunächst jeder Artikel einem *Preprocessing* unterzogen, bei dem der natürlich-sprachliche Text für die Verarbeitung mit einem Algorithmus zum maschinellen Lernen vorbereitet wird. In der Literatur sind viele verschiedene Schritte zu finden, die für ein Preprocessing genutzt werden [119, 100, 24, 61, 73]. Dazu werden mehrere Funktionen verwendet, die in unterschiedlicher Weise den Text verändern und in einen numerischen Vektorraum überführen. In vielen Fällen fehlt in den wissenschaftlichen Arbeiten eine Evaluation des Preprocessings, um sicherzustellen, dass die verwendeten Schritte des Preprocessings tatsächlich für den Einsatzzweck geeignet sind. Aus diesem Grund werden in [66] die verschiedenen Methoden für das Preprocessing und deren Kombination in insgesamt 432 Experimenten untersucht. Als Ergebnisse dieser Experimente zeigt sich, dass keine Konjunktionen aus dem Text entfernt werden sollten [66]. Es sollte auch keine Lemmatisierung und auch kein Stemming verwendet werden [66]. Es sollten Satzzeichen entfernt und Zahlen ersetzt werden [66]. Zur Auswahl der geeignetsten Features sollte das Verfahren *RDC* [92] verwendet werden [66]. Um die Features zu gewichten, sollte kein TF-IDF, sondern die reine Anzahl genutzt werden [66]. Mit diesem Modell erhält man folgende Performanceindikatoren: Recall 0,867 — Precision 0,668 — F_1 0,754 [66]. Die erhaltenen Precision- und Recall-Werte können in dem Anwendungskontext nicht als gleichwertig betrachtet werden. Wird ein Artikel fälschlicherweise als

Berichterstattung über einen Leak eingestuft, so wird dem Analysten dieser Artikel angezeigt. Der manuelle Aufwand, um die falsche Klassifikation festzustellen, ist gering. Viel wichtiger ist, dass möglichst viele Artikel, die über Leaks berichten, auch als solche klassifiziert werden. Werden Artikel nicht als eine Berichterstattung über Leaks klassifiziert, berichten jedoch über solche, dann besteht die Gefahr, dass der Analyst von einer relevanten Meldung nichts erfährt. Der Recall gibt genau diesen Anteil an. Er repräsentiert den Anteil der Artikel, die über Leaks berichten und auch als solche klassifiziert werden. Da der Recall mit 0,867 angegeben ist, werden somit 86,6 % aller über Leaks berichtende Artikel auch als solche erkannt und dem Analysten präsentiert. Über einen Leak wird nicht nur von einem Journalisten in einem Artikel berichtet. In der Regel berichten mehrere Nachrichtenagenturen über den gleichen Leak. Wird ein relevanter Artikel falsch klassifiziert, so ist die Wahrscheinlichkeit vorhanden, dass weitere Artikel anderer Nachrichtenagenturen über den gleichen Vorfall berichten, die dem Analysten angezeigt werden.

4.3 Zusammenfassung und Leistungsbewertung des Sammlungsprozesses

Mithilfe dieses Ansatzes konnten von März 2017 bis Mai 2020 insgesamt 604 Leaks mithilfe des manuellen Ansatzes gesammelt werden. Diese Daten haben eine Größe von 1.067 Gigabyte. In Abbildung 6 sind die zehn Identitätsdaten-Leaks mit den meisten enthaltenen Identitäten dargestellt. Zu sehen ist, dass in den Top 10 der größten gesammelten Identitätsdaten-Leaks nicht nur Collections wie *BigDB*, *Collection#1-2;4-5*, *Breach Compilation* und *Anti Public* enthalten sind. Die Dienste *MySpace*, *LinkedIn* und *Adobe* sind ebenfalls mit großen Identitätsdaten-Leaks vertreten. Die gesammelten Identitätsdaten-Leaks bestehen aus 84.802 Dateien, die in 3.752 verschiedenen Ordnern und Unterordnern abgelegt sind.

Der automatisierte Ansatz lieferte von April 2017 bis Juli 2017 Identitätsdaten-Leaks mit insgesamt 8,5 Millionen E-Mail-Adressen [72]. Mit dem automatisierten Ansatz konnten deutlich weniger Daten gesammelt werden als mit dem manuellen Ansatz. Einige der in [72] als Datenquellen verwendeten Dienste stellten während

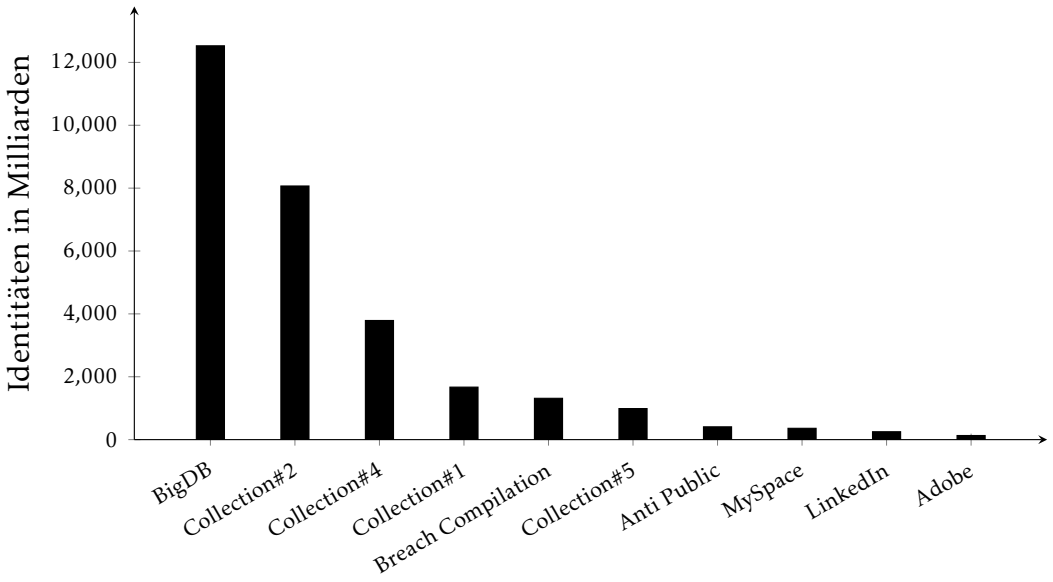


Abbildung 6: *Top 10 der Identitätsdaten-Leaks mit den meisten enthaltenen Identitäten.*

der Erstellung der hier vorliegenden Arbeit ihren Betrieb ein. Da einige automatisiert abgefragten Dienste ihren Betrieb eingstellt haben und weil der automatisierte Ansatz deutlich weniger Daten liefert als der manuelle Ansatz, wird im weiteren Verlauf dieser Arbeit der Fokus auf den manuellen Ansatz zum Sammlen von Identitätsdaten-Leaks gelegt.

5 Extraktion von Identitätsdaten

Dieses Kapitel basiert auf den bereits veröffentlichten Arbeiten „Ein Werkzeug zur automatisierten Analyse von Identitätsdaten-Leaks“ [68] und „Gathering and Analyzing Identity Leaks for a proactive Warning of affected Users“ [70].

Nachdem im vorherigen Kapitel ein Vorgehen zur Sammlung von Identitätsdaten-Leaks dargestellt wurde, steht nun die Frage im Mittelpunkt, mit welchem Verfahren die gesammelten Identitätsdaten automatisiert analysiert werden müssen, sodass auf Grundlage der Analyseergebnisse Warnungen an die Betroffenen herausgegeben werden können. Zur Beantwortung dieser Frage wird in diesem Kapitel ein Verfahren konzipiert, mit dem Identitätsdaten-Leaks vollautomatisiert analysiert und normalisiert werden können. Mithilfe dieses Verfahrens sollen einzelne Identitätsmerkmale abhängig von der Syntax und der Semantik aus einem Identitätsdaten-Leak extrahiert werden. Dazu müssen strukturierte Daten mit unbekannter und wechselnder Struktur analysiert und aufbereitet werden. Um die Komplexität dieser unbekannten Strukturen von Identitätsdaten-Leaks zu verstehen, wird zunächst auf den grundlegenden Aufbau von Identitätsdaten eingegangen (Kapitel 5.1). Dort wird auf die Heterogenität von Identitätsdaten-Leaks eingegangen, um zu zeigen, dass für die Verarbeitung solcher Leaks keine Standard-Software eingesetzt werden kann. Anschließend wird ein Konzept für einen Parser vorgestellt, der trotz der komplexen Eigenschaften von Identitätsdaten-Leaks dazu geeignet ist, um aus diesen Daten die einzelnen Datensätze zu extrahieren. Abschließend wird die Funktion des konzipierten Werkzeuges evaluiert.

5.1 Grundlegender Aufbau von Identitätsdaten-Leaks

Identitätsdaten-Leaks sind Sammlungen von Identitätsdaten, die in einer Datei zusammengefasst gespeichert sind. In solchen Leaks sind die unterschiedlichsten Identitätsdaten enthalten, zum Beispiel: E-Mail-Adressen, Passwörter im Klartext oder als Hash, Benutzername, Geburtsdatum, Vorname, Nachname, postalische Anschrift, sexuelle Orientierung, Kreditkartennummern oder IP-Adressen.

Identitätsdaten-Leaks sind Textdateien, in denen die Identitätsdaten meistens als CSV, seltener im SQL-Format, abgespeichert werden [72, 57]. Das **CSV-Format** steht für *Comma-Separated-Values*. Bei diesem Format wird pro Zeile ein Datensatz gespeichert, wobei jeder Datensatz aus mehreren Feldern bestehen kann. Die einzelnen Felder in einem Datensatz werden laut RFC durch ein Komma getrennt [101]. Die in Abschnitt 4.3 gesammelten Daten besitzen folgende Dateiendungen, die teilweise auf die enthaltene Datenstruktur hinweisen:

- Dateiendung `txt`: 93,66 %
- Dateiendung `csv`: 0,48 %
- Dateiendung `sql`: 0,10 %
- Website Dateiendungen wie `html, css, js, . . .`: 2,01 %
- Numerische Dateiendungen: 0,02 %
- Restliche Dateien: 0,56 %

Die Dateien mit der am häufigsten vorkommenden Dateiendung `txt` beinhaltet in den manuell begutachteten Fällen hauptsächlich Formate, die unter dem CSV-Format eingeordnet werden können. Zu sehen ist, dass komplexe Formate wie Datenbank-Dumps nur in einem sehr geringen Maß vorkommen. Aufgrund dieser Erkenntnis wird sich im weiteren Verlauf dieser Arbeit auf das Verarbeiten der Dateien mit `txt`-Dateiendung konzentriert.

In der Praxis werden häufig andere Zeichen zur Trennung einzelner Felder verwendet, als es durch den RFC definiert ist. Gerade bei Identitätsdaten-Leaks werden viele unterschiedliche Trennzeichen verwendet. Dort wird teilweise statt eines einzelnen Zeichens eine Zeichenkette bestehend aus mehreren Zeichen als Trenn-

```
1  melanie.mueller@web.de:sonne32!g-//-Memu32;23.02.1981
2  pascal.joee@yahoo.com:fJhe&2kc.ae23-//-DiamandJ;04.08.1975
3  seb-johansen@gmx.com:ForrestGump<3-//-Seb_87;15.11.1987
4  magic-manic99@gmail.com:qwer1234!-//-Carlo99;12.12.1969
5  jennifer-j.jacobs@icloud.com:password123!-//-JJJacobs;30.08.1972
```

Listing 5.1: *Beispielhafte Darstellung eines fiktiven Identitätsdaten-Leak mit verschiedenen Trennzeichen in einer Zeile.*

```
1  melanie.mueller@web.de:sonne32!g:Memu32:23.02.1981
2  pascal.joee@yahoo.com:fJhe&2kc.ae23:DiamandJ:04.08.1975
3  seb-johansen@gmx.com:ForrestGump<3:Seb_87:15.11.1987
4  magic-manic99@gmail.com:qwer1234!:Carlo99:12.12.1969
5  jennifer-j.jacobs@icloud.com:password123!:JJJacobs:30.08.1972
6  marc.meier@gmx.net;;mm123!1992;;MisterMarc;;13.09.1992
7  costa.n@architect-storm.com;;h!l=fje!83fa;;CostaN;;12.07.1957
8  sonja-sun@googlemail.com;;sunnysweetdream;;SunnySonja;;15.07.1985
9  jamesthomson@hotmail.com;;JT1964!UhGn;;;;
10 christina-kingston@epost-center.org;;abcdef123456;;Chrissy3;;05.03.1994
```

Listing 5.2: *Beispielhafte Darstellung eines fiktiven Identitätsdaten-Leak mit verschiedenen Blöcken.*

zeichen verwendet. Es kommt vor, dass die verschiedenen Felder pro Zeile in einer Datei durch unterschiedliche Trennzeichen separiert werden [72]. Dies ist in Listing 5.1 exemplarisch dargestellt. Die in diesem fiktiven Beispiel-Leak verwendeten Trennzeichen lauten: (1) „:“ (2) „-//-“ (3) „;“.

Jedoch gibt es auch Leaks, in denen in einzelnen Abschnitten die gleichen Trennzeichen genutzt werden, diese dann aber abschnittsweise wechseln. Durch die Verwendung unterschiedlicher Trennzeichen in verschiedenen Abschnitten entstehen zusammengehörige Blöcke mit unterschiedlichen Trennzeichen. Dies ist in Listing 5.2 dargestellt. Zu sehen ist ein Beispiel-Leak, bei dem der erste Block von Zeile eins bis fünf zu finden ist. In diesem Block wird das Trennzeichen „:“ genutzt. In Zeile sechs wechselt das Trennzeichen zu „;;“. Dieser Block verläuft anschließend von der sechsten bis zur zehnten Zeile. Eine Besonderheit ist in Zeile neun dargestellt. Dort fehlen die Inhalte der Felder *Benutzername* und *Geburtsdatum*. Diese Problematik, dass einzelne Zeilen eine andere Syntax aufweisen als die umgebenden Zeilen, kommt regelmäßig vor.

```
1  melanie.mueller@web.de:sonne32!g:Memu32:23.02.1981
2  pascal.joee@yahoo.com:fJhe&2kc.ae23:DiamandJ:04.08.1975
3  seb-johansen@gmx.com:ForrestGump<3:Seb_87:15.11.1987
4  magic-manic99@gmail.com:qwer1234!:Carlo99:12.12.1969
5  jennifer-j.jacobs@icloud.com:password123!:JJJacobs:30.08.1972
6  MisterMarc:marc.meier@gmx.net:mm123!1992:13.09.1992
7  CostaN:costa.n@architect-storm.com:h!l=fje!83fa:12.07.1957
8  SunnySonja:sonja-sun@googlemail.com:sunnysweetdream:15.07.1985
9  Jamie:jamesthomson@hotmail.com:JT1964!UhGn:16.04.1957
10 Chrissy3:christina-kingston@epost-center.org:abcdef123456:05.03.1994
```

Listing 5.3: *Identitätsdaten-Leak mit vertauschten Attributen.*

Als Trennzeichen werden häufig folgende Zeichen verwendet: Doppelpunkt, Semikolon, Komma, Tab, \r, Leerzeichen und Pipe [72]. Wie schon zuvor beschrieben, sind Kombinationen mehrerer Zeichen als Trennzeicheneinheit möglich. Dann werden öfters folgende Kombinationen gewählt: „-||-“, „//“, „;;“.

Eine weitere Problematik bei solchen Leaks ist, dass sich die Reihenfolge der Attribute in den Zeilen innerhalb eines Leaks verändern kann. Dies ist in Listing 5.3 abgebildet. Zu sehen ist, dass in den ersten fünf Zeilen des Beispiels die E-Mail-Adresse als erstes Attribut in jeder Zeile vorkommt. Ab Zeile sechs jedoch wird das Attribut *Benutzername* an erster Stelle einer jeden Zeile gesetzt. Der Positionswechsel von Attributen innerhalb eines Identitätsdaten-Leaks stellt die Verarbeitung vor Herausforderungen. Einzelne Identitätsmerkmale lassen sich zuverlässig durch ihre Syntax erkennen. Beispielsweise besitzen E-Mail-Adressen eine definierte Syntax, die mit regulären Ausdrücken detektiert werden kann. Andere Identitätsmerkmale lassen sich allerdings nicht über syntaktische Eigenschaften unterscheiden. Diese Problematik liegt unter anderem bei Benutzernamen und Passwörtern vor. Ganz allgemein können beide Merkmale aus einer beliebig langen Zeichenkette bestehen, die wiederum aus beliebigen Zeichen besteht.

Um die verwendeten Trennzeichen in einem Leak automatisiert zu erkennen und den gesamten Leak anschließend automatisiert verarbeiten zu können, müssen zunächst die Eigenschaften von Identitätsdaten-Leaks dargestellt werden, die bei einer Implementierung eines Parsers beachtet werden müssen. Aus den vorherigen Darstellungen lassen sich Eigenschaften von Identitätsdaten-Leaks ableiten, mit denen

ein automatisiertes Werkzeug umgehen können muss. Folgende Eigenschaften konnten durch eine manuelle Inspektion von mehreren hundert Identitätsdaten-Leaks festgestellt werden:

1. Keine Verwendung von einheitlichen Trennzeichen.
2. Trennzeichen können aus mehreren Zeichen bestehen.
3. Verwendung von verschiedenen Trennzeichen innerhalb einer Zeile.
4. Sonderzeichen werden am häufigsten für Trennzeichen verwendet.
5. Attribute können sich blockweise ändern.

5.2 Gesamtkonzept Identitätsdaten-Leak-Parser

Aufgrund der zuvor dargestellten inhaltlichen Eigenschaften ist eine Verarbeitung der Identitätsdaten-Leaks mit einem Standard-Framework für CSV-Dateien nicht möglich. Damit diese Daten jedoch trotzdem automatisiert verarbeitet werden können, muss ein spezielles System konzipiert werden, welches mit den besonderen Eigenschaften von Identitätsdaten-Leaks beim automatisierten Einlesen umgehen kann. Inhalt dieses Kapitels ist deshalb die Entwicklung eines Systems, welches automatisiert die einzelnen Identitätsmerkmale identifizieren und extrahieren kann.

Das Ergebnis dieses Kapitels ist das vollständige Konzept für ein System, welches Identitätsdaten-Leaks vollautomatisiert analysiert und verarbeitet. Dazu werden mit einem für diesen Anwendungsfall entwickelten Verfahren die Trennzeichen erkannt, um anschließend den gesamten Leak in die jeweiligen Bestandteile aufteilen zu können. Um festzustellen, welche semantische Bedeutung die im Leak enthaltenen Identitätsmerkmale besitzen, wird in einer anschließenden Analyse den im Leak enthaltenen Zeichenketten ein Identitätsdaten-Typ zugeordnet. Das zu Grunde liegende Vorgehen wird in diesem Abschnitt als Gesamtkonzept vorgestellt. Die einzelnen Bestandteile des Gesamtkonzepts werden in den anschließenden Kapiteln genauer beschrieben.

In Abbildung 7 ist das Gesamtkonzept für das benötigte System dargestellt, welches nachfolgend genauer erläutert wird. Das System hat die Aufgabe,

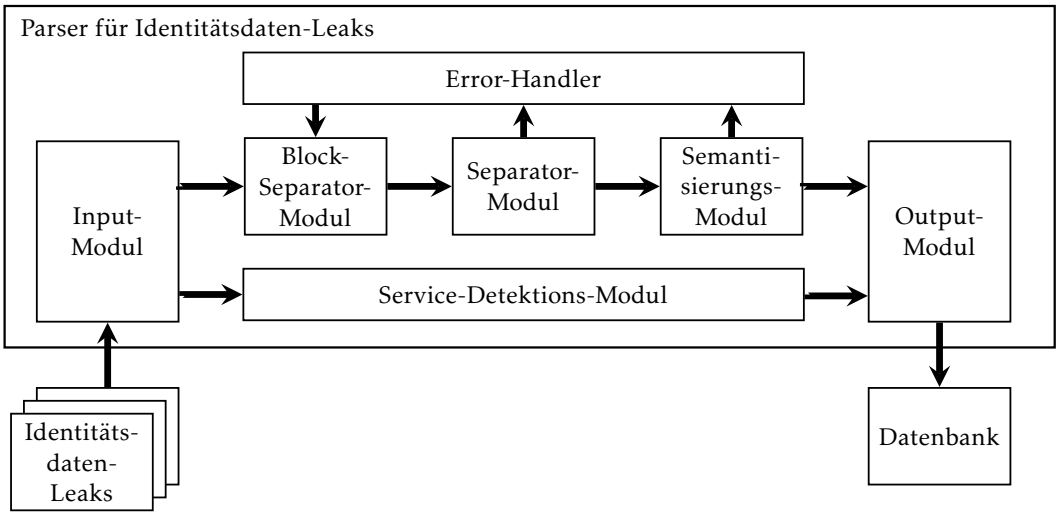


Abbildung 7: *Aufbau eines Parsers für Identitätsdaten-Leaks.*

Identitätsdaten-Leaks zu analysieren, um die Syntax und die Semantik zu erkennen. Aufgrund dieser Funktionalität wird das System auch als *Parser für Identitätsdaten-Leaks* bezeichnet. Initial liegen Identitätsdaten-Leaks in der Form vor, wie sie aus den in Kapitel 4 beschriebenen Quellen heruntergeladen werden. In der Regel liegen sie als reine Textdateien oder komprimiert in Archiven zum Beispiel als `zip`, `rar` oder `tar.gz` vor. Manche Leaks bestehen aus einzelnen Dateien, andere Leaks sind in viele Dateien aufgeteilt, die in Unterverzeichnissen sortiert sind. Damit keine manuelle Vorarbeit geleistet werden muss, wird ein **Input-Modul** benötigt, welches automatisiert die geladenen Dateien unabhängig von der vorliegenden Dateistruktur erfasst, eventuell entpackt und anschließend zur weiteren Verarbeitung bereithält.

Die Menge an in einem Leak enthaltenen Identitätsdaten unterscheidet sich abhängig von der entsprechenden Datei. Es sind Leaks zu finden, die nur einige wenige Identitätsdatensätze enthalten. Andere bestehen aus mehreren Millionen Datensätzen. Solche Leaks besitzen häufig eine Dateigröße von mehreren Gigabyte. Um solche großen Dateien unabhängig von den eingesetzten Ressourcen verarbeiten zu können, müssen Dateien solcher Größe sequenziell eingelesen und verarbeitet werden. Das bedeutet, dass eine Leak-Datei abschnittsweise analysiert wird. Im

Block-Separator-Modul findet eine Aufteilung der ursprünglichen Datei in einzelne **Blöcke** statt. Die Obergrenze für die Blocklänge sollte so gewählt werden, dass genügend Hauptspeicher vorhanden ist, um den kompletten Block für eine effiziente Verarbeitung zwischenspeichern zu können.

Jeder Block wird anschließend sequenziell tiefergehend analysiert, um die einzelnen Identitätsdaten mit den dazugehörigen Identitätsmerkmalen zu extrahieren. Dazu muss zunächst die Struktur erkannt werden, mit der die einzelnen Identitätsmerkmale in jeder Zeile gespeichert sind. Wie im vorherigen Abschnitt dargestellt, werden die einzelnen Merkmale als Zeichenketten repräsentiert, die mit Trennzeichen voneinander getrennt sind. Demnach müssen im ersten Analyseschritt die verwendeten Trennzeichen erkannt werden. Das **Separator-Modul** extrahiert die in einem Block enthaltenen Trennzeichen. Das genaue Vorgehen und die verwendeten Verfahren sind in Abschnitt 5.3 beschrieben. Anschließend werden die Trennzeichen vom Separator-Modul verwendet, um die Identitätsmerkmale aus den Zeilen zu extrahieren.

Diese nun extrahierten Identitätsmerkmale sind in diesem Verarbeitungsschritt einfache Zeichenketten ohne eine semantische Zuordnung. Das bedeutet, dass ein System nicht zwischen Merkmalen wie E-Mail-Adressen, Passwörtern und Benutzernamen unterscheiden kann. Da eine semantische Unterscheidung für das weitere Vorgehen notwendig ist, muss hierfür ein Verfahren konzipiert und entwickelt werden. In Abschnitt 5.4 wird das **Semantisierungs-Modul** vorgestellt, welches die häufigsten Identitätsdatentypen erkennen und zuordnen können soll.

Sollten bei der Verarbeitung einzelner Blöcke das Separator-Modul oder das Semantisierungs-Modul fehlschlagen, so liegen eventuell syntaktische oder semantische Unregelmäßigkeiten innerhalb des Blockes vor. Diese Blöcke werden an den **Error-Handler** weitergegeben. Der Error-Handler entscheidet, ob fehlgeschlagene Blöcke in einem zweiten Versuch erneut analysiert werden sollen, nachdem diese mittels des Block-Separator-Moduls in kleinere Blöcke zerlegt wurden. Um diese Entscheidung zu treffen, wird geprüft, ob eine Mindestgröße des Blocks eingehalten wird und ob sich gleiche Fehler bei dem vorherigen und nachfolgenden Block ergeben. Eine genauere Beschreibung des Moduls ist in Abschnitt 5.5 zu finden.

Konnte ein Block durch das Separator-Modul und das Semantisierungs-Modul erfolgreich verarbeitet werden, dann werden die Ergebnisse an das Output-Modul weitergegeben. Nach der Semantisierung besitzen die Identitätsmerkmale eine Typenzuordnung. Diese Daten müssen für die weitere Warnungskonstruktion persistiert werden. Die Speicherung der Daten wird von dem **Output-Modul** übernommen. Dieses Modul kann die an diesem Verarbeitungsstand vorliegenden Klartext-Identitätsdaten in einer Datenbank speichern. Da diese Daten sicherheitsrelevante und personenbezogene Informationen enthalten, ist eine Pseudonymisierung vor der Speicherung notwendig. Ein mögliches Konzept zur Speicherung dieser Daten wird in Abschnitt 6.5 vorgestellt.

Parallel zur eigentlichen Verarbeitung der Leak-Daten findet eine weitere Analyse statt. Das **Service-Detektions-Modul** versucht den kompromittierten Dienst eines Identitätsdaten-Leaks zu erkennen. Wird ein Dienst erkannt, so kann diese Information als Metadatum an das Output-Modul weitergegeben werden. Das dazugehörige Verfahren ist in Abschnitt 5.6 genauer beschrieben.

5.3 Strukturanalyse mittels Trennzeichendetektion

Identitätsdaten-Leaks sind strukturierte Daten von unbekannter Struktur. Wie zuvor in Abschnitt 5.1 dargestellt, ähnelt die in Identitätsdaten-Leaks verwendete Datenstruktur dem Format CSV. Das bedeutet, dass einzelne Identitätsmerkmale innerhalb einer Zeile durch Trennzeichen voneinander getrennt werden. Zur Extraktion der Identitätsmerkmale müssen zwei aufeinanderfolgende Prozesse im Separator-Modul durchlaufen werden. Im ersten Prozess namens **Trennzeichenerkennung** müssen die in einem Identitätsdaten-Leak verwendeten Trennzeichen erkannt werden. Erst im darauffolgenden Schritt können die erkannten Trennzeichen dazu verwendet werden, um die einzelnen Identitätsmerkmale aus einer Leak-Zeile zu extrahieren (**Merkmalsextraktion**). Dargestellt ist das Konzept des Separator-Moduls in Abbildung 8. Die Konzepte für die Umsetzung von Trennzeichenerkennung und Merkmalsextraktion werden in den folgenden Unterabschnitten genauer dargestellt.

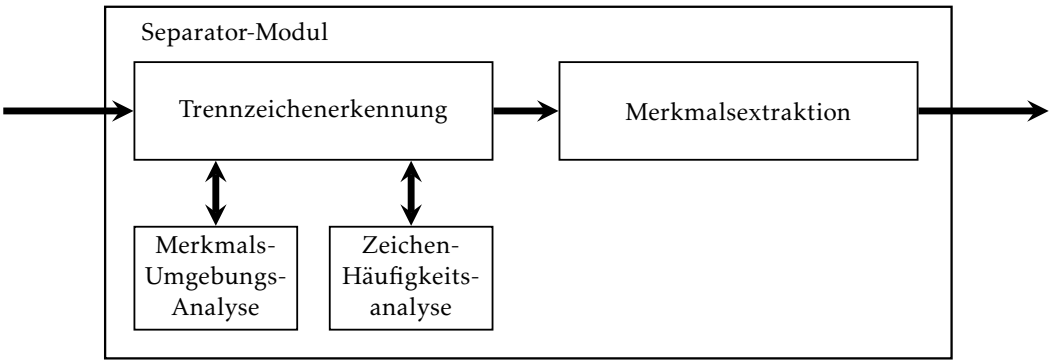


Abbildung 8: *Aufbau des Separator-Moduls.*

5.3.1 Trennzeichenerkennung

Um eine Trennzeichenerkennung bei Identitätsdaten-Leaks durchzuführen, muss ein Konzept für ein Werkzeug entwickelt werden, welches die genannten Eigenschaften der Identitätsdaten-Leaks berücksichtigt. Im Folgenden werden zwei Konzepte für die Trennzeichenerkennung vorgestellt. Beide Konzepte liefern valide Ergebnisse, allerdings benötigt das erste Verfahren deutlich mehr Ressourcen.

Konzept 1: *Zeichen-Häufigkeitsanalyse*

Eine naheliegende Idee für die Erkennung von Trennzeichen ist, eine statistische Auswertung durchzuführen, welche die Häufigkeit des Auftretens von verschiedenen Zeichen pro Zeile analysiert. Der Grundgedanke für diese Idee ist, dass das Vorkommen eines Zeichens pro Zeile über viele Zeilen hinweg konstant bleibt. Besteht jede Zeile eines Leaks beispielsweise aus der Syntax `email:passwort`, dann kommt der Doppelpunkt in den allermeisten Zeilen genau einmal vor. Um dies zu realisieren, muss folgendes Vorgehen implementiert werden.

Ein Leak D besteht aus n verschiedenen Zeilen, $L_1,\ldots,L_n$, also $D = \{L_1,\ldots,L_n\}$. Jede Zeile L besteht aus einer Multimenge von Zeichen, das heißt jedes in L vorkommende Zeichen z ist zusammen mit seiner absoluten Häufigkeit in L bekannt. Jedes mögliche Zeichen z ist in dem Kodierungsstandard für Unicode-Zeichen UTF8 kodiert. Mit $c(L,z)$ wird die Anzahl des Auftretens des Zeichens z in der

Zeile L bezeichnet. Für jedes Zeichen z erhält man somit die Zufallsvariable $X_z\colon D \to \mathbb{N}, X_z(L) = c(L,z)$. Die Zufallsvariable X_z gibt an, wie oft z in einer gegebenen Zeile vorkommt. Wie in der Einleitung erklärt, ist nun das Zeichen v gesucht, sodass X_v minimale Varianz besitzt. Dieses Zeichen stellt dann das gesuchte Trennzeichen dar.

Zu beachten bei diesem Vorgehen ist, dass gewisse Identitätsmerkmale auch spezielle Zeichen enthalten können, die als Trennzeichen erkannt werden. Beispielsweise enthält eine E-Mail-Adresse ein @, welches mit diesem Konzept als Trennzeichen erkannt würde, wenn jede Zeile eine E-Mail-Adresse enthält. Um diese Problematik auszubessern, muss zusätzlich analysiert werden, ob in einer Zeile eine E-Mail-Adresse enthalten ist. Sollte dies zutreffen, dann müssen die in der E-Mail-Adresse enthaltenen Zeichen aus der statistischen Analyse ausgeschlossen werden.

Dieses Verfahren liefert in ersten Tests zuverlässig die korrekten Trennzeichen zurück. Jedoch besitzt es zwei deutliche Nachteile. In den ersten Tests erweist sich dieses Verfahren als rechenintensiv. Besteht ein Trennzeichen aus mehr als einem Zeichen, so müssen nicht nur einzelne Zeichen, sogenannte Unigramme, mit der statistischen Auswertung überprüft werden, sondern auch Bi- und eventuell auch Trigramme. Wird eine Analyse mit Zeichenketten der Länge ≥ 2 durchgeführt, so steigt die benötigte Rechenzeit an. Auch wird die Erkennung von Trennzeichen mit diesem Verfahren schwierig, sobald verschiedene Trennzeichen innerhalb einer Zeile verwendet werden, da so mehrere Zeichen mit minimalen Varianzen ausgewählt werden müssten. Hierdurch wird das Verfahren deutlich fehleranfälliger.

Konzept 2: *Merkmals-Umgebungs-Analyse*

Das Konzept der *Merkmals-Umgebungs-Analyse* weist zur Erkennung der Trennzeichen einen vollständig anderen Ansatz auf. Bei der Gestaltung dieses Konzepts wird explizit darauf geachtet, den Ressourcenbedarf möglichst gering zu halten, um so die Geschwindigkeit des gesamten Parsers zu optimieren. Dazu wird von Beginn an darauf verzichtet, dass mit diesem Konzept restlos alle in Leaks vorkommenden Trennzeichen erkannt werden können. Die Trennzeichenerkennung wird so konstruiert, dass für die Funktionsweise zwei Einschränkungen hingenommen werden, welche jedoch zu einem deutlichen Leistungsanstieg verhelfen.

- **Einschränkung 1:** Es können nur Trennzeichen erkannt werden, wenn die Identitätsdatensätze Merkmale enthalten, die mittels eines regulären Ausdrucks erkannt werden können.
- **Einschränkung 2:** Es werden nur Trennzeichen erkannt, die ausschließlich aus Sonderzeichen bestehen.

Bei der *Merkmals-Umgebungs-Analyse* werden Identitätsmerkmale in den Zeilen des Leaks gesucht, welche sich aufgrund ihrer Syntax (E-Mail-Adressen, Hash-Werte, Telefonnummern, Kreditkartennummern) mittels eines regulären Ausdrucks erkennen lassen. Wird ein solches Identitätsmerkmal erkannt, so werden die links und rechts neben diesem Merkmal befindlichen Zeichen dahingehend untersucht, ob diese als Trennzeichen geeignet sind. In der Leak-Zeile `Benutzername:test@uni-bonn.de|testpasswort` würde mittels regulären Ausdrucks die E-Mail-Adresse `test@uni-bonn.de` erkannt. Werden anschließend die Zeichen neben der E-Mail-Adresse betrachtet, so lassen sich die Trennzeichen „:“ und „|“ ermitteln. In Algorithmus 1 ist die Prozedur zur Trennzeichenerkennung schematisch dargestellt.

In Zeile drei bis fünf wird ein Identitätsdatensatz (Zeile eines Leaks) nach den vordefinierten Identitätsmerkmalen mittels regulären Ausdrücken durchsucht. Bei einem Treffer wird das entsprechende Merkmal durch den String `\n` ersetzt, was dem Steuerzeichen für eine neue Zeile entspricht und nur als Platzhalter dient, da dieses Zeichen zu diesem Zeitpunkt nicht in einer Zeile vorkommen kann. In Zeile sechs wird bei jedem `\n` der String getrennt. In dem Block von Zeile acht bis elf wird mittels der regulären Ausdrücke ein Trennzeichen mit den Zeichen aus Zeile 1 gesucht. Wurden mittels dieses Vorgehens Trennzeichen gefunden, so wird die gesamte Identitätsdatenzeile erneut nach den gefundenen Trennzeichen durchsucht, damit alle Trennzeichen in der korrekten Reihenfolge abgespeichert werden können.

Damit die korrekten Trennzeichen für einen ganzen Block gefunden werden, müssten alle Zeilen nach diesem Vorgehen analysiert werden. Um den Rechenaufwand zu verringern, kann auch nur ein gewisser Anteil an Zeilen eines Leaks analysiert werden, um ein Trennzeichen zu finden. Liefert die in Algorithmus 1

Algorithmus 1: *Trennzeichenerkennung mittels Merkmals-Umgebungs-Analyse.*

Eingabe : Z – Zeile eines Leaks
Ausgabe : S – Liste von detektierten Separatoren

```
1  C ← [':', ';', ..., '–']                    /* Liste möglicher Separator-Zeichen */
2  R ← [emailRegex, hashRegex, ...]             /* Liste von regulären Ausdrücken für Merkmale */
3  for r in R do
4  |  suche r in Z und ersetze durch '\n'
5  end
6  A ← trenne Z an '\n' in eine Liste
7  S ← [ ]
8  for a in A do
9  |  S ← suche am Anfang von a nach Elementen von C
10 |  S ← suche am Ende von a nach Elementen von C
11 end
12 S ← bringe die Elemente von S in die richtige Reihenfolge
13 return S
```

vorgestellte Prozedur für einen gewissen Prozentsatz aller Zeilen das identische Ergebnis, kann darauf geschlossen werden, dass die gefundenen Trennzeichen auch in den restlichen Zeilen zur Anwendung kommen. Für den Identitätsdatensatz `Benutzer||test@uni-bonn.de:passwort;;0170-12345678` liefert die vorherige Prozedur folgende Trennzeichen zurück: `'||', ':', ';;'`.

Ein großer Vorteil dieses Konzepts ist die deutlich bessere Performance mit einer ähnlichen Zuverlässigkeit. Nur für die genannten Einschränkungen kann dieses Konzept nicht verwendet werden. Bei einem Vergleich der beiden implementierten Konzepte kann bei dem zweiten Konzept eine zwanzigfache Geschwindigkeitsverbesserung in einem vergleichenden Experiment festgestellt werden.

5.3.2 Merkmalsextraktion

In dem Schritt der Merkmalsextraktion werden die gefundenen Trennzeichen dazu genutzt, um den gesamten Identitätsdaten-Leak oder Teile davon in maschinenverarbeitbare Objekte zu überführen. Dazu wird jede einzelne Zeile an den Stellen der erkannten Trennzeichen aufgetrennt. Das Ergebnis hiervon ist eine Liste von einzelnen Identitätsmerkmalen. Jedoch ist an dieser Stelle für das System unklar, um welche Art von Merkmal es sich handelt. Bei der Implementierung dieses Moduls muss darauf geachtet werden, dass eine geeignete Fehlerbehandlung integriert wird. Fehler bzw. Unterschiede in einzelnen Zeilen kommen häufig vor, weswegen das Modul zur Merkmalsextraktion für diese Gegebenheiten geeignet sein muss. Tritt ein Fehler nur in einzelnen Zeilen auf, können diese erst einmal ignoriert werden und für eine spätere Analyse gespeichert werden. Alle einzelnen fehlerbehafteten Zeilen werden als eigener Block zusammengefasst und erneut analysiert. Tritt der Fehler in einem ganzen Block auf, wird der aktuelle Block in zwei Teilblöcke aufgeteilt, um diese beiden Blöcke getrennt voneinander zu verarbeiten.

5.4 Zuordnung der Semantik

In Abschnitt 5.3 wird gezeigt, wie die in den Identitätsdaten-Leaks verwendeten Trennzeichen automatisiert detektiert werden. Auf Grundlage dieser Trennzeichen werden die einzelnen Identitätsmerkmale extrahiert, sodass diese nachfolgend analysiert werden können. Nach der Merkmalsextraktion ist dem System noch unklar, um welchen Typ eines Identitätsmerkmals es sich bei den einzelnen Merkmalen handelt. Zur weiteren Verarbeitung muss eine Zuordnung der Bedeutung eines jeden Merkmals erfolgen. Deswegen werden in diesem Abschnitt verschiedene Methoden vorgestellt, mit denen sich die Typen der Identitätsmerkmale bestimmen lassen.

Es gibt Identitätsmerkmale, die sich mithilfe eines regulären Ausdrucks zuverlässig erkennen lassen. Eine Bestimmung von E-Mail-Adressen, Telefonnummern, Kreditkartennummern oder IBANs lässt sich mit geringem Aufwand realisieren. Allerdings existieren auch Identitätsmerkmale, die sich nicht direkt mit einem regulären Ausdruck abbilden lassen. Beispielsweise sind eine Erkennung und Un-

terscheidung von den Identitätsmerkmalen *Vorname*, *Benutzername* oder *Passwort* deutlich komplexer. Diese drei Merkmale sind in einzelnen Identitätsdatensätzen in der Realität mit ähnlichen oder teilweise identischen Zeichenketten gefüllt. Gerade die Unterscheidung von Benutzernamen und Passwörtern ist eine komplexe Aufgabe, da von den Benutzern für beide Merkmale beliebig lange Zeichenketten gewählt werden dürfen, die wiederum beliebige Zeichen beinhalten. Eine lokale Unterscheidung von Identitätsmerkmalen in einzelnen Zeilen ist deshalb unmöglich, da ein solches Verfahren in vielen Fällen unzureichende Ergebnisse liefern wird. Sinnvoller ist es, den Kontext einer einzelnen Zeile mit in eine Merkmalserkennung zu integrieren. Damit ist gemeint, dass auch die Zeilen darüber und darunter betrachtet werden, um einer ganzen Spalte eine Bedeutung zuzuordnen. Enthält eine einzelne Zeile tatsächlich nur sehr schwer unterscheidbare Zeichenketten für unterschiedliche Identitätsmerkmale, dann lässt sich eine Zuordnung durch den Menschen meist durch die Betrachtung der umgebenden Zeilen lösen.

Aus diesen Überlegungen sind vier verschiedene Module zur Zuordnung der Bedeutung entwickelt worden [70, 68], welche im Folgenden vorgestellt werden:

1. Regex-Modul
2. Wortlisten-Modul
3. Zeichenanalyse-Modul
4. API-Modul

Das **Regex-Modul** [68, 70] setzt für die Erkennung des Merkmalstyps reguläre Ausdrücke ein. Die Merkmalstypen, die mit diesem Modul erkannt werden können, haben eine definierte Syntax, sodass diese mit Automaten zuverlässig erkannt werden können. Eine Detektion für die folgenden Merkmalstypen lässt sich hiermit realisieren: *E-Mail-Adresse*, *Telefonnummer*, *IBAN*, *Kreditkartennummer*, *Hash-Wert*, *Datum*, *IP-Adresse*. IBANs und Kreditkartennummern enthalten Prüfsummen, welche auch bei der Detektion validiert werden können, um Zeichenketten mit gleichen Eigenschaften von diesen zu unterscheiden. Eine Berechnung der Prüfsummen ist jedoch nur mit aufwendigen Automaten möglich. Um eine Überprüfung der Prüfsummen zu realisieren, wird deshalb die reine Anwendung der regulären Ausdrücke

um eine softwarebasierte Überprüfung erweitert. Dieses Modul kann auf einzelne Zeilen angewandt werden, da die feste Syntax der Merkmale eine ausreichende Identifizierung ermöglicht.

Das **Wortlisten-Modul** [68, 70] erkennt einen Merkmalstyp durch Vergleiche der Identitätsmerkmale mit verschiedenen Wortlisten, welche typische und häufig verwendete Zeichenketten für den jeweiligen Merkmalstyp enthalten. Mit diesem Modul wird eine Erkennung der Merkmalstypen *Vornamen*, *Nachnamen*, *Benutzernamen*, *Passwörter* und *Bankleitzahlen* realisiert. Für jeden dieser Merkmalstypen wird eine eigene Wortliste verwendet, welche die häufigsten Einträge beinhaltet. Im Fall der Bankleitzahl-Wortliste enthält diese sämtliche in Deutschland gültigen Bankleitzahlen. Die Liste aller Bankleitzahlen kann von der deutschen Bundesbank bezogen werden [16]. Eine Erweiterung auf zusätzliche Länder lässt sich durch Ergänzungen dieser Liste realisieren.

Zur Detektion eines Merkmalstyps wird eine gewisse Anzahl an Einträgen einer Spalte eines Identitätsdaten-Leaks mit den Wortlisten verglichen. Es wird dabei auf eine vollständige Übereinstimmung getestet. Findet sich ein ausreichend großer Anteil von Identitätsmerkmalen in einer solchen Liste wieder, dann kann diese Spalte mit dem Merkmalstyp der Wortliste versehen werden. Das Vorgehen zur Ermittlung der Merkmalstypen mittels Wortlisten wird bereits in anderen Arbeiten beschrieben [42, 68].

Für die Erstellung der Nachnamens-Wortliste werden zwei Quellen gewählt, eine Liste mit den Top 500 deutschen Nachnamen [127] und eine mit den Top 1.000 US-amerikanischen Nachnamen [80]. Es wird eine deutsche und eine amerikanische Liste gewählt, um den deutschen als auch den amerikanischen Kontext abzubilden, da viele Leaks Daten von Amerikanern beinhalten. Die Daten aus beiden Quellen werden zusammengefügt und doppelte Einträge entfernt.

Ein ähnliches Vorgehen wird zur Erstellung der Vornamens-Wortliste angewendet. Für diese Liste werden folgende Quellen verwendet: Die Liste mit in Deutschland gängigen weiblichen und männlichen Vornamen wird aus folgender Quelle bezogen [55]. Eine Liste für in den Vereinigten Staaten gängige Vornamen wird von [80]

bezogen. Auch diese Listen werden zusammengefügt und doppelte Einträge entfernt. Die Liste für die Benutzernamen und die Passwörter wird aus mehreren gefundenen Leaks extrahiert und davon die Häufigsten gewählt.

Um für die Länge dieser Listen eine sinnvolle Größe zu wählen, wird eine Evaluation mehrerer Listenlängen durchgeführt [68]. Hierzu wird untersucht, in wie vielen Fällen die unterschiedlichen Listen (Vorname, Nachname, Passwort) in Verbindung mit deren Längenvariationen einzelne Einträge eines Testdatensatzes korrekt klassifizieren. Als Testdatensatz zur Evaluation der Vornamens- und Nachnamenslisten werden die ersten 100.000 Einträge aus dem Leak *Modern-Business-Solutions* [47] verwendet. Gewählt wird dieser Identitätsdaten-Leak, da Vornamen und Nachnamen enthalten sind. Zur Evaluation der Passwortliste wird der Leak der Dating-Plattform *Fling* [126] genutzt, da hier Klartextpasswörter enthalten sind. Für die Evaluation werden ausschließlich die genannten Spalten aus den zuvor genannten Leaks mit den verschiedenen Listen verglichen. Es wird überprüft, für wie viele Einträge der Leak-Listen eine Übereinstimmung in den Wortlisten gefunden werden kann.

In Abbildung 9 sind die prozentualen Detektionsraten der einzelnen getesteten Listen zu sehen. Mit Detektionsraten sind die prozentuale Übereinstimmung von Elementen einer Wortliste mit den Elementen einer Spalte eines Identitätsdaten-Leaks gemeint. Bei den Detektionsraten der Passwortspalte ist zu erkennen, dass alle Listen mit Passwörtern höhere Detektionsraten aufweisen als die anderen Listen. Die Liste mit 10.000 Einträgen weist die höchste Detektionsrate auf. Jedenfalls reichen die dargestellten Detektionsraten der Passwortlisten aus, um mittels dieser Listen den Typ der Spalte zu erkennen. Die Detektion der Vornamen mittels der Vornamenslisten zeigt sich als ebenso zielführend. An dieser Stelle erweist sich eine Liste mit den häufigsten männlichen und weiblichen deutschen und US-amerikanischen Vornamen als sinnvoll. Bei der Nachnamensdetektion ist genauso eine Detektion mittels der Nachnamensliste möglich. Lediglich die Liste mit den deutschen Nachnamen eignet sich nicht. Grund dafür ist vermutlich, dass es sich bei dem betreffenden Dienst des zur Evaluation genutzten Leaks um ein amerikanisches Unternehmen handelt und dort höchstwahrscheinlich nur eine geringe Anzahl von deutschen Nachnamen enthalten ist. Die geringen, aber deutlichen De-

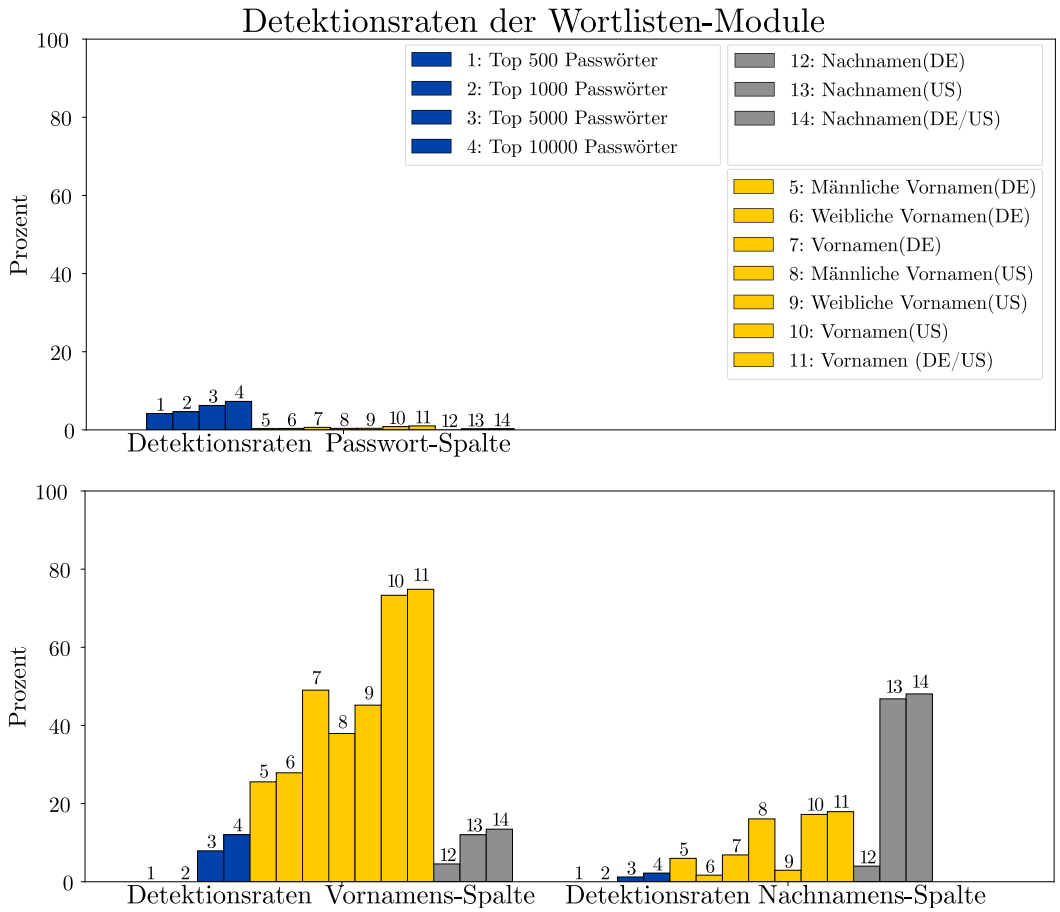


Abbildung 9: *Evaluation verschiedener Längen der Wortlisten [68].*

tektionsraten der anderen Listen bei der Vornamens- und Nachnamensspalte haben unter anderem die Ursache, dass die Typ-übergreifenden Listen identische Elemente beinhalten. Aus diesem Grund erscheint es sinnvoll, die Listen von gemeinsamen Elementen zu bereinigen.

Auf Grundlage dieser Ergebnisse werden die Wortlisten erstellt. Nach der Bereinigung von gleichen Einträgen in verschiedenen Listen enthält die Vornamens-Wortliste 2.915 Einträge, die Nachnamens-Wortliste 1.078 Einträge, die Liste für die Benutzernamen 48.705 Einträge, die Passwort-Wortliste 9.968 und die Bankleitzahl-Liste 3.600 Einträge. Die umfangreiche Länge der Benutzernamen-Liste lässt sich

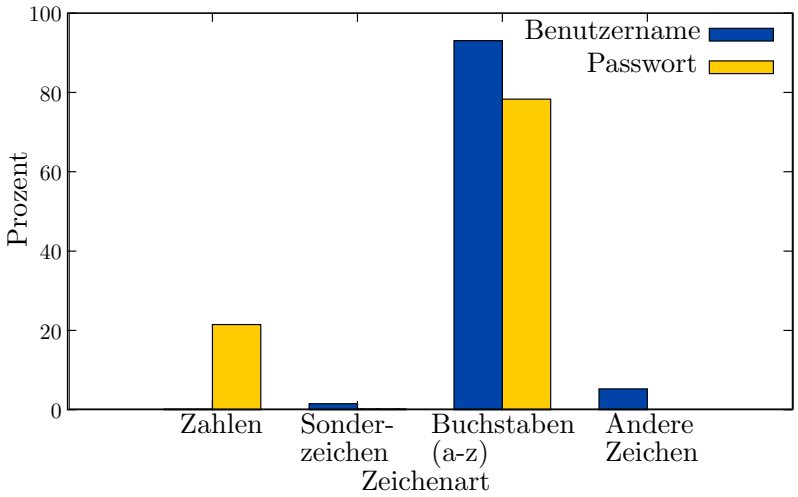


Abbildung 10: *Prozentuale Verteilung von Zeichenarten - Syntaktische Eigenschaften von Passwörtern und Benutzernamen im Leak badoo [68].*

damit begründen, dass bei Untersuchungen festgestellt wurde, dass Benutzernamen keine so hohe Entropie wie Passwörter enthalten, jedoch die Menge an häufig verwendeten Benutzernamen deutlich geringer ist.

Benutzernamen und Passwörter sind in erster Linie beliebig lange Zeichenketten, die in der Regel beliebige Zeichen enthalten können. Sollte aufgrund dieser Eigenschaft die Unterscheidung der beiden Typen mithilfe des Wortlisten-Moduls nicht möglich sein, so kann ein weiteres Modul zur genaueren Analyse herangezogen werden. Das **Zeichenanalyse-Modul** [70] untersucht die in einer Leak-Spalte verwendeten Zeichen und beurteilt anhand der Länge der vorliegenden Zeichenketten, ob es sich bei einer Spalte um ein Passwort oder einen Benutzernamen handelt. In den Abbildungen 10 und 11 sind die syntaktischen Eigenschaften von Passwörtern und Benutzernamen vergleichend dargestellt. Die in der Darstellung analysierten Daten stammen aus dem Identitätsdaten-Leak *badoo* mit 112 Millionen Datensätzen [68]. Anders als in der medialen Darstellung [95] sind auch Versionen des Leaks mit Klartextpasswörtern anstatt ungesalzenen MD5-Hashes zu finden. Eine solche Version des Leaks wurde für diese Analyse genutzt, da in diesem Leak Benutzernamen und Klartextpasswörter enthalten sind. In Abbildung 10 ist die Verteilung

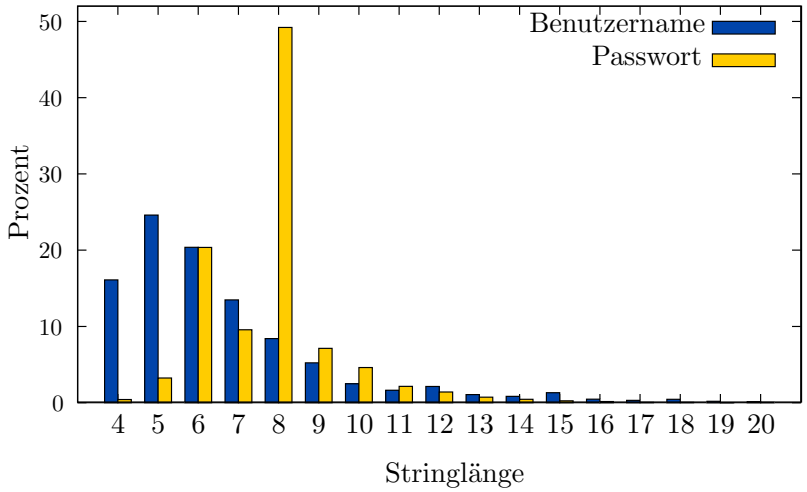


Abbildung 11: *Prozentuale Verteilung der Stringlänge - Syntaktische Eigenschaften von Passwörtern und Benutzernamen im Leak badoo [68].*

von Zeichenarten in die Kategorien *Zahlen*, *Sonderzeichen*, *Buchstaben von A bis Z* und andere Zeichen wie diakritische Zeichen unterteilt. Zu sehen ist, dass im Leak *badoo* Passwörter im Gegensatz zu Benutzernamen Zahlen enthalten. Ansonsten sind in Benutzernamen eine geringe Menge an Sonderzeichen und anderen Zeichen enthalten. Diese Eigenschaft kann aber speziell auf diesen Leak zurückzuführen sein. In [95] wird von einem Leak mit 127 Millionen Datensätzen ohne Klartextpasswörter, aber mit MD5-Hash-Werten berichtet. Der für diese Evaluation genutzte Leak enthält nur 112 Millionen Datensätze, jedoch mit Klartextpasswörtern. Vermutlich enthalten die fehlenden 15 Millionen Datensätze in deren Passwörtern Sonderzeichen, weswegen die enthaltenen Passwort-Hashes aufgrund ihrer Komplexität nicht gebrochen werden konnten. Dies würde das Fehlen der Sonderzeichen bei Passwörtern erklären.

Eine Entscheidung, ob es sich bei einer Leak-Spalte um einen Benutzernamen oder ein Passwort handelt, könnte anhand der Anzahl an enthaltenen Zahlen getroffen werden. Jedoch gibt eine Längenanalyse der Zeichenketten einen deutlicheren Aufschluss über die enthaltenen Datentypen. In Abbildung 11 ist die Verteilung der Längen der Zeichenketten von in dem *badoo*-Leak enthaltenen Passwörtern und

Benutzernamen dargestellt [68]. Zu sehen ist, dass fast die Hälfte aller enthaltenen Passwörter eine Länge von acht Zeichen besitzt. Dagegen verläuft die Verteilung der Längen der Benutzernamen ähnlich zu einer Art Normalverteilung. Die Eigenschaft, dass Passwörter häufig eine Länge von acht Zeichen besitzen, kann auch in anderen Leaks nachvollzogen werden. Eine Klassifikation über die Länge der Zeichenketten wird erst dann problematisch, wenn ein Dienstbetreiber eine Mindestlänge von mehr als acht Zeichen für Passwörter vorschreibt. Ein solcher Fall wurde jedoch in keinen vorliegenden Identitätsdaten-Leaks gesichtet. Das Zeichenanalyse-Modul nimmt somit eine Längenanalyse an einer Spalte vor und klassifiziert diese als Passwort-Spalte, wenn überdurchschnittlich viele Zeichenketten mit acht Zeichen enthalten sind.

Das **API-Modul** [68, 70] testet die Zeichenketten einer Spalte, indem es andere Dienste überprüfen lässt, ob diese der Zeichenkette eine Semantik zuordnen kann. In [68] wird OpenStreetMap[1] verwendet, um zu überprüfen, ob dieser Dienst in den Zeichenketten eine Beschreibung eines Ortes in Form einer Adresse oder Koordinaten erkennen kann. In diesem Test wurde eine Offline-Version von OpenStreetMap verwendet. Es konnten erfolgreich Adress-Spalten mit diesem Ansatz erkannt werden. Jedoch wurde dieser Ansatz in dieser Arbeit nicht weiterverfolgt, da Adressen für eine Warnung in diesem Kontext keine Relevanz besitzen. Trotzdem wurde dieser Ansatz genannt, um die Idee der Klassifikation mittels anderer Dienste aufzuzeigen.

In Abbildung 12 ist die Zusammenarbeit der vier beschriebenen Semantisierungs-Teilmodule dargestellt. Die einzelnen Module werden von dem Hauptmodul (in Abbildung 12 als *Semantisierung* bezeichnet) verwaltet. Das Hauptmodul liefert den Teilmodulen die jeweiligen Spalten aus den zu verarbeiteten Blöcken. Jedes dieser Teilmodule überprüft, ob es in den gelieferten Spalten Identitätsdatentypen erkennen kann. Das Hauptmodul bekommt von allen Teilmodulen anschließend Werte für die möglichen Merkmalstypen zurückgeliefert, abhängig davon, wie genau die einzelnen Merkmalstypen in den Spalten wiederzufinden sind. Erkennt das Wortlisten-Modul beispielsweise in jedem Element einer Spalte einen Vornamen,

[1] OpenStreetMap Foundation: OpenStreetMap, https://www.openstreetmap.org.

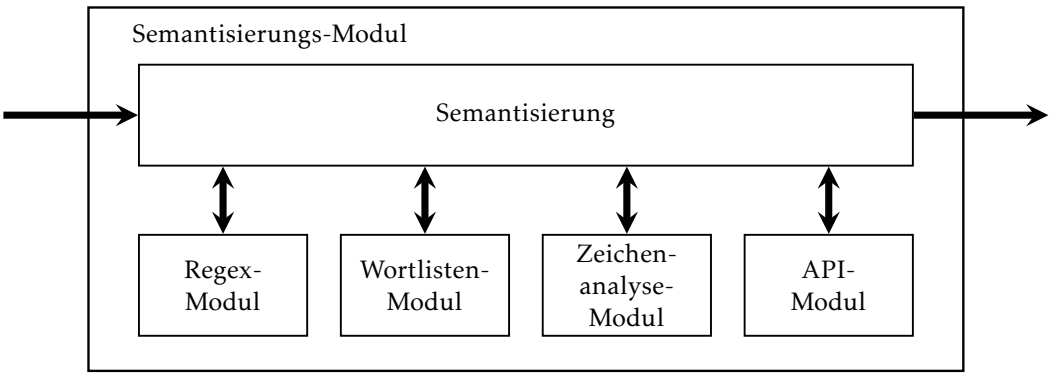


Abbildung 12: *Aufbau des Semantisierungs-Moduls.*

wird dieser Spalte der Identitätsdatentyp *Vorname* mit dem Wert 100 % zugeordnet. Aufgabe des Hauptmoduls ist es dann, anhand dieser numerischen Bewertungen den geeignetsten Typ zu selektieren. Wichtig hierbei ist, dass für jeden Merkmalstyp eine untere Schwelle definiert werden muss, die angibt, wie viele Einträge in einer Spalte als dieser Typ klassifiziert werden müssen, damit das Hauptmodul diesen Typ auswählen darf. Grund hierfür ist, dass Spalten mit unbekanntem oder vermischtem Inhalt nicht irrtümlich falsch klassifiziert werden.

5.5 Umgang mit Strukturveränderungen innerhalb eines Leaks

Zu Beginn dieses Kapitels wird beschrieben, dass strukturelle Veränderungen innerhalb eines Identitätsdaten-Leaks auftreten. Die möglichen Veränderungen müssen dabei in drei Kategorien unterschieden werden:

(a) Veränderungen der Reihenfolge der vorkommenden Identitätsmerkmalstypen in einem ganzen Block.

(b) Veränderung der eingesetzten Trennzeichen in einem ganzen Block.

(c) Lokale strukturelle Änderungen jeglicher Art in einzelnen Zeilen.

Veränderungen aus den Kategorien (a) und (b) sind solche, die sich teilweise über weite Teile eines Leaks erstrecken. Da von Beginn an eine Verarbeitung auf Blockebene stattfindet, besteht nicht die Gefahr, dass große Teile eines Leaks falsch oder fehlerhaft extrahiert und attribuiert werden. Ein Leak wird mit dem vorgestellten Verfahren in kleinere Blöcke unterteilt. Für jeden dieser Blöcke wird eine eigene Trennzeichenerkennung und Semantisierung durchgeführt. Ist in einem Leak eine einzige strukturelle Veränderung vorhanden, funktioniert die Trennzeichenerkennung und die Semantisierung fehlerfrei bei allen bis auf einen Block. Mit einer gewissen Wahrscheinlichkeit befindet sich der Übergang einer strukturellen Veränderung innerhalb eines Blocks und nicht zwischen zwei Blöcken. In einem solchen Fall tritt einer der drei Fälle ein:

1. **Fehler Trennzeichenerkennung:** Die Trennzeichenerkennung schlägt vollständig fehl, da sich innerhalb des analysierten Bereichs das Trennzeichen ändert und keine eindeutige Detektion möglich ist.
2. **Fehler Merkmalsextraktion:** Die Merkmalsextraktion schlägt für einen zu hohen Anteil an Zeilen des Leaks fehl. In diesem Fall wurde ein Trennzeichen erkannt, jedoch kann aufgrund der Veränderung im Block bei vielen Zeilen keine Separation am gefundenen Trennzeichen durchgeführt werden.
3. **Fehler Semantisierung:** Die Semantisierung ist erfolglos, obwohl der vorherige und der anschließende Block erfolgreich semantisiert werden konnten. Grund hierfür ist, dass die Detektionsmodule keine eindeutigen Identitätsmerkmalstypen erkennen können, da in einer Spalte mehrere Merkmalstypen enthalten sind.

Alle drei Fälle erzeugen Fehler, die im Programmablauf effektiv zu erkennen sind und mit einer geeigneten Fehlerbehandlung eventuell behoben werden können. Dazu wird der gesamte Block als ein Problem angesehen, welches nach dem *Teile-und-herrsche-Verfahren (englisch: divide and conquer)* [5] in kleinere Teilprobleme zerteilt wird. Tritt beim Verarbeiten eines Blocks einer der drei Fehler auf, wird dieser Block in der Mitte in zwei kleinere Teilblöcke zerlegt. Tritt in einem Teilblock wiederum einer dieser Fehler auf, wird auch dieser in kleinere Teilblöcke zerlegt. Dieses Vorgehen wird bei auftretenden Fehlern so lange rekursiv wiederholt bis eine

Mindestlänge eines Blocks erreicht wird, bei der das gesamte Verfahren nicht mehr anwendbar ist. Sollte ein Teilblock nicht verarbeitet werden können, so wird dieser verworfen. Als minimale Blocklänge wurden 10, 50, und 100 Zeilen in experimentellen Analysen untersucht. Ergebnis dieser Analyse ist, dass eine Mindestblocklänge von 100 Zeilen die Fehleranfälligkeit reduziert.

5.6 Identifikation des kompromittierten Onlinedienstes

Zur Warnung von Betroffenen ist es hilfreich, diesen mitzuteilen, bei welchem Dienst die Identitätsdaten abhandengekommen sind. Auch kann die Kenntnis, bei welchem Dienst die Identitätsdaten entwendet worden sind, genutzt werden, um den entsprechenden Dienst zu informieren. Auf Grund dessen soll in diesem Abschnitt untersucht werden, wie aus vorliegenden Identitätsdaten-Leaks ein Dienst detektiert werden kann, bei dem die Daten abhandengekommen sind. Dazu werden zwei verschiedene Ansätze vorgestellt, die *Domain-Detektion* in Unterabschnitt 5.6.1 und die *Dienstnamen-Detektion* in Unterabschnitt 5.6.2.

5.6.1 Domain-Detektion

Eine Benennung der Identitätsdaten-Leaks findet in aller Regel mittels der vorgefundenen Datei- oder Ordnernamen statt. Beispielsweise könnte ein Dateipfad eines Leaks folgendermaßen aufgebaut sein: */Collection #1_NEW combo Dumps/uni-bonn.de.txt*. Aus diesem Dateipfad kann anhand der Domain im Dateinamen darauf geschlossen werden, dass in diesem Beispiel der Identitätsdaten-Leak Identitäten enthält, die zu Diensten der Universität Bonn gehören. Zur Erkennung solcher Domains kann kein simpler regulärer Ausdruck verwendet werden. Problematisch ist die Unterscheidung zwischen Top-Level-Domains wie *.de* und Dateiendungen wie *.txt*. Es muss somit überprüft werden, ob eine Domain-Endung tatsächlich eine gültige Top-Level-Domain darstellt.

Zur Erkennung des Dienstes wird der Dateipfad als Zeichenkette mit einem regulären Ausdruck für Domains durchsucht. Handelt es sich bei der Syntax um eine gültige Domain, wird analysiert, ob die Top-Level-Domain gültig ist. Ist diese

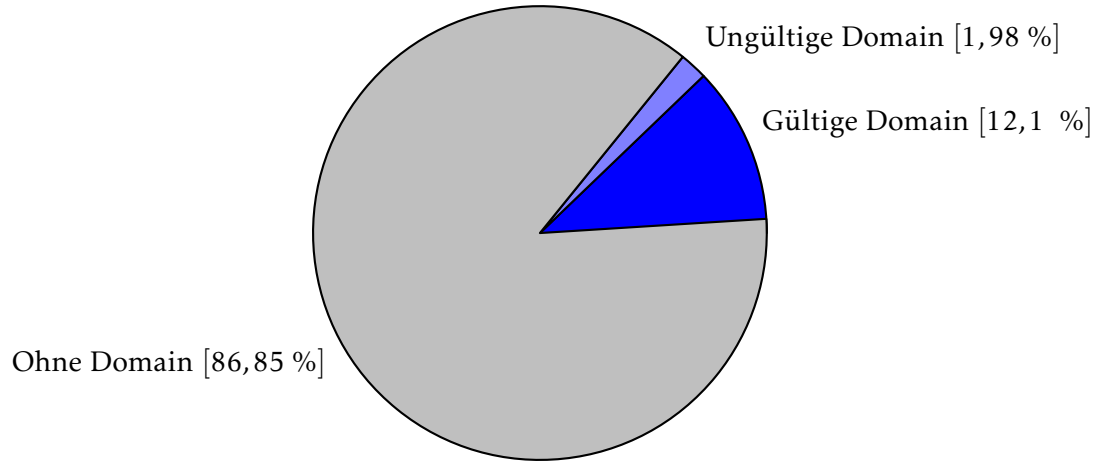


Abbildung 13: *Anteile an Dateipfaden mit integrierter Domain des betroffenen Dienstes.*

Anforderung erfüllt, wird abschließend getestet, ob es gültige DNS-Records zu dieser Domain gibt. Abgefragt wird der *A-Record* [77] und als Fallback der *MX-Record* [77]. Bei einer erfolgreichen Antwort des DNS-Servers kann davon ausgegangen werden, dass ein entsprechender Dienst unter der gefundenen Domain angeboten wird.

Eine Untersuchung aller vorliegenden 84.802 Dateipfade nach diesem Konzept wurde am 22.05.2020 durchgeführt. Verwendet wurde der DNS-Server `8.8.8.8` von `google.com`. Das Ergebnis ist, dass in 10.328 Dateipfaden eine Domain gefunden wurde, zu der ein DNS-Eintrag existiert. Das entspricht 12,18 % der gesamten Dateipfade. In weiteren 1.676 Dateipfaden wurden Domains mit vorhandener Top-Level-Domain identifiziert, jedoch existierte hierzu zum Zeitpunkt der Überprüfung kein DNS-Eintrag (1,98 %). Diese Aufteilung ist in Abbildung 13 dargestellt. Bei genauerer Analyse der Dateipfade fällt auf, dass eine große Menge von Leak-Dateien ausschließlich eine Nummer als Dateinamen besitzt, beispielsweise `8917859.txt`. Insgesamt besitzen 64.613 Dateien (76,19 %) einen solchen numerischen Dateinamen. Von diesen 64.613 Dateipfaden enthalten nur 81 eine gültige Domain. Alle gefundenen Domains sind dabei in Ordnernamen enthalten. Der Grund für die verhältnismäßig geringe Anzahl ist, dass sich diese große Menge von Dateien auf insgesamt nur 109 Ordner und Unterordner verteilt. Von diesen 109 Ordnerpfaden enthalten nur 33 eine gültige Domain.

5.6.2 Dienstnamen-Detektion

Ein anderer Ansatz zur Dienstidentifizierung im Dateipfad ist, den Pfad nach bekannten Dienstnamen zu durchsuchen. Dazu wurde die Liste *The Majestic Million*[2] verwendet (geladen am 23.05.2020), welche die Domains der eine Millionen meist besuchten Websites auflistet. Diese Liste enthält vollständige Domains. Um nun eine Liste mit bekannten Dienstnamen zu erhalten, werden die Second-Level-Domains aus dieser Liste extrahiert. Jedoch enthalten Listen dieser Größe auch Dienstnamen, die normale Wörter des Wörterbuchs (*Private*, *China*, *December*) als auch technische Begriffe (*Mail*, *Datenbank*, *Files*) darstellen. Diese Eigenschaft führt dazu, dass in fast jedem Dateipfad ein Dienstname aus der Top-Eine-Millionen-Liste enthalten ist. Um akzeptable Treffer zu erzielen, musste diese Liste auf die Top 10.000 begrenzt werden. Zusätzlich wurden folgende Begriffe aus dieser Liste aufgrund der False-Positive-Treffer entfernt: Datenbank, Mail, Files, China, Japan. Mit diesem Ansatz können 3.218 Dateipfade einem Dienst zugeordnet werden. Jedoch hätten 1.085 dieser Dateipfade auch mit dem Domain-Ansatz zugeordnet werden können (33,72 %). Muss eine Auswahl zwischen beiden Ansätzen getroffen werden, so ist der Domain-Ansatz deutlich effektiver. Insgesamt kann festgestellt werden, dass ein Großteil der vorliegenden Identitätsdaten-Leaks im Dateinamen keine Hinweise auf den kompromittierten Dienst enthält.

5.7 Evaluation des Parsers

Die Funktionsweise des vorgestellten Parsers soll in diesem Kapitel genauer untersucht und evaluiert werden. Dazu wird zunächst ein Beispieldatensatz analysiert, aus dem sich wichtige Kenngrößen ableiten lassen (siehe 5.7.1). Um die Effektivität des Parsers genauer beurteilen zu können, werden die ermittelten Kenngrößen mit Werten anderer Dienste verglichen, die eine ähnliche Analyse durchgeführt haben (siehe 5.7.2). Abschließend wird die Genauigkeit des Parser mit einer Stichprobenuntersuchung genauer quantifiziert (siehe 5.7.2).

[2] Majestic-12 Ltd.: The Majestic Million, https://de.majestic.com/reports/majestic-million (Sichtung: 23.05.2020).

5.7.1 Grundlegende Kenngrössen

Die in Abschnitt 4.3 gesammelten Identitätsdaten-Leaks bestehen aus 84.802 Dateien, die in 3.752 verschiedenen Ordnern und Unterordnern abgelegt sind. Die analysierten Daten bestehen aus insgesamt 28.939.264.258 Zeilen. Unter der Annahme, dass jede dieser Zeilen einen Identitätsdatensatz enthält, stellt diese Zahl das Maximum an extrahierbaren Identitätsdaten dar. Mittels des Parsers werden 23.907.894.602 Identitätsdatensätze aus den Leak-Daten extrahiert. Dies entspricht 82,61 %. In den Daten sind beispielsweise Zeilen enthalten, die ausschließlich eine E-Mail-Adresse enthalten und nicht als kompromittierter Datensatz gezählt werden, da hier ein dazugehöriges Authentifikationsmerkmal fehlt. Auch sind Zeilen enthalten, bei denen keine erkennbaren Identitätsmerkmale enthalten sind. Deshalb sind in den Leak-Daten weniger Identitätsdatensätze als Zeilen enthalten.

Um die Funktionalität des Parsers genauer zu beurteilen, sollen die gelieferten Ergebnisse mit Performancewerten anderer Projekte verglichen werden. Ein von verschiedenen Projekten untersuchter Leak ist *Collection #1*. Dieser Leak setzt sich aus 12.371 Dateien zusammen, hat eine Datenmenge von 91 Gigabyte und besteht aus 2.865.108.725 (2,8 Milliarden) Zeilen.

Für die Evaluation wird dieser Leak mit dem hier vorgestellten Parser verarbeitet. Dazu wird ein System mit 72 Kernen[3] und 756 Gigabyte Hauptspeicher verwendet. In unter 24 Stunden lässt sich der Leak mit diesem Setup verarbeiten. Aus den 2.865.108.725 Zeilen des Leaks extrahiert der Parser 2.649.591.931 Identitätsdatensätze. Somit können 92,47 % der gesamten Zeilen extrahiert werden. Die fehlenden 7,52 % sind dabei nicht zwingend auf Fehlfunktionen des Parsers zurückzuführen. Der Datensatz könnte auch Informationen beinhalten, die keine Identitätsdaten darstellen oder aufgrund syntaktischer Fehler nicht verarbeitbar sind.

Des Weiteren sind in 9.701 Dateien mindestens eine Kombination aus E-Mail-Adresse und Passwort enthalten. Das sind 79,37 % aller Dateien der Collection#1. Diese 79,37 % der gesamten Dateien enthalten 2.610.540.448 Identitätsdatensätze.

[3] 2 mal Intel(R) Xeon(R) CPU E5-2699 v3 @ 2.30GHz

Tabelle 2: *Anzahl Dateien, in denen folgende Identitätsmerkmale erkannt werden.*

Identitätsmerkmal	Anzahl
E-Mail-Adresse	11.725
Passwort	9.749
Hash	2.502
Nicht erkannt	1.720
Nummer	1.490
Gesamtname	526
Datum	302
Telefonnummer	263

Identitätsmerkmal	Anzahl
Domain	261
Benutzername	216
Vorname	151
Nachname	48
Kreditkarte	20
IP-Adressen	11
IBAN	1
Zeitstempel	0

Diese Menge entspricht wiederum 98,52 % aller Identitätsdatensätze. Nur 2.010.597 Identitätsdatensätze (0,08 %) enthalten keine E-Mail-Adresse.

In Tabelle 2 sind die Identitätsmerkmale dargestellt, die der Parser in den insgesamt 12.371 Dateien pro Datei erkennt. Zu sehen ist, dass alle Semantik-Module (siehe Abschnitt 5.4) mit Ausnahme des Moduls für Zeitstempel auch Identitätsmerkmale in der Collection#1 erkennen. E-Mail-Adressen und Passwörter kommen in deutlicher Mehrheit in den Dateien vor.

5.7.2 Vergleich mit anderen Diensten

Als Referenz zu diesen Analysewerten werden die Ergebnisse anderer Projekte herangezogen. Troy Hunt, der Betreiber des Identitätsdaten-Leak-Informationsdienstes *have i been pwned* [50], beschreibt die Auswertung der Collection#1 in einem Artikel [54]. Aus dem Artikel wird deutlich, dass Troy Hunt die in Collection#1 enthaltenen Daten manuell aufbereitet hat, um die Inhalte zu seinem Informationsdienst hinzufügen zu können. Dazu hat er die Daten sortiert, geordnet und mögliche Trennzeichen der Dateien identifiziert. Seine Ergebnisse fasst Troy Hunt folgendermaßen zusammen: „*I found a combination of different delimiter types including colons, semicolons, spaces and indeed a combination of different file types such as delimited text files, files containing SQL statements and other compressed archives*“ [54]. Der für diese Arbeit notwendige manuelle Ressourceneinsatz ist nur schwer zu schätzen, da es

sich um eine umfangreiche Datenmenge handelt. Folgende Werte wurden von ihm zu Collection#1 veröffentlicht:

- Gesamt 2.692.818.238 Zeilen
- 87 GB (vermutlich sind 87 GiB [Gibibytes] gemeint - eine Übereinstimmung mit den vorliegenden Daten wäre gegeben)
- 1.160.253.228 (43,09 %) einzigartige (gemeint ist *unique*) Kombinationen von E-Mail-Adressen und Passwörtern
- 772.904.991 (28,70 %) einzigartige E-Mail-Adressen
- 21.222.975 (0,79 %) einzigartige Passwörter

Eine weitere Untersuchung der Firma *Spy-Cloud* zur Collection#1 zeigt ähnliche Ergebnisse [105]:

- 87,16 GB
- 12.368 Dateien
- 2.692.818.238 Zeilen
- 1.013.050.906 (37,62 %) einzigartige Kombinationen von E-Mail-Adressen und Passwörtern

Zum genauen Vorgehen bei der Analyse wird jedoch nichts bekannt gegeben.

Die mit dem hier vorgestellten Parser extrahierten Daten besitzen 1.001.140.217 einzigartige Kombinationen aus E-Mail-Adressen und Passwörtern. Des Weiteren sind 788.702.092 einzigartige E-Mail-Adressen und 369.254.357 einzigartige Passwörter zu finden.

In Tabelle 3 sind die aus Collection#1 extrahierten einzigartigen E-Mail-Adressen, Passwörter und deren Kombination als Vergleich der Ergebnisse von *have i been pwned*, *Spycloud* und diesem Parser dargestellt. Angaben zu der Anzahl einzigartiger E-Mail-Adressen und Passwörter konnten nur bei *have i been pwned* gefunden werden. Der hier vorgestellte Parser extrahiert aus Collection#1 insgesamt 2,04 % mehr einzigartige E-Mail-Adressen und 1639 % mehr einzigartige Passwörter als *have i been pwned*. Die geringe Anzahl an extrahierten Passwörtern liegt vermutlich

Tabelle 3: *Vergleich von Extraktionsergebnissen bei Collection#1.*

	HIBP[A]	Spycloud[B]	Vorliegende Arbeit
Anzahl einzigartiger E-Mail-Adressen	772.904.991	k.A.	788.702.092
Anzahl einzigartiger Passwörter	21.222.975	k.A.	369.254.357
Anzahl einzigartiger E-Mail-Adressen & Passwort-Kombination	1.160.253.228	1.013.050.906	1.001.140.217

[A] Troy Hunt für *have i been pwned* [54].
[B] Spycloud [105].

an dem manuellen Vorgehen beim Verarbeiten der Daten, da bei Passwörtern keine syntaktische Erkennung möglich ist.

Werden die einzigartigen Kombinationen von E-Mail-Adressen und Passwörtern verglichen, liefert der hier präsentierte Parser 1,18 % weniger Kombinationen als *Spycloud* angibt. Dieser geringe Unterschied ist eventuell auf einen spezifischeren in dieser Arbeit verwendeten regulären Ausdruck zurückzuführen. Wird der Wert mit den Werten von Hunt [54] verglichen, werden 13,71 % weniger Kombinationen gefunden. Jedoch ist fraglich, wie die angegebene Menge an Daten mit einer manuellen Verarbeitung extrahiert wurde. Auffällig ist, das die gesamte Collection#1 nur 1.269.219.170 eindeutige Zeilen erhält. Werden aus diesen Zeilen alle Zeilen herausgesucht, die ein `@` enthalten, verbleiben eine Anzahl von 1.177.124.803 Stück. Dieser Wert weicht um 1,45 % von dem von Troy Hunt angegebenem Wert ab. Eventuell hat Hunt zur Ermittlung seiner Angabe zu den einzigartigen Kombinationen von E-Mail-Adresse und Passwort gar keine Identitätsdaten aus Collection#1 extrahiert, sondern ausschließlich die eindeutigen Zeilen in Collection#1 gezählt, die ein `@` enthalten und nicht länger als eine gewisse Längenbegrenzung sind. Aus diesem Grund wird die von Hunt angegebene Anzahl als kein Wert betrachtet, der sich zu einem sinnvollen Vergleich mit dem hier vorgestellten Parser eignet.

5.7.3 Genauigkeit des Parsers

Um die Funktionalität des Parsers besser beurteilen zu können, müssen die vom Parser extrahierten und klassifizierten Identitätsmerkmale auf Korrektheit überprüft werden. Da nicht alle 2,6 Milliarden extrahierten Datensätze aus Collection#1 manuell bewertet werden können, wird eine Stichprobe aus diesen Daten genauer untersucht. Dazu werden zufällige Zeilen aus den Rohdaten der Collection#1 ausgewählt, die dann manuell mit den Ergebnissen in der Datenbank verglichen werden. Bei der zufälligen Auswahl kann jede Zeile in den Rohdaten mit der gleich hohen Wahrscheinlichkeit ausgewählt werden. Das bedeutet, dass die Anzahl enthaltener Zeilen in einer Datei keinen Einfluss auf die Wahrscheinlichkeit hat, mit der die enthaltenen Zeilen ausgewählt werden. Um jedoch den Aufwand möglichst gering zu halten wird ein Werkzeug genutzt, welches eine zufällige Zeile aus den Rohdaten auswählt und anschließend den entsprechenden Eintrag aus der Datenbank heraussucht. Beides wird zur manuellen Überprüfung auf dem Bildschirm ausgegeben.

Zur Berechnung des benötigten Stichprobenumfangs wird die Gleichung 5.1 [13] herangezogen:

$$n = \frac{z^2 \cdot p(1-p)}{e^2} \tag{5.1}$$

Die Irrtumswahrscheinlichkeit wird gewählt als $\alpha = 0,01$, da davon auszugehen ist, dass eine manuelle Begutachtung nur in seltenen Fällen zu einem falschen Ergebnis führt. Mit α lässt sich aus Tabellen für die Standardnormalverteilung ein Konfidenzniveau von $z = 2,58$ ablesen. Als Fehlerspanne wird $e = 0,01$ gewählt, da die Aussage des Ergebnisses möglichst genau sein soll. Da nichts über die tatsächliche Genauigkeit des Parsers bekannt ist, wird $p = 0,5$ gewählt. Aus diesen Eingabeparametern lässt sich mit der gewählten Formel ein benötigter Stichprobenumfang von $n = 16.641$ ableiten. Aufgrund der Größe der Grundgesamtheit von $N = 2.649.591.931$ kann auf eine komplexere Gleichung zum Berechnen der Stichprobengröße verzichtet werden.

Insgesamt wurden 16.858 zufällig ausgewählte Zeilen der Rohdaten mit den Ergebnissen in der Datenbank verglichen. Davon wurden 16.506 Datenbankeinträge als *korrekt erkannt* bewertet. Lediglich 352 Datenbankeinträge sind als *falsch erkannt* bewertet worden. Hieraus lässt sich eine Genauigkeit des Parsers von 97,91 % ableiten.

5.8 Zusammenfassung

In diesem Kapitel wird ein vollständiges Konzept für einen Parser vorgestellt, der vollautomatisiert Identitätsdaten-Leaks analysiert und die darin enthaltenen Identitätsdaten extrahiert. Dazu werden Eigenschaften von Identitätsdaten-Leaks manuell erhoben, um aufbauend auf diesem Wissen ein Konzept für einen Parser zu erarbeiten. Aufbauend darauf wird ein Konzept vorgestellt, welches Trennzeichen in Identitätsdaten-Leaks erkennt, Identitätsmerkmale extrahiert und ihnen eine semantische Bedeutung zuordnet.

Zur Evaluation dieses Konzepts wird ein Identitätsdaten-Leak ausgewählt, zu dem zwei veröffentlichte Analysen zu finden sind. Dieser Identitätsdaten-Leak wird mit dem vorgestellten Parser verarbeitet. Die Resultate werden anschließend mit den Werten aus den genannten Analysen verglichen. Ergebnis hiervon ist, dass der hier vorgestellte Parser bessere Ergebnisse liefert als ein manuelles Verfahren von Hunt (vgl. [54]). Des Weiteren liefert dieser Parser ähnliche Ergebnisse wie die von *Spycloud* (vgl. [105]). Abschließend wird die Genauigkeit des Parsers mit einer Stichprobenuntersuchung festgestellt. Das hier vorgestellte Konzept klassifiziert Collection#1 zu 97,91 % korrekt.

6 Proaktive Warnung von Betroffenen

Dieses Kapitel basiert auf den bereits veröffentlichten Arbeiten „Warning of Affected Users About an Identity Leak“ [69] und „Effektive Warnung bei Identitätsdiebstahl an Hochschulen“ [71].

In Kapitel 5 wird gezeigt, wie große Datenmengen an Identitätsdaten-Leaks automatisiert für die Identifikation betroffener Personen aufbereitet werden können. Auf Grundlage der gegebenen Datenbasis wird in diesem Kapitel ein Konzept entworfen, mit dem eine Warnung der betroffenen Personen umgesetzt werden kann. Dazu wird zunächst diskutiert, welche Kommunikationskanäle für eine solche Warnung geeignet sind. Auf Basis dieser Kommunikationskanäle wird anschließend ein Konzept erarbeitet, welches das Ziel zur automatisierten Frühwarnung verfolgt. Nachdem das Konzept vorgestellt wurde, werden die genutzten Protokolle erläutert.

6.1 Grundlegende Ideen und Herleitung des Konzepts

In dieser Arbeit soll ein System zum Schutz von Benutzern vor Identitätsdatenmissbrauch entwickelt werden. Zur Realisierung dieses Systems muss ein geeignetes Vorgehen erarbeitet werden. Dazu wird in diesem Abschnitt diskutiert, welche Arten von Kommunikationskanälen sich für eine Warnung von Betroffenen grundsätzlich eignen.

Liegt eine umfangreiche Menge an gestohlenen Identitätsdaten (siehe Kapitel 4) vor, muss ein automatisiertes Vorgehen zur Warnung entwickelt werden, um die hohe Stückzahl an kompromittierten Identitätsdaten bearbeiten zu können. Eine Idee,

die mit geringen Mitteln realisierbar wäre, ist die direkte Warnung der Betroffenen per E-Mail. Die meisten Datensätze enthalten eine E-Mail-Adresse, die theoretisch genutzt werden kann, um die betroffene Person zu erreichen. Praktisch kann eine Warnung per E-Mail von Beginn an ausgeschlossen werden. Der erste Grund hierfür ist, dass ein Großteil der Benutzer eine solche E-Mail als Spam klassifizieren würde, da sie den Absender nicht kennen und ihm höchstwahrscheinlich kein Vertrauen schenken würden. Ein zweiter Grund ist, dass das Versenden von E-Mails keine triviale Aufgabe darstellt, sobald mehrere Millionen bis Milliarden Nachrichten versendet werden müssten. Die Wahrscheinlichkeit ist sehr groß, dass viele Spamfilter solche Massen-E-Mails zurückhalten würden.

Der Hauptgrund für das Scheitern des genannten Vorgehens ist das Fehlen eines Vertrauensverhältnisses zwischen dem betroffenen Benutzer und der warnenden Instanz. Ein Benutzer wird einem unbekannten Absender vermutlich weniger vertrauen als beispielsweise einem bereits bekannten Onlinedienst, wenn dieser über kompromittierte Identitätsdaten berichtet. Ist eine Person bereits Kunde bei einem Onlinedienst, ist der Empfang von Nachrichten dieses Onlinedienstes in der Regel nichts Ungewöhnliches. Es eignen sich somit keine Kommunikationskanäle, bei denen der Betroffene kein Vertrauen zum Kommunikationspartner besitzt. Aus diesen Überlegungen kann abgeleitet werden, dass es sinnvoll ist, die warnende Instanz und den für die Opfer unbekannten Frühwarndienst voneinander zu trennen.

Es ist anzunehmen, dass bei einem Benutzer ein gewisses Vertrauen zu einem Onlinedienst vorhanden ist, wenn dieser ein Benutzerkonto bei diesem Dienst besitzt. Sollte ein solcher Onlinedienst eine Warnung über einen zwischen Benutzer und Dienst etablierten Kommunikationskanal versenden, ist das Öffnen und Lesen dieser Nachricht deutlich wahrscheinlicher. Die Idee für ein solches System ist, dass ein zentralisierter Frühwarndienst bei einem Fund eine Warnung an kooperierende Onlinedienste herausgibt, die dann die betroffenen Personen warnen.

Vorteilhaft an diesem Konzept ist, dass die Opfer von jemandem informiert werden, zu dem ein gewisses Vertrauensverhältnis besteht, da der Benutzer dem Onlinedienst mindestens eine E-Mail-Adresse und ein Passwort anvertraut hat und diesen Dienst kennt. Zusätzlich hat der kooperierende Onlinedienst die Möglichkeit,

technische und organisatorische Maßnahmen bei gefährdeten Benutzerkonten zu ergreifen, um den Kunden und die eigene Infrastruktur zu schützen. Eine solche Maßnahme könnte das Deaktivieren einzelner sicherheitskritischer Funktionen betroffener Benutzerkonten sein wie das Bezahlen auf Rechnung oder das Ändern der hinterlegten E-Mail-Adresse. Ebenso ist die Deaktivierung eines Benutzerskontos denkbar, bis die betroffene Person sich erneut legitimiert und ein neues Passwort gesetzt hat.

Die kooperierenden Onlinedienste bieten deren Benutzern in der Regel einen Login an, bei dem ein Identifikator wie E-Mail-Adresse oder Telefonnummer und ein Passwort zur Authentifikation verwendet wird. Diese Kombination aus Identitätsdaten ist häufig in Identitätsdaten-Leaks enthalten. Zum Schutz vor unberechtigtem Zugriff auf Benutzerkonten sollte ein Onlinedienst genau diese Kombinationen von Identitätsmerkmalen dahingehend überprüfen, ob sie bereits in Identitätsdaten-Leaks enthalten sind.

6.2 Konzept eines Frühwarnsystems

Im vorherigen Abschnitt werden erste Ideen für die Funktionsweise eines Frühwarnsystems genannt. Aufbauend darauf werden im Folgenden genauere Überlegungen und das konkrete Konzept für ein solches System vorgestellt.

Die Hauptidee des Konzepts [71, 69] ist der Betrieb eines zentralisierten Dienstes, welcher die Aufgabe besitzt, die neusten Identitätsdaten-Leaks nach dem Vorgehen aus Kapitel 4 zu sammeln und diese mittels der in Kapitel 5 vorgestellten Konzepte eines Parsers automatisiert für entsprechende Warnungen aufzubereiten. Dieser zentralisierte Dienst wird im Folgenden als *Frühwarndienst* bezeichnet.

In Abbildung 14 sind die Interaktionen des Frühwarndienstes dargestellt. Das Sammeln neuer Leaks wird in der Grafik durch (1) dargestellt. Online-Services wie Online-Shops und Soziale Netzwerke können die Ergebnisse des Frühwarndienstes nutzen, wenn sie sich dem Frühwarndienst als Kooperationspartner zur Verfügung stellen. Im Falle eines neuen Leaks würde der Frühwarndienst die aufbereiteten Identitätsdaten an die kooperierenden Onlinedienste weiterleiten (2). Die Online-

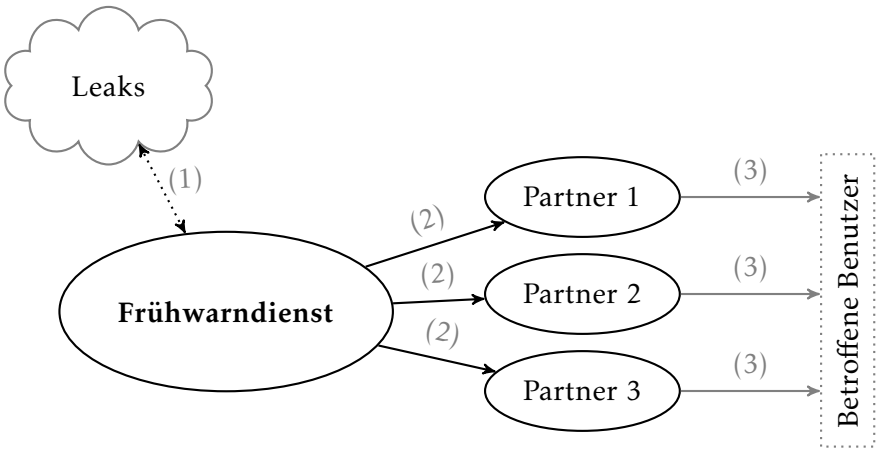


Abbildung 14: *Aufbau eines Frühwarndienstes [71, 69].*

dienste können anschließend überprüfen, ob in den enthaltenen Datensätzen eigene Benutzer bzw. Kunden enthalten sind. Sollte dieser Fall eintreten, können geeignete Maßnahmen eingeleitet werden (3). Abhängig von der Art des betroffenen Benutzerkontos sind verschiedene Gegenmaßnahmen notwendig. Eventuell reicht eine Warnung des Kunden per E-Mail. Jedoch besitzt der kooperierende Onlinedienst an dieser Stelle die Möglichkeit weitere Maßnahmen zu ergreifen, um die eigene Infrastruktur zu schützen.

An ein solches Verfahren gibt es bestimmte Anforderungen, von denen die wichtigsten kurz beschrieben werden sollen. Zunächst muss sichergestellt werden, dass tatsächlich die betroffene Person benachrichtigt wird. Mit der Umsetzung dieses Aspekts sollen Falschwarnungen vermieden werden. Solche Falschwarnungen könnten entstehen, wenn beispielweise nur die Aliasse der E-Mail-Adressen genutzt werden, um ein kompromittiertes Benutzerkonto festzustellen. An dieser Stelle können Überschneidungen entstehen, wenn verschiedene Benutzer einen identischen Alias bei verschiedenen Mail-Providern verwenden. Beispielsweise können die E-Mail-Adressen `max@gmx.de` und `max@googlemail.com` zwei verschiedenen Personen gehören. Wird bei einer Überprüfung der Kompromittierung nur der Alias `max` verwendet, besteht die Gefahr, dass der Falsche der beiden E-Mail-Inhaber informiert wird.

Die in den Identitätsdaten-Leaks enthaltenen personenbezogenen Daten stellen hochsensible Informationen dar. Aus diesem Grund ist ein bestmöglicher Schutz dieser Daten wünschenswert. Europäische Verordnungen wie die DSGVO fordern ebenfalls einen besonderen Schutz dieser Daten. Das Konzept sieht jedoch vor, dass die aufbereiteten Leak-Daten mit den kooperierenden Onlinediensten geteilt werden. Um zu verhindern, dass diese einen ungehinderten Einblick in all diese Daten bekommen, können kryptografische Verfahren verwendet werden.

Eine weitere Anforderung an ein solches System ist die Einhaltung datenschutzrechtlicher Aspekte. Um Onlinedienste zur Teilnahme an einem solchen Netzwerk zu motivieren, muss das eingesetzte Verfahren so gestaltet werden, dass der zentrale Frühwarndienst zu keinem Zeitpunkt Kenntnis über die Benutzer und Kunden des kooperierenden Onlinedienstes bekommt. Der Frühwarndienst wäre sonst bei genügend kooperierenden Onlinediensten in der Lage, Benutzerprofile zu bilden. Auch wenn sich ein solch konstruierter Dienst ohne datenschutzrechtliche Einwände umsetzen lässt, ist die Akzeptanz zur Nutzung des Frühwarndienstes bei potenziellen wirtschaftlichen Kooperationspartnern vermutlich geringer.

Wichtig bei einer Warnung ist außerdem, dass sie nur erfolgt, wenn die in den Leaks enthaltenen Zugangsdaten tatsächlich für eine Anmeldung bei einem Dienst genutzt werden können. Es ist nicht ausreichend festzustellen, dass beispielsweise die E-Mail-Adresse eines Kunden enthalten ist und ihn darüber zu informieren, obwohl das im Leak vorhandene Passwort gar nicht für einen erfolgreichen Login genutzt werden kann. Die Gefahr hierbei ist zu groß, dass Benutzer von vielen kooperierenden Onlinediensten Mehrfachwarnungen erhalten, die alle nicht valide sind. Hierdurch könnten Benutzer eine persönliche Resistenz gegen solche Informationen entwickeln. Bei einer späteren Betroffenheit könnten diese Benutzer die Warnungen ignorieren, anstatt notwendige Maßnahmen einzuleiten. Onlinedienste sollten in ihrer Benutzerdatenbank keine Klartextpasswörter, sondern nur Hash-Werte abgespeichert haben. Damit haben die Onlinedienste keinen Zugriff auf die Klartextpasswörter ihrer Benutzer. Diese technische Eigenschaft muss bei der Konzeption eines Frühwarnsystems beachtet werden.

Die zuvor dargestellten Anforderungen lassen sich folgendermaßen zusammenfassen:

(A1) Betroffene müssen eindeutig identifiziert werden können.

(A2) Personenbezogene Informationen aus Leak-Daten müssen angemessen geschützt werden.

(A3) Der Frühwarndienst darf keine Kenntnis über Kunden der kooperierenden Onlinedienste erhalten.

(A4) Warnung darf nur erfolgen, wenn gestohlene Zugangsdaten beim kooperierenden Onlinedienst zur Anmeldung genutzt werden können.

Diese Anforderungen müssen in ein technisches Konzept überführt werden. Zunächst muss überlegt werden, auf welchem Übertragungsweg die Daten zwischen dem Frühwarndienst und den kooperierenden Onlinediensten übermittelt werden. Da eine vollständige Automatisierung aufgrund der umfangreichen Datenmengen erforderlich ist, bietet sich an, eine REST-API zu nutzen. Vorteil hierbei ist die Transportverschlüsselung mittels `https`. Darüber hinaus lässt sich eine genaue Spezifikation erstellen, wie die Endpunkte dieser API definiert sind. Um die Last auf der Seite des Frühwarndienstes zu reduzieren, wird durch jeden kooperierenden Onlinedienst eine eigene API betrieben. In einem solchen Fall kann der Frühwarndienst abhängig von der eigenen Auslastung entscheiden, wann er einem kooperierenden Onlinedienst neue Daten übermittelt. Eine Lastverteilung ist auf diese Weise gezielter umsetzbar.

Ein weiterer Vorteil dieses Ansatzes ist, dass zu keinem Zeitpunkt Benutzerdaten von kooperierenden Onlinediensten an den Frühwarndienst gesendet werden. Ein kooperierender Onlinedienst erhält ausschließlich Identitätsdaten. Lediglich eine statistische Rückmeldung von Trefferquoten ist vorgesehen. Es besteht somit keine Möglichkeit, dass das Frühwarnsystem Informationen oder Meta-Informationen über Kunden oder den Kundenstamm gewinnen kann. Damit ist die Anforderung *A3* erfüllt. Die weiteren Anforderungen werden nachfolgend untersucht.

6.3 Leak-Warn-Protokoll

Bei der genauen Umsetzung des vorgestellten Konzepts müssen die zuvor formulierten Anforderungen berücksichtigt werden. Insbesondere die Anforderungen *A2* und *A4* schließen einige denkbare Umsetzungsmöglichkeiten aus. Nach dem vorgestellten Konzept werden alle aus Identitätsdaten-Leaks extrahierten Identitätsdaten an die kooperierenden Onlinedienste geschickt. Würden alle extrahierten Daten im Klartext übertragen, würden die Onlinedienste an personenbezogene Daten von Personen gelangen, die nicht zum eigenen Kundenstamm gehören. Aus diesem Grund fordert Anforderung *A2*, dass das Erlangen unnötiger Kenntnis über Identitätsdaten durch kooperierende Onlinedienste verhindert wird. Dies kann durch den Einsatz kryptographischer Verfahren realisiert werden, indem Klartextdaten verschleiert werden. Dazu bieten sich Techniken zur Pseudonymisierung an [98]. Konkret können die in Kapitel 2 genannten Hash-Verfahren zur Pseudonymisierung verwendet werden. Eine Hash-Funktion überführt einen Klartext eines Identitätsmerkmals in eine Zeichenkette, die den Klartext eindeutig repräsentiert. Die hier benötigte Eigenschaft von Hash-Funktionen ist, dass der Hash-Wert nicht mehr in den Klartext zurückgerechnet werden kann. Aus diesem Grund wird der Hash-Wert an dieser Stelle als nicht aufdeckbares Pseudonym bezeichnet.

Werden die in den Identitätsdaten-Leaks enthaltenen Daten vor der Übertragung mittels Hash-Verfahren pseudonymisiert, können die kooperierenden Onlinedienste keine Informationen über fremde Personen erlangen. Wollen die kooperierenden Onlinedienste überprüfen, ob eigene Benutzer in den Daten enthalten sind, müssen diese ebenfalls Hash-Werte der vorliegenden Benutzerdaten erzeugen. Dazu nutzen sie die in der eigenen Benutzerdatenbank gespeicherten Identifikatoren der Benutzer. Dies können beispielsweise die E-Mail-Adressen, Benutzernamen oder Telefonnummern sein. Von diesen Klartextdaten werden unter Verwendung der gleichen Hash-Verfahren wie zuvor die entsprechenden Pseudonyme generiert. Wenn eine Übereinstimmung zwischen einem durch den Onlinedienst generierten Hash-Wert mit einem aus den empfangenen Daten vorliegt, kann der Onlinedienst darauf schließen, dass die Daten des betreffenden Kunden in den Leak-Daten enthalten sind.

Will der Onlinedienst überprüfen, ob ein zum eigenen Dienst passendes Passwort in dem auffälligen Datensatz enthalten ist, kann das Verfahren der nicht aufdeckbaren Pseudonymisierung nicht verwendet werden. Grund dafür ist, dass der kooperierende Onlinedienst dazu das Klartextpasswort des betroffenen Kunden kennen muss, da sonst kein Hash-Wert des Passworts zur Überprüfung berechnet werden kann. Eine Idee zur Lösung dieses Problems wäre, dass der Frühwarndienst die Passwort-Hash-Werte mit dem identischen Vorgehen erzeugt, mit dem der kooperierende Onlinedienst seine Passwort-Hash-Werte erstellt, um sie in der Benutzerdatenbank abzuspeichern. Jedoch müssten dazu die Onlinedienste das verwendete Verfahren mit allen Parametern offenlegen. Dazu werden die wenigsten Onlinedienste bereit sein, da die Geheimhaltung der Verfahren häufig als Schutz der Klartextpasswörter betrachtet wird. Deshalb muss ein anderes Verfahren verwendet werden, um zu überprüfen, ob ein Leak-Datensatz valide Anmeldedaten für den Dienst eines Kooperationspartners enthält.

Als Verfahren eignet sich an dieser Stelle eine Verschlüsselung des Klartextpassworts. Ist ein Passwort verschlüsselt, kann das erhaltene Chiffrat nicht ohne den passenden Schlüssel zurück in das Klartextpasswort überführt werden. Als Schlüssel eignet sich ein Klartext eines Identifikators, da dieser einem kooperierenden Onlinedienst nur bekannt ist, wenn der Inhaber des Identifikators selbst Benutzer bei dem Onlinedienst ist. Liegen in einem Datensatz eine E-Mail-Adresse und ein Passwort vor, wird das Passwort mit dem Klartext der E-Mail-Adresse verschlüsselt. An den kooperierenden Onlinedienst wird das Chiffrat des Passworts und der Hash-Wert der E-Mail-Adresse übertragen. Zur Überprüfung vergleicht der Onlinedienst den Hash-Wert der E-Mail-Adresse mit allen Hash-Werten, die sich aus allen E-Mail-Adressen der eigenen Benutzer berechnen lassen. Gibt es eine Übereinstimmung, weiß der Onlinedienst, welcher Benutzer betroffen ist. Der Onlinedienst kennt an der Stelle die Klartext-E-Mail-Adresse und kann diese als Schlüssel verwenden, um das Chiffrat des Passworts zu entschlüsseln. Analog zur E-Mail-Adresse kann dieses Verfahren für alle Identifikatoren eingesetzt werden.

In Abbildung 15 ist das dargestellte Verfahren als Kommunikationsprotokoll abgebildet. Dieses wird im Folgenden genauer erläutert [71, 69]. In der Abbildung

werden im initialen Schritt (1) zwischen dem Frühwarndienst und dem kooperierenden Onlinedienst Geheimnisse ausgetauscht, sogenannte *Shared-Secrets*. Diese werden als zusätzliches Attribut bei den Hash-Verfahren genutzt, um die Hash-Werte resistenter gegen Wörterbuch-Angriffe zu machen. Dieses Shared-Secret wird dabei als sogenannter *Salt* eingesetzt. In einem zweiten initialen Schritt muss der Onlinedienst die Benutzeridentifikatoren aus seiner Benutzerdatenbank mit dem vereinbarten Hash-Verfahren unter Verwendung des Shared-Secrets in Hash-Werte überführen, die ebenfalls in der Benutzerdatenbank abgespeichert werden können (2). Der Frühwarndienst sammelt neue Identitätsdaten-Leaks (3) und verarbeitet diese entsprechend Kapitel 5 (4). Liegen die verarbeiteten Leak-Daten dem Frühwarndienst vor, beginnt die Datenübertragung an den Onlinedienst. Zunächst werden Metadaten über den betreffenden Leak gesendet (5). In diesen Daten ist die Größe des Leaks als auch eine Beispielliste von im Leak vorhandenen E-Mail-Adressen als Hash-Wert enthalten (5). Dies dient dazu, um den Systemen der Onlinedienste eine Einschätzung zu ermöglichen, welche Last ihnen bevorsteht, wenn sie die Übertragung der Daten beginnen lassen. Sollten einmal nicht genügend Ressourcen beim kooperierenden Onlinedienst zur Verfügung stehen, kann an dieser Stelle die Übertragung des Leaks auf einen späteren Zeitpunkt verschoben werden. Wird der Leak vom Onlinedienst akzeptiert, werden die Daten in (6) abschnittsweise vom Frühwarndienst an den Onlinedienst übertragen. Zuletzt überprüft der Onlinedienst, ob in den empfangenen Daten die Hash-Werte von Identifikatoren eigener Benutzer enthalten sind. Ist dies der Fall, kann der Klartext des Identifikators genutzt werden, um das Chiffrat des Passworts zu entschlüsseln. Wenn die Kombination aus Identifikator und Passwort zur Anmeldung genutzt werden kann, können durch den kooperierenden Onlinedienst geeignete Maßnahmen eingeleitet werden.

6.4 Technische Umsetzung des Protokolls

In diesem Abschnitt soll auf eine technische Umsetzung dieses Konzepts unter Einsatz kryptografischer Verfahren eingegangen werden. Dabei wird versucht, die technischen Details so grob zu beschreiben, dass jeder technisch versierte Leser die Inhalte nachvollziehen kann, jedoch soll dabei darauf verzichtet werden zu umfangreich auf die Grundlagen einzugehen.

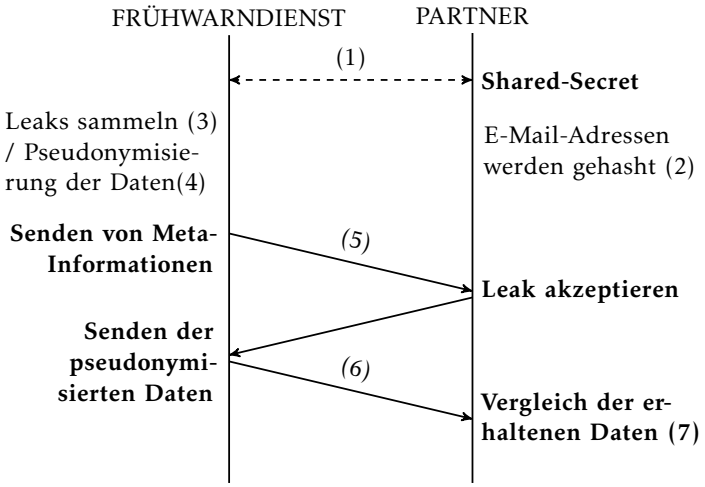


Abbildung 15: *Kommunikationsprotokoll zum Identitätsdatenaustausch [71, 69].*

6.4.1 Protokoll beim Frühwarndienst

Eine konkrete Umsetzung des technischen Konzepts [18] ist in Abbildung 16 dargestellt und wird in diesem Abschnitt erläutert. Diese Umsetzung wird bereits mit einem kooperierenden Onlinedienst im Kontext des Forschungsprojektes verwendet [18]. In der Grafik ist das Verfahren in fünf Phasen eingeteilt: *Ergebnis Parser*, *Vorbereitung Speicherung*, *Speicherung Datenbank*, *Vorbereitung Übertragung* und *Übertragung Partner*. In der ersten Phase liefert der Parser aus Kapitel 5 extrahierte Identitätsmerkmale. Von Interesse sind an dieser Stelle nur die in den Datensätzen enthaltenen Identifikatoren und Passwörter. Da die Daten noch nicht in den persistenten Speicher abgelegt wurden, wird im nächsten Schritt die Speicherung vorbereitet. Um die personenbezogenen Daten geeignet zu schützen, werden diese vor der Speicherung pseudonymisiert. Dazu werden die Identifikatoren mit dem ressourcenintensiven Hash-Verfahren `Argon2` [8] zunächst in Hash-Werte überführt.

Anschließend muss wie vorgestellt das Passwort (PW) mit einem Identifikator (ID) verschlüsselt werden. Als Verschlüsselungsalgorithmus wird `AES-128-CBC` [35] eingesetzt. Dieser Verschlüsselungsalgorithmus fordert, dass die verwendeten Schlüssel eine Länge von 128, 192 oder 256 Bits besitzen. Da ein Identifikator jedoch eine fast beliebige Länge besitzen kann, muss aus diesem zunächst ein geeigneter Schlüssel

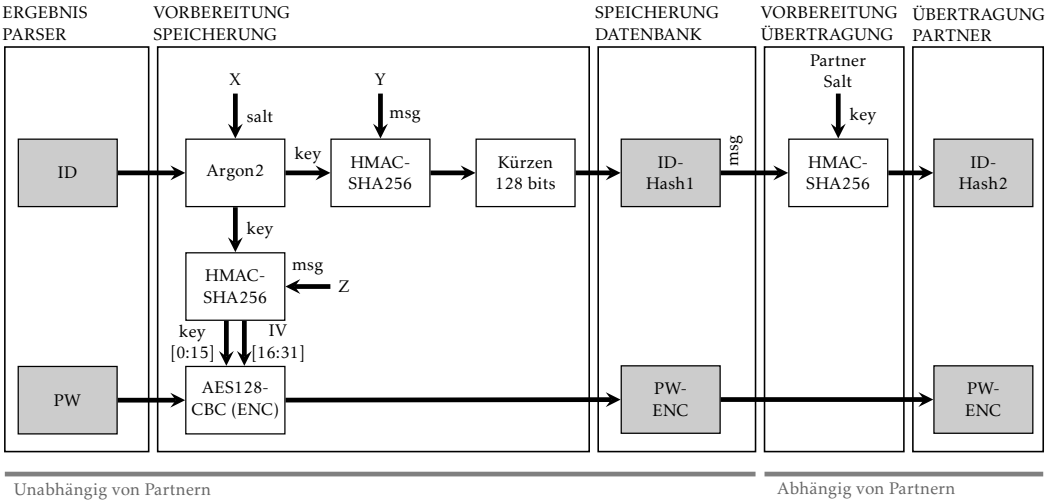


Abbildung 16: *Kryptografische Umsetzung des Protokolls seitens des Frühwarndienstes [18].*

mit spezifizierter Länge abgeleitet werden. `Argon2` bietet sich ebenfalls als Funktion zur Schlüsselableitung an [8]. Da bereits ein mit `Argon2` berechneter Hash-Wert eines Identifikators vorliegt, wird dieser zur Ressourcenschonung als Schlüssel für die Verschlüsselung verwendet. Würde jedoch dieser Hash-Wert unverändert als Schlüssel verwendet und als Identifikator in der Datenbank abgespeichert, würden die Chiffrate der Passwörter direkt neben den zugehörigen Schlüsseln abgespeichert. Aus diesem Grund werden mittels der beiden `HMAC-SHA-256` [62, 1] in Schritt *Vorbereitung Speicherung* zwei getrennte Hash-Werte aus dem zuvor berechneten `Argon2`-Hash-Wert abgeleitet. Da beide Funktionen unumkehrbar sind, kann der in der Datenbank gespeicherte Hash-Wert nicht als Schlüssel verwendet werden.

Für die Verschlüsselung des Klartextpassworts wird der aus dem Identifikator generierte `HMAC-SHA-256` in zwei Teile getrennt. Die ersten 128 Bits werden als Schlüssel verwendet. Die letzten 128 Bits werden als Initialisierungsvektor (IV) für den `CBC`-Modus genutzt. Mit dieser Eingabe wird das vorliegende Passwort (PW) verschlüsselt und in der Datenbank als Chiffrat (PW-ENC) abgelegt.

Der andere HMAC wird im Schritt *Kürzen 128 Bits* auf eine Länge von 128 Bits reduziert. Grund hierfür ist die Reduzierung des benötigten Speicherbedarfes in der Datenbank auf die Hälfte. Die Gefahr von daraus entstehenden Hash-Kollisionen wird aufgrund des beschränkten Eingabe-Universums als gering und für diesen Anwendungsfall als vertretbar eingeschätzt. Dieser 128 Bit lange Hash-Wert wird zusammen mit dem Passwort-Chiffrat (PW-ENC) in der Datenbank abspeichert (ID-Hash1), um diese für den Versand an kooperierende Onlinedienste bereitzuhalten. Die bis hierhin beschriebenen Schritte werden pro Identitätsdaten-Leak einmal durchlaufen und sind von den kooperierenden Onlinediensten unabhängig.

Werden Daten aus der Datenbank an einen Onlinedienst versendet, findet vor dem Versand eine für den jeweiligen Onlinedienst individuelle Verarbeitung statt. Dazu wird der Hash-Wert aus der Datenbank (ID-Hash1) mit einem weiteren HMAC-SHA-256 verarbeitet. Der hieraus resultierende Hash-Wert (ID-Hash2) wird zusammen mit dem verschlüsselten Passwort (PW-ENC) an den Partner übertragen. Dieser letzte HMAC-SHA-256 wird mit einem Onlinedienst-spezifischen Salt versehen, um das Vorberechnen von Hash-Werten als Angriff zu erschweren.

6.4.2 Protokoll bei kooperierenden Onlinediensten

Nachdem die technische Umsetzung auf der Seite des Frühwarndienstes besprochen wurde, ist im Folgenden dargestellt, welche Operationen ein kooperierender Onlinedienst durchführen muss, um eine Überprüfung der Login-Daten durchzuführen. Dazu muss ein Onlinedienst zunächst alle Identifikatoren seiner Benutzer in einen Hash-Wert überführen. Das Verfahren ist in Abbildung 17 abgebildet.

Ein Onlinedienst wendet auf jeden seiner Benutzeridentifikatoren die identischen Verarbeitungsschritte wie der Frühwarndienst an, um anschließend sämtliche Identifikatoren als Hash-Wert (ID-Hash2) vorliegen zu haben. Dazu wendet er zunächst Argon2 auf einen Identifikator an. Dieses Ergebnis wird mit einem HMAC-SHA-256 verarbeitet und auf die Hälfte gekürzt. Abschließend wird dieser gekürzte Hash-Wert mit einem weiteren HMAC-SHA-256 verarbeitet, jedoch mit dem Onlinedienst-spezifischen Salt. Dieses Ergebnis wird in der Datenbank im Eintrag des entsprechenden Benutzers abgespeichert. Wichtig ist, dass dabei dieselben

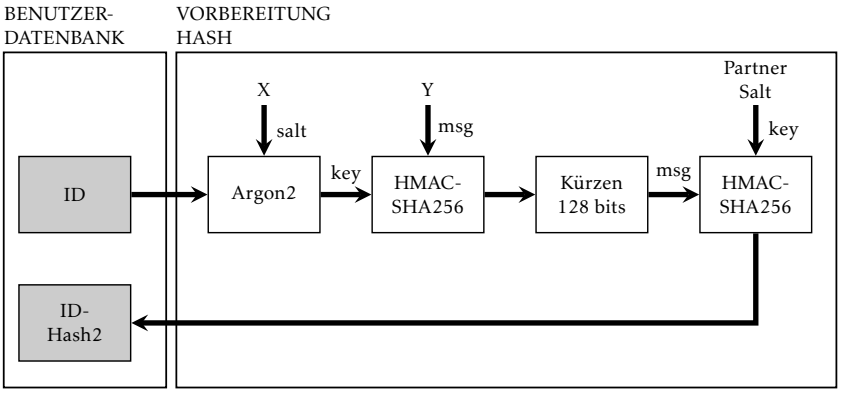


Abbildung 17: *Kryptografische Umsetzung der Vorbereitung beim kooperierenden Onlinedienst.*

Shared-Secrets verwendet werden, wie sie durch den Frühwarndienst eingesetzt werden. Sendet der Frühwarndienst einem Onlinedienst Identitätsdaten-Leaks, erhält der Onlinedienst einen Hash-Wert (ID-Hash2) und ein verschlüsseltes Passwort (PW-ENC). Dies ist in Abbildung 18 auf der linken Seite dargestellt.

Um zu überprüfen, ob ein einzelner Datensatz in den empfangenen Daten kompromittiert wurde, muss der kooperierende Onlinedienst den enthaltenen Hash-Wert des Identifikators mit allen Hash-Werten in der eigenen Benutzerdatenbank vergleichen (Lookup). Gibt es dabei einen Treffer, kann aus dem erhaltenen Hash-Wert des Identifikators auf den Klartext des Identifikators geschlossen werden. Der nun vorliegende Identifikator wird verwendet, um das verschlüsselte Passwort zu entschlüsseln. Allerdings muss zuerst aus dem Identifikator ein geeigneter Schlüssel abgeleitet werden. Dazu wird analog zum Vorgehen beim Frühwarndienst der Identifikator mit `Argon2` verarbeitet und anschließend mit einem `HMAC-SHA-256`. Der erhaltene Schlüssel (Key) und der erhaltene Initialisierungsvektor (IV) können verwendet werden, um mittels `AES-128-CBC` das Chiffrat des Passworts (PW-ENC) zu entschlüsseln (AES-128-CBC (DEC)). Als Ergebnis des ganzen Verfahrens liegen final ein Identifikator und ein Passwort vor. Diese Anmeldedaten müssen überprüft werden, ob mit ihnen eine erfolgreiche Anmeldung möglich ist. Ist dies der Fall, kann der Onlinedienst wie zuvor beschrieben geeignet darauf reagieren.

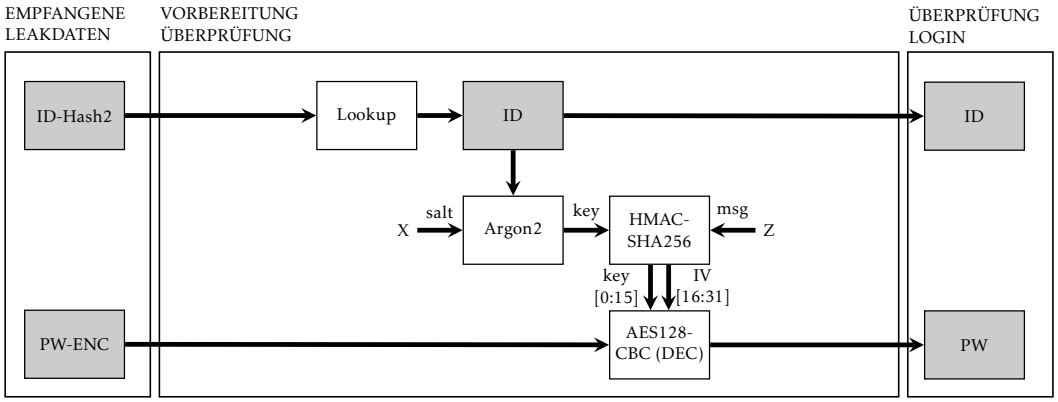


Abbildung 18: *Kryptografische Umsetzung der Überprüfung von Leak-Daten durch kooperierenden Onlinedienst.*

6.5 Speicherung der Identitätsdaten-Leaks

In Abschnitt 5.2 wird das Gesamtkonzept für den Identitätsdaten-Leak-Parser vorgestellt. Die Ergebnisse des dort vorgestellten Systems werden an das *Output-Modul* weitergegeben. Da die hier beschriebenen kryptografischen Elemente in Abschnitt 5.2 noch nicht besprochen werden, wird das genaue Konzept des *Output-Moduls* in diesem Abschnitt genauer beschrieben.

Generell können die Ergebnisse des Parsers direkt in einer Datenbank gespeichert werden. Allerdings liegt dann eine Datenbank vor, die eine sehr umfangreiche Menge an Klartext-Identitätsdaten enthält. Da dies datenschutzrechtlich problematisch ist und ein solches Frühwarnsystem kein attraktives Ziel für Angreifer werden will, müssen die Daten in einer pseudonymisierten Form abgespeichert werden. Dazu bietet sich das in diesem Kapitel vorgestellte Verfahren an (siehe Unterabschnitt 6.4.1). Hierbei werden Hash-Werte der Identifikatoren und die mit dem Identifikator verschlüsselten Passwörter in der Datenbank abgelegt.

Wichtig ist hierbei, dass sowohl beim Frühwarndienst als auch bei den kooperierenden Onlinediensten eine Normalisierung der Identifikatoren vor der Berechnung der Hash-Werte erfolgt. Hiermit ist gemeint, dass die Identifikatoren vor der Verarbeitung in ein einheitliches Format gebracht werden. Grund dafür ist, dass ein

gleicher Identifikator in unterschiedlichen Darstellungsformen vorliegen kann. Ein Beispiel hierfür sind folgende E-Mail-Adressen: 1) `max.mustermann@uni-bonn.de` 2) `Max.Mustermann@uni-bonn.de`. Beide dieser E-Mail-Adressen beschreiben den gleichen Identitätsmerkmals-Besitzer, jedoch würden beide E-Mail-Adressen zwei völlig verschiedene Hash-Werte erhalten. Aus diesem Grund muss ein Vorgehen zur Normalisierung definiert sein, um Fehler basierend auf einer unterschiedlichen Formatierung zu vermeiden.

Das vorgestellte Konzept verschlüsselt das Klartextpasswort mit einem Schlüssel, der aus dem dazugehörigen Identifikator abgeleitet wird. Für Brute-Force- oder Wörterbuch-Angriffe ist der Suchraum des Schlüssels zum Entschlüsseln der Klartextpasswörter somit auf all die Zeichenketten beschränkt, die einen validen Identifikator repräsentieren. Angreifer müssten für einen erfolgreichen Angriff nur sämtliche Identifikatoren durchprobieren, anstatt alle beliebigen Zeichenketten als Schlüssel zu testen. Da jedoch der Schlüssel mittels `Argon2` aus dem Identifikator abgeleitet wird, müssten Angreifer für jeden Versuch bei Brute-Force- oder Wörterbuch-Angriffen einen `Argon2`-Hash ermitteln. Dieses Vorgehen benötigt einen hohen Aufwand, da die Berechnung von `Argon2`-Hash-Werten wie zuvor dargestellt mit einem hohen Ressourceneinsatz verbunden ist. Der Aufwand für einen solchen Angriff übersteigt den Aufwand um ein Vielfaches, den ein Angreifer benötigen würde, um sich die Rohdaten selbst herunterzuladen. Aus diesem Grund wird davon ausgegangen, dass mit diesem Verfahren die Identitätsdaten für den vorliegenden Anwendungsfall ausreichend geschützt sind. Eine detailliertere Sicherheitsbetrachtung ist in Abschnitt 6.7 zu finden.

6.6 Schnittstelle

In dem vorgestellten Konzept betreibt jeder kooperierende Onlinedienst eine API, welche von dem Frühwarndienst dazu genutzt wird, um an diese die verarbeiteten Daten zu senden. Eine solche Schnittstelle kann als *REST-API* gestaltet und implementiert werden [19]. Eine *REST-API* [33] ist ein Architektur-Paradigma für Schnittstellen, um verteilte Systeme mittels *HTTP* und *HTTPS* [26, 31, 32, 30, 27, 28, 29] und *URI* [6, 82] kommunizieren zu lassen. Hierzu werden *URIs* als Res-

sourcen gesehen, welche mittels der HTTP-Abfragemethoden *GET*, *POST*, *PUSH*, *DELETE* [32] abgefragt, verändert, angelegt oder gelöscht werden können. Diese im Projekt eingesetzte REST-API stellt für jeden in Abschnitt 6.3 beschriebenen Verarbeitungsschritt einen eigenen API-Endpunkt dar.

Die genaue API setzt sich aus sechs Endpunkten zusammen, die wie folgt aufgebaut sind [19]:

1. `GET /supported_id_types`
   Diese Ressource gibt eine Liste aller vom Onlinedienst verwendbaren Identifikationsmerkmale an, die für die Warnkanäle des Onlinedienstes nützlich sind. So müssen beispielsweise einem Onlinedienst keine Kreditkartennummern gesendet werden, wenn dieser gar keine Kreditkartennummern von den Kunden abgespeichert hat.
2. `POST /upload_sample`
   Bei diesem API-Endpunkt werden Meta-Informationen und eine Beispielliste von Hash-Werten von im Identitätsdaten-Leak enthaltenen Identifikatoren mitgesendet. Die Datensätze werden im Datenformat *JSON* in den *Request Body* der Anfrage integriert.
3. `GET /check_sample/{sample_token}`
   Nachdem eine Beispielliste an den vorherigen API-Endpunkt gesendet wurde, kann bei dieser Ressource abgefragt werden, ob die Beispielliste bereits von den Systemen des Onlinedienstes verarbeitet worden ist. Ist das der Fall, liefert dieser Endpunkt einen `Upload-Token` zurück, wenn der Onlinedienst den Leak erhalten möchte. Die Anforderung an die Möglichkeit zum Übertragen eines Samples vor der eigentlichen Übertragung ist von einem kooperierenden Onlinedienst im Forschungsprojekt *EIDI* formuliert worden.
4. `POST /upload/{upload_token}`
   Mit dem gelieferten `Upload-Token` können an diesem Endpunkt die pseudonymisierten Leak-Daten an den kooperierenden Onlinedienst übertragen werden. Hierbei muss nicht der ganze Datensatz auf einmal übertragen werden, sondern es kann jeder Leak abschnittsweise im JSON-Format übertragen werden. Dazu wird dieser Endpunkt mehrfach aufgerufen.

5. `POST /finish_upload/{upload_token}`
   Hat der Frühwarndienst alle Abschnitte eines Identitätsdaten-Leaks übertragen, dann wird dies dem Onlinedienst mit dem Aufruf dieses Endpunktes mitgeteilt. Als Rückgabewert wird ein `Result-Token` übergeben, der im nächsten Schritt zur Anwendung kommt.
6. `GET /result/{result_token}`
   Will der Frühwarndienst Feedback zu einem übertragenen Leak erhalten, kann dieser Endpunkt dazu genutzt werden. Abhängig vom kooperierenden Onlinedienst können hier unterschiedliche Informationen mitgeteilt werden.

6.7 Sicherheit und Angriffsvektoren

Die genannten kryptografischen Maßnahmen sollen den Schutz der personenbezogenen Daten und des Warnsystems erhöhen. In diesem Abschnitt sollen die beschriebenen Maßnahmen auf mögliche Angriffsvektoren untersucht werden. Besprochen werden solche Risiken, die eine Bedrohung für die Vertraulichkeit der Identitätsdaten darstellen. Grundlegende Risiken des Betreibens von technischen Infrastrukturen werden nicht diskutiert.

Kompromittierung des Frühwarndienstes

Zunächst muss überlegt werden, welche Auswirkung eine Kompromittierung des Frühwarndienstes auf die Vertraulichkeit der personenbezogenen Identitätsdaten besitzt. Ausgehend von dem Fall, dass einem Angreifer gelingt, Zugriff auf die Datenbank zu erhalten, in der alle Identitätsdaten gespeichert sind, können Auswirkungen diskutiert werden. Sollte es gelingen, die vollständige Datenbank zu entwenden, ist fraglich, welcher Nutzen sich für den Angreifer aus diesen Daten ziehen lässt. Gelingt es ihm nur die Datenbank zu entwenden, sind die Daten für ihn nahezu wertlos, da die verwendeten Shared-Secrets nicht in der Datenbank gespeichert werden. Sollte dieser zusätzlich Kenntnis über die verwendeten Shared-Secrets erhalten, indem er diese im System des Parsers ausfindig macht oder über den Weg des Onlinedienstes an sie herankommt, ist er in der Lage Wörterbuchangriffe auf

die Datenbank zu starten. Das heißt, er kann große Listen von E-Mail-Adressen und anderen Identifikatoren ausprobieren, indem er die Hash-Werte der einzelnen Identifikatoren berechnet. Sollten Übereinstimmungen gefunden werden, kann für den betreffenden Datensatz das Passwort entschlüsselt werden. Um alle in der Datenbank enthaltenen Datensätze in den Klartext zu überführen, müssten dem Angreifer sämtliche Identifikatoren im Klartext vorliegen.

An dieser Stelle muss auf das Aufwand-Nutzen-Verhältnis hingewiesen werden, welches sich für einen solchen Angreifer ergibt. Die Aufwände für den Angreifer, um die Datenbank zu entwenden und anschließend einzelne Einträge zu rekonstruieren, sind deutlich umfangreicher als die Aufwände, um sich die Rohdaten im Klartext im Internet zu besorgen. Die in der Datenbank pseudonymisiert gespeicherten Daten sind aus öffentlichen und frei zugänglichen Quellen geladen worden. Für einen Angreifer sollte dieser Weg deutlich effizienter sein.

Kompromittierung eines kooperierenden Onlinedienstes

Werden die Systeme eines kooperierenden Onlinedienstes mit dem Ziel angegriffen, die vom Frühwarndienst erhaltenen Daten zu entwenden, steht der Angreifer vor folgenden Problemen: Ein kooperierender Onlinedienst speichert die vom Frühwarndienst erhaltenen Daten nur so lange, bis eine vollständige Überprüfung der Daten abgeschlossen ist. Danach müssen die Daten auf der Seite eines kooperierenden Onlinedienstes aus Datenschutzgründen gelöscht werden, da sie nach Überprüfung ihren Zweck erfüllt haben. Eine Sammlung von Identitätsdaten in einem Umfang wie beim Frühwarndienst sollte bei den Onlinediensten nicht zu finden sein. Ein Angreifer kann lediglich die Daten mitlesen, die ab dem Zeitpunkt der Kompromittierung empfangen werden. Da die Daten jedoch nur vollständig pseudonymisiert gesendet werden, muss der Angreifer entsprechende Identifikatoren vorliegen haben, um an die in den Datensätzen enthaltenen Passwörter zu gelangen. An dieser Stelle gestaltet es sich für einen Angreifer als deutlich effizienter, sich die entsprechenden Identitätsdaten-Leaks selbst im Internet zu beschaffen und damit einen Vollzugriff auf die Klartextdaten zu erlangen.

Maliziöser Onlinedienst

Sollte es einem Angreifer gelingen, selbst als maliziöser Onlinedienst Teil des Warn-Netzwerks zu werden, steht dieser vor den gleichen Herausforderungen wie die zuvor dargestellten Angreifer. Der Angreifer bekommt als Onlinedienst nur pseudonymisierte Daten, die mittels aufwändigen Wörterbuchangriffen in Klartext überführt werden müssen. Die genaue Ausgestaltung, wie jemand zum Kooperationspartner des Warn-Netzwerks wird, ist nicht Teil dieser Arbeit. Allerdings kann gesagt werden, dass es als nicht sinnvoll erscheint, diesen Dienst für jeden frei zugänglich zu machen. Hier bieten sich Zugangshürden an wie beispielsweise eine Vertragsschließung nur mit Unternehmen, welche im Handelsregister geführt werden.

Wörterbuchangriffe im Allgemeinen

Bei der Gestaltung des kryptografischen Konzepts wird explizit *Argon2* als Hash-Verfahren gewählt. *Argon2* besitzt den Vorteil, dass unter anderem der für eine Berechnung benötigte Arbeitsspeicher und die benötigte Ausführungszeit konfigurierbar sind [8]. Der größte Rechenaufwand liegt beim Frühwarndienst, da dieser beim Verarbeiten von neuen Identitätsdaten-Leaks jeden extrahierten Identifikator mit *Argon2* verarbeiten muss. Die kooperierenden Onlinedienste dagegen müssen lediglich sämtliche Identifikatoren ihrer Benutzer mit diesem Verfahren verarbeiten. Ein durchschnittlicher Onlinedienst besitzt bestenfalls ein paar Millionen Benutzer. Ein Identitätsdaten-Leak kann schnell mehrere Milliarden Datensätze enthalten. Anhand dieser Überlegungen können die bei *Argon2* verfügbaren Parameter an die vorhandene Hardware beim Frühwarndienst angepasst werden, sodass ein durchschnittlich großer Leak in einer annehmbaren Zeit verarbeitet werden kann. Pro enthaltenem Datensatz muss der Frühwarndienst einmal einen Hash-Wert mittels *Argon2* berechnen.

Will ein Angreifer die in der Datenbank des Frühwarndienstes enthaltenen Hash-Werte auf ihre Klartexte zurückführen, muss dieser Wörterbuch- oder sogar Brute-Force-Angriffe durchführen. Um einen Treffer bei einem solchen Angriff zu erzielen, muss er eine deutlich größere Anzahl an Versuchen starten. Dabei ist die Erfolgsquo-

te stark abhängig von den eingesetzten Wörterbüchern, jedoch kann angenommen werden, dass wesentlich mehr als ein Versuch pro Treffer notwendig ist.

6.8 Vergleich mit anderen Konzepten

In der Literatur werden Protokolle vorgestellt, die ähnliche, aber nicht identische Ziele wie das vorgestellte Verfahren verfolgen. Der schon zuvor genannte Identitätsdaten-Leak-Informationsdienst *have i been pwned* bietet ebenfalls eine API an, um clientseitige Abfragen an den Server stellen zu können. Dabei sind zwei verschiedene Typen von Abfragen möglich. Mit der ersten Möglichkeit können E-Mail-Adressen und Benutzernamen abgefragt werden, um herauszufinden, in welchen Identitätsdaten-Leaks diese enthalten sind [51]. Dabei werden ausschließlich Meta-Informationen zurückgeliefert. Die zweite Abfragemöglichkeit ist die Abfrage von Passwörtern, um zu überprüfen, ob diese bereits in Leaks verbreitet wurden [51]. Hierbei kommt ein Protokoll zum Einsatz, welches mittels des Konzepts der *k-Anonymität* den genauen Inhalt der Abfrage verbergen soll. Das Verfahren ist so aufgebaut, dass das abzufragende Passwort mit einem festgelegten Hash-Verfahren in einen Hash-Wert überführt wird [52]. Von diesem Hash-Wert werden nur die ersten Zeichen der Hexadezimaldarstellung des Hash-Werts für eine Abfrage verwendet. Hierbei kommt eine Länge des Hash-Präfixes von fünf Zeichen zum Einsatz [51, 52]. An die API wird eine Abfrage für diesen Hash-Präfix gestellt. Als Ergebnis wird eine Liste von Hash-Werten zurückgeliefert, die genau mit dem angefragten Hash-Präfix beginnen [51, 52]. Durch die Anzahl an Daten, die dieser Dienst in der Datenbank vorhält, werden mindestens 381 und maximal 584 Hash-Werte zurückgeliefert [52]. Somit ist hierbei eine *k-Anonymität* von 381 gegeben. Diese API wird unter anderem von *Mozilla* für den *Firefox Monitor* [78] verwendet. Die API ist ebenfalls im Passwortmanager *1Password* integriert [102]. Eine schematische Darstellung des Kommunikationsablaufs ist in Abbildung 19 zu finden.

Ein weiteres Protokoll wird von *Google* in ihrer *Password-Checkup*-Browser-Erweiterung eingesetzt [40, 91]. Dieses Protokoll setzt genauso eine API ein, die clientseitig abgefragt werden kann. Anders als bei der API von *have i been pwned* lässt sich mit dieser API die Kompromittierung einer Kombination aus E-Mail-Adresse

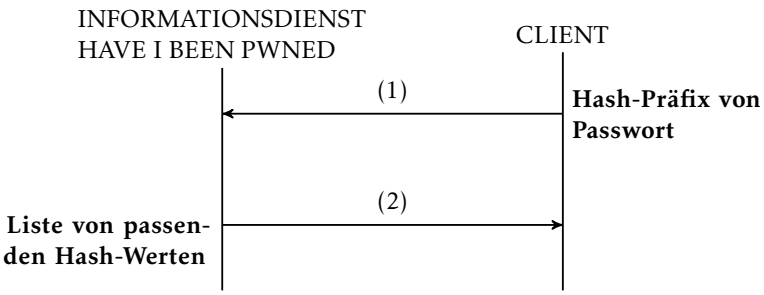


Abbildung 19: *Kommunikationsprotokoll der API von have i been pwned [50].*

und Passwort feststellen [114]. Hierzu wird auf Seiten des Clients die E-Mail-Adresse zusammen mit dem Passwort in einen gemeinsamen Hash-Wert überführt, von dem wiederum ein Präfix an die API gesendet wird [114]. Zusätzlich kommen bei diesem Verfahren noch kryptografische Techniken des Blindings zum Einsatz [114], um die Identitätsdaten des Dienstes stärker zu schützen. Blinding ist ein Verfahren, um beispielsweise Dokumente zu signieren, ohne dabei die Möglichkeit des Einblicks in das Dokument zu bekommen. Dazu wird die eigentliche Nachricht vor dem Versand an den Empfänger mit einem Blinding-Faktor multipliziert [97]. Enthält der Absender die signierte Nachricht vom Empfänger zurück, kann der Blinding-Faktor mittels Division wieder aus der Nachricht entfernt werden [97]. Bei Googles Plugin wird ein blinded Hash-Wert aus E-Mail-Adresse und Passwort ebenfalls an die API übertragen. Auf die genaue Beschreibung der kryptografischen Ausgestaltung dieses Protokolls soll an dieser Stelle verzichtet werden. Jedoch ist in Abbildung 20 eine schematische Darstellung der API-Nutzung dargestellt. Gemeinsam haben beide vorgestellten Protokolle, dass sie Präfixe der Hash-Werte nutzen, um Anfragen an die jeweilige API zu stellen. In der Literatur gibt es Kritik daran, bei solchen Protokollen eine Abfrage mittels Hash-Präfix zu realisieren, da dadurch die Gefahr von Credential-Stuffing-Angriffen deutlich zunimmt [64]. Beispielsweise könnte ein Angreifer seine Credential-Stuffing-Angriffe deutlich effizienter gestalten, wenn er mögliche Passwörter von Beginn an ausschließen kann, weil diese einen falschen Hash-Präfix besitzen. Für solche Angriffe muss der Angreifer allerdings Kenntnis über den angefragten Hash-Präfix bekommen.

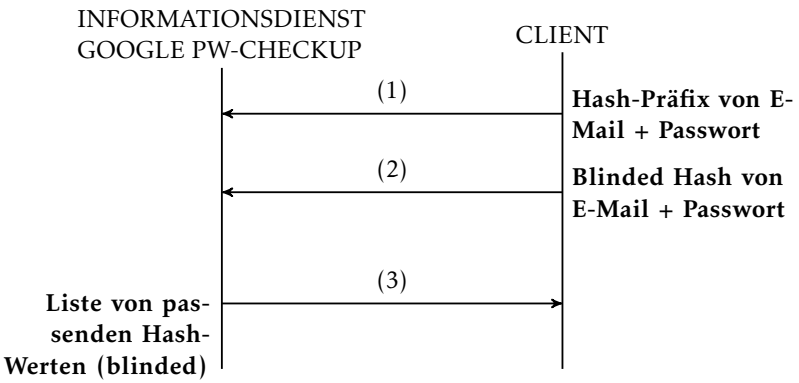


Abbildung 20: *Kommunikationsprotokoll der API von Googles Password-Checkup [114].*

Ein weiteres Protokoll mit gleichem Zweck, aber etwas anderer Umsetzung ist durch die API des Unternehmens *Enzoic* definiert [21]. Die durch dieses Unternehmen betriebene API erlaubt unter anderem ebenso eine Abfrage, ob die Kombination aus Identifikator und Passwort bereits in einem Identitätsdaten-Leak enthalten ist. Eine Überprüfung setzt sich aus insgesamt zwei API-Abfragen zusammen. Zunächst wird der Klartext oder der Hash-Wert eines Identifikators an die API gesendet [21]. Als Ergebnis werden Hash-Verfahren und Shared-Secrets zurückgeliefert, die für die zweite API-Abfrage notwendig sind [21]. Für die zweite Abfrage werden die Konkatenation von Identifikator und Passwort mit dem erhaltenen Verfahren in einen Hash-Wert überführt [21]. Von diesem Hash-Wert werden dann in der zweiten Anfrage die ersten zehn Stellen der Hexadezimaldarstellung an die API gesendet, woraufhin anschließend alle Datensätze mitgeteilt werden, die mit diesem Hash-Präfix in der Leak-Datenbank des Unternehmens enthalten sind [21].

An diesem Protokoll können mehrere Aspekte kritisch beurteilt werden, da aus diesen ein Bedrohungspotenzial entsteht, welches bei den zuvor dargestellten Protokollen nicht vorhanden ist. Der erste kritische Aspekt dieses Protokolls ist, dass eine Überprüfung im ersten Teil vorsieht, dem Dienst den Identifikator des Benutzers mitzuteilen, welcher überprüft werden soll. So wäre die Firma *Enzoic* in der Lage, Profile über alle angefragten Identifikatoren anzulegen, bei welchen Diensten diese Identifikatoren genutzt werden. Zusätzlich ist die Möglichkeit vorhanden,

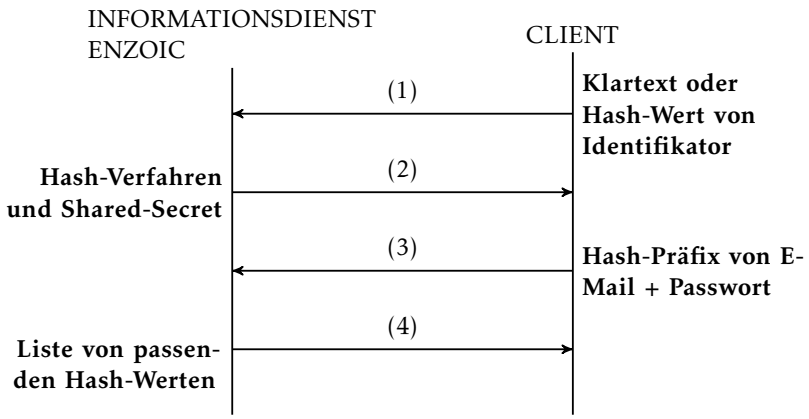


Abbildung 21: *Kommunikationsprotokoll der API von Enzoic [21].*

festzustellen, bei welchen Diensten die gleiche Kombination aus Identifikator und Passwort genutzt wird. Wäre *Enzoic* ein maliziöser Dienst, dann ermöglichen die durch erhaltene Anfragen gesammelten Daten einen hilfreichen Ausgangspunkt für Credential-Stuffing-Angriffe. Ein maliziöser Dienstanbieter erhält den Identifikator, einen Präfix des Hash-Werts von Identifikator und Passwort als auch den Dienst, bei dem Identifikator und Passwort zur Anmeldung genutzt werden können. Das Kommunikationsprotokoll ist in Abbildung 21 zu finden. Zu diesem Dienst sei abschließend genannt, dass der bekannte Passwort-Manager *LastPass*[1] den vorgestellten Dienst zum Schutz der eigenen Kunden nutzt [23, 111].

Ein weiterer am Markt tätiger US-amerikanischer Dienst heißt *SpyCloud*. Dieser bietet ebenfalls eine API an, mit der Identitätsdaten auf eine Kompromittierung überprüft werden können [106]. Bei der API wird jedoch auf sämtliche Pseudonymisierungsmethoden verzichtet. Eine API-Anfrage wird laut der Website mit einem Klartext eines Identifikators gestellt. Die API antwortet mit allen Inhalten, die der Dienst zu diesem Identifikator in der Datenbank gespeichert hat. Alle Daten werden dabei im Klartext übertragen. *SpyCloud* extrahiert Identitätsmerkmale aus Leak-Daten, die zu Informationen des persönlichen Lebensbereichs zählen wie beispielsweise die Religionszugehörigkeit [2]. Vermutlich sind diese Informationen im Klartext in der Antwort der API enthalten. Der Ablauf der API-Anfrage ist in

[1] LastPass: https://www.lastpass.com.

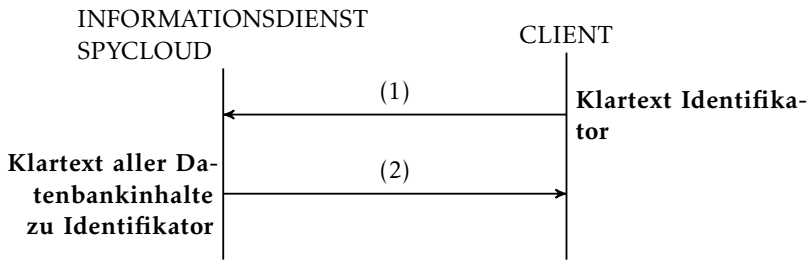


Abbildung 22: *Kommunikationsprotokoll der API von Spycloud [106].*

Abbildung 22 dargestellt. Ein API-Nutzer ist damit in der Lage, Informationen über alle in der Datenbank gespeicherten Personen zu erhalten. Das Betreiben eines solchen Dienstes müsste in einer gesonderten Arbeit ethisch diskutiert werden.

Die vier vorgestellten Leak-Informationsdienste haben alle gemeinsam, dass ein Dienst, der seine Benutzer und Mitarbeiter schützen will, für alle zu überprüfenden Identitätsdaten einzelne API-Anfragen stellen muss. Bei jedem dieser Dienste müssen dazu Informationen des Benutzers preisgegeben werden, seien es Hash-Präfixes oder Benutzernamen. Inwieweit ein in der europäischen Union ansässiges Unternehmen diese Dienste nutzen darf, soll in dieser Arbeit aufgrund des juristischen Schwerpunkts nicht diskutiert werden. Jedoch sei angemerkt, dass die Herausgabe von personenbezogenen Daten ohne Einwilligung äußerst problematisch aufgrund der Datenschutzgrundverordnung sein könnte.

Troy Hunt, der Betreiber von *have i been pwned*, hat diese Problematik erkannt und bietet seitdem eine vollständige Liste mit Hash-Werten aller dem Dienst vorliegenden Passwörter zum Download an [53]. So können Passwörter überprüft werden, ob sie bereits in einem Leak vorhanden sind. Da sämtliche Überprüfungen offline stattfinden, müssen keine Informationen an einen externen Dienst gegeben werden.

Im Weiteren sollen die Vor- und Nachteile der eben vorgestellten Leak-Informationsdienste genannt werden, um diese mit den Vor- und Nachteilen des in diesem Kapitel vorgestellten Konzepts zu vergleichen. Ein Vorteil der vorgestellten Leak-Informationsdienste ist, dass sie dazu genutzt werden können, um die von Benutzern gewählten Anmeldedaten bei einer Neuregistrierung oder einem Passwortwechsel zu überprüfen. Darüber hinaus ist eine Überprüfung der

Anmeldedaten bei jeder Anmeldung eines Benutzers denkbar. Hierbei muss jedoch die verwendete API eine gewisse Performance bieten, um die Anmeldung möglichst nicht für den Benutzer spürbar zu verlangsamen. Da für die Verwendung der vorgestellten APIs der Klartext des Passworts vorliegen muss, kann eine solche Überprüfung nur stattfinden, wenn der Benutzer das Passwort beispielsweise auf einer Anmeldeseite eingegeben hat. Eine regelmäßige Überprüfung ohne Eingabe des Passworts des Benutzers ist nicht möglich, da beispielsweise Online-Services nur einen Hash-Wert der Benutzerpasswörter gespeichert haben sollten und mit diesen Hash-Werten keine Anfrage an die API gestellt werden kann. Ein Nachteil dieses Konzepts ist, dass Benutzer, die nur selten einen Dienst benutzen erst bei ihrer nächsten Anmeldung von der Betroffenheit erfahren. Frühzeitige aber eventuell sehr erfolgreiche Maßnahmen sind so unter Umständen nicht möglich. Erfährt ein Benutzer nicht rechtzeitig von einem Identitätsdaten-Leak, kann er beispielsweise nicht bei anderen Diensten Maßnahmen ergreifen, bei denen er das gleiche Passwort verwendet hat.

Das in dieser Arbeit vorgestellte Protokoll besitzt die Vorteile, dass Benutzer direkt und ohne Verzögerung gewarnt werden können und dabei keine Informationen über die Benutzer geteilt werden müssen. Eine eventuelle Schwäche des Protokolls ist aber, dass die Passwörter von neuen Benutzern nicht mehr überprüft werden können. Registriert sich ein Benutzer bei einem kooperierenden Onlinedienst und wählt bei dem Registrierungsprozess ein Passwort aus, welches vom Benutzer bereits bei anderen Diensten verwendet wird und zusätzlich bereits in Leaks enthalten ist, so kann dieser Dienst nicht davor schützen.

Das passende Konzept ist stark von dem Einsatzszenario abhängig. Dabei muss eventuell abgewogen werden, welche Vorteile als wichtiger eingeschätzt werden. Sicherlich gibt es Anwendungsfälle, in denen Kombinationen aus mehreren Konzepten hilfreich sind.

6.9 Zusammenfassung

Dieses Kapitel stellt ein Konzept für einen zentralen Warndienst vor, der die Aufgabe besitzt, Unternehmen aktuell kompromittierte Identitätsdaten zur Verfügung zu stellen, die in öffentlich verfügbaren Identitätsdaten-Leaks enthalten sind. Unternehmen können durch diese Kooperation die eigene Infrastruktur und die eigenen Benutzer gezielt vor unterschiedlichen Bedrohungen schützen. Zum effizienten Betrieb eines solchen zentralen Frühwarnsystems wird eine vollständige Automatisierung der Verarbeitungsschritte benötigt. Deshalb wird in diesem Kapitel ein Schnittstellen-Design beschrieben, welches für den Austausch von gestohlenen Identitätsdaten genutzt werden kann. Damit die in den Identitätsdaten-Leaks enthaltenen personenbezogenen Daten bei der gesamten Verarbeitung so umfassend wie möglich geschützt werden, werden kryptografische Verfahren verwendet, um die Daten durch Pseudonymisierung vor unautorisierter Kenntnisnahme abzusichern. Im Anschluss werden die Beschränkungen dieses Schutzes und die daraus resultierenden Angriffsvektoren diskutiert. Abschließend wird das entwickelte Protokoll mit anderen in der Literatur beschriebenen Konzepten verglichen. Festzustellen ist, dass diese weiteren Konzepte für andere Kontexte entwickelt wurden und sich mit diesen nicht die hier vorgestellte Idee eines zentralen Frühwarndienst direkt abbilden lässt.

7 Gesamtevaluation

In Kapitel 4 wird ein Verfahren zum Sammeln von Identitätsdaten vorgestellt. Mithilfe dieses Vorgehens konnten 1067 Gigabyte an Identitätsdaten-Leaks gesammelt werden. Das in Kapitel 5 beschriebene System wird anschließend dazu genutzt, um die gesammelten Daten zu analysieren. Mit diesem System können aus den gesammelten Leak-Daten insgesamt 23.907.894.602 Datensätze extrahiert werden. In Kapitel 6 wird ein technisches Konzept vorgestellt, welches dazu genutzt werden kann, um die betroffenen Personen zu schützen. Um die gesamten Konzepte, Verfahren und Systeme in der Praxis zu testen, wird ein Kooperationspartner benötigt.

Ein Mitglied des Projektkonsortiums des Forschungsprojektes *EIDI* [116] ist das Unternehmen *New Work SE*[1]. Das Unternehmen *New Work SE* ist der Betreiber der Plattform *XING*[2]. *XING* ist ein Karriere-Netzwerk und bietet seinen Benutzern unter anderem die Möglichkeit, sich mit anderen Benutzern zu vernetzen. Für diese Plattform werden folgende Benutzerzahlen angegeben [81]:

- Deutschland: Ungefähr 15,5 Millionen
- Schweiz: Mehr als 1 Millionen
- Österreich: Ungefähr 1,5 Millionen

Hieraus lässt sich eine ungefähre Benutzerzahl von insgesamt 18 Millionen Benutzern für den deutschsprachigen Raum ableiten. Aus den vorangegangenen Gründen bietet es sich an, die in dieser Arbeit beschriebenen Verfahren mit der *New Work SE* als Kooperationspartner zu testen.

[1] New Work SE: https://www.new-work.se/de/.
[2] XING: https://xing.com.

7.1 Vorbedingungen

Vor der Durchführung einer Evaluation soll zunächst überlegt werden, was mit *XING* als Kooperationspartner getestet werden kann und welche Vorbedingungen erfüllt sein müssen. Mit dieser Evaluation soll überprüft werden, wie viele Identitätsdatensätze aus den Leak-Daten extrahiert werden können und ob diese Identitätsdatensätze valide Zugangsdaten bei dem Dienst *XING* darstellen.

Eine wichtige Einschränkung, die an dieser Stelle zu nennen ist, bezieht sich auf die zu überprüfenden Leak-Daten und soll an dieser Stelle sehr prägnant dargestellt werden, damit es nicht zu Missverständnissen kommt.

HINWEIS — Kein Leak bei XING
In den gesammelten Identitätsdaten-Leaks konnten **keine Leaks** gefunden werden, die erkennbare Anzeichen oder Hinweise enthalten, dass dieser Leak seinen Ursprung beim ***XING*-Netzwerk** hat. Auch ist in der gesamten Vergangenheit nicht medial darüber berichtet worden, dass Benutzerdaten des *XING*-Netzwerks abhandengekommen sind. Es ist somit davon auszugehen, dass in den gesammelten Leak-Daten keine Datensätze enthalten sind, die aus der Benutzerdatenbank von *XING* stammen oder in irgendeiner anderen Form bei *XING* entwendet wurden.

In der angedachten Untersuchung wird die Validität von in Identitätsdaten-Leaks enthaltenen Benutzerdaten für den Dienst *XING* überprüft. Die zu testenden Benutzerdaten haben ihren Ursprung jedoch nicht beim Dienst *XING*, sondern stammen aus Identitätsdaten-Leaks, die beispielsweise bei anderen Diensten entstanden sind. Wenn ein Datensatz tatsächlich dazu genutzt werden kann, um sich bei *XING* mit den enthaltenen Anmeldedaten zu authentifizieren, hat es höchstwahrscheinlich den Grund, dass der entsprechende Benutzer dieselben Anmeldedaten auch bei anderen Diensten nutzt oder genutzt hat, bei denen es in der Vergangenheit zu einem Identitätsdaten-Leak gekommen ist. Eine weitere Ursache für die erfolgreiche Anmeldung könnte sein, dass die Anmeldedaten während einer Benutzereingabe auf einem maliziösen Client-System mitgelesen, kompromittiert und als Leak veröffentlicht wurden. Auch andere Angriffe sind denkbar. Die Ergebnisse dieser Evaluation

liefern Hinweise darauf, wie viele Benutzer ihr Passwort mehrfach verwenden, obwohl es in einem Leak enthalten ist oder die Benutzerdaten auf alternativen Wegen abhandengekommen sind.

7.2 Durchführung

Im Rahmen des Forschungsprojektes *EIDI* [116] sind Daten an das *XING*-Netzwerk mit dem in Kapitel 6 beschriebenen Verfahren übermittelt worden, um die von Identitätsdatendiebstahl betroffenen Benutzer ausfindig zu machen und diese anschließend zu warnen. Die aus dieser Übertragung entstandenen Metadaten werden für diese Evaluation herangezogen.

Hierzu betreibt das *XING*-Netzwerk eine in Abschnitt 6.6 beschriebene API, an welche pseudonymisierte Datensätze aus Identitätsdaten-Leaks gesendet werden können. Da beim *XING*-Netzwerk eine Anmeldung mit den Identifikatoren *E-Mail-Adresse* oder *Telefonnummer* möglich ist, werden diese beiden Identifikatoren übertragen. Andere Identifikatoren werden vor der Übertragung aus Datenschutzgründen herausgefiltert, da bei *XING* hierfür keine Verwendung besteht. Mit der Evaluation soll überprüft werden, welche Identitätsdaten valide Zugangsdaten für *XING* darstellen. Hierzu sind nur geeignete Identifikatoren und ein Klartextpasswort notwendig. Alle anderen Identitätsmerkmale werden ebenfalls vor der Übertragung herausgefiltert.

Auf der Seite des Frühwarndienstes wählt ein System zufällig einen Identitätsdaten-Leak in der Datenbank aus, der anschließend übertragen werden soll. Übertragen werden immer alle Identitätsdatensätze, welche aus einer Datei mittels Parser (Kapitel 5) extrahiert werden können und den zuvor genannten Anforderungen entsprechen. Ist eine Übertragung einer Datei abgeschlossen, so wird die nächste zufällig ausgewählte Datei übertragen. Die zufällige Auswahl der Dateien sorgt dafür, dass mit den in der Evaluation gewonnenen Erkenntnissen eine möglichst breite Aussage über alle gesammelten Daten getroffen werden kann.

Seit dem 21. Februar 2020 werden Daten an das *XING*-Netzwerk übermittelt. *XING* überprüft die erhaltenen Leak-Daten auf eine Übereinstimmung der erhalte-

nen Pseudonyme der Identifikatoren mit den Informationen der eigenen Benutzer. Wenn es eine Übereinstimmung gibt, kann das entsprechende Passwort entschlüsselt und daraufhin auf Validität überprüft werden. Dazu werden Identifikator und Passwort an einer internen Login-Schnittstelle getestet. Handelt es sich bei erhaltenen Datensätzen um valide Zugangsdaten, ist eine erfolgreiche Anmeldung an der Login-Schnittstelle möglich. In einem solchen Fall leitet *XING* unverzüglich geeignete und vorher festgelegte Maßnahmen ein. Zum Schutz der eigenen Infrastruktur und zum Schutz des Benutzers wird ein kompromittiertes Benutzerkonto gesperrt, da offensichtlich Dritte die Möglichkeit zum Zugriff auf den Dienst über ein fremdes Benutzerkonto besitzen. Die betroffenen Benutzer erhalten eine Warn-E-Mail, in der der Vorfall genauer erläutert wird (siehe Anhang A). Bei einer Benutzeranmeldung mit den kompromittierten Zugangsdaten wird ein kurzer Informationstext angezeigt (siehe Anhang B.3), der besagt, dass das Passwort geändert werden muss. Der weitere Zugriff auf das Benutzerkonto ist bis zur erneuten Verifikation des Benutzers deaktiviert. Um wieder Zugriff auf das eigene Benutzerkonto zu erhalten, müssen Benutzer ein neues Kennwort setzen und den Besitz ihrer Identifikatoren nachweisen. Zur Auswertung der Resultate werden in dieser Arbeit die fertig übertragenen Daten bis zum 15.09.2020 berücksichtigt. Die erhaltenen Trefferquoten werden vom *XING*-Netzwerk erhoben und für diese Ausarbeitung dankenswerterweise zur Verfügung gestellt.

7.3 Auswertung

In dem Zeitraum vom 21. Februar bis zum 15. September 2020 sind insgesamt 2.512 pseudonymisierte und gefilterte Identitätsdaten-Leaks an *XING* übertragen worden. Diese übertragenen Leaks enthalten 5,05 Milliarden Identitätsdatensätze, bestehend aus Identifikator und Passwort. Von den übertragenen Identifikatoren kennt *XING* 36.229.163 Stück (0,72 %). Von diesen 36 Millionen Datensätzen können 5.366.408 Stück (14,81 %) genutzt werden, um sich tatsächlich bei *XING* zu authentifizieren. Da in den übertragenen Daten auch Duplikate enthalten sind, stimmt die Anzahl gültiger Datensätze nicht mit der Anzahl der betroffenen Benutzer überein. Damit ein Benutzer aufgrund von Duplikaten nicht mehrfach gewarnt

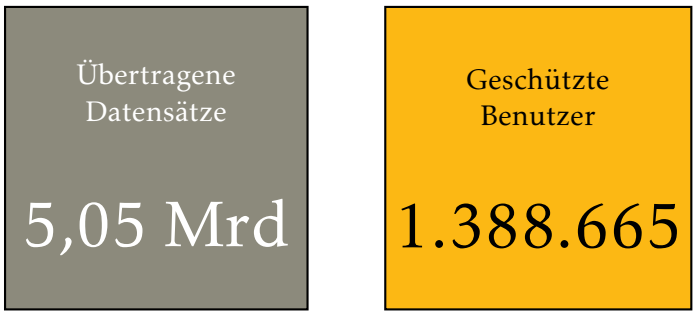


Abbildung 23: *Anzahl von übertragenen Datensätzen und die Anzahl von geschützten Benutzern.*

und das Benutzerkonto gesperrt wird, vermerkt *XING* sich eine bereits durchgeführte Schutzmaßnahme bei einem Benutzer. Die Zahl der tatsächlich geschützten Benutzer beläuft sich auf 1.388.665 Stück (siehe Abbildung 23).

XING gibt an, dass sie in Deutschland, Österreich und der Schweiz insgesamt 18 Millionen *XING*-Nutzer haben. Allerdings ist die Plattform auch in weiteren Ländern verfügbar, für die aber keine Benutzerzahlen von *XING* veröffentlicht wurden. Deshalb lässt sich kein Prozentsatz der betroffenen Benutzer berechnen.

Aus dem Verhältnis der validen übertragenen Datensätze und den tatsächlich betroffenen Benutzerkonten lässt sich eine Duplikatsquote von 286,44 % ableiten. Das bedeutet, dass *XING* die Zugangsdaten der zu schützenden Benutzer im Durchschnitt 3,86 Mal erhalten hat. Allerdings muss bei der Interpretation dieser Zahlen beachtet werden, dass während des Übertragungszeitraums Veränderungen in der Benutzerdatenbank von *XING* stattgefunden haben. Beispielsweise wurden neue Benutzerkonten angelegt, gelöscht oder das Passwort geändert. Für genauere Zahlen hätte die Überprüfung mit einer sich nicht verändernden Benutzerdatenbank durchgeführt werden müssen.

In dem genannten Zeitraum sind nur 5,05 Milliarden Datensätze übertragen worden, weil die eingesetzten Systeme und Verfahren aufgrund des Produktivbetriebs nicht überlastet werden sollten und Erfahrungen gesammelt werden mussten. Von den gesamten gesammelten und extrahierten Daten entsprechen die 5,05 Milliar-

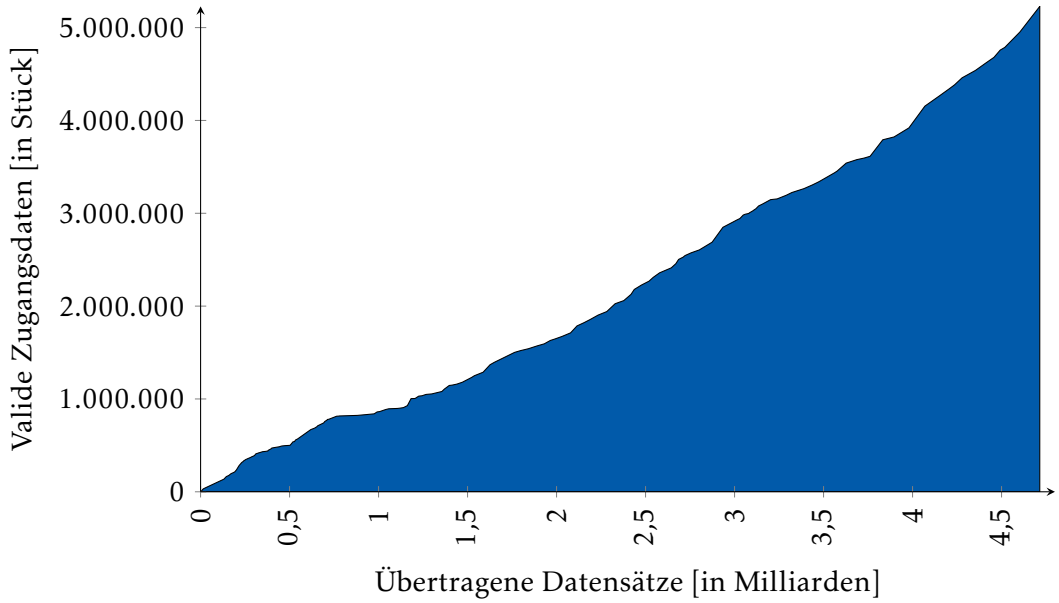


Abbildung 24: *Geschützte Benutzerkonten durch Detektion kompromittierter Zugangsdaten in Bezug zu den übertragenen Leak-Daten.*

den Datensätze 21,13 %. Es ist jedoch anzunehmen, dass bei steigender Anzahl an übertragenen Daten auch die Duplikatsquote ansteigen wird.

In Abbildung 24 sind die aus den übertragenen Datensätzen resultierenden validen Zugangsdaten für *XING* dargestellt. Zu sehen ist, dass die Zahl der validen Zugangsdaten für *XING* in den bereits übertragenen Datensätzen zwar ansteigt, jedoch einige übertragene Leaks fast gar keine validen Zugangsdaten enthalten. Gerade im Bereich von 0,8 bis 1 Milliarde Datensätzen ist ein Plateau erkennbar. Insgesamt sind in den 3.222 übertragenen Leak-Dateien nur 378 Dateien enthalten, die keine validen Zugangsdaten enthalten. Daraus lässt sich schließen, dass in 88,27 % der übertragenen Dateien tatsächlich valide Zugangsdaten enthalten sind.

In Abbildung 25 sind die Übereinstimmungen von den Identifikatoren allein und der Kombination aus Identifikator plus Passwort dargestellt. Zu sehen ist, dass eine deutlich größere Menge von Datensätzen zwar bekannte Identifikatoren enthält, diese jedoch keine validen Passwörter besitzen. In den Datensätzen, bei

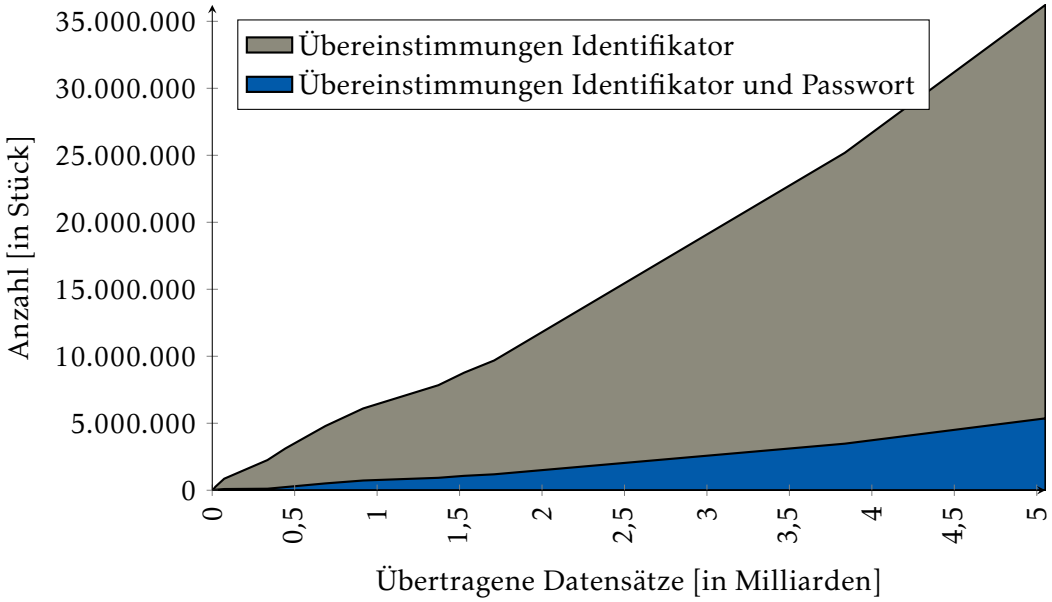


Abbildung 25: *Darstellung der Übereinstimmungen von* `Identifikator` *und* `Identifikator + Passwort`.

denen Identifikator und Passwort mit den Benutzerdaten von *XING* übereinstimmen, sind Duplikate enthalten. Diese werden aus Datenschutzgründen und zur Ressourcenschonung nicht zuvor herausgefiltert. In Abbildung 26 ist die Anzahl der in den übertragenen Leak-Daten enthaltenen Duplikate zu erkennen. Während der Übertragung der ersten 400.000.000 Datensätze bleibt die Anzahl der Duplikate gering und konstant. Danach steigt die Duplikatsquote kontinuierlich an. Dieser Effekt kann seine Ursache in der Reihenfolge der übertragenen Datensätze besitzen.

In Abbildung 27 sind die an *XING* übertragenen Dateien nach ihrer Größe und enthaltenen validen Zugangsdaten dargestellt. Die meisten Dateien bestehen aus weniger als 10 Millionen Identitätsdatensätzen und enthalten weniger als 10.000 valide Zugangsdaten. Jedoch enthalten einige Identitätsdaten-Leaks auch deutlich mehr valide Zugangsdaten. Die drei Identitätsdaten-Leaks mit den meisten validen Zugangsdaten bestehen aus folgender Anzahl an gesamten Identitätsdatensätzen:

1. 51.354.061 Datensätze mit 181.320 validen Zugangsdaten.

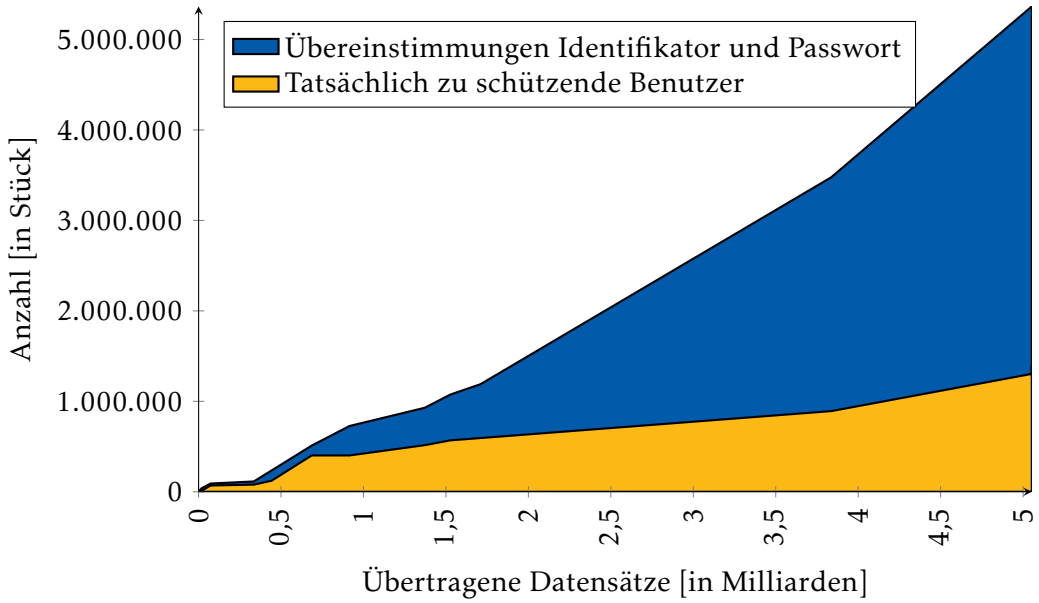


Abbildung 26: *Verhältnis der Duplikatsquote in den übertragenen Daten.*

2. 38.035.272 Datensätze mit 147.560 validen Zugangsdaten.
3. 113.798.351 Datensätze mit 106.920 validen Zugangsdaten.

Auch wenn eine Analyse der Herkunft von den Identitätsdaten-Leaks mit den meisten Treffern von wissenschaftlichem Interesse ist, soll an dieser Stelle darauf verzichtet werden. Der Grund dafür ist, dass *XING* durch die Veröffentlichung dieser Arbeit erfahren würde, bei welchen Diensten ihre eigenen Kunden identische Passwörter benutzen. Das Protokoll zum Übertragen der Daten wurde explizit so gestaltet, dass den Kooperationspartnern nicht mitgeteilt wird, aus welchen Identitätsdaten-Leaks die erhaltenen Daten stammen. Dies trägt zum Schutz der Privatsphäre der betroffenen Personen bei.

Abschließend soll auf die in den Daten enthaltene Duplikatsquote eingegangen werden. Die aus den übertragenen Daten abgeleitete Duplikatsquote weist eine hohe Ungenauigkeit aufgrund der genannten Faktoren auf. Für eine genauere Ermittlung der Anzahl muss eine gesonderte Untersuchung durchgeführt werden. Da eine vollständige Untersuchung der gesamten Leak-Datenbank aufgrund des

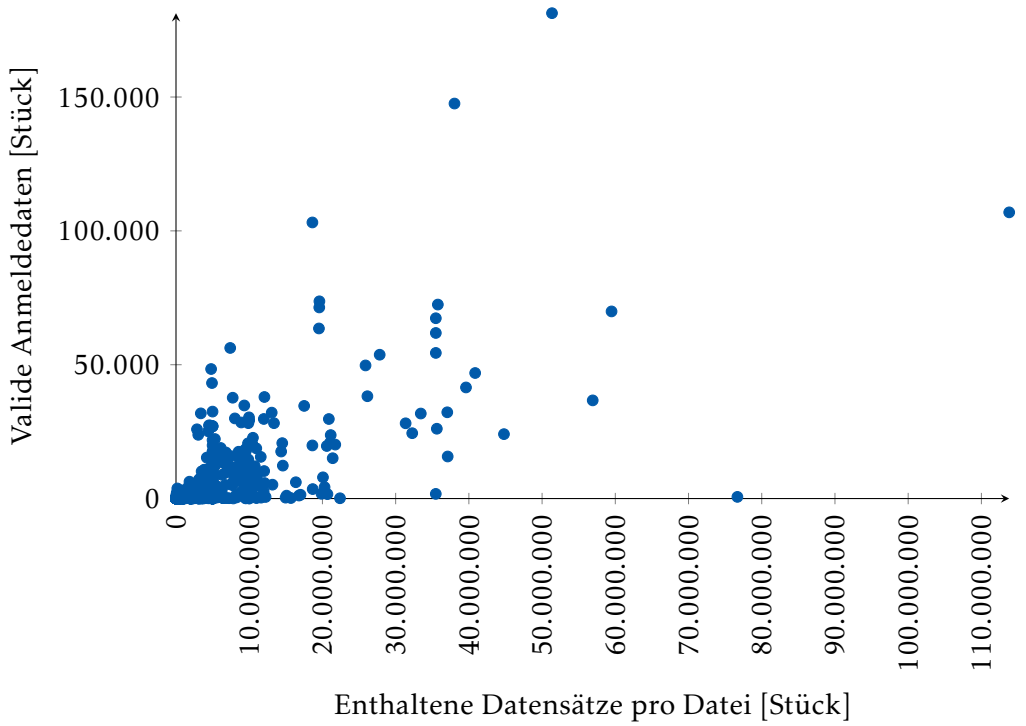


Abbildung 27: *Anzahl der in übertragenen Identitätsdaten-Leaks enthaltenen validen Zugangsdaten.*

Datenumfangs zu ressourcenintensiv ist, wird eine Stichprobe aus dem gesamten Datensatz entnommen und analysiert. Hierfür werden zufällig ausgewählte Einträge in der Datenbank ausgewählt. Die darin enthaltene Kombination aus Identifikator und Passwort wird genutzt, um abzufragen, wie häufig diese Kombination in der Datenbank vorhanden ist. Bei der zufälligen Auswahl der Datensätze wird sichergestellt, dass eine Kombination aus Identifikator und E-Mail-Adresse lediglich einmal ausgewählt wird. Problematisch ist, dass Kombinationen aus E-Mail-Adresse und Passwort mit vielen Duplikaten mit einer höheren Wahrscheinlichkeit bei der zufälligen Auswahl gezogen werden. Das bedeutet, dass eine Stichprobe mit hoher Wahrscheinlichkeit viele Elemente enthält, die eine überdurchschnittliche Anzahl an Duplikaten besitzen. Ein statistisches Vorgehen zum Umgang mit dieser Eigenschaft konnte für diesen Anwendungsfall in der Literatur nicht gefunden werden. Aus diesem Grund können die statistischen Ergebnisse lediglich als grobe

Zielgrößen betrachtet werden. Aufgrund der zu hohen Anzahl der Elemente mit vielen Duplikaten in der Stichprobe stellen die Ergebnisse eine obere Schranke für die Resultate dar. In einer Stichprobe mit 100.000 Elementen konnten folgende statistische Größen ermittelt werden: 54.621 Kombinationen aus E-Mail-Adresse und Passwort aus dieser Stichprobe enthalten keine Duplikate. Für eine einzelne Kombination aus E-Mail-Adresse und Passwort ist in der gesamten Datenbank ein Maximum von 87.165 Duplikaten zu finden. Aufgrund einiger Ausreißer wird das arithmetische Mittel verzerrt und liegt bei 21,70 Duplikaten. Deshalb ist die Interpretation des Median hilfreicher. Dieser liegt bei 12,00 Duplikaten. Aufgrund der statistischen Verteilung sind diese Werte als obere Schranken zu betrachten. Wird die Stichprobengröße erweitert, sollten das arithmetische Mittel sowie der Median sinken.

7.4 Zusammenfassung

In diesem Kapitel werden die in dieser Arbeit vorgestellten Konzepte zum Sammeln, Analysieren und Versenden von Identitätsdaten-Leaks in einem realen Szenario mit einem Kooperationspartner getestet. Im Testzeitraum werden 5,05 Milliarden Identitätsdatensätze an das Karriere-Netzwerk *XING* übertragen. Von diesen 5,05 Milliarden Datensätzen stellen insgesamt 5.366.408 valide Zugangsdaten zur *XING*-Plattform dar. In diesen 5,37 Millionen validen Datensätzen sind jedoch aufgrund der Struktur von Identitätsdaten-Leaks eine hohe Quote an Duplikaten enthalten. Von diesen validen Datensätzen waren bei *XING* insgesamt 1.388.665 Benutzerkonten betroffen, die durch dieses Projekt geschützt werden konnten, indem der Zugriff mit den kompromittierten Kennwörtern deaktiviert wurde.

XING war zuvor nicht von einem Identitätsdaten-Leak betroffen. Daraus lässt sich schließen, dass die betroffenen Benutzer ihr bei *XING* verwendetes Passwort eventuell zusätzlich bei anderen Diensten einsetzen, bei denen Benutzerdaten abhandengekommen sind. Diese Anzahl an *XING*-Benutzern, deren Betroffenheit auf die Problematik der Passwortwiederverwendung zurückzuführen ist, ist sicherlich nicht nur bei dem Dienst *XING* vorhanden. Diese Werte verdeutlichen, wie umfassend die Problematik der Mehrfachverwendung von Passwörtern ist. *XING*

hat im Gegensatz zu anderen Diensten die betroffenen Benutzerkonten durch die stattgefundene Überprüfung abgesichert und die Benutzer bereits informiert.

Andere Dienste haben vermutlich Mengen an ungeschützten Benutzerkonten in ihrem Bestand, was eventuell zu regelmäßigen Schäden in vielfacher Form auf Seiten des Dienstes oder der betroffenen Benutzer führen kann. Es wird mit dieser Untersuchung gezeigt, dass bei Onlinediensten eine Überprüfung von Benutzerkonten auf eine Kompromittierung zu einer massiven Verbesserung der Sicherheit führen kann. Wünschenswert wäre, wenn viele Onlinedienste das hier vorgestellte Verfahren zum Schutz der Betroffenen nutzen würden, um zu einer gesamten Sicherheitsverbesserung beizutragen.

8 Zusammenfassung, Fazit & Ausblick

In diesem Kapitel werden die wichtigsten Beiträge dieser Arbeit dargestellt. Dazu werden die Ergebnisse zunächst zusammengefasst dargestellt (siehe Abschnitt 8.1), um anschließend den Zielerreichungsgrad der gesamten Arbeit zu diskutieren (siehe Abschnitt 8.2). Abschließend werden Forschungsideen vorgestellt, die sich aus dieser Arbeit ergeben haben und in zukünftigen Arbeiten untersucht werden sollten.

8.1 Zusammenfassung

Die Anzahl von Sicherheitsvorfällen, bei denen Identitätsdaten gestohlen oder missbraucht werden, nimmt stetig zu. In der Literatur sind keine Ansätze zu finden, um eine möglichst große Anzahl an betroffenen Personen zu identifizieren und zeitnah zu schützen. Die vorliegende Arbeit entwickelt technische Komponenten, um Betroffene vor den Gefahren von Identitätsdaten-Leaks zu schützen. Dazu wird in Kapitel 4 ein Vorgehen erarbeitet, um Quellen für öffentlich verfügbare Identitätsdaten-Leaks zu identifizieren, aus denen die entsprechenden Leak-Daten bezogen werden können. Hierbei kommt ein manuelles als auch ein automatisiertes Vorgehen zum Einsatz. Es wird gezeigt, dass mit dem manuellen Vorgehen deutlich größere öffentlich verfügbare Identitätsdaten-Leaks gesammelt werden können.

Darüber hinaus wird ein Threat-Intelligence-Service vorgestellt, welcher den Betreiber eines Frühwarndienstes darüber informiert, dass bei einem Unternehmen Identitätsdatensätze abhandengekommen sind. Dieses System hilft den Mitarbeitern

des Frühwarnsystems dabei, zeitnah gezielt nach konkreten Identitätsdaten-Leaks suchen zu können, um so einen möglichst frühzeitigen Schutz der Betroffenen zu realisieren. Mit diesem Vorgehen wurden in der Zeit von März 2017 bis Mai 2020 insgesamt 604 Identitätsdaten-Leaks bestehend aus 84.802 Dateien identifiziert und heruntergeladen.

Die Formate der geladenen Leak-Daten sind stark heterogen und nicht standardisiert, weswegen die Daten nicht mit standardisierten Werkzeugen verarbeitet werden können. Aufgrund der unbekannten Struktur dieser strukturierten Daten wird ein eigenes System benötigt, welches die vorliegende Datenstruktur erkennt und anschließend die enthaltenen Identitätsdaten extrahiert. In Kapitel 5 wird ein Parser konzipiert, welcher die Leak-Daten vollautomatisiert analysiert und verarbeitet. Mit einer speziell entwickelten Strukturanalyse können die enthaltenen Identitätsmerkmale aus den Identitätsdaten-Leaks extrahiert werden (Abschnitt 5.3). In einem weiteren Verarbeitungsschritt wird den extrahierten Identitätsmerkmalen ein Merkmalstyp und damit eine Bedeutung zugeordnet. Hierfür werden Verfahren vorgestellt, die eine Erkennung des Merkmaltyps ermöglichen (Abschnitt 5.4). In einer Evaluation des Parsers wird festgestellt, dass mit diesem Verfahren eine Genauigkeit von 97,91 % erreicht wird.

Die Ergebnisse des Parsers können genutzt werden, um Betroffene zu identifizieren und weitere Schutzmaßnahmen einzuleiten. In Kapitel 6 wird ein Warnsystem vorgestellt, mit dem eine proaktive Warnung von betroffenen Personen möglich ist. Hier soll ein zentraler Frühwarndienst Identitätsdaten-Leaks sammeln und verarbeiten, um die Resultate kooperierenden Onlinediensten zur Verfügung zu stellen. Die Onlinedienste werden so in die Lage versetzt, die eigene Infrastruktur und die eigenen Benutzer schützen zu können. Für dieses Konzept wird ein Protokoll entwickelt, mit dem die gesammelten Identitätsdaten-Leaks an kooperierende Onlinedienste datenschutzkonform übermittelt werden können.

Abschließend werden die vorgestellten Systeme in der Praxis getestet. Dazu werden die verarbeiteten Identitätsdaten-Leaks mit dem Protokoll aus Kapitel 6 an das Karriere-Netzwerk *XING* übertragen. *XING* überprüft nach Erhalt, ob eigene Benutzer in den Daten enthalten sind. Das Ergebnis dieser Gesamtevaluation ist,

dass mithilfe der hier vorgestellten Systeme bei *XING* insgesamt 1.388.665 Benutzerkonten identifiziert werden konnten, die von Identitätsdatendiebstahl betroffen waren und anschließend geschützt wurden.

8.2 Fazit

In diesem Abschnitt wird der Erreichungsgrad der Zielsetzung dieser Arbeit diskutiert. Ziel dieser Arbeit ist der Entwurf eines Warnsystems, welches Unternehmen und Benutzer vor den Bedrohungen schützt, die aus öffentlich verfügbaren Identitätsdaten-Leaks resultieren. Abgeleitet aus dieser Zielbeschreibung sind in Abschnitt 2.5 drei Forschungsfragen dargestellt, die in dieser Arbeit beantwortet werden sollen.

Die erste Forschungsfrage bezieht sich darauf, wie Identitätsdaten-Leaks gesammelt werden können. Zur Beantwortung dieser Frage wird in Kapitel 4 ein Konzept vorgestellt, welches mehrere Ansätze zum Sammeln von Identitätsdaten-Leaks beinhaltet. Es wird ein automatisiertes Verfahren als auch ein manuelles Vorgehen zum Sammeln von Identitätsdaten-Leaks präsentiert. Aufgrund der Heterogenität der Datensenken eignet sich der manuelle Ansatz zum Sammeln aktueller Leak-Daten deutlich besser.

Die zweite Forschungsfrage fordert ein Verfahren, mit dem Identitätsdaten-Leaks vollautomatisiert analysiert und verarbeitet werden können. Beantwortet wird diese Forschungsfrage mit dem in Kapitel 5 vorgestellten System, welches Identitätsdaten-Leaks vollautomatisiert verarbeiten kann. Dazu werden Verfahren entwickelt, welche die Syntax und Semantik der Identitätsdaten-Leaks ermitteln.

Die letzte Forschungsfrage sucht ein Verfahren, um die durch Identitätsdatendiebstahl betroffenen Personen zu identifizieren und zu schützen. Zur Beantwortung dieser Forschungsfrage wird ein technisches Verfahren entwickelt, mit dem Betroffene mithilfe von kooperierenden Onlinediensten ausfindig gemacht werden können, um geeignete Schutzmaßnahmen einzuleiten.

Damit werden in dieser Arbeit alle drei gestellten Forschungsfragen vollständig beantwortet. Bei der Bearbeitung dieser Arbeit sind weitere Ideen für ergänzende oder zusätzliche Forschungsfragen entstanden, die im nachfolgenden Abschnitt abschließend präsentiert werden.

8.3 Ausblick

In dieser Arbeit werden mehrere aufeinander aufbauende Verfahren präsentiert, mit denen Betroffene vor Identitätsdiebstahl geschützt werden können. Der vorgestellte Ansatz sieht vor, dass ein zentraler Frühwarndienst die gefundenen Identitätsdatensätze einmalig mit jedem kooperierenden Onlinedienst abgleicht. Sollte sich ein neuer Benutzer bei einem kooperierenden Onlinedienst mit kompromittierten und bereits in der Vergangenheit abgeglichenen Zugangsdaten registrieren, so kann dieser nicht geschützt werden. Hierfür ist eine separate Erweiterung des vorgestellten Ansatzes notwendig.

Aus der Literatur und den Ergebnissen der Gesamtevaluation (siehe Kapitel 7) ist zu erkennen, dass die Mehrfachverwendung von Passwörtern ein weit verbreitetes Vorgehen bei Benutzern ist. Dieses Verhalten verstärkt die gesamte Gefahr von Identitätsdiebstahl. Zur deutlichen Reduktion der Fallzahlen kann eine Veränderung des Benutzerverhaltens, als auch der Einsatz von weiteren technischen Sicherheitsmaßnahmen beitragen. Sowohl das Benutzerverhalten als auch alternative Authentifikationsverfahren werden in der Literatur bereits ausführlich diskutiert (siehe Kapitel 3).

Solange keine flächendeckende Veränderung des Benutzerverhaltens herbeigeführt wurde oder die Zweifaktor-Authentifizierung zum Einsatz kommt, sind reaktive Sicherheitsmaßnahmen notwendig wie beispielsweise das in dieser Arbeit präsentierte Verfahren. Damit der vorgestellte Dienst immer die neusten Identitätsdaten-Leaks zur Verfügung hat, bedarf es einer ständigen Weiterentwicklung der Verfahren zum Sammeln dieser Daten. Ebenfalls muss untersucht werden, was aus Sicht der Benutzer geschieht, wenn tatsächlich mehrere Onlinedienste an das vorgestellte Frühwarnsystem angeschlossen werden und es zu zeitgleichen Warnungen und

Schutzmaßnahmen kommt. Beispielsweise könnten Mehrfachwarnungen von verschiedenen Onlinediensten eine negative Auswirkung auf das Benutzerverhalten haben. Erste Ansätze zur Vermeidung von Mehrfachwarnungen werden bereits diskutiert [60].

Literaturverzeichnis

[1] 3rd, D. Eastlake ; Hansen, T.: *US Secure Hash Algorithms (SHA and SHA-based HMAC and HKDF)*. RFC 6234 (Informational). Internet Engineering Task Force, 2011. URL: http://www.ietf.org/rfc/rfc6234.txt.

[2] Apiary: *SpyCloud Data Schema*. 2020. URL: https://spyclouddataschema.docs.apiary.io/#introduction/introduction (Gesichtet am 28. 09. 2020).

[3] Avast Software s.r.o.: *Avast Hack Check*. 2020. URL: https://www.avast.com/hackcheck (Gesichtet am 28. 09. 2020).

[4] Beiersmann, Stefan: *Facebook speichert mehrere Hundert Millionen Passwörter im Klartext*. 2019. URL: https://www.zdnet.de/88356927/facebook-speichert-mehrere-hundert-millionen-passwoerter-im-klartext/ (Gesichtet am 28. 09. 2020).

[5] Bentley, Jon Louis: „Multidimensional Divide-and-Conquer“. In: *Commun. ACM* 23.4 (1980), S. 214–229. URL: https://doi.org/10.1145/358841.358850.

[6] Berners-Lee, T. ; Fielding, R. ; Masinter, L.: *Uniform Resource Identifier (URI): Generic Syntax*. RFC 3986 (INTERNET STANDARD). Updated by RFCs 6874, 7320. Internet Engineering Task Force, 2005. URL: http://www.ietf.org/rfc/rfc3986.txt.

[7] Bieker, Felix ; Bremert, Benjamin ; Hansen, Marit: „Die Risikobeurteilung nach der DSGVO“. In: *Datenschutz und Datensicherheit - DuD* 42.8 (2018), S. 492–496.

[8] Biryukov, Designers Alex ; Dinu, Daniel ; Khovratovich, Dmitry: *Argon2 : the memory-hard function for password hashing and other applications*. 2015. URL: https://www.cryptolux.org/images/0/0d/Argon2.pdf.

[9] Blocki, Jeremiah ; Harsha, Benjamin ; Zhou, Samson: „On the Economics of Offline Password Cracking“. In: *2018 IEEE Symposium on Security and Privacy (SP)*. Bd. 2018. San Francisco, CA, 2018, S. 853–871. arXiv: 2006.05023.

[10] Brogada, Michael Angelo D. ; Sison, Ariel M. ; Medina, Ruji P.: „Cryptanalysis on the head and tail technique for hashing passwords“. In: *Proceeding - 2019 IEEE 7th Conference on Systems, Process and Control, ICSPC 2019*. December. IEEE, 2019, S. 137–142.

[11] Brogada, Michael Angelo D. ; Sison, Ariel M. ; Medina, Ruji P.: „Head and Tail Technique for Hashing Passwords“. In: *Proceeding - 2019 IEEE 11th International Conference on Communication Software and Networks* (2019), S. 137–142.

[12] Cao, Yunbo ; Xu, Jun ; Liu, Tie Yan ; Li, Hang ; Huang, Yalou ; Hon, Hsiao Wuen: „Adapting ranking SVM to document retrieval“. In: *Proceedings of the Twenty-Ninth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Bd. 2006. 49. 2006, S. 186–193.

[13] Charan, Jaykaran ; Biswas, Tamoghna: „How to calculate sample size for different study designs in medical research?“ In: *Indian Journal of Psychological Medicine* 35.2 (2013), S. 121. URL: https://doi.org/10.4103/0253-7176.116232.

[14] Cimpanu, Catalin: *Robinhood admits to storing some passwords in cleartext*. 2019. URL: https://www.zdnet.com/article/robinhood-admits-to-storing-some-passwords-in-cleartext/ (Gesichtet am 28. 09. 2020).

[15] Deblasio, Joe ; Savage, Stefan ; Voelker, Geoffrey M. ; Snoeren, Alex C.: „Tripwire: Inferring Internet Site Compromise“. In: *Internet Measurement Conference (IMC)*. 2017, S. 341–354.

[16] Deutsche Bundesbank: *Download - Bankleitzahlen*. 2017. URL: https://www.bundesbank.de/Redaktion/DE/Standardartikel/Aufgaben/

Unbarer_Zahlungsverkehr/bankleitzahlen_download.html (Gesichtet am 28. 09. 2020).

[17] Eckert, Claudia: *IT-Sicherheit : Konzepte - Verfahren - Protokolle*. Munich, Germany: De Gruyter Oldenbourg, 2014.

[18] EIDI Projektkonsortium: „EIDI-Crypto“. (Unveröffentlichtes Projektergebnis). 2019.

[19] EIDI Projektkonsortium: „EIDI-REST-API“. (Unveröffentlichtes Projektergebnis). 2019.

[20] Eikenberg, Ronald: *Neue Passwort-Leaks: Insgesamt 2,2 Milliarden Accounts betroffen*. 2019. url: https://www.heise.de/security/meldung/Neue-Passwort-Leaks-Insgesamt-2-2-Milliarden-Accounts-betroffen-4287538.html (Gesichtet am 28. 09. 2020).

[21] Enzoic: *Credentials API*. 2020. url: https://www.enzoic.com/docs-credentials-api/ (Gesichtet am 28. 09. 2020).

[22] Enzoic: *Detect Compromised Passwords*. 2019. url: https://www.enzoic.com/ (Gesichtet am 28. 09. 2020).

[23] Enzoic: *LastPass Selects Enzoic for Compromised Credential Screening*. 2017. url: https://www.enzoic.com/lastpass-selects-passwordping-for-compromised-credential-screening/ (Gesichtet am 28. 09. 2020).

[24] Etaiwi, Wael ; Naymat, Ghazi: „The Impact of applying Different Preprocessing Steps on Review Spam Detection“. In: *Procedia Computer Science* 113 (2017), S. 273–279.

[25] Europäische Union: *DSGVO : EU-Datenschutz-Grundverordnung; 2018; aktuelle Gesetze*. 2018.

[26] Fielding, R. ; Gettys, J. ; Mogul, J. ; Frystyk, H. ; Masinter, L. ; Leach, P. ; Berners-Lee, T.: *Hypertext Transfer Protocol – HTTP/1.1*. RFC 2616 (Draft Standard). Obsoleted by RFCs 7230, 7231, 7232, 7233, 7234, 7235, updated by RFCs 2817, 5785, 6266, 6585. Internet Engineering Task Force, 1999. url: http://www.ietf.org/rfc/rfc2616.txt.

[27] Fielding, R. ; Lafon, Y. ; Reschke, J.: *Hypertext Transfer Protocol (HTTP/1.1): Range Requests*. RFC 7233 (Proposed Standard). Internet Engineering Task Force, 2014. URL: http://www.ietf.org/rfc/rfc7233.txt.

[28] Fielding, R. ; Nottingham, M. ; Reschke, J.: *Hypertext Transfer Protocol (HTTP/1.1): Caching*. RFC 7234 (Proposed Standard). Internet Engineering Task Force, 2014. URL: http://www.ietf.org/rfc/rfc7234.txt.

[29] Fielding, R. ; Reschke, J.: *Hypertext Transfer Protocol (HTTP/1.1): Authentication*. RFC 7235 (Proposed Standard). Internet Engineering Task Force, 2014. URL: http://www.ietf.org/rfc/rfc7235.txt.

[30] Fielding, R. ; Reschke, J.: *Hypertext Transfer Protocol (HTTP/1.1): Conditional Requests*. RFC 7232 (Proposed Standard). Internet Engineering Task Force, 2014. URL: http://www.ietf.org/rfc/rfc7232.txt.

[31] Fielding, R. ; Reschke, J.: *Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing*. RFC 7230 (Proposed Standard). Internet Engineering Task Force, 2014. URL: http://www.ietf.org/rfc/rfc7230.txt.

[32] Fielding, R. ; Reschke, J.: *Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content*. RFC 7231 (Proposed Standard). Internet Engineering Task Force, 2014. URL: http://www.ietf.org/rfc/rfc7231.txt.

[33] Fielding, Roy Thomas: „Architectural Styles and the Design of Network-based Software Architectures“. Diss. University of California, 2000.

[34] Florencio, Dinei ; Herley, Cormac: „A Large-Scale Study of Web Password Habits“. In: *Proceedings of the 16th International Conference on World Wide Web*. Banff, Alberta, Canada: ACM, 2007, S. 657–666.

[35] Frankel, S. ; Glenn, R. ; Kelly, S.: *The AES-CBC Cipher Algorithm and Its Use with IPsec*. RFC 3602 (Proposed Standard). Internet Engineering Task Force, 2003. URL: http://www.ietf.org/rfc/rfc3602.txt.

[36] Golla, Maximilian ; Filipe, Lydia ; Wei, Miranda ; Dürmuth, Markus ; Ur, Blase ; Hainline, Juliette ; Redmiles, Elissa: „'What was that site doing with my Facebook password?' Designing password-reuse notifications“. In: *Proceedings of the ACM Conference on Computer and Communications Security* (2018), S. 1549–1566.

[37] Gollmann, Dieter: *Computer security*. Chichester, West Sussex: Wiley, 2011.

[38] Gonscherowski, Susan ; Rack, Fabian ; Vettermann, Oliver: „Forschungsprojekt zum digitalen Identitätsdiebstahl-Research in Progress“. In: *MKWI 2018 - Multikonferenz Wirtschaftsinformatik* 2018-March (2018), S. 1402–1408.

[39] Gonscherowski, Susan ; Vettermann, Oliver ; Wübbeling, Matthias ; Malderle, Timo: „Datenkrake Leak-Checker – Lösung in Sicht?“ In: *digma - Zeitschrift für Datenrecht und Informationssicherheit* 18.2 (2018). Hrsg. von Bruno Baeriswyl ; Beat Rudin ; Bernhard M. Hämmerli ; Rainer J. Schweizer ; Karjoth Günter ; David Vasella, S. 60–64.

[40] google.com: *Password Checkup extension*. 2020. url: https://chrome.google.com/webstore/detail/password-checkup-extensio/pncabnpcffmalkkjpajodfhijclecjno (Gesichtet am 28. 09. 2020).

[41] Gott, Amber: *LastPass Reveals 8 Truths about Passwords in the New Password Exposé*. 2017. url: https://blog.lastpass.com/2017/11/lastpass-reveals-8-truths-about-passwords-in-the-new-password-expose.html/ (Gesichtet am 28. 09. 2020).

[42] Graupner, Hendrik ; Jaeger, David ; Cheng, Feng ; Meinel, Christoph: „Automated Parsing and Interpretation of Identity Leaks“. In: *Proceedings of the ACM International Conference on Computing Frontiers*. Bd. CF '16. New York, NY, USA: Association for Computing Machinery, 2016, S. 102–115.

[43] Gruss, Daniel ; Schwarz, Michael ; Wübbeling, Matthias ; Guggi, Simon ; Malderle, Timo ; More, Stefan ; Lipp, Moritz: „Use-After-FreeMail: Generalizing the Use-After-Free Problem and Applying It to Email Services“. In: *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*. Incheon, Republic of Korea: ACM, 2018, S. 297–311.

[44] Hasso-Plattner-Institut: *Wurden Ihre Identitätsdaten ausspioniert?* 2020. url: https://sec.hpi.de/ilc (Gesichtet am 28. 09. 2020).

[45] Heen, Olivier ; Neumann, Christoph: „On the Privacy Impacts of Publicly Leaked Password Databases“. In: *DIMVA 2017*. Hrsg. von M. Polychronakis ; M. Meier. Springer, Cham, 2017, S. 347–365.

[46] Heim, Patrick: *An inside look at how we keep customer data safe*. 2016. URL: https://blog.dropbox.com/topics/product-tips/dropbox-customer-data-safety (Gesichtet am 28. 09. 2020).

[47] Hirsch, Christian: *Offene Datenbank: 58 Millionen Datensätze im Umlauf*. 2016. URL: https://www.heise.de/newsticker/meldung/Offene-Datenbank-58-Millionen-Datensaetze-im-Umlauf-3351104.html (Gesichtet am 28. 09. 2020).

[48] Hitaj, Briland ; Gasti, Paolo ; Ateniese, Giuseppe ; Perez-Cruz, Fernando: „PassGAN: A Deep Learning Approach for Password Guessing“. In: *17th International Conference, ACNS*. Bd. 1. Springer International Publishing, 2019, S. 217–237.

[49] Huh, Jun Ho ; Kim, Hyoungshick ; Rayala, Swathi S.V.P. ; Bobba, Rakesh B. ; Beznosov, Konstantin: „I'm Too Busy to Reset My LinkedIn Password: On the Effectiveness of Password Reset Emails“. In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2017, S. 387–391.

[50] Hunt, Troy: *';–have i been pwned?* 2020. URL: https://haveibeenpwned.com/ (Gesichtet am 28. 09. 2020).

[51] Hunt, Troy: *API v3*. 2020. URL: https://haveibeenpwned.com/api/v3 (Gesichtet am 28. 09. 2020).

[52] Hunt, Troy: *I've Just Launched "Pwned Passwords"V2 With Half a Billion Passwords for Download*. 2018. URL: https://www.troyhunt.com/ive-just-launched-pwned-passwords-version-2/ (Gesichtet am 28. 09. 2020).

[53] Hunt, Troy: *Pwned Passwords*. 2020. URL: https://haveibeenpwned.com/Passwords (Gesichtet am 28. 09. 2020).

[54] Hunt, Troy: *The 773 Million Record „Collection #1“ Data Breach*. 2019. URL: https://www.troyhunt.com/the-773-million-record-collection-1-data-reach/ (Gesichtet am 28. 09. 2020).

[55] IA7.de - InternetAgentur: *Die schönsten Vornamen*. 2016. URL: http://www.mybabysitter.de/extras/vornamen/ (Gesichtet am 28. 09. 2020).

[56] Jaeger, David ; Graupner, Hendrik ; Pelchen, Chris ; Cheng, Feng ; Meinel, Christoph: „Fast Automated Processing and Evaluation of Identity Leaks“. In: *International Journal of Parallel Programming*. Bd. 46. 2. Springer US, 2018, S. 441–470.

[57] Jaeger, David ; Graupner, Hendrik ; Sapegin, Andrey ; Cheng, Feng ; Meinel, Christoph: „Gathering and Analyzing Identity Leaks for Security Awareness“. In: *PASSWORDS 2014*. Hrsg. von S. Mjølsnes. Bd. 9551. 2015, S. 102–115.

[58] Jaeger, David ; Pelchen, Chris ; Graupner, Hendrik ; Cheng, Feng ; Meinel, Christoph: „Analysis of Publicly Leaked Credentials and the Long Story of Password (Re-)use“. In: *Proceedings of the 11th International Conference on Passwords (PASSWORDS2016)*. Bochum: Springer, 2016, S. 1–19.

[59] Johansen, Alison Grace: *4 Lasting Effects of Identity Theft*. 2018. url: https://www.lifelock.com/education/4-lasting-effects-of-identity-theft/ (Gesichtet am 28. 09. 2020).

[60] Kasem-Madani, Saffija ; Malderle, Timo ; Boes, Felix ; Meier, Michael: „Privacy-Preserving Warning Management for an Identity Leakage Warning Network“. In: *European Interdisciplinary Cybersecurity Conference*. Im Veröffentlichungsprozess. 2020.

[61] Khader, Mariam ; Awajan, Arafat ; Al-Naymat, Ghazi: „The impact of natural language preprocessing on big data sentiment analysis“. In: *International Arab Journal of Information Technology* 16.3ASpecial Issue (2019), S. 506–513.

[62] Krawczyk, H. ; Bellare, M. ; Canetti, R.: *HMAC: Keyed-Hashing for Message Authentication*. RFC 2104 (Informational). Updated by RFC 6151. Internet Engineering Task Force, 1997. url: http://www.ietf.org/rfc/rfc2104.txt.

[63] Le Bras, Tom: *[INFOGRAPHIC] Online Overload – It's Worse Than You Thought*. 2015. url: https://blog.dashlane.com/infographic-online-overload-its-worse-than-you-thought/ (Gesichtet am 28. 09. 2020).

[64] Li, Lucy ; Sullivan, Nick ; Pal, Bijeeta ; Chatterjee, Rahul ; Ali, Junade ; Ristenpart, Thomas: „Protocols for checking compromised credentials“. In:

Proceedings of the ACM Conference on Computer and Communications Security (2019), S. 1387–1403. arXiv: 1905.13737.

[65] Madarie, Renushka ; Ruiter, Stijn ; Steenbeek, Wouter ; Kleemans, Edward: „Stolen account credentials: an empirical comparison of online dissemination on different platforms“. In: *Journal of Crime and Justice* 42.5 (2019), S. 551–568.

[66] Malderle, Timo ; Boes, Felix ; Muuss, Gina ; Wübbeling, Matthias ; Meier, Michael: „Credential Intelligence Agency - A Threat Intelligence Approach to Mitigate Identity Theft“. In: *ICISSP 2020 - Revised Selected Papers*. Hrsg. von Steven Furnell ; Paolo Mori ; Edgar Weippl ; Oliver Champ. In Submission. Springer, 2020.

[67] Malderle, Timo ; Knauer, Sven ; Lang, Martin ; Wübbeling, Matthias ; Meier, Michael: „Track Down Identity Leaks Using Threat Intelligence“. In: *ICISSP 2020 - Proceedings of the 6th International Conference on Information Systems Security and Privacy*. Hrsg. von Steven Furnell ; Paolo Mori ; Edgar Weippl ; Oliver Champ. Valetta, Malta: SCITEPRESS – Science and Technology Publications, 2020.

[68] Malderle, Timo ; Wübbeling, Matthias ; Knauer, Sven ; Meier, Michael: „Ein Werkzeug zur automatisierten Analyse von Identitätsdaten-Leaks“. In: *SICHERHEIT 2018*. Hrsg. von Hanno Langweg ; Michael Meier ; Bernhard C. Witt ; Delphine Reinhardt. Bonn: Gesellschaft für Informatik e.V., 2018, S. 43–54.

[69] Malderle, Timo ; Wübbeling, Matthias ; Knauer, Sven ; Meier, Michael: „Warning of Affected Users About an Identity Leak“. In: *Proceedings of the Tenth International Conference on Soft Computing and Pattern Recognition (SoCPaR 2018)*. Hrsg. von Ana Maria Madureira ; Ajith Abraham ; Niketa Gandhi ; Catarina Silva ; Mário Antunes. Springer International Publishing, 2019, S. 278–287.

[70] Malderle, Timo ; Wübbeling, Matthias ; Knauer, Sven ; Sykosch, Arnold ; Meier, Michael: „Gathering and Analyzing Identity Leaks for a Proactive

Warning of Affected Users“. In: *Proceedings of the 15th ACM International Conference on Computing Frontiers*. Ischia, Italy: ACM, 2018, S. 208–211.

[71] Malderle, Timo ; Wübbeling, Matthias ; Meier, Michael: „Effektive Warnung bei Identitätsdiebstahl an Hochschulen“. In: *Sicherheit in vernetzten Systemen - 27. DFN-Konferenz*. Hrsg. von Albrecht Ude. Hamburg: BOOKS ON DEMAND, 2020.

[72] Malderle, Timo ; Wübbeling, Matthias ; Meier, Michael: „Sammlung geleakter Identitätsdaten zur Vorbereitung proaktiver Opfer-Warnung“. In: *Multikonferenz Wirtschaftsinformatik 2018 : Data driven X - Turning Data into Value*. Hrsg. von Paul Drews. Lüneburg: Leuphana Universität Lüneburg, Institut für Wirtschaftsinformatik, 2018, S. 1381–1393.

[73] Malik, Sayyam ; Sani, Sana Ahmad ; Baqir, Anees ; Ahmad, Usman ; Mustafa, Faizan ul: „Preprocessing Techniques in Text Categorization: A Survey“. In: *Communications in Computer and Information Science*. Bd. 1198. Springer Nature Singapore, 2020, S. 502–509.

[74] Mayer, Peter ; Volkamer, Melanie: „Addressing misconceptions about password security effectively“. In: *Proceedings of the 7th Workshop on Socio-Technical Aspects in Security and Trust*. New York, NY, USA: Association for Computing Machinery, 2018, S. 16–27.

[75] McCandless, David ; Evans, Tom ; Barton, Paul ; Starling, Stephanie ; Geere, Ducan: *World's Biggest Data Breaches & Hacks*. 2020. url: https://www.informationisbeautiful.net/visualizations/worlds-biggest-data-breaches-hacks/ (Gesichtet am 28. 09. 2020).

[76] Milka, Grzegorz: *Anatomy of Account Takeover*. 2018. url: https://www.usenix.org/node/208154 (Gesichtet am 28. 09. 2020).

[77] Mockapetris, P.V.: *Domain names - implementation and specification*. RFC 1035 (INTERNET STANDARD). Updated by RFCs 1101, 1183, 1348, 1876, 1982, 1995, 1996, 2065, 2136, 2181, 2137, 2308, 2535, 2673, 2845, 3425, 3658, 4033, 4034, 4035, 4343, 5936, 5966, 6604. Internet Engineering Task Force, 1987. url: http://www.ietf.org/rfc/rfc1035.txt.

[78] Mozilla Corporation: *Firefox Monitor*. 2020. URL: https://monitor.firefox.com/ (Gesichtet am 28. 09. 2020).

[79] Naiakshina, Alena ; Danilova, Anastasia ; Gerlitz, Eva ; Von Zezschwitz, Emanuel ; Smith, Matthew: „'If you want, I can store the encrypted password.' A Password-Storage Field Study with Freelance Developers". In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2019, S. 1–12.

[80] namecencus.com: *Name Census: United States Demographic Data*. 2020. URL: https://namecensus.com/ (Gesichtet am 28. 09. 2020).

[81] New Work SE, Xing: *Xing - Mitgliederzahlen DACH*. 2019. URL: https://werben.xing.com/daten-und-fakten (Gesichtet am 28. 09. 2020).

[82] Nottingham, M.: *URI Design and Ownership*. RFC 7320 (Best Current Practice). Internet Engineering Task Force, 2014. URL: http://www.ietf.org/rfc/rfc7320.txt.

[83] Onaolapo, Jeremiah ; Mariconti, Enrico ; Stringhini, Gianluca: „What Happens After You Are Pwnd: Understanding the Use of Leaked Webmail Credentials in the Wild". In: *IMC '16 Proceedings of the 2016 Internet Measurement Conference*. Santa Monica, CA, USA: Association for Computing Machinery, 2016, S. 65–79.

[84] Pal, Bijeeta ; Daniel, Tal ; Chatterjee, Rahul ; Ristenpart, Thomas: „Beyond credential stuffing: Password similarity models using neural networks". In: *2019 IEEE Symposium on Security and Privacy (SP)*. Bd. 2019-May. San Francisco, CA, USA: IEEE, 2019, S. 417–434.

[85] Pasquini, Dario ; Gangwal, Ankit ; Ateniese, Giuseppe ; Bernaschi, Massimo ; Conti, Mauro: „Improving Password Guessing via Representation Learning". In: *Proceedings of the 42nd IEEE Symposium on Security and Privacy*. 2. IEEE, 2019.

[86] Pearman, Sarah ; Thomas, Jeremy ; Naeini, Pardis Emani ; Habib, Hana ; Bauer, Lujo ; Christin, Nicolas ; Cranor, Lorrie Faith ; Egelman, Serge ; Forgetz, Alain: „Let's Go in for a closer look: Observing passwords in their natural habitat". In: *Proceedings of the ACM Conference on Computer and*

Communications Security. New York, NY, USA: Association for Computing Machinery, 2017, S. 295–310.

[87] Peng, Peng ; Xu, Chao ; Quinn, Luke ; Hu, Hang ; Viswanath, Bimal ; Wang, Gang: „What Happens After You Leak Your Password: Understanding Credential Sharing on Phishing Sites“. In: *Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security*. New York, NY, USA: Association for Computing Machinery, 2019, S. 181–192.

[88] Petsas, Thanasis ; Tsirantonakis, Giorgos ; Athanasopoulos, Elias ; Ioannidis, Sotiris: „Two-Factor Authentication: Is the World Ready? Quantifying 2FA Adoption“. In: *Proceedings of the Eighth European Workshop on System Security*. EuroSec '15. New York, NY, USA: Association for Computing Machinery, 2015, S. 1–7.

[89] Pierce, Dean: *Robinhood admits to storing some passwords in cleartext*. 2015. url: https://www.pxdojo.net/2015/08/what-i-learned-from-cracking-4000.html (Gesichtet am 28. 09. 2020).

[90] PricewaterhouseCoopers GmbH: *Identitätsklau - die Gefahr aus dem Netz*. 2016. url: https://www.pwc.de/de/handel-und-konsumguter/assets/cyber-security-identitaetsdiebstahl-2016.pdf (Gesichtet am 28. 09. 2020).

[91] Pullman, Jennifer ; Thomas, Kurt ; Bursztein, Elie: *Protect your accounts from data breaches with Password Checkup*. 2019. url: https://security.googleblog.com/2019/02/protect-your-accounts-from-data.html (Gesichtet am 28. 09. 2020).

[92] Rehman, Abdur ; Javed, Kashif ; Babri, Haroon A. ; Saeed, Mehreen: „Relative discrimination criterion – A novel feature ranking method for text data“. In: *Expert Systems with Applications* 42.7 (2015), S. 3670–3681.

[93] Rogers, Katie: *After Ashley Madison Hack, Police in Toronto Detail a Global Fallout*. 2015. url: https://www.nytimes.com/2015/08/25/technology/after-ashley-madison-hack-police-in-toronto-detail-a-global-fallout.html (Gesichtet am 28. 09. 2020).

[94] Saleem, Hamza ; Naveed, Muhammad: „SoK : Anatomy of Data Breaches“. In: *Proceedings on Privacy Enhancing Technologies*. Bd. 2020. 4. Berlin: Sciendo, 2020, S. 153–174.

[95] Scherschel, Fabian A.: *Dating-Seite Badoo: 127 Millionen Passwort-Hashes im Netz*. 2016. url: https://www.heise.de/security/meldung/Dating-Seite-Badoo-127-Millionen-Passwort-Hashes-im-Netz-3228893.html (Gesichtet am 28. 09. 2020).

[96] Schmerer, Kai: *Twitter hat Nutzer-Passwörter im Klartext abgespeichert*. 2018. url: https://www.zdnet.de/88332925/twitter-hat-nutzer-passwoerter-im-klartext-abgespeichert/ (Gesichtet am 28. 09. 2020).

[97] Schneier, Bruce: *Applied cryptography : protocols, algorithms, and source code in C*. New York, USA: John Wiley & Sons, 2015.

[98] Schwartmann, Rolf ; Weiss, Steffen: *Whitepaper zur Pseudonymisierung der Fokusgruppe Datenschutz der Plattform Sicherheit, Schutz und Vertrauen für Gesellschaft und Wirtschaft im Rahmen des Digital-Gipfels 2017*. 2017. url: https://www.gdd.de/downloads/whitepaper-zur-pseudonymisierung (Gesichtet am 28. 09. 2020).

[99] Scott, Cory: *Protecting Our Members*. 2016. url: https://blog.linkedin.com/2016/05/18/protecting-our-members (Gesichtet am 28. 09. 2020).

[100] Sergienko, Roman ; Shan, Muhammad ; Schmitt, Alexander: „A Comparative Study of Text Preprocessing Techniques for Natural Language Call Routing Roman“. In: *Dialogues with Social Robots: Enablements, Analyses, and Evaluation*. Hrsg. von Kristiina Jokinen ; Graham Wilcock. Bd. 427. Singapore: Springer Singapore, 2017, S. 23–37.

[101] Shafranovich, Y.: *Common Format and MIME Type for Comma-Separated Values (CSV) Files*. RFC 4180 (Informational). Updated by RFC 7111. Internet Engineering Task Force, 2005. url: http://www.ietf.org/rfc/rfc4180.txt.

[102] Shiner, Jeff: *Finding pwned passwords with 1Password*. 2020. url: https://blog.1password.com/finding-pwned-passwords-with-1password/ (Gesichtet am 28. 09. 2020).

[103] Sood, Gaurav ; Cor, Ken: „Pwned: The Risk of Exposure From Data Breaches“. In: *Proceedings of the 10th ACM Conference on Web Science - WebSci '19*. New York, New York, USA: Association for Computing Machinery, 2019, S. 289–292.

[104] Sotirov, Alexander ; Stevens, Marc ; Appelbaum, Jacob ; Lenstra, Arjen ; Molnar, David ; Osvik, D A ; De Weger, B: „MD5 considered harmful today“. In: *Announced at the 25th Chaos Communication Congress*. Berlin, Germany: ChaosComputerClub, 2008, S. 1–25.

[105] SpyCloud, Team: *Our Perspective on the "Collection" Combo Lists*. 2019. url: https://spycloud.com/our-perspective-on-the-collection-combo-lists/ (Gesichtet am 28. 09. 2020).

[106] SpyCloud Inc.: *Data Delivered However You Need It*. 2020. url: https://spycloud.com/products/spycloud-api/ (Gesichtet am 28. 09. 2020).

[107] SpyCloud Inc.: *Protect Employees and Consumers from Account Takeover*. 2020. url: https://spycloud.com/ (Gesichtet am 28. 09. 2020).

[108] Statista GmbH: *Waren Sie schon einmal von Identitätsdiebstahl betroffen, hat also schon einmal jemand Ihre persönlichen Daten missbräuchlich genutzt und Ihnen Schaden zugefügt?* 2019. url: https://de.statista.com/prognosen/953397/umfrage-in-deutschland-zu-personen-die-opfer-eines-identitaetsdiebstahls-geworden-sind (Gesichtet am 28. 09. 2020).

[109] Stobert, Elizabeth ; Biddle, Robert: „The password life cycle“. In: *ACM Transactions on Privacy and Security* 21.3 (2018).

[110] Stobert, Elizabeth ; Biddle, Robert: „The Password Life Cycle: User Behaviour in Managing Passwords“. In: *10th Symposium On Usable Privacy and Security (SOUPS 2014)*. Bd. 10. Menlo Park, CA: USENIX Association, 2014.

[111] Stockton, Rachael: *LastPass Selects Enzoic for Compromised Credential Screening*. 2020. url: https://blog.lastpass.com/2020/01/all-your-passwords-every-device-for-free.html/ (Gesichtet am 28. 09. 2020).

[112] Symantec Corporation: *2017 - Norton Cyber Security Insights Report - Global Results*. 2017. URL: https://www.nortonlifelock.com/content/dam/nortonlifelock/docs/about/2017-ncsir-global-results-en.pdf (Gesichtet am 28. 09. 2020).

[113] Thomas, Kurt ; Moscicki, Angelika ; Margolis, Daniel ; Paxson, Vern ; Bursztein, Elie ; Li, Frank ; Zand, Ali ; Barrett, Jacob ; Ranieri, Juri ; Invernizzi, Luca ; Markov, Yarik ; Comanescu, Oxana ; Eranti, Vijay: „Data Breaches, Phishing, or Malware?“ In: *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security - CCS '17*. New York, NY, USA: Association for Computing Machinery, 2017, S. 1421–1434.

[114] Thomas, Kurt ; Pullman, Jennifer ; Yeo, Kevin ; Raghunathan, Ananth ; Kelley, Patrick Gage ; Invernizzi, Luca ; Benko, Borbala ; Pietraszek, Tadek ; Patel, Sarvar ; Boneh, Dan ; Bursztein, Elie: „Protecting accounts from credential stuffing with password breach alerting“. In: *28th {USENIX} Security Symposium, {USENIX} Security 2019, Santa Clara, CA, USA, August 14-16, 2019*. Santa Clara, CA, USA: USENIX Association, 2019, S. 1556–1571.

[115] Universität Bonn: *identity leak checker*. 2020. URL: https://leakchecker.uni-bonn.de/ (Gesichtet am 28. 09. 2020).

[116] Universität Bonn - Institut für Informatik IV: *Effektive Information nach digitalem Identitätsdiebstahl*. 2019. URL: https://itsec.cs.uni-bonn.de/eidi/ (Gesichtet am 28. 09. 2020).

[117] Ur, Blase ; Noma, Fumiko ; Bees, Jonathan ; Segreti, Sean M ; Shay, Richard ; Bauer, Lujo ; Christin, Nicolas ; Cranor, Lorrie Faith: „'I Added '!' at the End to Make It Secure': Observing Password Creation in the Lab“. In: *Proceedings of the eleventh Symposium On Usable Privacy and Security*. Ottawa: USENIX Association, 2015, S. 123–140.

[118] Vettermann, Oliver: „Self-made data protection – is it enough ? Prevention and afer-care of identity theft“. In: *European Journal of Law and Technology* 10.1 (2019).

[119] Vijayarani, S ; Ilamathi, J ; Nithya ; Phil, M: „Preprocessing Techniques for Text Mining -An Overview“. In: *International Journal of Computer Science & Communication Networks* 5.1 (2015), S. 7–16.

[120] Wang, Chun ; Jan, Steve T.K. ; Hu, Hang ; Bossart, Douglas ; Wang, Gang: „The next domino to fall: Empirical analysis of user passwords across online services“. In: *CODASPY 2018 - Proceedings of the 8th ACM Conference on Data and Application Security and Privacy*. Bd. 2018. New York, NY, USA: Association for Computing Machinery, 2018, S. 196–203.

[121] Wang, Ding ; Zhang, Zijian ; Wang, Ping ; Yan, Jeff ; Huang, Xinyi: „Targeted Online Password Guessing: An Underestimated Threat Ding“. In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. Bd. 24-28. New York, NY, USA: Association for Computing Machinery, 2016, S. 1242–1254.

[122] Wang, Feng ; Yang, Canqun ; Wu, Qiang ; Shi, Zhicai: „Constant memory optimizations in MD5 Crypt cracking algorithm on GPU-accelerated supercomputer using CUDA“. In: *ICCSE 2012 - Proceedings of 2012 7th International Conference on Computer Science and Education* Iccse (2012), S. 638–642.

[123] Wang, Ke Coby ; Reiter, Michael K.: „Detecting stuffing of a user's credentials at her own accounts“. In: *Cryptography and Security*. 2019, S. 1–16. arXiv: 1912.11118.

[124] Wash, Rick ; Rader, Emilee ; Berman, Ruthie ; Wellmer, Zac: *Understanding Password Choices: How Frequently Entered Passwords Are Re-used across Websites*. Denver, CO, USA, 2016. url: https://www.usenix.org/conference/soups2016/technical-sessions/presentation/wash.

[125] Web.de: *Was ist Ihre bevorzugte Methode, die notwendige Menge an Passwörtern zu verwalten?* 2019. url: https://www.slideshare.net/WEBDE_DEUTSCHLAND/passwortstudie-59-der-deutschen-internetnutzer-verwenden-passwrter-mehrfach (Gesichtet am 28. 09. 2020).

[126] Whittaker, Z. (ZDNet): *A dating site leaked over a million accounts because of shoddy security*. 2016. URL: http://www.zdnet.com/article/dating-site-leaked-one-million-accounts-because-of-shoddy-security/ (Gesichtet am 28. 09. 2020).

[127] Wikimedia Foundation: *Verzeichnis: Deutsch/Namen/die häufigsten Nachnamen Deutschlands*. 2020. URL: https://de.wiktionary.org/wiki/Verzeichnis:Deutsch/Namen/die_h%5C%C3%5C%A4ufigsten_Nachnamen_Deutschlands (Gesichtet am 28. 09. 2020).

[128] Xie, Zhijie ; Zhang, Min ; Yin, Anqi ; Li, Zhenhan: „A New Targeted Password Guessing Model“. In: *Information Security and Privacy*. Cham: Springer International Publishing, 2020, S. 350–368.

[129] Zetter, Kim: *Hackers Finally Post Stolen Ashley Madison Data*. 2015. URL: https://www.wired.com/2015/08/happened-hackers-posted-stolen-ashley-madison-data/ (Gesichtet am 28. 09. 2020).

Abbildungsverzeichnis

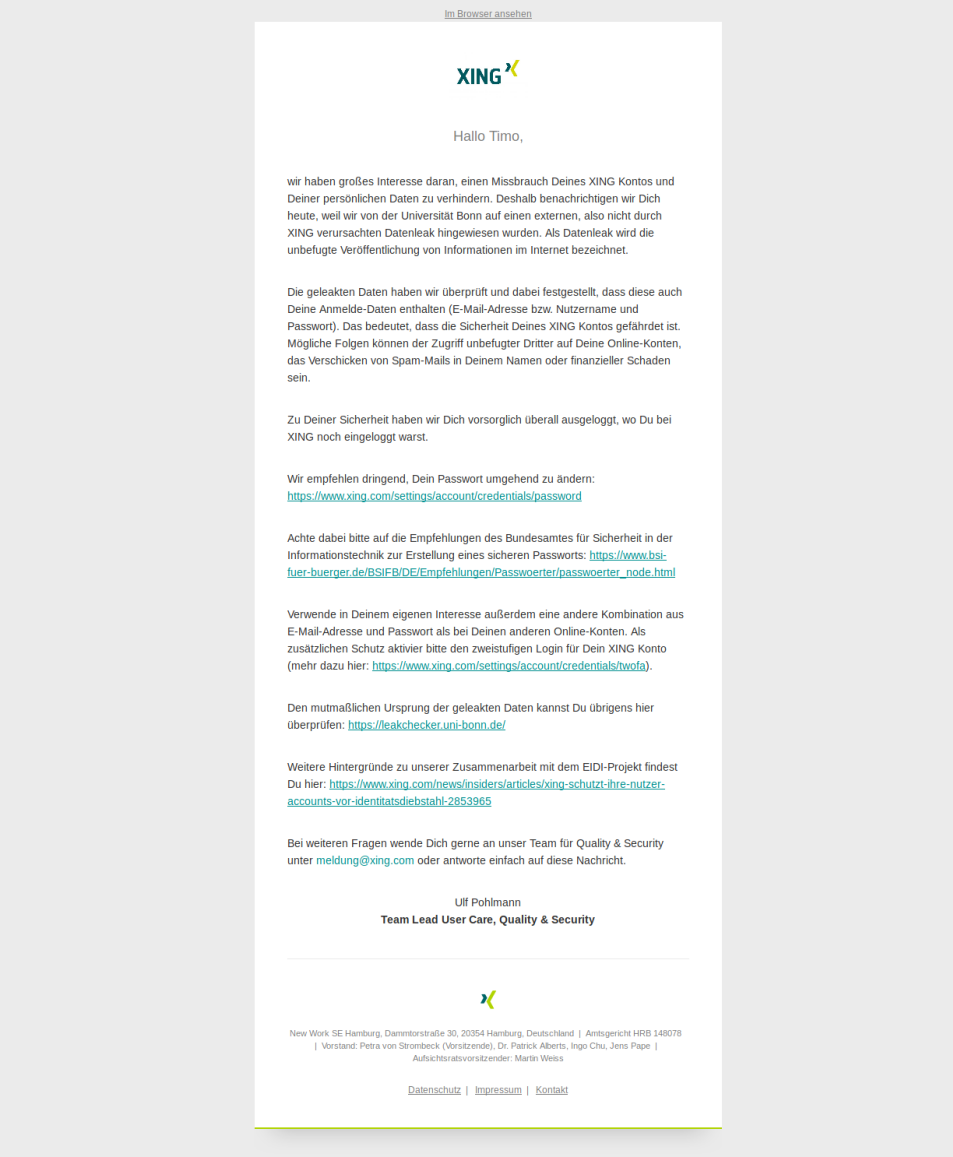
Tabellenverzeichnis

Listings

Liste der Algorithmen

Anhang

A Warn-E-Mail



Im Browser ansehen

XING

Hallo Timo,

wir haben großes Interesse daran, einen Missbrauch Deines XING Kontos und Deiner persönlichen Daten zu verhindern. Deshalb benachrichtigen wir Dich heute, weil wir von der Universität Bonn auf einen externen, also nicht durch XING verursachten Datenleak hingewiesen wurden. Als Datenleak wird die unbefugte Veröffentlichung von Informationen im Internet bezeichnet.

Die geleakten Daten haben wir überprüft und dabei festgestellt, dass diese auch Deine Anmelde-Daten enthalten (E-Mail-Adresse bzw. Nutzername und Passwort). Das bedeutet, dass die Sicherheit Deines XING Kontos gefährdet ist. Mögliche Folgen können der Zugriff unbefugter Dritter auf Deine Online-Konten, das Verschicken von Spam-Mails in Deinem Namen oder finanzieller Schaden sein.

Zu Deiner Sicherheit haben wir Dich vorsorglich überall ausgeloggt, wo Du bei XING noch eingeloggt warst.

Wir empfehlen dringend, Dein Passwort umgehend zu ändern:
https://www.xing.com/settings/account/credentials/password

Achte dabei bitte auf die Empfehlungen des Bundesamtes für Sicherheit in der Informationstechnik zur Erstellung eines sicheren Passworts: https://www.bsi-fuer-buerger.de/BSIFB/DE/Empfehlungen/Passwoerter/passwoerter_node.html

Verwende in Deinem eigenen Interesse außerdem eine andere Kombination aus E-Mail-Adresse und Passwort als bei Deinen anderen Online-Konten. Als zusätzlichen Schutz aktivier bitte den zweistufigen Login für Dein XING Konto (mehr dazu hier: https://www.xing.com/settings/account/credentials/twofa).

Den mutmaßlichen Ursprung der geleakten Daten kannst Du übrigens hier überprüfen: https://leakchecker.uni-bonn.de/

Weitere Hintergründe zu unserer Zusammenarbeit mit dem EIDI-Projekt findest Du hier: https://www.xing.com/news/insiders/articles/xing-schutzt-ihre-nutzer-accounts-vor-identitatsdiebstahl-2853965

Bei weiteren Fragen wende Dich gerne an unser Team für Quality & Security unter meldung@xing.com oder antworte einfach auf diese Nachricht.

Ulf Pohlmann
Team Lead User Care, Quality & Security

New Work SE Hamburg, Dammtorstraße 30, 20354 Hamburg, Deutschland | Amtsgericht HRB 148078 | Vorstand: Petra von Strombeck (Vorsitzende), Dr. Patrick Alberts, Ingo Chu, Jens Pape | Aufsichtsratsvorsitzender: Martin Weiss

Datenschutz | Impressum | Kontakt

Abbildung A.2: *E-Mail, die von XING an deren Benutzer zur Warnung versendet wird [eigener Screenshot von erhaltener E-Mail versendet durch* `mailrobot@mail.xing.com` *an eigenes Postfach, empfangen am 04.08.2020].*

Hier wird der Text aus Abbildung A.2 zur besseren Lesbarkeit in Textform abgebildet[1]:

Hallo Timo,
wir haben großes Interesse daran, einen Missbrauch Deines XING Kontos und Deiner persönlichen Daten zu verhindern. Deshalb benachrichtigen wir Dich heute, weil wir von der Universität Bonn auf einen externen, also nicht durch XING verursachten Datenleak hingewiesen wurden. Als Datenleak wird die unbefugte Veröffentlichung von Informationen im Internet bezeichnet.
Die geleakten Daten haben wir überprüft und dabei festgestellt, dass diese auch Deine Anmelde-Daten enthalten (E-Mail-Adresse bzw. Nutzername und Passwort). Das bedeutet, dass die Sicherheit Deines XING Kontos gefährdet ist. Mögliche Folgen können der Zugriff unbefugter Dritter auf Deine Online-Konten, das Verschicken von Spam-Mails in Deinem Namen oder finanzieller Schaden sein.
Zu Deiner Sicherheit haben wir Dich vorsorglich überall ausgeloggt, wo Du bei XING noch eingeloggt warst.
Wir empfehlen dringend, Dein Passwort umgehend zu ändern: https://www.xing.com/settings/account/credentials/password
Achte dabei bitte auf die Empfehlungen des Bundesamtes für Sicherheit in der Informationstechnik zur Erstellung eines sicheren Passworts: https://www.bsi-fuer-buerger.de/BSIFB/DE/Empfehlungen/Passwoerter/passwoerter_node.html
Verwende in Deinem eigenen Interesse außerdem eine andere Kombination aus E-Mail-Adresse und Passwort als bei Deinen anderen Online-Konten. Als zusätzlichen Schutz aktivier bitte den zweistufigen Login für Dein XING Konto (mehr dazu hier: https://www.xing.com/settings/account/credentials/twofa).
Den mutmaßlichen Ursprung der geleakten Daten kannst Du übrigens hier überprüfen: https://leakchecker.uni-bonn.de/
Weitere Hintergründe zu unserer Zusammenarbeit mit dem EIDI-Projekt findest Du hier: https://www.xing.com/news/insiders/articles/xing-schutzt-ihre-nutzer-accounts-vor-identitatsdiebstahl-2853965
Bei weiteren Fragen wende Dich gerne an unser Team für Quality & Security unter meldung@xing.com oder antworte einfach auf diese Nachricht.
Ulf Pohlmann
Team Lead User Care, Quality & Security

[1] Textkopie von erhaltener E-Mail versendet durch mailrobot@mail.xing.com an eigenes Postfach, empfangen am 04.08.2020

B Warnmeldung nach Login

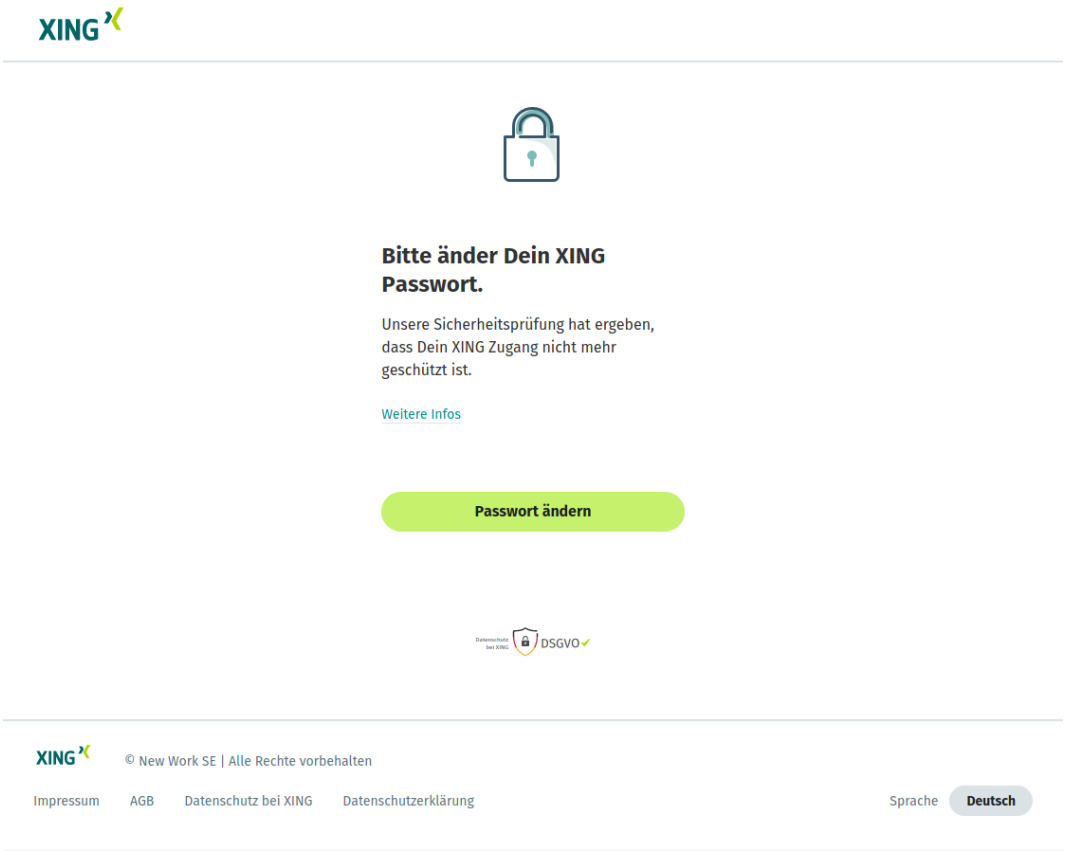


Abbildung B.3: *Warnmeldung nach einer erfolgreichen Anmeldung mit kompromittierten Zugangsdaten [Quelle: eigener Screenshot der Website XING (`xing.com`) nach Anmeldung mit eigenen Zugangsdaten, abgerufen am 07.08.2020].*