Lufthansa Systems

DHBW
Duale Hochschule
Baden-Württemberg
Mannheim

Duale Hochschule Baden-Württemberg
Mannheim

Bachelorarbeit

Analyse und Integration einer passwortlosen Authentifizierung im Unternehmenskontext

Studiengang Cyber Security

Verfasser:	Luka Tsipitsoudis
Matrikelnummer:	4110112
Kurs:	TINF20CS1
Bearbeitungszeitraum:	06.06.2023 – 29.08.2023
Abgabedatum:	29.08.2023
Betreuer:	Stefan Köster

Ehrenwörtliche Erklärung

Ich versichere hiermit, dass ich meine Projektarbeit mit dem Thema: „*Analyse und Integration einer passwortlosen Authentifizierung im Unternehmenskontext*“ selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Ich versichere zudem, dass die eingereichte elektronische Fassung mit der gedruckten Fassung übereinstimmt.

Ort, Datum

Abstract

Deutsch

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Abkürzungsverzeichnis

LSY Lufthansa Systems GmbH & Co. KG
FIDO Fast Identity Online
W3C World Wide Web Consortium
SFA Single-Factor Authentication
MFA Multi-Factor Authentication
CTAP2 Client-to-Authenticator Protocol 2
ITU-T International Telecommunication Union Telecommunication Standardization Sector
MITM Man In The Middle
EUF-CMA Existential Unforgeability under a Chosen Message Attack
PQ Post-Quantum
KEM Key Encapsulation Mechanism
TLS Transport Layer Security
puvProtocol PIN/UV Auth Protocol
SUF Strongly Unforgeable
UF Unforgeable
OTP One-Time Password
HOTP HMAC-based One-Time Password
TOTP Time-based One-Time Password
HMAC Hash-based Message Authentication Code
AD Active Directory
SSO Single Sign-On
U2F Universal Second Factor
IDS Intrusion Detection System
DDoS Distributed Denial of Service
2FA Two-Factor Authentication
API Application Programming Interface

ECDH Elliptic Curve Diffie-Hellman

TEE Trusted Execution Environment

1 Einführung

Diese Arbeit beschäftigt sich mit passwortlosen Authentifizierungsverfahren. Im Folgenden werden zunächst die Problemstellung und das Ziel der Arbeit erläutert. Anschließend wird der Aufbau der Arbeit beschrieben und auf verwandte Arbeiten eingegangen:

1.1 Problemstellung & Ziel der Arbeit

Die Problemstellung dieser Arbeit bezieht sich auf den aktuellen, passwortlosen Ansatz der Authentifizierung im Unternehmenskontext der Lufthansa Systems GmbH & Co. KG (LSY). Trotz ihrer hohen Etablierung und Verbreitung bieten passwortlose Authentifizierungsverfahren nicht nur Vorteile, sondern auch eine hohe Anzahl an Angriffsvektoren.

Ziel dieser Arbeit ist es daher passwortlose Authentifizierungsverfahren als Alternative genauer zu betrachten. Verschiedene passwortlose Verfahren werden vorgestellt und ihre individuellen Vor- und Nachteile aufgezeigt. Dabei soll ein besonderes Augenmerk auf den Vergleich der Angriffsvektoren von passwortlosen und passwortbasierten Verfahren gelegt werden. Ein besonderer Fokus liegt auf der Analyse von FIDO2 in Kombination mit einem Security Key. Dabei wird analysiert, ob das Verafahren für die LSY geeignet ist und welche Anpassungen vorgenommen werden müssen. Betrachtet werden insbesondere die Aspekte der Sicherheit und der Benutzerfreundlichkeit. Der Fokus liegt auf der Frage, ob passwortlose Verfahren eine Alternative darstellen, welche Passwörter gänzlich ersetzen.

1.2 Aufbau der Arbeit

Die Arbeit gibt zunächst eine Einführung in die verschiedenen Arten der Authentifizierung und stellt eine Verknüpfung zu den Schutzzielen der Informatik her. Anschließend wird die Passwortbasierte Authentifizierung genauer betrachtet. Dabei werden die typischen Schwachstellen und Angriffsvektoren detailiert aufgezeigt und beschrieben. Auf Basis dieser Erkentnisse werden mögliche passwortlose Alternativen vorgstellt. Diese werden kurz beschrieben und ihre Vor- und Nachteile betrachtet. Spezifischer wird auf die Nutzung von Security Keys (Yubikeys) eingegangen. Ein Fokus liegt dabei auf der Recherche zum Thema der Benutzerfreundlichkeit. Anschließend wird das FIDO2-Protokoll detailiert betrachtet. Die unterliegenden Protkollo WebAuthn und CTAP2 werden dargestellt und deren Funktionsweise erläutert. Hierbei wird ebenfalls der Aspekt der Sicherheit analysiert. So sollen die Unterschiede der passwortbasierten und passwortlosen Authentifizierung verdeutlicht werden, insbesondere im Bezug auf die Sicherheit.

Nach der auf Fachliteratur basierenden Ergebnisse wird die aktuelle Lage der LSY dargestellt. Der Fokus liegt dabei auf den aktuellen Prozessen der Authentifizierung. Um die Ergebnisse der Fachliteratur mit der Praxis zu vergleichen, wird eine Implementierung von FIDO2 in Kombination mit einem Security Key vorgenommen innerhalb einer Abteilung der LSY vorgenommen. Dabei wird betrachtet, ob und wie gut sich FIDO2 in den Unternehmenskontext der LSY integrieren lässt. Die vorgenommene Umsetzung wird aufgezeigt und mit den aktuellen Prozessen verglichen.

Um die Benutzerfreundlichkeit besser bewerten zu können wird auf Basis der erarbeiten Ergebnisse der Fachliteratur ein interaktiver Fragebogen erstellt. Dieser wird innerhalb einer Abteilung der LSY durchgeführt. Die Ergebnisse werden ausgewertet und mit den Ergebnissen der Fachliteratur verglichen. Ziel ist es dabei eine individuelle Empfehlung für die LSY auszusprechen.

Abschließend soll auf Basis der Literaturrecherche und der Auswertung des Fragebogens eine Empfehlung für die LSY ausgesprochen werden. Dabei wird die Frage beantwortet, ob passwortlose Authentifizierungsverfahren eine aktuelle Alternative für die LSY darstellen.

1.3 Referenzierte Arbeiten

2 Grundlagen

Im Folgenden werden auf Basis der fachlichen Literaturrechere die benötigten Grundlagen dieser Arbeit beschrieben und dargestellt:

2.1 Einführung in cFront

cFront ist eine Anwendung, welche von der Abteilung cGroup Solutions innerhalb der LSY entwickelt wird. Es handelt sich bei der Anwendung um ein Tool, welches an Flughafen Terminals eingesetzt wird.

Es agiert dabei als Interface für mehrere Anwendung und ermöglicht es dem Nutzer, diese Anwendungen zu starten. Welche Anwendungen sichtbar sind, können je nach Kunde individuell konfiguriert werden. Ein Beispiel für ein solches Startportal wird in **2.1** dargestellt.

2.2 CIA-Triade

Die CIA-Triade gehört zu den wichtigsten Darstellungen von Sicherheitszielen innerhalb der Informationssicherheit. Sie beschreibt die drei Schutzziele *Confidentiality* (Vertraulichkeit), *Integrity* (Integrität) und *Availability* (Verfügbarkeit). Im Folgenden werden diese kurz beschrieben:

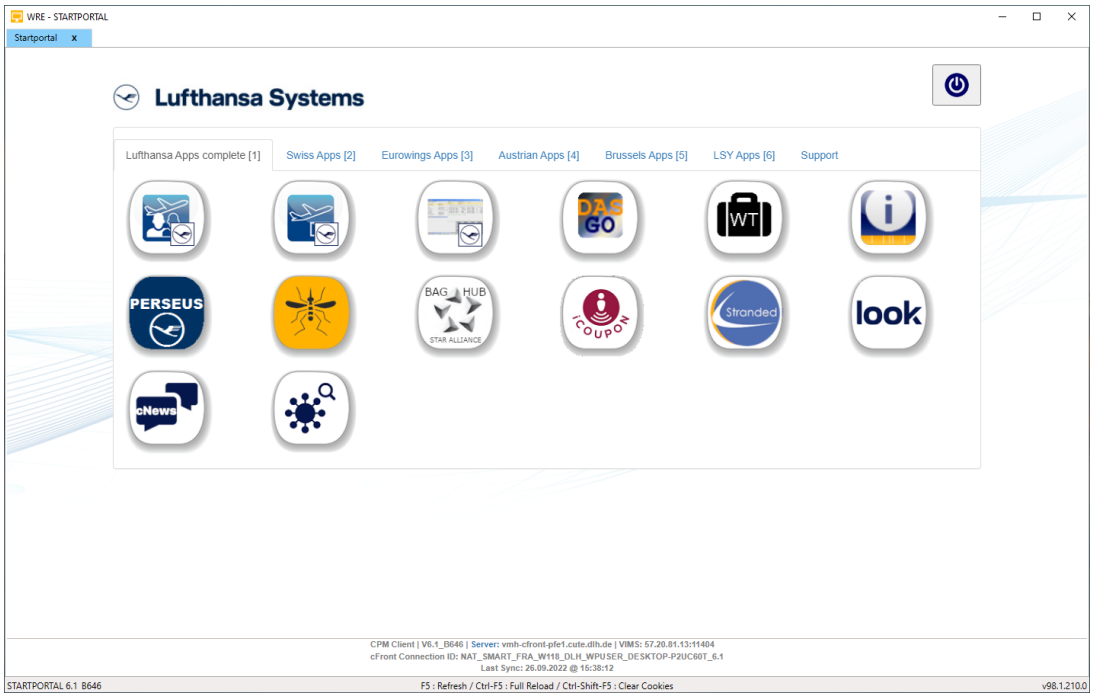

Abbildung 2.1: cFront Startportal

Confidentiality Die Vertraulichkeit gehört zu den wichtigsten Schutzzielen in der Informationssicherheit [1]. Das Wort *Confidentiality* kommt vom lateinischen Wort *confidere* und bedeutet so viel wie *vertrauen* [2] [1]. Das Schutzziel besagt, dass Informationen und Daten so geschützt sein müssen, dass diese nur von autorisierten Personen und für autorisierte Zwecke genutzt werden können [1]. Dies beinhaltet beispielsweise Einschränkungen des Zugriffs auf Informationen und Daten, um die Privatsspähre und persönliches Eigentum zu schützen [1]. Aber auch Verschlüsselungen, eine sichere Authentifizierung und Sicherheitsprotokolle können zur Gewährleistung der Vertraulichkeit beitragen [3]. Aufgrund der steigenden Wichtigkeit von wirtschaftlichen Aspekten hat die Vertraulichkeit im Vergleich zu früher an Bedeutung verloren [1]. Häufig werden Schutzziele vernachlässigst, um im Gegenzug eine erhöhte Benutzerfreundlichkeit oder Wirtschaftlichkeit zu erreichen.

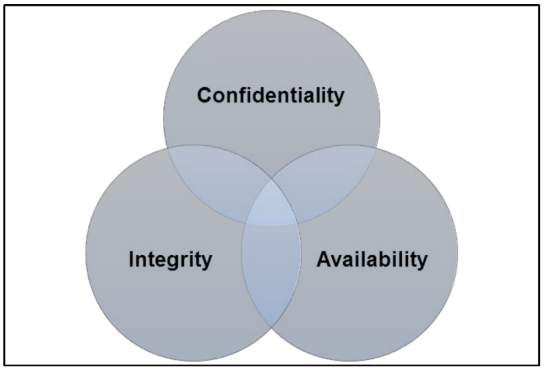

Abbildung 2.2: CIA-Triad

Integrity Das Wort Integrity leitet sich vom lateinischen Wort *tangere* ab und bedeutet so viel wie *berühren* [2]. Durch die Vorsilbe *In-* soll eine Gegenteilige Bedeutung entstehen im Sinne von *Unberührbarkeit* [1]. Die Integrität soll somit garantieren, dass Daten nicht verändert werden können, ohne dass dies bemerkt wird [3]. Schickt ein Sender S beispielsweise eine Nachricht an einen Empfänger E, so soll die Nachricht identisch beim Empfänger E ankommen, wie sie vom Sender S gesendet wurde [3]. Umsetzungsmöglichkeiten die Integrität zu schützen beinhalten Maßnahmen wie beispielsweise das Verwenden einer Firewall, Intrusion Detection System (IDS) oder auch digitale Signaturen [3].

Availability Das Wort Availability leitet sich vom lateinischen *valere* ab und bedeutet so viel wie *kräftig sein*. Die Verfügbarkeit bezieht sich also auf einen zeitnahen und und zuverlässigen Zugriff auf Informationen und Daten [1]. Zuverlässig bedeutet dabei auch, dass ein Zugriff möglichst ohne Unterbrechungen und unabhängig vom Standort möglich ist [3]. Verfügbarkeit kann beispielsweise durch Netzwerksicherheit (z.B. Schutz vor Distributed Denial of Service (DDoS)) oder Fehlertoleranz (z.B. durch Limitierung von Authentifizierungsversuchen) gewährleistet werden [3].

Ein wichtiger Aspekt der Schutzziele ist, dass diese nicht unabhängig voneinander betrachtet werden dürfen. Vielmehr handelt es sich, um ein Zusammenspiel

der verschiedenen Schutzziele (siehe 2.2). So kann eine Maßnahme beispielsweise mehrer Schutzziele schützen. Ebenfalls lassen sich weitere Schutzziele aus den drei bestehenden ableiten. Häufig werden erweiterte Schutzziele wie beispielsweise Authenticity (Authentizität) oder Non-repudiation (Nicht-Abstreitbarkeit) [1] definiert. Diese können dabei zumeist von einem oder mehreren Schutzzielen der CIA-Triade abgeleitet werden. Die folgende Grafik zeigt beispielhaft einige erweiterte Schutzziele und deren Bezug zu der CIA-Triade:

Additional Tenets	Relation to CIA triad
Authenticity	Integrity
Non-repudiation	Integrity
Correctness in specification	Integrity and Availability
Responsibility	Integrity
Integrity of people	Integrity
Trust	Confidentiality and Integrity
Ethicality	Integrity
Identity management	Confidentiality, Integrity and Availability

Abbildung 2.3: Erweiterte Schutzziele

2.3 Arten der Authentifizierung

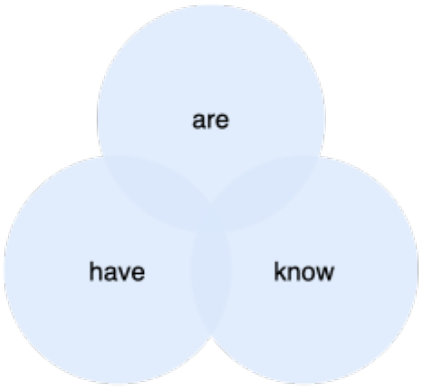

Abbildung 2.4: Faktoren der Authentifizierung

Die Authentifizerung dient häufig als erste Verteidigungslinie von Systemen [4]. Die Authentifizierung gilt als erweitertes Schutzziel und ist eine der wichtigsten Schutznmaßnahmen von Systemen. Sie übernimmt die Kontrolle über die Zugänge von Systemen und bestimmt wer oder was autorisiert ist diese zu nutzen. In der Fachliteratur wird häufig zwischen drei verschiedenen Arten der Authentifizierung unterschieden, welche umgangssprachlich auch als *Faktoren* bekannt sind. Diese sollen im Folgenden beschrieben werden:

Something you know: Die meistgenutzte Art der Authentifizierung basiert auf dem Wissen des Nutzers. Diese Methode nutzt Informationen - welche nur dem Nutzer bekannt sind - und bestätigt somit seine Identität [4]. Das bekannteste Verfahren ist dabei die Nutzung von Passwörtern, welche nur dem Nutzer bekannt sein sollten. Weitere Verfahren dieser Kategorie wären allerdings auch Sicherheitsfragen. Diese werden initial vom Nutzer beantwortet und im weiteren Verlauf zur Authentifizierung abgefragt.

Something you have: Diese Art der Authentifizierung nutzt physische Objekte, um die Identität des Nutzers zu verifizieren. Es handelt sich um Objekte die sich lediglich im Besitz des Nutzers befinden [4]. Mögliche Beispiele für diese Methode sind Smartcards, welche an physische Zutrittskontrollen gehalten werden müssen oder Hardware Tokens, die für die Anmeldung an Systemen genutzt werden.

Something you are: Diese Art der Authentifizierung basiert auf der Inhärenz. Das bedeutet, dass zur Verifizierung der Identität des Nutzers biometrische Merkmale verwendet werden [4]. Dazu gehören u.a. Fingerabdrücke, Geischtserkennung und Iris-Scans. Diese Methode hat sich besonders im Bereich der mobilen Systeme etabliert, so bietet Apple bei seinen Smartphones beispielsweise eine Authentifizierung per Fingerabdruck (*Touch ID*) oder Gesichtserkennung

(*Face ID*) an CITE. Aber auch Microsoft bietet mittlerweile eine Authentifizierung mittels biometrischer Daten an (*Windows Hello* und *Hello for Business*) CITE.

Wie schon bei der CIA-Triade lassen sich auch hier weitere Arten ergänzen oder ableiten. Eine weitere Art ist beispielsweise **something you produce** [4]. Diese Art der Authentifizierung leitet sich teilweise von dem Faktor *something you are* ab. Sie nutzt beispielsweise die Stimme des Nutzers oder seine (digitale) Unterschrift, um seine Identität zu verifizieren [4].

Die verschiedenen Arten der Authentifizierung spielen auch bei der Unterscheidung zwischen einer Single-Factor Authentication (SFA) und einer Multi-Factor Authentication (MFA) eine wichtige Rolle. Wird eine einzelner Faktor genutzt, so bezeichnet man dies als SFA. Werden mehrere Faktoren genutzt handelt es sich um eine MFA.

2.4 Passwortbasierte Authentifizierung

Die heutzutage am häufigsten genutzte Methode zur Authentifizierung ist die passwortbasierte Authentifizierung [5] [4] [6]. Diese basiert auf dem Faktor *something you know*, also auf dem Wissen der Nutzer. Zumeist handelt es sich um alphanumerische Passwörter, welche aus einer Kombination von Groß- und Kleinbuchstaben, Zahlen und Sonderzeichen bestehen [5]. Die Sicherheit informationstechnischer Systeme ist somit abhängig von der Sicherheit der genutzten Passwörter [4]. Trotz ihrer weitreichenden Verbreitung gelten Passwörter als eine der größten Sicherheitsrisiken für Systeme, da sie vile Schwachstellen und Angriffsvektoren bieten [6] [7]. Laut einer Studie von *Verizon* basierten 2017 81% der Hackerangriffe auf der Kompromittierung von Passörtern [8] [9]. Eine weitere Studie zeigt auf, dass 2017 Phishing E-Mails die Angriffsmethode darstellte [10] [8]. Diese sind darauf ausgelegt an Passwörter von Nutzern zu gelangen.

Eine Vielzahl von großen Unternehmen wurden bereits Opfer von der Veröffentlichung von Passwörtern, obwohl ein hoher Aufwand betrieben wird, um diese zu schützen [4]. Da sich die Enthüllung der Passwörter allerdings als Angriffsziel bei Angreifern etabliert hat, ist selbst ein hoher Aufwand nicht mehr immer ausreichend, um jene zu schützen [4]. Der entstehende Schaden ist immens, da es sich um einen hohen Geldwert, aber u.a. auch um einen Reputationsschaden handeln kann. Trotz der bekannten Schwachstellen und bereits entwickelten alternativen Ansätzen, bleibt das Passwort weiterhin genutzt [11]. Dies liegt insbesondere an der Einfachheit und dem geringen Aufwand, welche die Nutzung von Passwörtern mit sich bringt [6].

Passwörter können durch verschiedene Arten von Angriffen kompromittiert werden. So können Angreifer beispielsweise Zugriff auf die Datenbank erhalten, in welcher die Passwörter gespeicghert werden, aber auch auf persönlicher Ebene können Passörter erlangt werden. Dabei spielt das sog. Social Engineering eine große Rolle. Durch Shoulder Surfing können Angreifer versuchen Nutzern beim Passwort eintippen zuschauen. Mit Hilfe von Dumpster Diving können beispielsweise aufgeschriebene Passwörter erlangt werden. Zu den häufigsten Social Engineering Angriffen gehören allerdings die bereits beschriebenen Phishing Mails. Auf technischer Ebene ist ebenfalls ein Einsatz von Keyloggern möglich, welche alle Tastendrücke des Nutzers speichert. Ein häufig gewähltes und sehr effektives Mittel bei schlechten Passwörtern sind allerdings Brute-Force- und Dictionary-Angriffe. Diese kompromittieren Passwörter durch das stupide Ausprobieren aller möglichen Kombinationen oder die Nutzung von Tabellen, welche die meistgenutzten Passwörter beinhalten [5] [12].

Um Passwörter resistenter gegen Brute-Force-Angriffe zu gestalten, kann eine Erweiterung des Zeichenraums oder der Passwort-Länge genutzt werden. So wird die mögliche Anzahl an Kombinationen des Passworts erhöht. Je mehr mögliche Kombinationen es gibt, desto schwieriger wird es Passwörter durch Erraten zu komprommittieren [5]. Wichtig ist hierbei, dass die Erweiterung der Passwortlänge deutlich effektiver ist als die Erweiterung des Zeichenraums. Betrachtet

man die Anzahl aller Elemente des Zeichenraums Z und die Passwortlänge L, so wird die Komplexität eines Passwortes durch Z^L abgebildet. Während die Erweiterung der Passwortlänge ein exponentielles Wachstum aufweist, steigt bei einer Erweiterung des Zeichenraums die Steigung lediglich linear. Die Effektivität längerer Passwörter wird ebenfalls in **2.5** und **2.6** dargestellt. **2.5** stellt die Entropie von Passwörtern in Abhängigkeit ihrer Länge dar. Dabei werden ebenfalls verschieden große Zeichenräume betrachtet. Es wird deutlich, dass selbst eine hohe Differenz des Zeichenraumes lediglich einen geringen Einfluss auf die Entropie hat. Unabhängig vom Zeichenraum aber dennoch eine hohe Entropie durch ein größere Länge möglich ist. **2.6** stellt die benötigte Zeit zum Brechen von Passwörtern in Abhängigkeit zu ihrer Länge dar. Bei beiden Varianten handelt es sich um eine zu kurze Länge eines Passwortes, allerdings ist der signifikante Unterschied durch die Erweiterung der Passwortlänge um eins deutlich erkennbar.

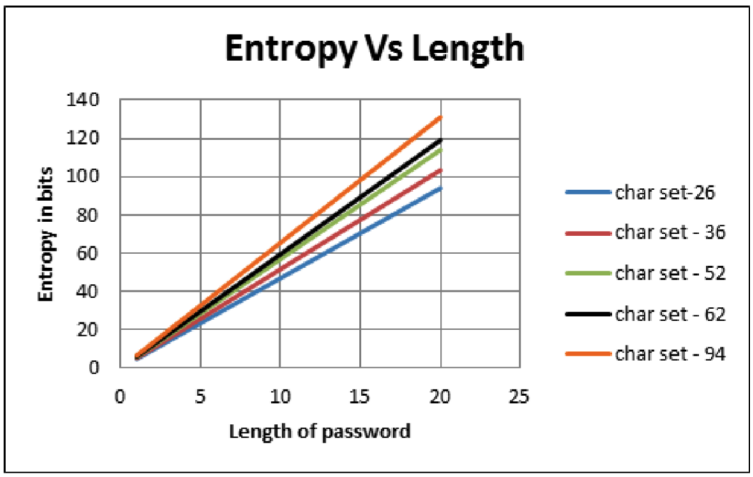

Abbildung 2.5: Entropie in Abhängigkiet der Passwortlänge

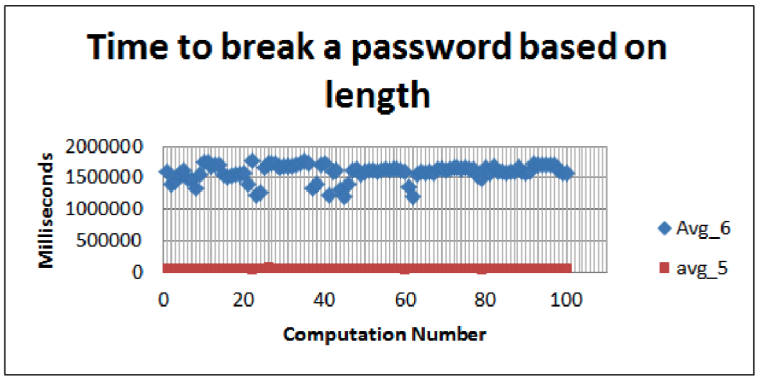

Abbildung 2.6: Zeit, um ein Passwort zu brechen in Abhängigkeit zu der Länge

Zwei Angriffsvektoren sind dabei zumeist betroffen: die Speicherung und der Mensch. Im Folgenden soll präziser erläutert werden, was diese beiden Angriffsvektoren so verwundbar machen:

Speicherung:

Viele Angreifer versuchen Passwörter zu kompromittieren, indem sie Zugriff auf die Datenbank erhalten, in welcher die Passwörter gespeichert sind. Mit Hilfe der erlangten Passwörtern erhoffen sie sich zumeist einen erweiterbaren Zugriff auf Systeme oder nutzen die Passwörter, um ihre Opfer zu erpressen [4]. Der wichtigste Faktor für den Erfolg solcher Angriffe spielt die Art der Speicherung. Abhängig von der Art wie Passwörter gespeichert sind offenbaren sich auch verschiedene Schwachstellen [5]. Die schlechteste, aber dennoch immer noch genutzte Art Passwörter zu speichern ist die Speicherung von Passwörtern im Klartext. Die Passwörter werden also in lesbarer Form gespeichert. Haben Angreifer also Zugriff auf die Datenbank, so können sie alle gespeicherten Passwörter ohne weiteren Aufwand auslesen [5].

Eine bessere Variante - allerdings weitaus nicht optimale - ist die Verschlüsselung der gespeicherten Passwörter. Der größte Kritikpunkt an dieser Variante ist allerdings, dass Verschlüsselungen zurückführbar sind. Das bedeutet mit dem Besitz des benötigten Schlüssels, lassen sich alle gespeicherten Daten ebenfalls

in Klartext umwandeln. Hierbei müssen Angreifer also einen weiteren Aufwand erbringen, um an den benötigten Schlüssel zu gelangen. Sind sie allerdings im Besitz dieses Schlüssels können sie ebenfalls alle gespeicherten Passwörter auslesen [5].

Um die Sepicherung weiter zu optimieren sollte somit keine Zurückführbarkeit bestehen. Dies kann mit Hilfe von Hashing umgesetzt werden. Sog. Hashfunktionen erhalten einen Eingabewert und bilden diesen auf (im Optimalfall) einen einzigen Ausgabewert ab. Dieser Ausgabewert ist nicht zurückführbar auf den Eingabewert. Komprommittieren Angreifer also die Datenbank, in welcher die Passwörter gespeichert sind, können diese die gespeicherten Werte nicht direkt weiterverwenden. Auch dieser Ansatz birgt allerdings Schwachstellen. So lassen sich beispielsweise sog. Rainbow-Tables nutzen, um Hash-Werte zurückzuführen. Dies wird ermöglicht indem häufig genutzte Passwörter gehashed werden und dann mit Hash-Werten innerhalb der Datenbank verglichen werden [5].

Um auch diese Schwachstelle zu verhindern, wird ein sog. Salt benötigt. Dabei wird an jedes Passwort, bevor es gehashed wird, ein individueller randomisierter Wert gehangen. Somit wird verhindert, dass sich der gespeicherte Hashwert mit Hilfe von Rainbow-Tables vergleichen lässt. Auch eine Umsetzung mit zwei Salt-Werten ist möglich. Dabei ist ein Salt öffentlich und der andere privat. So kann ebenfalls ein Schutz gegen offline-Angriffe geboten werden [5].

Faktor Mensch:

Neben den aufgezählten technischen Aspekten, stellt der Mensch selbst eine der größten Angriffsvektoren bezogen auf Passwörter dar [11] [6]. Eins der größten Problem stellt der Aspekt dar, dass von Menschen erstellte Passwörter keine echten Zufallswerte sind. Das liegt insbesondere daran, dass Nutzer sich ihre Passwörter merken müssen. Je komplexer ein Passwort gestaltet ist, desto schwieriger wird es für Nutzer sich dieses zu merken - insbesonere, wenn sie sich mehrere verschiedene Passwörter merken müssen. Daher beinhalten Passwörter häufig

Informationen, welche einen Bezug zum Inhaber haben. Dazu gehören Beispielsweise Namen, Geburtsdaten, Adressen, oder andere persönliche Informationen. Auch Passwörter, welche einfache Muster beinhalten sind sehr beliebt. Dazu gehören beispielsweise *qwertz*, welches die ersten Buchstaben auf der Tastatur darstellt und *123456*. Solche Passwörter können sich Menschen besser einprägen, was notwendig ist, wenn Passwörter häufig genutzt werden müssen. Aus dem identischen Grund neigen Nutzer ebenfalls dazu ein Passwort für mehrere Systeme zu nutzen [5] [4] [6].

Die genannten Faktoren führen dazu, dass die Anzahl an genutzten Kombinationen für ein Passwort deutlich geringer ist als die gesamte Menge an möglichen Kombinationen [4]. Das macht von Menschen erstellte Passwörter deutlich anfälliger für Angriffe, da diese einfacher zu erraten sind [5]. Dies liegt häufig auch daran, dass die Motivation der Nutzer häufig gering ist, komplexe Passwörter zu erstellen, weil sie sich der Gefahr von schwachen Passwörtern nicht bewusst sind [6]. Kontraproduktiv wirken in diesem Zusammenhang auch Policies und Richtlinien zur Erstellung von Passwörtern [6]. Sind die Richtlinien zur Erstellung von Passwörtern zu komplex, tendieren Nutzer bewusst dazu Muster in das Passwort einzubauen, um sich dieses zu merken. Dies führt zu einem gegenteiligen Effekt, da die Sicherheit und die Komplexität der Passwörter dadurch sinkt. Die These, dass solche Richtlininien zwangsweise zu einer erhöhten Sicherheit beitragen ist somit ein Irrglaube [6] [12].

Ein weiteres Problem stellt die die bereits genannte mehrfache Nutzung eines Passwortes für verschiedene Systeme dar. Aktive Internet-Nutzer verwalten im Durchschnitt 15 Passwörter pro Tag [11]. Um sich also das Einprägen verschiedener Passwörter zu ersparen, wählen Nutzer tendenziell lieber ein Passwort. Das führt häufig zu einem Domino-Effekt im Falle einer Passwort-Kompromittierung. Gelangen Angreifer an ein einzelnes Passwort des Nutzers, ist es häufig möglich mit diesem auch Zugriff auf andere Systeme zu gelangen [11] [12].

2.5 Passwortlose Authentifizierung

Unter dem Sammelbegriff der passwortlosen Authentifizierung werden verschiedene Verfahren zusammengefasst, welche die Nutzung von Passwörten ersetzen sollen. Im Gegensatz zur passwortbasierten Authentifizierung steht also nicht mehr der Faktor *something you know* im Vordergrund, da das Wissen des Nutzers nicht mehr die Grundlage zur Verifizierung seiner Identität darstellen soll. Die Fast Identity Online (FIDO) Allianz nutzt den Begriff passwortlose Authentifizierung beispielsweise, um eine eine SFA oder MFA mit der Hilfe eines Security Keys zu beschreiben [7].

Passwortlose Verfahren werden dabei als sicherer im Vergleich zur passwortbasierten Alternative angesehen, da viele der in **2.4** aufgeführten Angriffsvektoren passwortlosen Ansätze nicht existieren [13] [14]. Zudem erhofft man sich eine zusätzliche erhoffte Benutzerfreundlichkeit durch passwortlose Verfahren - insbesondere, weil Nutzer sich keine Passwörter mehr merken müssen und so ein geringerer Aufwand besteht [13].

Passwortlose Verfahren haben sich jedoch noch nicht flächendeckend durchgesetzt und sind nicht annähernd so weit verbreitet wie die Nutzung von Passwörtern. Dies lässt sich auf mehrere Faktoren zurückführen. Häufig genannte Gründe innerhalb der Fachliteratur sind die Umgewöhnung der Nutzer an eine neuartige Authentifizierung, welches als Hürde zur Etablierung der passwortlosen Verfahren angesehen wird. Aber auch zusätzliche entstehende Kosten durch die Integration der neuen Verfahren können eine Verbereitung ausbremsen [13]. Ein detailierter Einblick in die Vor- und Nachteile in Bezug auf der Benutzerfreundlichkeit wird in **2.6** gegeben.

Es gibt dabei eine Vielzahl an Möglichkeiten eine passwortlose AUthentifizierung umzusetzen. Eine der am häufigsten genutzen Varianten ist die Nutzung von Security Keys in Kombination mit FIDO2. Diese liegen im Fokus dieser Arbeit

und werden in **2.6** und **2.7** genauer beschrieben. Dennoch sollen auch mögliche Alternativen kurz vorgestellt werden:

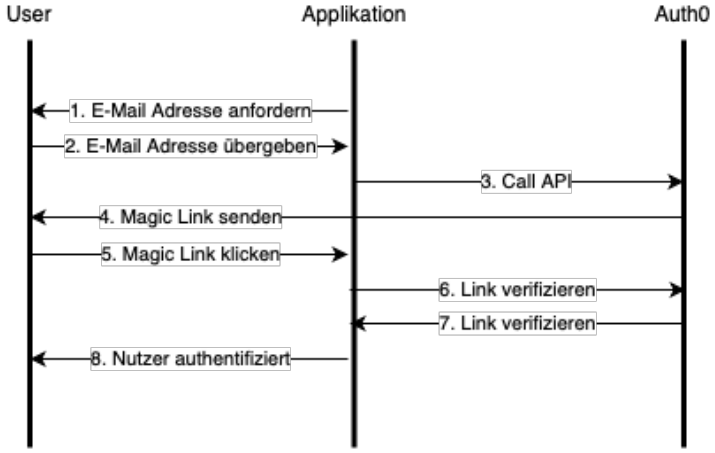

Abbildung 2.7: Beispielhafte Umsetzung eines Magic Links

Magic Link:

Bei einem Magic Link handelt es sich um eine Authentifizierungsmöglichkeit, bei welcher Nutzer lediglich ihren Benutzernamen oder ihre E-Mail-Adresse zur Anmeldung angeben müssen. Anschließend erhält der Nutzer eine E-Mail mit einem dazugehörigen Link, welcher genutzt wird, um seine Identität zu verifizieren [13] [14]. Dieser Link beinhaltet einen Authentication Code, welcher im Hintergrund abgeglichen und validiert wird. Ist die Validierung erfolgreich wird der Nutzer authentifiziert und angemeldet. Nach der Anmeldung verliert der AUthentication Code seine Gültigkeit und somit auch der Link selbst [13]. Der Ablauf des Verfahrens wird ebenfalls vereinfacht in **2.7** dargestellt. Die Sicherheit dieses Verfahrens basiert dabei auf der Annahme, dass der Mail-Server bzw. der Zugang zum Account des Nutzers ausreichend geschützt ist. Ist diese Annahme nicht gegeben können sich auch andere Personen mit dem Link des eigentlichen Nutzers authentifizieren ohne autorisiert zu sein [13].

Vorteile: Ein Passwort bleibt zwar in den meisten Fällen für den Zugriff auf den E-Mail Zugang notwendig, würde aber zumindest die Anzahl an benötigten Passwörtern für Nutzer reduzieren. Zudem handelt es sich bei einem Magic Link um eine sehr benutzerfreundliche und einfach verständliche Art der Authentifizierung [14]. Auch die Implementierung und die Kosten zur Instandshaltung sind verhältnismäßig gering einzuorden [14].

Nachteile: Insbesondere im Unternehmenskontext kann die Nutzung von Spam-Filtern die Benutzerfreundlichkeit von Magic Links stark beeinträchtigen. So können beispielsweise die zugehörigen Mails fälschlicherweise als Spam klassifiziert werden oder eine erhöhte Wartezeit auf die E-Mail entstehen [14]. Auch im Bezug auf das Thema Sicherheit sind einige Aspekte fragwürdig. So hängt die Sicherheit des Verfahrens von der Sicherheit des Mail-Servers ab. Ist dieser nicht ausreichend geschützt, können Angreifer Zugriff auf die Mails erhalten und sich ebenfalls mit dem Link authentifizieren [13]. Dies kann geschehen, ohne dass der Nutzer dies überhaupt bemerkt [13].

One-Time Password (OTP):

Das Konzept hinter OTPs ähnelt dem des Magic Links. Nutzer geben ihre E-Mail-Adresse oder ihre Handynummer an (diese können ebenfalls einem Benutzernamen zugewiesen sein) und erhalten eine E-Mail/SMS, welche ein OTP beinhaltet [13] [14]. Dieses wird vom System abgeglichen und validiert. Ist die Validierung erfolgreich wird der Nutzer authentifiziert und angemeldet. Nach der Anmeldung verliert das OTP seine Gültigkeit [13]. Häufig werden OTPs allerdings nicht für eine oben beschrieben SFA genutzt, sondern dienen als zusätzlicher Faktor für eine MFA [13]. So können beispielsweise Authenticator Apps zur Bereitstellung von OTPs genutzt werden, um die etabliertere passwortbasierte Authentifizierung sicherer zu gestalten. Im Gegensatz zu statischen, von Anwendern gewählten Passwörtern sind OTPs dynamisch erzeugt und haben nur eine geringe Lebensdauer. So wird eine höhere Sicherheit gewährleistet, da OTPs nur schwierig durch stupides Erraten oder Brute Force Attacken erbeutet werden

können [13]. Für die Umsetzung von OTPs gibt es mehrere Möglichkeiten. Zwei häufig verwendete Optionen sind HMAC-based One-Time Password (HOTP) und Time-based One-Time Password (TOTP) [13]. HOTPs basieren auf der technischen Spezifikation RFC 4226. Sie werden mit Hilfe von Hash-based Message Authentication Code (HMAC) und unabhängig von der Zeit generiert. Neue HOTPs können Event-basiert von dem Nutzer angefordert werden [13]. TOTPs basieren auf der technischen Spezifikation RFC 6238 und werden in Abähngigkeit zu der Zeit erstellt. Sie ändern sich nach einem vordefinierten Zeitintervall und sind somit sehr kurzlebig [13].

Vorteile: Die Nutzung von OTPs ist sehr effektiv für eine MFA, da die Sicherheit im Vergleich zu einer passwortbasierten SFA signifikant erhöht werden kann [13] [14]. Zudem handelt es sich um eine sehr benutzerfreundliche und einfachz anwendbare Methode, welche sich bereits für die Nutzung von MFA weitreichend etabliert hat [14]. Dies liegt auch an der Vielzahl an Umsetzungsmöglichkeiten von OTPs, da diese beispielsweise via E-Mail, SMS, Authenticator App oder auch Security Key an den Nutzer übermittelt werden können [13] [14].

Nachteile: Da sich diese Arbeit auf die Nutzung einer passwortlosen Authentifizierung als SFA fokussiert, wird es hier als Nachteil eingeordnet, dass sich die Nutzung von OTPs hauptsächlich für die Umsetzung einer MFA anbietet. Eine Nutzung von OTPs als SFA wird häufig nicht unterstützt. Zudem ist je nach Implementierung wie auch bei einem Magic Link eine Abhängigkeit auf einen anderen Dienst gegeben, welche die Sicherheit des Verfahrens beeinträchtigen können.

Biometrische Daten:

Eine bereits weitreichend etablierte Methode zur passwortlosen AUthentifizierung ist die Nutzung von biometrischen Daten. Diese wird insbesondere im Bereich der mobilen Endgeräte häufig genutzt [14]. Dabei werden einzigartige bio-

metrische Merkmale des Nutzers genutzt um seine Identität zu verifizieren. Dazu gehören beispielsweise Fingerabdrücke oder eine Gesichtserkennung [14]. Im Bereich der Smartphones wird dieses Beispielsweise von Apple durch die Technologien *Touch ID* und *Face ID* umgesetzt. Aber auch Microsoft bietet mittlerweile eine Authentifizierung mittels biometrischer Daten an (*Windows Hello* und *Hello for Business*). Diese Methode lässt sich ebenfalls für eine Integration in den Unternehmenskontext nutzen und ist nicht nur für mobile Endgeräte verfügbar. Wichtig ist hierbei, dass lediglich der reine Zugriff mit biometrischen Daten ermöglicht wird. Die Authentifizierung selbst basiert im Verlauf auf der Nutzung von öffentlich/privaten Schlüsselpaaren.

Vorteile: Viele mobile Endgeräte arbeiten bereits mit biometrischen Daten. Daher ist für eine Vielzahl an Nutzern keine große Umgewöhnung an die neue Art der Authentifizierung notwendig [14]. Da biometrische Daten nahezu einzigartig sind, sind diese ebenfalls deutlich schwieriger anzugreifen als Passwörter. Auch durch die Unterstützung von Microsoft über Windows Hello for Business ist diese Methode bereits für den Unternehmenskontext verfügbar [14].

Nachteile: Äußere Bedingungen können die Erkennung von biometrischen Daten beeinträchtigen. So kann beispielsweise schlechtes Licht bei einer Gesichtserkennung oder staubige Umgebungen bei einem Fingerabdruckscanner die Erkennung beeinträchtigen [14]. Zudem können sich biometrische Daten im Laufe der Zeit verändern. Auch Verletzungen oder Krankheiten können zu Veränderungen der biometrischen Daten beitragen [4]. Auch wenn Microsoft bereits biometrische Daten unterstützt ist es im Unternehmenskontext häufig so, dass verschiedene Hersteller und Geräte genutzt werden. Diese unterstützen nicht alle biometrischen Daten oder sind nicht untereinander kompatibel [14].

Öffentliche/Private Schlüsselpaare:

Bei der Nutzung von öffentlichen und privaten Schlüsselpaaren handelt es sich um eine asymmetrische Verschlüsselung. Dabei wird ein öffentlicher und ein privater Schlüssel generiert. Der öffentliche Schlüssel wird dabei an den Server übermittelt und der private Schlüssel wird auf dem Gerät des Nutzers gespeichert. Die AUthentifizierung erfolgt durch die Nutzung des privaten Schlüssels. Mit Hilfe des öffentlichen Schlüssels kann der Server die Identität des Nutzers verifizieren. Eine genaue Beschreibung des Verfahrens an Hand des FIDO2 Protokolls wird in **2.7** gegeben.

2.6 YubiKey

Ein Security Key ist eine Hardware, welche es ermöglicht einen Nutzer zu authentifizieren, indem dieser mit dem Security Key interagiert (beispielsweise durch einen Knopfdruck) [15]. In der Fachliteratur lassen sich viele verschiedene Bezeichnungen für Security Keys finden. Dazu gehören beispielsweise *Security Token*, *Hardware Token*, *Authentifizierungsgerät*. Um eine einheitliche Bezeichnung zu gewährleisten, wird in dieser Arbeit der Begriff *Security Key* genutzt. Beispielhaft wird in dieser Arbeit ein YubiKey der Series 5 (siehe **2.8**) als Referenzmodell genutzt. Dies basiert auf der Entscheidung, welche in **3.2** getroffen wird. Grundsätzlich sind die Funktionsweisen der verschiedenen Security Keys allerdings sehr ähnlich.

Abbildung 2.8: Yubikey der Series 5

Der YubiKey 5 ermöglicht grundsätzlich drei Arten der Authentifizierung:

1. Eine SFA, welche Passwörter durch ein passwortloses *tap-n-go* Verfahren ersetzt [16].
2. Eine Nutzung des Secruity Keys als zusätzlicher Faktor für eine Two-Factor Authentication (2FA). Somit wird das Passwort zusätzlich abgesichert. Der Security entspricht somit dem zweiten Faktor (*something you have*) [16].
3. Eine passwortlose MFA mit Hilfe einer zusätzlichen PIN für den Security Key [16].

2.6.1 Usability

Es gibt bereits Fachliteratur, welche sich mit der Benutzerfreundlichkeit von Security Keys beschäftigen. Diese beziehen sich zumeist auf die Implementierung von Security Keys in Kleinunternehmen. Die gesammelte Recherche soll genutzt werden um im Folgenden Vor- und Nachteile der Nutzung von Security Keys im Bezug auf deren Benutzerfreundlichkeit zu erläutern. Dies wird in **3.4.4** als Basis für die Evaluation eines Fragebogens für die LSY genutzt.

Vorteile:

- Ergebnisse zeigen, dass Nutzer grundsätzlich bereit sind, Passwörter durch passwortlose Verfahren zu ersetzen [17].
- Passwortlose Verfahren mit Security Key wurden mehr akzeptiert als tradionelle passwortbasierte Verfahren [17].
- Es handelt sich um eine implizite Garantie, dass sich lediglich Nutzer authentifizieren können, welche auch im Besitz des Security Keys sind [17].
- Durch die Nutzung von FIDO2 kann die Benutzerfreundlichkeit erhöht werden, da Nutzer sich keine Passwörter mehr merken müssen. Häufig wird das Verwalten der immer höher werdenden Anzahl an Passwörtern als Problem angesehen [17] [7].

- Es wird ein deutlich geringerer kognitiver Aufwand benötigt, da Nutzer keine neuen Passwörter mehr erstellen und merken müssen [17].
- Zum aktuellen Zeitpunkt wird FIDO2 bereits von einer Vielzahl an Browsern unterstützt (und somit auch die Nutzung von Security Keys). Zusätzlich bieten immer mehr Online-Dienste die Möglichkeit an sich mit Hilfe von FIDO2 zu authentifizieren [17] [7].
- Es handelt sich um offene und standardisierte Protokolle. Das verhindert verschieden Lösungsansätze verschiedener Hersteller und führt zu einer unabhängigeren und universellen Lösung [7].

Nachteile:

- Im Falle einer SFA wird der Verlust des Security Keys als größtes Problem angesehen. Bei Verlust hat auch der Nutzer keinen Zugriff mehr und aktuell gibt es noch keine sichere und effiziente Möglichkeiten, um den Zugriff wiederherzustellen [17].
- Da es sich um zusätzliche Hardware handelt kann diese ebenfalls kaputt gehen [7].
- Im Unternehmenskontext, kann die Verwaltung und Verteilung der Authentifizierungsgeräte zu einem Problem werden [7].
- Da es sich um Hardware handelt, können Zugänge nicht an vertraute Personen weitergegeben werden, da der Zugang nur mit dem Authentifizierungsgerät möglich ist [17].
- Ohne das Authentifizierungsgerät sind keine spontanen Logins möglich [17].
- Bereits das aus der Tasche holen des Authentifizierungsgerätes ist für manche Nutzer bereits eine Hürde [7].

- Es wird ein physischer Aufwand benötigt, da das Authentifizierungsgerät mitgeführt werden muss [17].
- Authentifizierungsgeräte sind häufig mit Kosten verbunden, welche vom Nutzer getragen werden müssen [17].
- Nutzer haben Probleme ein neues Verfahren für die Authentifizierung zu nutzen, da sie sich an das alte Verfahren gewöhnt haben. Das führt dazu, dass Nutzer das neue Verfahren als kompliziert und ungewohnt empfinden. Sie verfügen häufig nicht über das nötige Wissen, um die Funktion und Sicherheit des Verafahrens zu verstehen [17].
- Selbst Nutzern, welchen das Konzept der passwortlosen Authentifizierung gefällt, nutzen in der Praxis häufig weiterhin Passwörter [7].
- Nutzer wollen keine Angewohnheiten verändern, wenn die nicht dazu gezwungen sind [7].
- Nutzer verwenden lieber Passwörter, da sie das Konzept und die Technologie besser verstehen [17].
- Nicht zwangsweise schneller als die Nutzung von Passwortmanagern [7].
- Allgemein fällt das Feeback von Nutzern weniger positiv aus, wenn diese vorher bereits Passwortmanager genutzt haben [7].

Insgesamt lassen sich noch nicht alle Szenarien mit Security Keys abdecken. Es gibt noch spezielle Fälle, in welchen die Nutzung von Passwörtern weiterhin notwendig ist [17]. Auffälig jedoch ist, dass die Teilnehmer der Studien häufig einen skeptischen Blick auf die neuartige Authentifizierung werfen. Trotz der gesammelten Vor- und Nachteile kann keine Annahme darüber getroffen werden, ob die Vor- oder die Nachteile überwiegen und ob das Verfahren grundsätzlich gegenüber der passwortbasierten Alternative bevorzugt wird. Dies wird in **3.4.4** mit Hilfe der hier erarbeiteten Grundlage genauer analysiert.

2.7 FIDO2

Es gibt bereits viele Alternativen zur passwortbasierten Authentifizierung. Einzelne wurden in **2.5** bereits beispielhaft beschrieben. Der Großteil dieser Alternativen wird allerdings nur in einem sehr geringen Ausmaß genutzt [7]. FIDO2 gehört zu den passwortlosen Verfahren, welche am meisten unterstützt werden. Das Protokoll wird von der FIDO Allianz und dem World Wide Web Consortium (W3C) entwickelt und bereitgestellt. Die FIDO Allianz ist eine Organisation mit weltweit über 250 Mitgliedern. Darunter befinden sich Unternehmen wie Google, Microsoft, Apple, Amazon, Visa und viele mehr. Das W3C wurde 1994 - mit dem Ziel das Wachstum des Internets zu gewährleisten - von Tim Berners-Lee gegründet. Es hat über 300 Mitglieder, darunter auch die hier aufgezählten Mitglieder der FIDO Allianz [17] [7] [18]. Aus diesem Grund wird FIDO2 von vielen Browsern standardmäßig unterstützt. Zudem gibt es eine Vielzahl an FIDO2 kompatiblen Authentifizierungsgeräte, darunter u.A. Security Keys oder auch Smartphones [17].

Das Ziel bei der ENtwicklung von FIDO2 ist es Nutzer zu authentifizieren, ohne dass diese ein Passwort verwenden müssen. Statt der Nutzung von Passwörtern basiert das Protokoll auf der Nutzung von Schlüsselpaareen. Dabei können Security Keys genutzt werden, aber auch das Gerät selber oder andere Authentifizierungsgeräte [8] [12]. Diese können zusätzlich ebenfalls mit einer PIN oder biometrischen Daten abgesichert werden. Dabei ist die PIN nicht mit einem Passwort gleichzusetzen. Die PIN wird lediglich für das Authentifizierungsgerät genutzt und wird auch nur auf diesem gespeichert [7] [8]. FIDO2 unterstützt somit sowohl SFA als auch MFA [17] [7]. Zudem wird eine hohe Sicherheit ermöglicht, da Zugangsdaten bereitgestellt werden, welche nicht von Phishing oder Datenlecks betroffen sein können [17]. Dies ist der Fall, da keine geteilten Geheimnisse zwischen Nutzer und Dienst existieren, welche auf einem Server gespeichert werden [12].

In **2.9** werden die möglichen Vor- und Nachteile von FIDO2 aufgezeigt. Da-

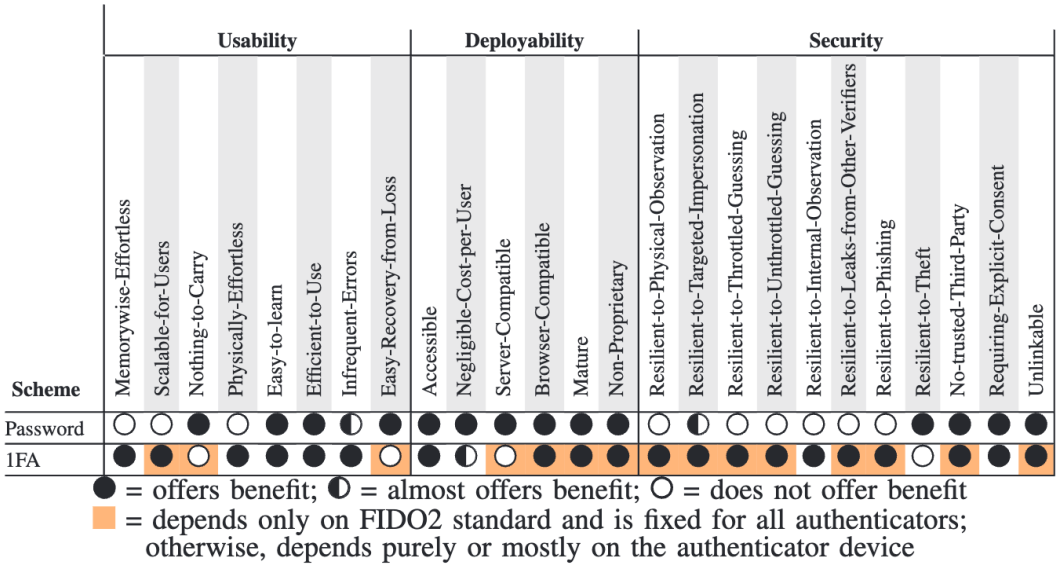

Abbildung 2.9: Mögliche Vorteile von FIDO2

bei steht die Zeile *1FA* für eine SFA mit Hilfe von FIDO2, welche mit der Nutzung von Passwörtern verglichen wird. Orange markiert in der *1FA*-Zeile sind Bereiche, welche grundsätzlich von FIDO2 abhängig sind. Weiß hinterlegte Bereiche sind zusätzlich abhängig vom genutzten Authentifizierungsgerät. Auffällig ist, dass insbesondere der Bereich *Sicherheit* viele Vorteile bietet, welche grundsätzlich durch FIDO2 ermöglicht werden und unabhängig vom genutzten Authentifizierungsgerät sind. Einige dieser Vorteile wurden bereits im Rahmen dieser Arbeit betrachtet. So lassen sich FIDO2 Nutzerdaten beispielsweise nicht erraten oder phishen. Ebenfalls können diese nicht betroffen sein von Datenlecks und auch nicht durch Shoulder Surfing erlangt werden. Ein aufgezeigter Nachteil im Vergleich zu Passwörtern ist jedoch die Möglichkeit des Diebstahls eines Authentifizierungsgerätes. Dies ist allerdings abhängig vom genutzten Gerät und ergibt sich nicht durch die Nutzung von FIDO2. Auch aufällig ist die große Abhängigkeit zum Authentifizierungsgerät im Bereich der Benutzerfreundlichkeit. Dort sind grundsätzlich zwar Vorteile gegenüber der Nutzung von Passwörtern möglich, aber nicht automatisch durch die Nutzung von FIDO2 gegeben.

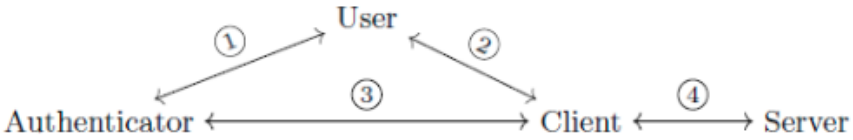

Abbildung 2.10: FIDO2 Teilnehmer

Bei der Nutzung von FIDO2 werden vier Kommunikationskanäle genutzt. Diese werden in **2.10** abgebildet. Es besteht eine Kommunikation zwischen User, Authentifizerungsgerät und dem Client. Der CLient ist üblicherweise ein Browser und übernimmt die Kommunikation mit dem Server. Vereinfacht wird der FIDO2 Ablauf in **2.11** dargestellt. Der User meldet sich über den Client beim Server an (oder registriert sich erstmalig). Daraufhin sendet der Server eine Challenge an den Client weiter, welche vom Authentifizierungsgerät signiert werden muss. Der Client gibt die Challenge an das Authentifizierungsgerät weiter. Bevor dieser die Challenge signieren kann, muss der User die Aktion mit Hilfe einer Geste am Authentifizierungsgerät autorisieren (z.B. ein Knopfdruck). Anschließend signiert das Authentifizierungsgerät die Challenge und sendet diese an den Client zurück. Der Client sendet die signierte Challenge an den Server weiter. Der Server kann die Signatur mit Hilfe des öffentlichen Schlüssels verifizieren und den User authentifizieren [17] [7]. Dieser Ablauf wird im weiteren Verlauf genauer erläutert.

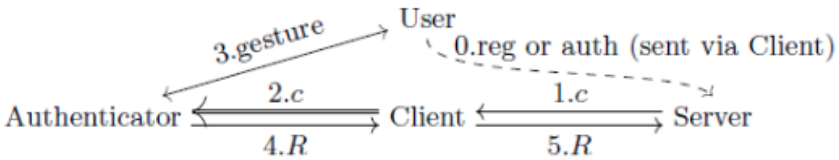

Abbildung 2.11: Vereinfachter FIDO2 Ablauf

Dieser aufgezeigte Ablauf besteht aus den zwei Subprotokollen Client-to-Authenticator Protocol 2 (CTAP2), welches für die Kommunikation zwischen Client und Authentifizierungsgerät genutzt wird und WebAuthn, welches für die Kommunikation zwischen Client und Server zuständig ist. Dabei wird WebAuthn von dem

W3C spezifiziert und CTAP2 von der FIDO Allianz [7].

2.7.1 WebAuthn

Bei WebAuthn handelt es sich um einen Standard, welcher es Webanwendungen ermöglicht Nutzer zu authentifizieren [17]. Seit 2019 gehört WebAuthn zu den offiziellen Webstandards [7]. Er spezifiziert eine standardiserte, vom Browser unabhängige JavaScript Application Programming Interface (API) zur Authentifizierung von Nutzern für Webanwendungen. Ein großer Vorteil von WebAuth ist, dass es Webanwendungen ermöglicht wird eine Authentifizierung zu integrieren, welche resistent gegenüber Phsishing, Datenlecks und Passwortdiebstahl ist. Anstelle von geteilten Geheimnissen nutzt WebAuthn public-key Kryptographie, um einzigartige Zugangsdaten für jede Webanwendung zu erstellen, welche nur auf dem Gerät des Nutzers gespeichert werden [7]. Es handelt sich dabei um ein sog. Challenge-Response-Verfahren zwischen einem Client und einem Server [8]. Die Sicherheit des Standards basiert auf dem Beweis, dass RSASSA PKCS1-v1_5 und RSASSA-PSS als Existential Unforgeability under a Chosen Message Attack (EUF-CMA) gelten und der Annahme, dass SHA-256 kollisionsresistent ist [8].

WebAuthn unterstützt zwei Operationen: Registrierung und Anmeldung. In der Registrierungsphase sendet der Server dem Authentifizierungsgerät über den Client eine zufällige Challenge. In dieser Phase signiert das Authentifizierungsgerät mit Hilfe seines privaten Schlüssels die Challenge und sendet zusätzlich öffentliche Anmeldedaten für zukünftige Anmeldungen an den Server. Meldet sich ein bereits registrierter Nutzer an, wird die Challenge des Servers erneut von dem Authentifizierungsgerät signiert zurück an den Server gesendet. Der Server kann die Signatur mit Hilfe des öffentlichen Schlüssels verifizieren und den Nutzer authentifizieren [8].

Dieser Prozess wird in **2.12** genauer beschrieben (als schwarz dargestellt). Der Server S sendet eine challenge message m_{rch} über den Client C an das Authenti-

fizierungsgerät. Diese Challenge beinhaltet eine randomisierte Nonce, Parameter UV (beispielsweise, ob eine Nutzerverifizierung notwendig ist) und optional einen wert tb, welcher den zugrunde liegenden Kanal eindeutig identifiziert (typischerweise eine Transport Layer Security (TLS) Verbindung). Der Client C erhält die challenge message m_{rch} und wandelt diese in eine command message m_{rcom} und eine client mesasage m_{rcl} um. Die command message m_{rcom} wird an das Authentifizierungsgerät T übermittelt. Das Authentifizierungsgerät T erzeugt ein öffentlich-privates Schlüsselpaar, welches an den Server S gebunden ist und diesem ermöglicht, eine Verifizierung während der folgenden Authentifizierungsphase durchzuführen. Die erzeugten Daten werden dabei nur auf dem Authentifizierungsgerät T gespeichert. Zudem gibt das Authentifizierungsgerät T eine response message m_{rrsp} aus. Der Client übergibt diese und die client message m_{rcl} an den Server S. Die response message m_{rrsp} beinhaltet einen *attestation type*, welcher es dem Server S ermöglicht eine Verifizierung während der Registrierungsphase durchzuführen und beinhaltet den öffentlichen Schlüssel. WebAuthn 2 unterstützt fünf *attestation types*. Häufig werden die types *None* und *Basic* verwendet. Die restlichen types sind *Self*, *AttCA* und *AnonCA*. [19]

Ist ein Nutzer bereits registriert, so empfängt der Client eine challenge message m_{ach} von Server S und wandelt diese in eine command message m_{acom} und eine client message m_{acl} um. Die command message m_{acom} wird an das Authentifizierungsgerät T übermittelt. Das Authentifizierungsgerät T erzeugt eine response message m_{arsp}, welche mit dem privaten Schlüssel signiert wird und sendet diese an den Server S (über den Client C). Der Server S akzeptiert die response message m_{arsp} und die client message m_{acl} nur, wenn sie sich mit dem dazugehörigen öffentlichen Schlüssel und privaten Schlüssel des Servers S verifizieren lassen. [19]

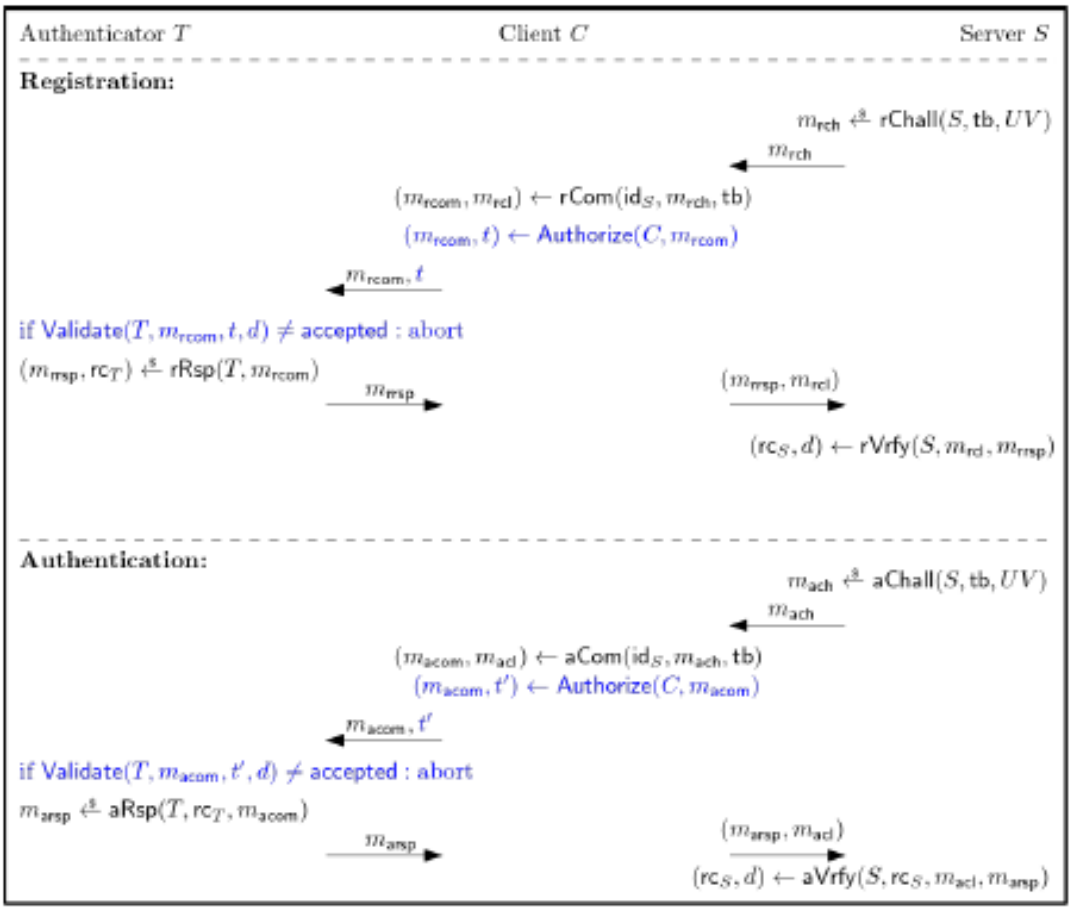

Abbildung 2.12: FIDO2 Darstellung

2.7.2 CTAP2

CTAP2 wurde 2018 als internationaler Standard der International Telecommunication Union Telecommunication Standardization Sector (ITU-T) anerkannt [8]. Es handelt sich um ein Protokoll auf der Anwendungsebene, welches für die Kommunikation zwischen eines WebAuthn Clients und eines konformen Authentifizierungsgerätes genutzt wird. Das Authentifizierungsgerät kann dabei ein externes Gerät sein, z.B. ein Security Key, welcher über USB, Bluetooth oder NFC eine Verbindung mit dem Client aufbaut. Aber auch interne Hardware eines

Gerätes wie beispielsweise ein Fingerabdruckscanner oder ein Trusted Platform Module können als Authentifizierungsgerät genutzt werden [17].

Mit Hilfe von CTAP2 wird die Kommunikation zwischen einem Authentifizierungsgerät und einem Client spezifiziert. Der Client ist dabei üblicherweise ein Webbrowser. Das Ziel ist es zu garantieren, dass der Client das Authentifizierungsgerät nur nutzen darf, wenn der Nutzer dies autorisiert. Dafür muss der Nutzer beispielsweise einen Knopf am Authentifizierungsgerät drücken und/oder sich mit Hilfe eines PINs oder eines biometrischen Merkmals beim Authentifizierungsgerät authentifizieren. Das Ziel von CTAP2 ist es zunächst den Client an das Authentifizierungsgerät zu binden. Ist dies nicht der Fall, so wird es dem Client nicht ermöglicht sich zu authentifizieren. Dies wird auch in **2.12** abgebildet. Die blauen Darstellungen betreffen den Prozess von CTAP2. Bevor die command message m_{rcom} an das Authentifizierungsgerät T gelangt, muss der Nutzer mit Hilfe der persönlichen PIN t die Aktion über den Client C autorisieren. Nach der Eingabe erfolgt die Übergabe an das Authentifizierungsgerät T, welcher diese validiert. Bei nicht erfolgreicher Validierung wird der Prozess abgebrochen [8].

CTAP2 besteht dabei aus vier Phasen. Diese werden im Folgenden kurz erläutert. Da eine zu detailierte Beschreibung den Rahmen dieser Arbeit übertreffen würde, werden die Phasen lediglich in verinfachter Form erläutert. Eine detaillierte Darstellung ist **2.13** und wird genauer in [8] beschrieben. Folgende Phasen beinhaltet CTAP2:

1. In der Reboot Phase wird mit Hilfe eines unauthentifizierten Elliptic Curve Diffie-Hellman (ECDH) nach der NIST P-256 Spezifikation ein Schlüsselpaar (a, aG) generiert (in **2.13** bezeichnet als ECKG). Zudem wird ein pin-Token pt erstellt und der mismatch counter m auf den Wert drei gesetzt. Der mismatch counter m wird genutzt, um die Anzahl der fehlgeschlagenen PIN Eingaben zu zählen [8].

2. In der Setup Phase erfolgt ein unauthentifizierter ECDH Schlüsselaustausch, gefolgt von der Übertragung des verschlüsselten PIN c_p des Nutzers an das Authentifizierungsgerät T. Der geteilte Schlüssel K für die Entschlüsselung ermittelt sich durch das Hashen der x-Koordinate $abG.x$ des ECDH. Durch die Nutzung von AES-256-CBC CBC_0 und des Schlüssels K lässt sich die PIN c_p entschlüsseln. Diese wird vom Authentifizierungsgerät T validiert. Ist die Validierung erfolgreich, wird Hash des PIN pin_u als statisches Geheimnis $st_T.s$ lokal auf dem Authentifizierungsgerät T gespeichert und der retries counter $st_T.n$ auf den standard Wert von acht gesetzt. Fallen diese acht Versuche fehl, so wird das Authentifizierungsgerät T gesperrt [8] [19].

3. Die Bind Phase beginnt ebenfalls mit einem unauthentifizierten ECDH. Darauf folgt eine Übertragung der erst gehashten und dann verschlüsselten PIN c_{ph} vom Client C an das Authentifizierungsgerät T. Diese wird vom AUthentifizierungsgerät T entschlüsselt und mit dem statischen Geheimnis $st_T.s$ verglichen. Ist dieser Vergleich erfolgreich wird der pinToken pt sowohl beim Authentifizerungsgerät T als auch beim Client C als binding state gesetzt [8] [19].

4. Zuletzt findet die Validierungs- und Autorisierungsphase statt. Der Client C schickt einen Befehl mit einem HMAC Tag an das Authentifizerungsgerät T. Dieses validiert den Tag und autorisiert den Befehl nur, wenn eine *positive decision* d des Nutzers vorliegt (beispielsweise einem Knopfdruck) [8] [19].

Eine in der Fachliteratur häufig genannte Kritik an CTAP2 ist die Verwendung des unauthentifizierten ECDH, welcher während der Binding und Setup Phase genutzt wird. Dieser kann von Man In The Middle (MITM) Angriffen betroffen sein [8]. Zudem wird auch die Umsetzung der pinTokens kritisiert. Jedem Authentifizierungsgerät wird beim Hochfahren je ein pinToken zugeteilt. Das bedeutet mehrere Authentifizierungsgeräte können den gleichen pinToken besitzen. Dadurch wird die Sicherheit von CTAP2 limitiert [8].

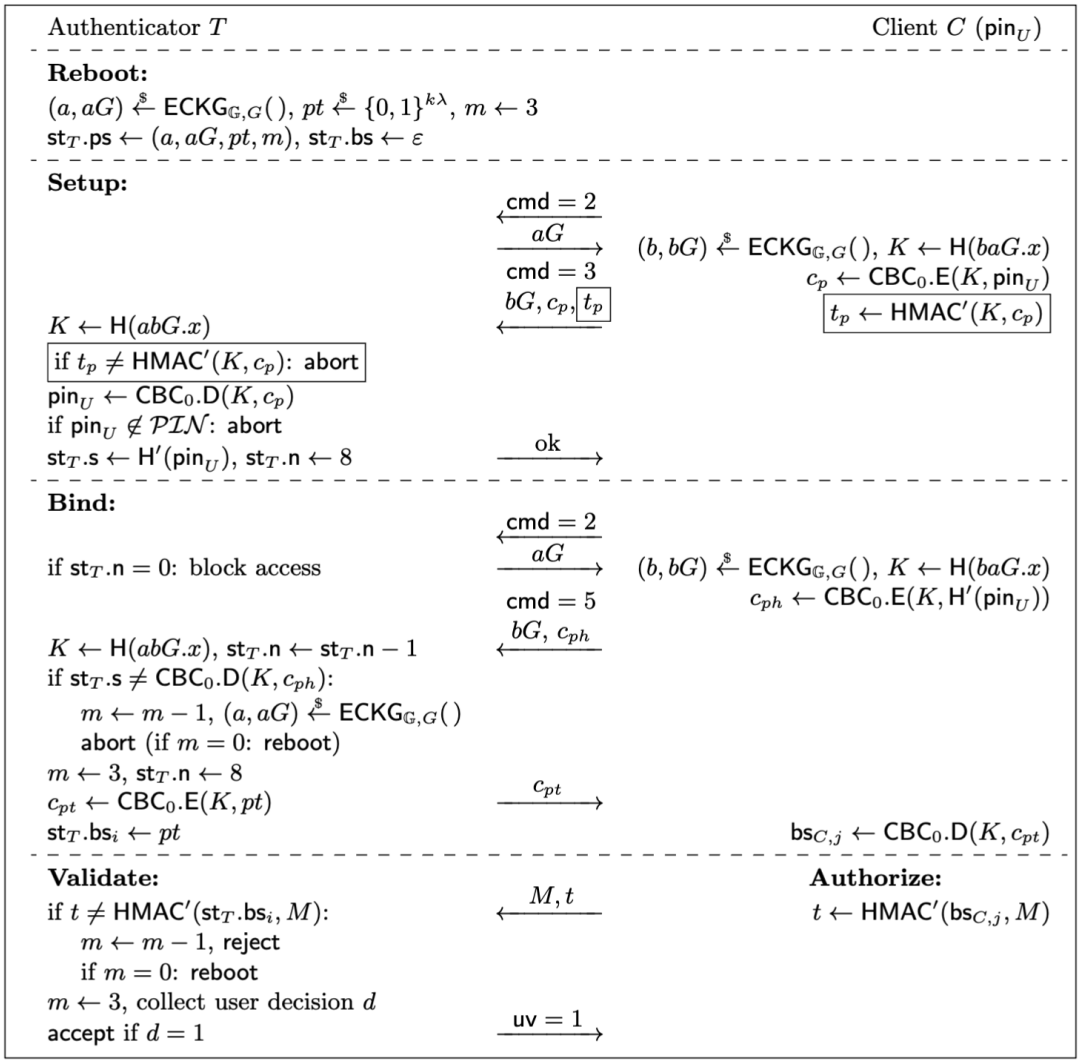

Abbildung 2.13: CTAP2

CTAP2.1

Um diese Kritikpunkte zu beheben wurde CTAP2.1 entwickelt. CTAP2.1 basiert nicht mehr auf unauthentifizierten ECDH, sondern auf einem sog. PIN/UV Auth Protocol (puvProtocol). In Verbindung mit WebAuthn gilt CTAP2.1 als Post-Quantum (PQ) bereit, da ein Operationsmodus ermöglicht wird, welcher

nur kryptographische Primitive und digitale Signaturen und Key Encapsulation Mechanism (KEM) verwendet [19].

Zusätzlich werden die pinTokens im Vergleich zu CTAP2 anders erstellt. Während der pinToken bei CTAP2 aus mehreren 128 Bit-Blöcken besteht und keine maximale Begrenzung der Länge besitzt, wird der pinToken bei CTAP2.1 als pinUvAuthToken definiert. Dieser hat eine feste Länge von 128 oder 256 Bit. Der pinToken von CTAP2 wird bis zum nächsten Neustart wiederverwendet. Der pinUvAuthToken von CTAP2.1 wird nach jeder erfolgreichen Authentifizierung neu generiert. So besteht ein verringertes Risiko, dass sich mehrere Authentifizierungsgeräte den gleichen pinToken teilen. Das führt dazu, dass CTAP2.1 eine Strongly Unforgeable (SUF)-t' Sicherheit aufweist und CTAP2 lediglich eine Unforgeable (UF)-t' Sicherheit [19]. Ebenfalls ermöglicht CTAP2.1 die Nutzung von biometrischen Merkmalen, anstelle von einer individuellen PIN [19].

2.7.3 Sicherheit

FIDO2 wird in der Fachliteratur als grundsätzlich sicheres Protokoll angesehen. Es ist eine Erweiterung des FIDO Universal Second Factor (U2F) Protokolls und es handelt sich um geprüfte asymmetrische Kryptographie [17] [7]. Im Folgenden sollen kurz die wichtigsten Vorteile im Aspekt der Sicherheit aufgezeigt werden:

- Es gibt keine geteilten Geheimnisse zwischen Usern und Diensten, welche durch Phishing oder Datenlecks kompromittiert werden können [17] [7].
- Dasas selbe Authentifizierungsgerät für mehrere Dienste nutzbar, ohne dass sich dabei eine Verknüpfung zurückführen lässt [17] [7].
- es kann lediglich die Session kompromittiert werden und nicht die Zugangsdaten selbst [12].

- Authentifizierungsgeräte lassen sich mit zusätzlichen PINs oder biometrischen Merkmalen absichern, um sich ebenfalls vor Diebstahl schützen [8].
- Zugangsdaten können nicht durch systematisches erraten kompromittiert werden [8].
- Wird ein Security Key gestohlen, kann dieser nur genutzt werden, wenn ebenfalls der PIN bekannt ist [8].

In folgendem Szenario:

1. Der Nutzer besitzt ein Authentifizierungsgerät, welcher mit einem drückbaren Knopf oder ähnlichen ausgestattet ist.
2. Das Authentifizierungsgerät ist mit einem geheimen PIN oder biometrischen Merkmalen geschützt.
3. Der Nutzer autorisiert vertrauten Clients auf das Authentifizierungsgerät zuzugreifen.
4. Der Nutzer verbindet sein Authentifizierungsgerät mit mehreren Clients und nutzt diese um sich bei mehreren Webdiensten zu registrieren/anzumelden.

Dann ist versichert, dass:

1. Die Authentifizierung von dem Authentifizierungsgerät durchgeführt wurde, welcher die genutzen Zugangsdaten bei dem Webdienst registriert hat.
2. Ein autorisierter Befehl auf das Authentifizierungsgerät zugegriffen hat.
3. Dieser autorisierte Befehl von einem autorisierten Client beauftragt wurde (sollte der Nutzer den korrekten PIN eingegeben haben).

Unter der Voraussetzung, dass:

1. Das Authentifizierungsgerät nicht gestohlen wurde.

2. Der PIN des Authentifizierungsgerätes nicht kompromittiert wurde.
3. Der autorisierte Client nicht kompromittiert wurde (korrekte Ausführung von CTAP2 und Client ist nicht von böswilliger Software betroffen).

3 Umsetzung

Im Folgenden wird die Umsetzung einer Integration eines Security Keys in Kombination mit FIDO2 innerhalb der LSY beschrieben. Dabei wird der aktuelle Stand berücksichtigt und begründet ein passender Ansatz für die Integration gewählt. Zusätzlich wird die Umsetzung zum Aspekt der Benutzerfreundlichkeit analysiert.

3.1 Aktueller Stand der LSY

Innerhalb der LSY werden aktuell verschiedene Verfahren zur Authentifizierung genutzt. Grundsätzlich besteht eine zentrale Nutzerverwaltung innerhalb der LSY, welche innerhalb einer Microsoft Azure AD verwaltet wird. Wie diese an eine Applikation gebunden ist, ist dabei nicht vorgegeben und kann von den jeweiligen ABteilungen individuell festgelegt werden. Auch Lösungsansätze ohne die Nutzung der Azure AD sind je nach Abteilung möglich. Die einzige zentrale Schnittstelle ist somit die genannte Azure AD.

Diese bestehende Nutzerverwaltung basiert auf einer passwortbasierten Authentifizierung inklusive MFA. Das Passwort muss von allen Nutzern in regelmäßigen Abständen (drei Monate) geändert werden. Der weitere Faktor für die Nutzung der MFA kann frei von den Nutzern gewählt werden, solange er mit der Azure AD kompatibel ist. Zusätzlich zur regelmäßigen Passwortänderungen, gibt es Richtlinien zur Passworterstellung, welche von den Nutzern eingehalten werden müssen.

Grundsätzlich ist eine Integration eines Security Keys als SFA mit Hilfe einer Azure AD möglich. Diese Funktion ist innerhalb der LSY allerdings nicht aktiviert, da diese Art der Authentifizierung nicht den Richtlinien des Unternehmens

entspricht. Eine Änderung dieser Richtlinien ist möglich, erfordert allerdings einen sehr hohen Aufwand und viel Zeit. Lediglich die Verwendung eines Security Keys als zweiten Faktor für eine MFA wird von der LSY unterstützt und ist auch mit der Azure AD kompatibel (siehe **3.1**) Da der Bearbeitungsraum dieser Arbeit zeitlich limitiert ist, wird aus den genannten Gründen keine LSY weite Integration eines Security Keys betrachtet. Dennoch soll eine Integration in einem kleinerem Rahmen auf Abteilungsebene getestet werden. Dies soll eine Aussage ermöglichen, ob eine Umstellung der aktuellen Richtlinien sinnvoll ist und eine Integration eines Security Keys in der gesamten LSY möglich ist.

Abbildung 3.1: Security Key als zweiter Faktor in der Azure AD

Für diese Testphase wird die Abteilung cGroup Solutions ausgewählt. Diese ist zuständig für das Produkt cFront, welches in **2.1** beschrieben ist. Innerhalb der Abteilung wird aktuell eine passwortbasierte Authentifizierung gegen die Azure AD inklusive MFA genutzt. Vorteilhaft für die Integration eines Security Keys ist, dass die Abteilung zusätzlich Keycloak für die Identitäts- und Zugriffsverwaltung nutzt. Keycloak ist eine Open-Source-Plattform für Identitäts- und Zugriffsmanagement, die Single Sign-On, Benutzerverwaltung und Sicherheitsfunktionen bietet, um die Authentifizierung und Autorisierung in Anwendungen zu erleichtern. Es wird verwendet, um Benutzer sicher anzumelden, ihre Zugriffsrechte zu verwalten und die Integration mit anderen Identitätsdiensten zu unterstützen [20]. Somit bietet sich eine neue Schnittstelle für die testweise Integration eines Security Keys an, da Keycloak zwar die Schnittstelle zur Azure AD nutzt, allerdings zusätzlich auch eine eigene Nutzerverwaltung besitzt. Dies erfordert keine Änderungen der aktuellen Richtlinien oder große technische Umstellungen innerhalb der gesamten LSY, sondern ermöglicht eine Integration auf Abteilungsebene.

Die aktuelle Umsetzung der Abteilung zur Authentifizierung ist vereinfacht in **3.2** abgebildet. Die Applikation kommuniziert nicht direkt mit dem Keycloak-Server, sondern nutzt eine selbst entwickelte Anwendung, welche sich als Keycloak-Client ausgibt. Der Nutzer gibt seine Zugangsdaten bei der Anmeldung an den Keykloak-Client weiter. Dieser wandelt die Zugangsdaten in einen validen JWT-Token um und übergibt diesen an den Keycloak-Server. Der Keycloak-Server validiert die Zugangsdaten gegen die Azure AD und authentifiziert den Nutzer, indem er einen oAuth2-Token erstellt und diesen an die Applikation zurückgibt.

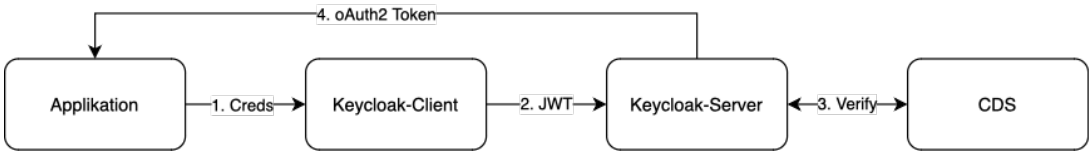

Abbildung 3.2: Aktuelle Umsetzung der Abteilung

3.2 Wahl des Security Keys

Für die Umsetzung der passwortlosen Authentifizierung innerhalb der LSY wurde ein Yubikey der Series 5 mit NFC gewählt. Dieser wurde in Kapitel vorgestellt. Hingewiesen sei an dieser Stelle, dass auch andere Hersteller Security Keys anbieten, welche das FIDO2-Protokoll unterstützen.

Dazu gehören unter anderem:

- *Feitian ePass* des Herstellers FEITIAN Technologies Co., Ltd.
- *Titan* des Herstellers Google
- *SafeNet eToken* des Herstellers Thales Group

Die Wahl des Security Keys richtete sich allerdings unter anderem an der Kompatibilitätsliste [21] von Microsoft. Diese listet alle Security Keys auf, welche

für eine passwortlose Authentifizierung gegen eine Microsoft Azure AD genutzt werden können. Nicht auffindbar in der Liste ist beispielsweise der Google Titan. Dieser unterstützt aktuell nicht FIDO2, sondern lediglich FIDO und U2F. Microsoft ist allerdings nicht abwärtskompatibel, was bedeutet, dass der Google Titan nicht für eine passwortlose Authentifizierung gegen das Azure AD genutzt werden kann .

Die endgültige Auswahl basiert auf dem bestehenden Bestand eines Yubikeys der Series 5 mit NFC. Dieser ist mit der Azure AD nutzbar. Grundsätzlich ist allerdings auch eine Nutzung eines anderen Security Keys möglich, sofern dieser das FIDO2-Protokoll unterstützt, in der Kompatibilitätsliste von Microsoft aufgeführt ist und und offiziel von der FIDO Allianz zertifiziert wurde.

Sollte die Nutzung eines Security Keys innerhalb der LSY in Zukunft erweitert werden, so wird eine neue weitreichende Analyse notwendig. Da im Rahmen dieser Arbeit lediglich eine Testphase auf Abteilungsebene stattfindet und die grundsätzliche Funktionsweise der unterschiedlichen Security Keys ähnlich ist, wird auf eine detaillierte Analyse der unterschiedlichen Security Keys verzichtet.

3.3 Integration eines Yubikeys in die LSY

Grundsätzlich bieten sich zwei Möglichkeiten für die Integration eines Security Keys in die Abteilung cGroup Solutions an. Die erste Option wurde bereits **3.1** beschrieben. Hierbei würde eine Authentifizierung der Nutzer ohne Umwege gegen die Azure AD stattfinden, wie in **3.3** vereinfacht dargestellt. Diese Option ist nativ mit der Azure AD kompatibel und erfordert technisch lediglich eine Anpassung der aktuellen Konfiguration. Dies würde zu einer LSY weiten Integration führen, da die Azure AD als zentrale Nutzerverwaltung genutzt wird.

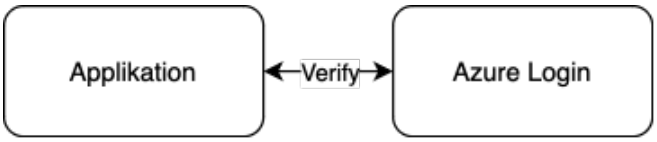

Abbildung 3.3: Umsetzungsmöglichkeit mit Azure AD

Die zweite Option würde Gebrauch von der Schnittstelle des Keycloak-Servers innerhalb der Abteilung cGroup Solutions zu machen. Diese Option würde somit nur die Abteilung betreffen und erfordert eine technische Veränderung der aktuellen Konfiguration des Keycloak-Servers. Hierbei würden nur bestimmten Nutzern die Möglichkeit gegeben werden, sich mit Hilfe eines Security Keys anzumelden, da der Security Key lediglich vom Keycloak-Server dem Nutzer zugeordnet wird. Also hat diese Option keine Auswirkungen auf die zentrale Nutzerverwaltung.

Da es sich wie in **3.1** beschrieben lediglich um eine Testphase handelt, welche nur die Abteilung cGroup Solutions betrifft, wird die zweite Option gewählt. Dies liegt insbesondere an dem hohen organisatorischen Aufwand die aktuellen Richtlinien der LSY verändern zu lassen und die technische Umstellung der Azure AD zu beantragen. Da die zentrale Nutzerverwaltung ein wichtiger Bestandteil aller Applikationen ist, wäre zudem ein zu hohes Risiko vorhanden, sollten technische Probleme auftreten.

Um die genannte Option mit Hilfe von Keycloak umzusetzen, müsste der aktuelle Prozess aus **3.2** wie folgt angepasst werden:

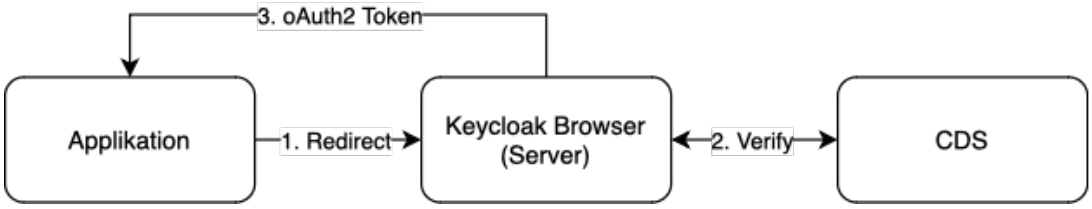

Abbildung 3.4: Veränderter Keycloak-Login

Statt die Anmeldung eines Client zu simulieren, lässt sich mit Keycloak eine

Anmeldung über eine Nutzeroberfläche realisieren. Diese wird dabei selbst von Keycloak gestellt. Dafür wird ein redirect auf die Keycloak Login-Seite durchgeführt. Bei einer erfolgreichen Verifizierung wird der Nutzer zurück auf die Applikation geleitet und vom Keycloak-Server mit Hilfe eines oAuth2-Tokens authentifiziert.

Diese Umstellung muss in Keycloak selbst konfiguriert werden. Dafür muss der sog. *Authentication Flow* modifiziert werden. Statt einer Client-Anmeldung erfolgt eine Anmeldung via Browser. Für die Testphase werden zwei Optionen in den Authentication Flow integriert, welcher in **3.5** dargestellt sind. Der obere Pfad wird für die passwortlose Authentifizierung genutzt. Dabei gibt der Nutzer zunächst seinen Username ein und authentifiziert sich anschließend mit Hilfe eines Security Keys und WebAuthn (inklusive CTAP2.1). Der Username wird dabei benötigt, um Nutzern die Möglichkeit zu geben sich mit einem Security Key für mehrere Zugänge zu authentifizieren. Darf ein Nutzer lediglich einen Zugang zur Anwendung besitzen, so kann die Eingabe des Username grundsätzlich entfallen. Der untere Pfad wird für die typische passwortbasierte Authentifizierung genutzt. Diese wird nur integriert da es sich um eine Testphase handelt und dient als Absicherung im Falle von technischen Problemen. Grundsätzlich ist eine Authentifizierung entsprechend des oberen Pfades ausreichend.

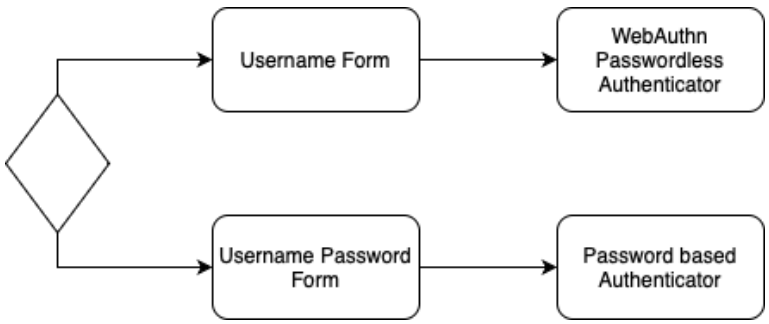

Abbildung 3.5: Authentication Flow

Nach der einer erfolgreichen Konfiguration des Authentication Flows, entstehen zwei neue Prozesse für die Anmeldung und für die Registrierung eines Nutzers.

Diese werden in den folgenden Grafiken dargestellt:

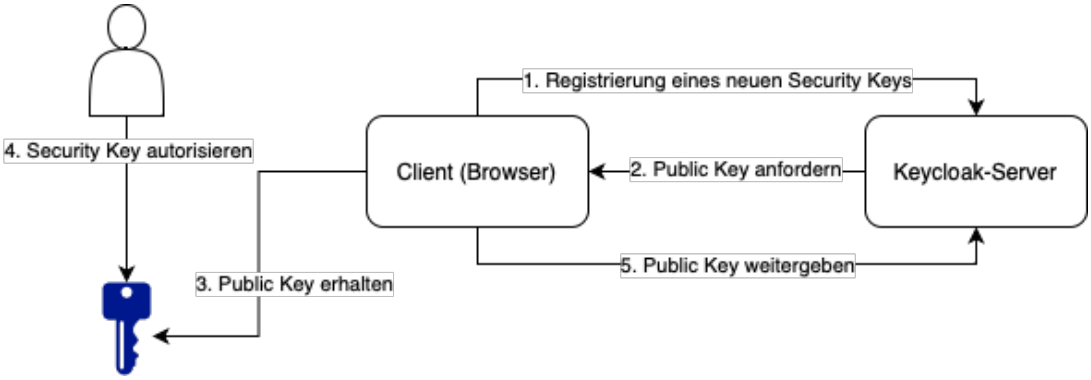

Abbildung 3.6: Registrierung (vereinfacht)

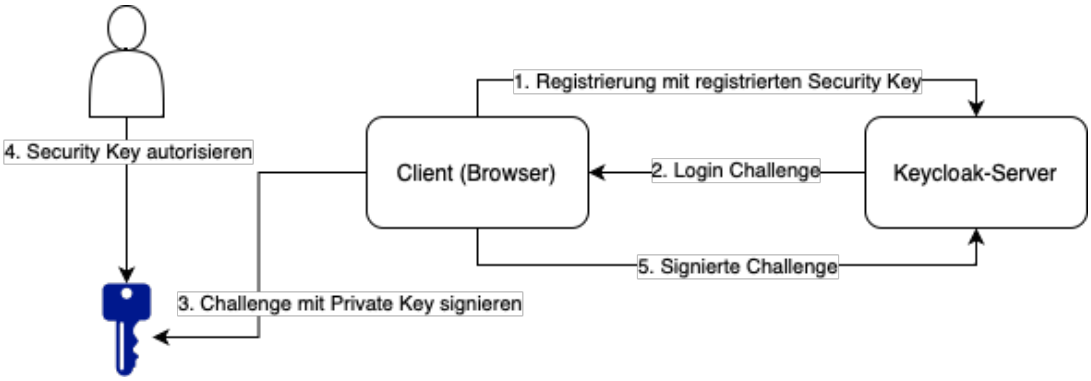

Abbildung 3.7: Anmeldung (vereinfacht)

Die Grafiken stellen den vereinfachten Ablauf der Registrierung und Anmeldung mit Hilfe eines Security Keys dar. Eine detailierte technische Erläuterung und Darstellung der Funktionsweise ist **2.7**zu finden. Der entscheidende Unterschied der beiden Prozesse ist allerdings, dass bei der Registrierung lediglich der öffentliche Schlüssel übergeben wird, während bei der Anmeldung der private Schlüssel benötigt wird. Dieser wird allerdings nicht übergeben, sondern signiert eine Login Challenge, welche vom Keycloak-Server generiert wird. Kann der Keycloak-Server die Signatur mit Hilfe des gespeicherten öffentlichen Schlüssels

verifizieren, wird der Nutzer authentifiziert. Sowohl die Registrierung als auch die Anmeldung erfolgen hierbei also nicht über die Anwendung selbst, sondern über den Keycloak-Server und dessen Nutzeroberfläche.

3.4 User Feedback

Um eine Aussage über die Akzeptanz und die Benutzerfreundlichkeit der aufgezeigten Umsetzung zu treffen, wird ein Feedback von den Nutzern der Abteilung cGroup Solutions eingeholt. Um eine wissenschaftliche Aussage zu treffen wird ein Fragebogen erstellt. Es handelt sich dabei um eine Mischform aus einer qualitativen und einer quantitativen Befragung. So wird es ermöglicht eine numerische Auswertung der Antworten zu erhalten, sowie eine qualitative Auswertung der Kommentare. Im Folgenden wird die Durchführung des Fragebogens beschrieben.

3.4.1 Rahmen des Feedbacks

Da zum Zeitpunkt der Erstellung dieser Arbeit die Nutzung keine Umsetzung einer passwortlosen Authentifizierung innerhalb der gesamten LSY möglich ist (siehe Kapitel) wird das Feedback auf die Abteilung cGroup Solutions beschränkt. Diese ist zuständig für das in Kapitel beschriebene Produkt cFront, in welchem die passwortlose Authentifizierung testweise implementiert wurde. Die Abteilung besteht aus 15 Personen.

Über einem Zeitraum von zwei Wochen werden alle Mitglieder eingeladen an der Befragung teilzunehmen. Eine Teilnahme ist freiwillig. Die Befragung findet im Büro der Abteilung statt und wird von dem Autor dieser Arbeit durchgeführt. Jeder Teilnehmer wird einzeln und vor Ort befragt. Dies ermöglicht es mit jedem Teilnehmer eine Live-Demonstration durchzuführen. So wird ebenfalls ermöglicht, dass Teilnehmer bereits während der Befragung und der Demonstration

Kommentare hinterlassen können. Diese werden auf dem Fragebogen festgehalten und werden für die qualitative Auswertung genutzt werden.

Während der gesamten Demonstration und Befragung werden den Teilnehmern keine Informationen zum Fido2 Protkoll vermittelt, da sonst die Aussagekraft des Feedbacks verfälscht werden könnte. Ziel ist es den ersten Eindruck aller Teilnehmer zu erhalten, ohne dass diese eine erzwungene Einführung in die Thematik erhalten. Die Live-Demonstration beinhaltet die Registrierung und Anmeldung mit Hilfe eines Security Keys, sowie eine Demonstration einer möglichen Anmeldung mit Hilfe eines Passkeys. Zusätzlich erhalten die Teilnehmer die Möglichkeit den Security Key physich zu betrachten. Ein detailierter Verlauf der Demonstartion wird im weiteren Verlauf der Arbeit beschrieben.

3.4.2 Auswahl der Teilnehmer

Zur Durchführung des Fragebogens wurden alle Mitglieder des Teams eingeladen, eine Teilnahme war jedoch freiwillig. Zwei der Mitglieder der Abteilung konnten auf Grund eines Urlaubs nicht an der Befragung teilnehmen. Vor der Durchführung wurden alle Teilnehmer darüber informiert zu welchem Zweck die Daten für diese Arbeit erhoben werden. Die Befragung stand dabei nicht anonym statt, um einen Austausch zwischen dem Autor und den Teilnehmern zu ermöglichen. Da die Befragung die Abteilung der Teilnehmer betrifft sollten diese somit eine Möglichkeit bekommen, ihre Gedanken zu dem modifiziertem Anmeldevorgang zu teilen.

14 Mitglieder der Abteilung stimmten der Teilnahme an der Befragung zu. Die letzte Person befand sich während des möglichen Zeitraumes der Befragung im Urlaub. Das durchschnittsalter der Teilnehmer beträgt 45,4 Jahre. Die genaue Verteilung wird in der folgenden Grafik sichtbar:

Dabei ist auffällig, dass die Teilnehmer der Befragung überwiegend der Gruppe 50-60 Jahre zugehörig sind. Auch die Gruppe 20-30 Jahre ist häufig vertreten.

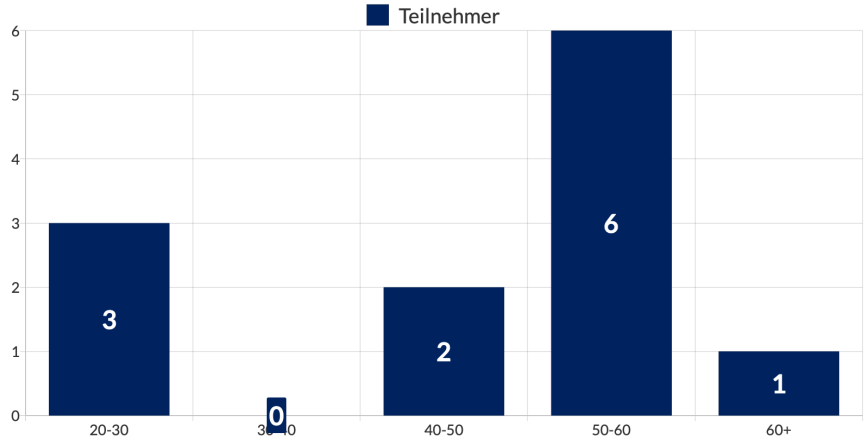

Abbildung 3.8: Alter der Teilnehmer

Lediglich die Gruppe 40-50 Jahre ist wenig vertreten. Daraus lässt sich schließen, dass die Teilnehmer der Befragung überwiegend entweder neu in das Berufsfeld eingestiegen sind oder bereits eine langjährige Erfahrung in diesem Bereich haben.

Vor Beginn der Befragung wurden die Teilnehmer gebeten anzugeben, welche Rolle sie innerhalb des Teams einnehmen. Daraus lassen sich zwei Gruppen bilden: Development und Operations. Die Verteilung der Teilnehmer auf die beiden Gruppen ist in der folgenden Grafik dargestellt:

3.4.3 Inhalt der Demonstration

Allen Teilnehmern wurde vor der Befragung eine Live-Demonstration der Registrierung und Anmeldung mit Hilfe eines Security Keys gezeigt. Der Security Key wurde zu Beginn der Demonstration in einen üblichen USB-Slot eines Firmenlaptops eingesteckt und nach der Demonstration an die Teilnehmer übergeben.

Die Anmeldung/Registrierung ist in mehrere Schritte unterteilt. Zunächst bestätigt der Nutzer, dass er sich mit Hilfe eines Security Keys anmelden/registrieren

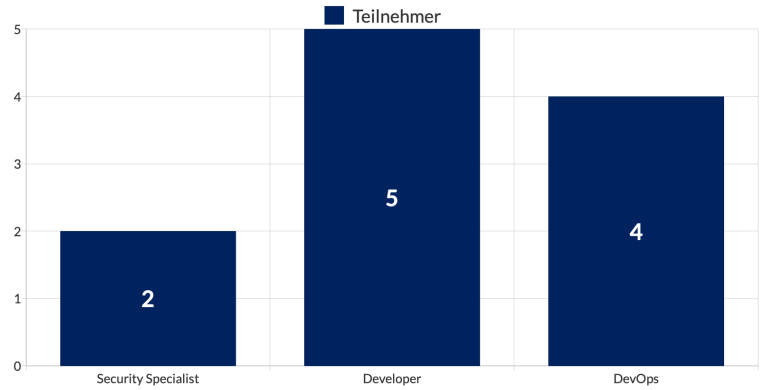

Abbildung 3.9: Alter der Teilnehmer

möchte:

Abbildung 3.10: Veränderter Keycloak-Login

Darauf folgt ein Dialogfeld des Browsers, welcher den Nutzer dazu auffordert zu bestätigen, dass der Security Key registriert wird. Dieser Schritt ist einmalig und findet nur bei der Registrierung statt. Ist der Security Key bereits registriert, wird dieser Schritt übersprungen:

Nach der Bestätigung des Dialogs muss der Nutzer den PIN des Security Keys eingeben:

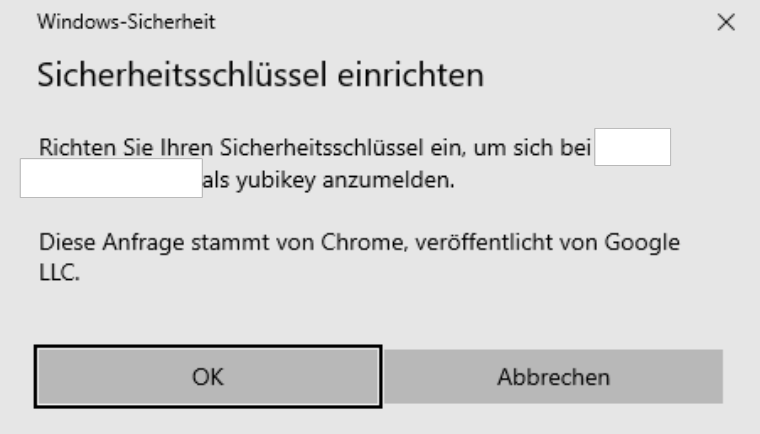

Abbildung 3.11: Veränderter Keycloak-Login

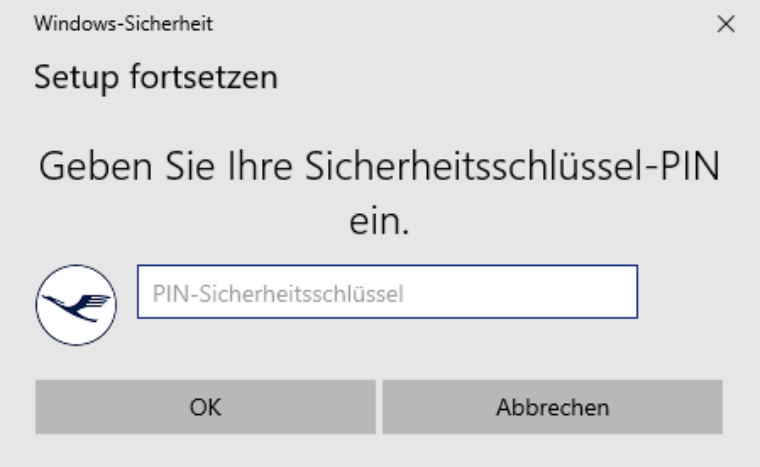

Abbildung 3.12: Veränderter Keycloak-Login

Ist die richtige PIN eingegeben wurden, erscheint ein letztes Fenster, welches den Nutzer dazu auffordert den Knopf des Security Keys zu drücken. Erst danach ist der Browser dazu autorisert sich mit Hilfe des Security Keys gegen den Keycloak-Server zu registrieren oder anzumelden:

Sobald der Knopfdruck erfolgt, wird der Nutzer erfolgreich eingeloggt. Diese Informationen wurden den Teilnehmern ebenfalls während der Durchführung des Fragebogens mitgeteilt.

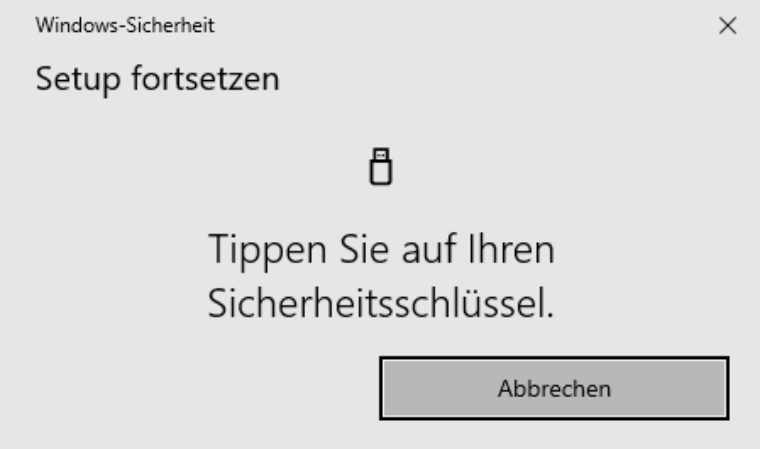

Abbildung 3.13: Veränderter Keycloak-Login

Nachdem die Registrierung und Anmeldung mit Hilfe eines Security Keys demonstriert wurde, wurde den Teilnehmern ebenfalls eine mögliche Anmeldung mit Hilfe eines Passkeys gezeigt. Der Prozess beginnt ebenfalls bei Abbildung xy und hat lediglich einen Folgeschritt:

Für die Demonstration wurde hierbei ein privates Gerät genutzt (Apple Macbook Air M1), welches mit einem Touch ID Scanner ausgestattet ist.

3.4.4 Herleitung der Fragen

Aufgrund des Ziels der Befragung, eine Aussage über die Akzeptanz und die Benutzerfreundlichkeit einer passwortlosen Authentifizierung zu treffen, werden lediglich Fragen gestellt, die sich auf diese beiden Punkte beziehen. Um eine hohe Teilnahme zu gewährleisten, werden die Fragen möglichst kurz gehalten und nur wenige Fragen gestellt. Die Fragen werden so gestaltet, dass sie dem Teilnehmer die Möglichkeit bietet Kommentare zu hinterlassen oder seine Antwort zu begründen. Die Fragen werden so gestellt, dass sie einfach zu verstehen sind und kein Vorkenntnisse im Bereich der passwortlosen Authentifizierung voraussetzen. Im folgenden werden die Fragen begründet aufgelistet und erläutert:

Abbildung 3.14: Veränderter Keycloak-Login

Frage 1:

Hast du schonmal einen Security Key genutzt?

Antwörtmöglichkeiten: Ja; Nein;

Diese Frage leitet sich aus [7] ab. Die Antwortmöglichkeiten werden im Vergleich aber angepasst und reduziert. Durch die Reduzierung auf zwei Antwortmöglichkeiten wird eine bessere Auswertung ermöglicht. Antworten Teilnehmer mit *Ja*, werden sie gefragt in welchem Kontext sie den Security Key genutzt haben. So lassen sich zusätzliche Informationen über die Nutzungsdauer und den Zweck der Nutzung zu erhalten.

Frage 2:

Bist du generell bereit deine Passwörter durch eine andere Art der Authentifizierung zu ersetzen?

Antwörtmöglichkeiten: Ja; Nein;

Diese Frage ergibt sich aus einer Umfrage von Statista, in welcher Teilnehmer gefragt wurden, durch welche Art der Authentifizierung sie das Passwort ersetzen würden. Lediglich 22% der Teilnehmer gaben an, dass sie ihr Passwort lieber beibehalten würden [22]. Daraus folgt die Annahme, dass eine Vielzahl an Nutzern grundsätzlich dazu bereit wären ihr Passwort zu ersetzen. Die Frage soll eine bessere Analyse der folgenden Fragen ermöglichen und zielt auf die Akzeptanz einer passwortlosen Authentifizierung im generellen ab.

Frage 3:

Benutzt du auf der Arbeit aktuell einen Passwort Manager?

Antwörtmöglichkeiten: Ja; Nein;

Verwandte Studien zeigen, dass Nutzer eines Passwort Managers teilweise eine geringere Anmeldezeit auf Grund eines Passwort Managers aufweisen (insbesondere bei einer Nutzung von autofill) [7]. Die Frage soll einen Zusammenhang zwischen der Nutzung eines Passwort Managers und der Einschätzung der Benutzerfreundlichkeit einer passwortlosen Authentifizierung ermöglichen.

Frage 4:

Kennst du das FIDO2-Protokoll und weißt du grob wie es funktioniert?

Antwörtmöglichkeiten: Ja; Nein;

Diese Frage bezieht sich auf die in Kapitel xy aufgeführte Problematik, dass Nutzer lieber Passwörter nutzen, da sie die Funktionsweise und Technologie im Hintergrund besser verstehen. Dieser mögliche Zusammenhang soll betrachtet

werden. Antworten Teilnehmer mit *Ja*, werden sie gefragt, ob sie die Funktionsweise des FIDO2-Protokolls erklären können. So lässt sich eine Aussage über die Kenntnisse der Teilnehmer treffen.

Frage 5:

> *Wie bewertest du die Benutzerfreundlichkeit der Registrierung mit Hilfe eines Security Keys?*

Antwörtmöglichkeiten: Besser als mit einem Passwort; Gleich gut wie mit einem Passwort; Schlechter als mit einem Passwort;

Diese Frage soll einen Vergleich zwischen der Benutzerfreundlichkeit einer passwortlosen Authentifizierung und einer passwortbasierten Authentifizierung ermöglichen. Aus diesem Grund wurden die Antwortmöglichkeiten bewusst so gewählt, dass sie einen Vergleich ermöglichen. Eine generelle Bewertung würde die Auswertung erschweren, da die Teilnehmer unterschiedliche Vergleichswerte wählen könnten.

Frage 6:

> *Wie bewertest du die Benutzerfreundlichkeit der Anmeldung mit Hilfe eines Security Keys?*

Antwörtmöglichkeiten: Besser als mit einem Passwort und MFA; Gleich gut wie mit einem Passwort und MFA; Schlechter als mit einem Passwort und MFA;

Wie auch Frage fünf zielt diese Frage auf die Benutzerfreundlichkeit ab. Die Unterteilung in zwei Fragen ergibt sich vor allem aus der Tatsache, dass sich die Registrierung und die Anmeldung, insbesondere bei einer passwortbasierten Authentifizierung, deutlich unterscheiden. Während es sich bei einer Anmeldung lediglich um eine Wissensabfrage handelt, muss bei der Registrierung zunächst

ein eigenes Passwort erstellt werden. Dies könnte dazu führen, dass die beiden Abläufe unterschiedlich bewertet werden und somit der Vergleich zur passwortlosen Authentifizierung erschwert wird.

Frage 7:

> *Wärst du dazu bereit einen Security Key für den privaten Gebrauch zu kaufen, wenn der Preis bei ca. 50€ liegt?*

Antwörtmöglichkeiten: Ja; Nein;

Diese Frage zielt auf die Akzeptanz einer passwortlosen Authentifizierung im privaten Kontext ab und basiert auf dem Ergebnis aus Kapitel xy. Dort wurde festegestellt, dass der Kaufpreis eines Security Keys ebenfalls eine Hürde für die Nutzung darstellen kann. Als Richtwert für den Kaufpreis wird hierbei der ungefähre Preis eines Yubikeys der Series 5 mit NFC gewählt, da dieser ebenfalls für die Umsetzung genutzt wird. Die Frage soll im Weiteren auch auf für die Nutzung im Unternehmenskontext genutzt werden, da die Akzeptanz im Generellen auch eine Auswirkung auf die Etablierung von Security Keys hat. Eine erhöhte Etablierung kann ebenfalls zu einer breiteren Unterstüzung führen.

Frage 8:

> *Hältst du einen Security Key für sicherer als ein Passwort?*

Antwörtmöglichkeiten: Ja; Nein;

Diese Frage basiert auf der in Kapitel xy beschriebenen Annahme, dass Nutzer an der Sicherheit von Security Keys zweifeln, da sie die Funktionsweise der Technologie nicht verstehen. Dies soll im Zusammenhang mit Frage 4 betrachtet werden. Bewusst wird dabei auf die Antwortmöglichkeit *Ich weiß es nicht*

verzichtet, da Teilnehmer auf der Basis ihres aktuellen Wissensstands eine intuitive Entscheidung treffen sollen. Dies ermöglicht ebenfalls eine Aussage über die Akzeptanz der Teilnehmer.

Frage 9:

> *Findest du eine Anmeldung per Passkey besser als eine Anmeldung per Security Key?*

Antwörtmöglichkeiten: Ja; Nein; Gleich;

Diese Frage soll für einen Ausblick genutzt werden, ob eine Anmeldung per Passkey eine Alternative zu einer Anmeldung per Security Key darstellt. In **2.9** wurde ebenfalls deutlich, dass die Benutzerfreundlichkeit weniger von FIDO2 abhängig ist, sondern viel mehr vom genutzten authentifizerungsgerät. Mit Hilfe von kommentaren der Teilnehmer sollen konkrete Vor- und Nachteile der beiden Verfahren in Bezug auf deren Benutzerfreundlichkleit ermittelt werden.

3.4.5 Auswertung

Die Auswertung der Fragebögen bestätigt in vielen Teilen die bereits erarbeiteten Annahmen aus Kapitel xy. Lediglich zwei der Teilnehmer geben an, dass sie bereits einen Security Key genutzt haben. Dies allerdings nur testweise und nicht im alltäglichen Gebrauch. Die restlichen Teilnehmer geben an, dass sie noch keinen Security Key genutzt haben bzw. lediglich einen gesehen haben. Dies bestätigt, dass die Nutzung von Security Keys aktuell noch nicht weit verbreitet ist und Passwörter weiterhin die dominierende Methode der Authentifizierung darstellen.

Alle Teilnehmer geben an, dass sie dazu bereit wären ihr Passwort durch eine andere Art der Authentifizierung zu ersetzen. Dies übertrifft das Ergebnis aus [22]. Dies kann daran liegen, dass alle Teilnehmer in einem sehr technischen Kontext

arbeiten und somit eine höhere Akzeptanz für neue Technologien aufweisen und ein höheres Bewusstsein für Sicherheit innerhalb der Informatik aufweisen.

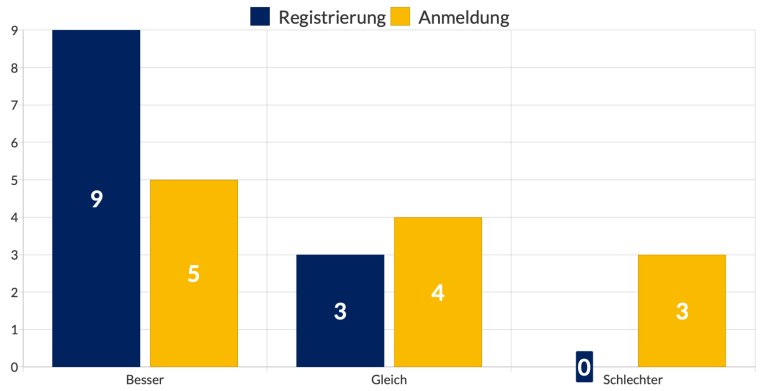

Abbildung 3.15: Veränderter Keycloak-Login

Bei der Bewertung der Registrierung und der Anmeldung mit Hilfe eine Security Keys sind deutliche Unterschiede sichtbar. Die deutliche Mehrheit der Teilnehmer bevorzugte die Registrierung per Security Key gegenüber der passwortbasierten Alternative. Keiner der Teilnehmer fand die Registrierung im Vergleich schlechter. Bei der Anmeldung hingegen ist ein ausgeglicheneres Ergebnis sichtbar. Im Vergleich geben drei der Teilnehmer an, dass sie Anmeldung schlechter finden als die aktuelle Alternative mit Hilfe eines Passwortes und MFA. Es wird also deutlich, dass die beiden Abläufe der Registrierung und der Anmeldung differenziert betrachtet werden müssen. Eine Abhängigheit zwischen der Nutzung eines Passwort Managers und der Bewertung der Benutzerfreundlichkeit lässt sich nicht feststellen, da lediglich drei Teilnehmer keinen Passwort Manager nutzen und diese sehr verschiedene Bewertungen abgeben. Eine aussagekräftige Auswertung ist somit nicht möglich.

Zehn der Teilnehmen geben an, dass sie nicht bereit wären 50€ für einen Security Key auszugeben. Dies bestätigt die Annahme aus Kapitel xy, dass der Preis eine Hürde für die Nutzung und Etablierung darstellen kann. Eine vermehrte Nutzung

im privaten Kontext würde zu mehr Akzeptanz führen, da die Teilnehmer bereits mit der Technologie vertraut sind.

Bis auf zwei Teilnehmer wird die Nutzung von Security Keys sicherer eingeschätzt als die Nutzung von Passwörtern. Da lediglich zwei Nutzer die Nutzung als weniger sicher betrachten lässt sich keine Aussage über einen Zusammenhang zwischen der Kenntnis des FIDO2-Protokolls und der Einschätzung der Sicherheit treffen.

Zwölf der Teilnehmer geben an eine Anmeldung per Passkey besser zu finden als eine Anmeldung per Security Key.

Aus den Kommentaren lassen sich ebenfalls einige Erkenntnisse ziehen:

- Ein Teilnehmer stufte die Registrierung und ANmeldung als „*sehr aufregend*“, da es etwas neues ist und begründete so seine positive Bewertung. Dieses Beispiel zeigt auf, dass die Gewöhnung an eine neue Technologie nicht zwangsweise negativ ist, sondern auch positiv bewertet werden kann.
- Mehrere Teilnehmer kritisierten die Notwendigkeit den Security Key immer dabei haben zu müssen und spontane Logins nicht möglich sind. Dies deckt sich mit den Ergebnissen aus Kapitel xy.
- Daraus resultiere auch der Kritikpunkt, dass zusätzliche Hardware verloren gehen kann. Dies ist ein weiterer Kritikpunkt, welcher bereits in Kapitel xy aufgeführt wurde.
- Mehrere Teilnehmer wiesen darauf hin, dass sie ihr Handy und somit ihre Authenticator App im dabei haben. Selbst, wenn sie den Prozess des anmeldens mit Hilfe eines Security Keys als besser bewerten, würden sie weiterhin die Nutzung eines Passwortes mit MFA bevorzugen.
- Ein neuer Punkt der in den Kommentaren aufgeführt wurde ist, dass Single Sign-On (SSO) ein wichtiger Faktor ist. Da somit eine geringere Abfrage der Passwörter gegeben ist und auch weniger Passwörter erstellt werden

müssen. Dies wirkt sich auch auf die EInschätzung der Benutzerfreundlichkeit aus.

- Die Mehrheit der Teilnehmer war der Meinung, dass sie die Benutzerfreundlichkeit der Security Keys erst nach einer längeren Testphase bewerten können.
- Ein Kritikpunkt der Teilnehmer am Anmeldevorgang mit Hilfe eines Security Keys ist, dass die EIngabe der PIN und des Benutzernamens zu viel sind. Dadurch bewerteten sie die Alternative als gleich oder schlechter als die aktuelle Lösung. Sie wünschen sich eine Lösung, bei dem der Security Key lediglich eingesteckt und gedrückt werden muss.
- Ein Teilnehmer begründete seine Antwort bezogen auf die Passkeys damit, dass er zusätzliche Hardware für sicherer hält und sich bei der Nutzung von Passkeys nicht sicher ist, ob diese wirklich nur auf dem Gerät gespeichert werden.
- Einige Teilnehmer erläuterten, dass sie die Anmeldung per Security Key benutzerfreundlicher als ein Passwort mit MFA finden, allerdings nicht besser als lediglich einem Passwort.
- Ebenfalls wurde die mechanische Belastung des Security Keys und des USB-Ports genannt. Beide könnten durch die Nutzung beschädigt werden. Auch dieser Punkt wurde bereits in Kapitel xy aufgeführt.
- Auch der Preis wurde häufig als Kritikpunkt genannt. Ergänzend erwähnte allerdings ein Teilnehmer, dass er den Preis bezahlen würde, wenn es weiter etabliert ist. Ein anderer Teilnehmer erwähnte, dass er zum aktuellen Zeitpunkt maximal 20€ für einen Security Key ausgeben würde.
- Zur reinen Benutzerfreundlichkeit merkte ein Teilnehmer an, dass ein schlechtes und einfach gewähltes Passwort deutlich benutzerfreundlicher sei, wenn man den Faktor der Sicherheit nicht betrachtet.

3.4.6 Fazit & Reflexion

Aus der Auswertung der Kommentare lässt sich ebenfalls eine Reflexion des Fragebogens ableiten. Ein entscheidenes Feedback ist dabei, die Testphase zu verlängern. Da der Bearbeitungszeitraum dieser Arbeit begrenzt ist und zunächst eine Implementierung erfolgen musste war dies nicht möglich. Aus diesem Grund wurde lediglich dieser Fragebogen erstellt, um einen ersten Eindruck zu erhalten. Für eine weitere Ausarbeitung sollte allerdings eine längere Testphase eingeplant werden.

3.5 Wirtschaftlichkeit

Ein entscheidender Faktor für den Einsatz einer passwortlosen Authentifizierung mit Hilfe eines Security Keys ist die Wirtschaftlichkeit. Im Folgenden soll eine Analyse der Wirtschaftlichkeit im Bezug auf die Abteilung cGroup Solutions durchgeführt werden.

Betrachtet man die Teamgröße von 15 Personen und geht von den aktuellen Kosten eines Yubikeys aus (50€) so ergibt sich ein Gesamtpreis von 750€. Dieser Preis ist einmalig und muss nicht wiederholt werden. Auf Grund der geringen Backup-Möglichkeiten eines Security Keys sollte allerdings ein Backup-Key pro Nutzer angeschafft werden. Dieser kann im Falle eines Verlustes des ersten Keys genutzt werden. Dies erhöht die Kosten auf 1500€. Lässt man alternativ weiterhin eine Anmeldung per Passwort zu, würden 750€ entfallen, allerdings wäre der eigentliche Zweck der Anschaffung nicht mehr erfüllt. Sollte ein Security Key verloren oder kaputt gehen muss dieser ebenfalls ersetzt werden. Zusätzlich zu den Materialkosten müssen die Kosten für die Implementierung betrachtet werden. Die Migration von Passwörtern auf Security Keys muss von erfahrenen Entwicklern und Architekten durchgeführt werden. Die genauen Kosten dafür lassen sich allerdings nicht definieren und sind stark abhängig von der gewünschten Implementierung.

Sollte eine Anschaffung für das gesamte Unternehmen LSY erfolgen so würde sich der Preis auf 280.000€ belaufen. Dies ergibt sich aus der aktuellen Mitarbeiterzahl von 2.800€ und den Kosten eines Security Keys von 50€, sowie einem Backup-Key pro Nutzer.

Im Gegenzug muss ein möglicher Kostenvorteil eingerechnet werden. Da Passwörter wie in Kapitel xy beschrieben eine der größten Schwachstellen darstellen, sind sie ein grundlegender Faktor für erfolgreiche Angriffe. Würde eine Nutzung der Security Keys dies verhindern oder eindämmen, so würde dies zu einer Kostenreduktion führen. Auch hier lassen sich allerdings nur schwierig konkrete Zahlen nennen. Diese sind unter anderem abhängig von der Schwere und Art des Angriffs. Einen Richtwert liefert der *Cost of a Data Breach Report 2023* von IBM [23]. Dort werden die durchschnittlichen Kosten eines Data Breach für bestimmte Regionen aufgezählt. Der Wert für die Region Deutschland liegt bei 4,67 Millionen US-Dollar [23]. Daraus lässt sich folgern, dass ein erfolgreicher Angriff auf die LSY zu einem deutlich höheren Kostenaufwand führen kann, als die Anschaffung von Security Keys.

4 Fazit & Empfehlung

Das Fazit der Arbeit lässt sich in drei Teile Aspekte gliedern. Diese leiten sich aus den Zielen der Arbeit ab, welche in **1.1** definiert wurden. Es handelt sich dabei um die Aspekte der Sicherheit, der Benutzerfreundlichkeit und der Umsetzbarkeit im Bezug auf einer passwortlosen SFA. Das gesamte Fazit dieser Arbeit bezieht sich dabei lediglich auf die passwortlose Authentifizierung mit FIDO2, da die anderen passwortlosen Verfahren nicht weiter betrachtet wurden.

Sicherheit: Grundsätzlich handelt es sich bei FIDO2 Protokoll an sich um eine deutlich sicherere Alternative zu einer passwortbasierten Authentifizierung. So übersteigt eine Nutzung von FIDO2 Zugangsdaten auch die Sicherheit einer Nutzung von Passwörtern inklusive MFA. Dies liegt insbesondere an der Nutzung von geprüfter asymmetrischer Kryptografie. Es gibt keine geteilten Geheimnisse und der private Schlüssel wird nur lokal auf dem Gerät des Nutzers gespeichert. So werden die meisten Angriffsvektoren der klassischen Passwortauthentifizierung eliminiert. FIDO2 Zugangsdaten lassen sich nicht erraten, phishen und können nicht von Datenlecks betroffen sein. Zudem gilt CTAP2.1 in Verbindung mit WebAuthn als PQ bereit. Es handelt sich also auch um eine zukunftsfähige Technologie. Dennoch besteht eine große Abhängigkeit zum genutzten Authentifizierungsgerät. Eine Nutzung von Security Keys führt zu physischen Angriffsvektoren. So kann dieser beispielsweise von Diebstahl betroffen sein. Allerdings lassen sich Security Keys meisten mit einer PIN oder einem biometrischen Merkmal absichern. Dennoch wird die Sicherheit im Vergleich zur passwortbasierten Alternative als höher eingestuft.

Benutzerfreundlichkeit: Wie auch im Aspekt der Sicherheit besteht hier eine große Abhängigkeit zum genutzten Authentifizierungsgerät. Grundsätzlich ist

FIDO2 in Form von CTAP2.1 und WebAuthn bereits weitreichend verbreitet. Es wird von den meisten Browsern nativ unterstützt und auch Anbieter wie Apple, Microsoft und Google ermöglichen die Nutzung von FIDO2. Nutzer müssen sich keine Passwörter mehr ausdenken und sich diese merken, was einen großen Vorteil darstellt. Dennoch zeigt diese Arbeit auch einige Kritikpunkte auf - insbesondere im Bezug auf die Nutzung von Security Keys. Diese stellen für viele Nutzer eine große Hürde dar. Nutzer kritisierten den Security Key immer bei sich haben zu müssen. Das physische Objekt kann zudem verloren oder kaputt gehen. In solchen Fällen besteht ebenfalls oftmals kein effektiver und sicherer Proezess für eine Wiederherstellung des Zugangs. Auch der entstehende Kostenaufwand und die Verwaltung einer Vielzahl an Security Keys kann im Unternehmenskontext eine Hürde darstellen. Die meisten Nutzer sind zusätzlich so sehr an die Nutzung von Passwörtern gewöhnt, dass es Prozess der Umgewöhnung benötigt wird, um die Akzeptanz zu erhöhen.

Umsetzbarkeit: Die Umsetzbarkeit einer Integration von Security Keys mit Hilfe von FIDO2 innerhalb der LSY stellt sich in dieser Arbeit als größte Hürde dar. Dies liegt insbesondere an der festen Verankerung der passwortbasierten Authentifizierung in das Sicherheitskonzept und die Richtlinien der LSY. Diese zu verändern benötigt einen langen und aufwändigen Prozess. Auch auf technischer Ebene kann sich eine Umstellung in einem so großem Ausmaß als schwierig darstellen. Diese ist aber grundsätzlich möglich und wird von dem Großteil der Systeme unterstützt. Dennoch bestehen auch Sonderfälle, in welchen weiterhin eine Nutzung von Passwörtern notwendig ist. Trotz der weitreichen Unterstützung von FIDO2 ist diese nicht vergleichbar mit der Etablierung von Passwörtern. Dennoch zeigt die Testphase dieser Arbeit auch, dass eine Umstellung der Authentifizierung möglich ist und diese mit einem verhältnismäßig geringen Aufwand umgesetzt werden kann. Mit der aktuellen Umsetzung innerhalb der LSY ist eine Nutzung von Security Keys lediglich für eine MFA möglich.

Aus dem Fazit der einzelnen Faktoren wird deutlich, dass FIDO2 insbesondere

im Aspekt der Sicherheit viele Vorteile bietet. Auch eine höhere Benutzerfreundlichkeit wird durch FIDO2 grundsätzlich ermöglicht. Hierbei ist allerdings die hohe Abhängigkeit zu den genutzten Authentifizierungsgeräten auffällig. Je nach Auswahl können sowohl Vorteile als auch Nachteile entstehen. In diesem Bereich sollte genauer betrachtet werden, welche Geräte sich in Zukunft etablieren können. Die Ergebnisse dieser Arbeit zeigen, dass die Nutzung von Security Keys noch nicht komplett die ermöglichten Vorteile von FIDO2 ausnutzen können. Auch wenn diese im Bereich der Benutzerfreundlichkeit bereits einige Vorteile bieten, entstehen auch einige Nachteile im Vergleich zu passwortbasierten Alternativen. Eine bessere Variante könnten Passkeys darstellen. Da es sich bei Passkeys allerdings um eine relativ neue Technologie handelt, bleibt ihre Entwicklung abzuwarten. Ein Ausblick zur Nutzung von Passkeys wird in **5** gegeben.

Daraus leitet sich folgende Empfehlung für die Integration einer passwortlosen Authentifizierung innerhalb der LSY ab:

Eine Nutzung des FIDO2 Protkolls wird grundsätzlich empfohlen, da dieses eine deutlich höhere Sicherheit bietet und eine höhere Benutzerfreundlichkeit ermöglicht. Allerdings lässt sich noch keine klare Empfehlung für die Nutzung von Security Keys aussprechen. Vielmehr wird empfohlen die Entwicklung der Authentifizierungsgeräte und ihre Verbreitung zu beobachten. Sollte ein Authentifizierungsgerät in der Lage sein die vollen Vorteile auszureizen und weitreichend verbreitet sein, wird eine Nutzung empfohlen. Aus diesem Grund sollte die LSY bereits eine Änderung der Richtlinien vornehmen, welche eine Nutzung von FIDO2 ermöglicht. So wird eine signifikant schnellere Handlungszeit ermöglicht. Passwortbasierte Alternativen können weiterhin eingesetetzt werden, könnten aber durch eine vorzeitige Änderung der Richtlininien deutlich schneller ersetzt werden. Zudem würde eine Änderung der Richtlinien eine Testphase in einem größeren Rahmen ermöglichen Dies stellt im Bereich der Umsetzbarkeit aktuell eine große Hürde dar.

5 Ausblick Passkeys

Ein Problem der Nutzung von FIDO2 mit Hilfe von Security Keys ist die Abhängigkeit an ein einzelnes Gerät. Mit Passkeys stellt FIDO Allianz eine Alternative, welche multi-device fähige Zugangsdaten ermöglicht **usecasfido**. Technisch ist diese Möglichkeit der Authentifizierung bereits in FIDO2 integriert. Es besteht allerdings eine Abhängigkeit an die Anbieter der Authentifizierungsgeräte oder Authenticator Apps. Diese müssen die Möglichkeit der Nutzung von Passkeys integrieren **usecasfido**. Dies ist zum aktuellen Zeitpunkt noch nicht weitreichend gegeben.

Die FIDO2-Zugangsdaten werden bei Passkeys nicht auf einem externen Authentifizierungsgerät gespeichert, sondern werden lokal auf dem Gerät selbst gespeichert. Dies führt zu einer deutlich verbesserten Benutzfreundlichkeit im Vergleich zur Nutzung von Security Keys, da keine zusätzliche externe Hardware benötigt wird. Zudem können Passkeys auf mehreren Geräten genutzt werden. So können diese beispielsweise auf einem Smartphone und einem Laptop genutzt werden. Dabei werden die Passkeys zwischen den Geräten synchronisiert (siehe **5.1**). Dies ermöglicht eine Nutzung von Passkeys auf mehreren Geräten. Dadurch besteht allerdings auch eine Abhängigkeit an die Anbieter der Synchronisierungsdienste sowie an Hersteller der Geräte. Diese müssen für die Sicherheit des Betriebssystems, der Hardware und der Synchronisierung sorgen **usecasfido**.

Die Nutzung von Passkeys ist vergleichbar mit der Nutzung von Passwortmanagern, welche ein automatisches Ausfüllen der Zugangsdaten ermöglichen. Dabei bietet die Nutzung von Passkeys allerdings zusätzlich eine erhöhte Sicherheit, selbst ohne eine Nutzung von MFA **usecasfido passkeysgoogle**. Für die benötigte Autorisierung des CTAP2.1 Standards wird typischerweise ein biometrisches Merkmal des Nutzers genutzt **usecasfido**. Zusätzlich ermöglicht der FIDO2 Standard die Nutzung von Bluetooth. So wird dem Nutzer die Möglichkeit

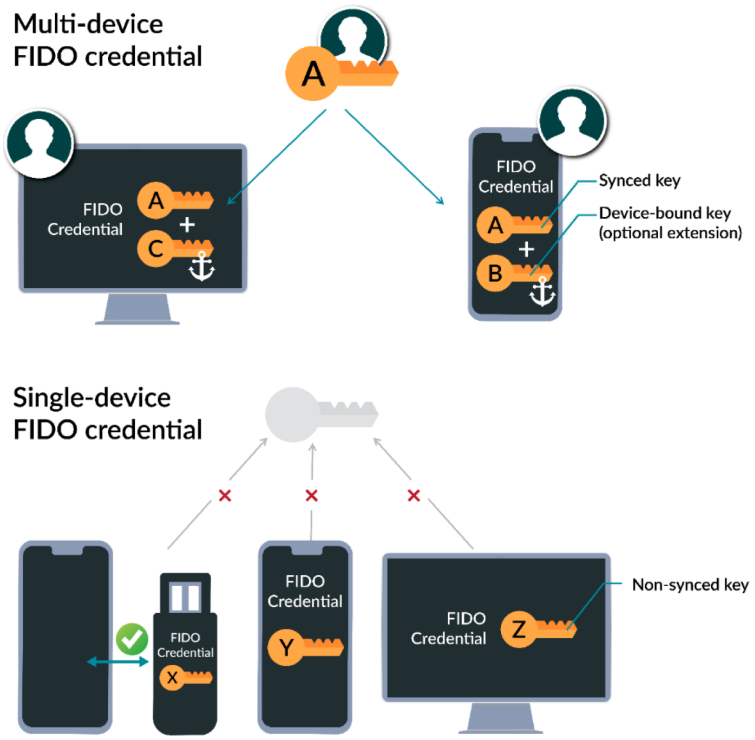

Abbildung 5.1: Multi-device FIDO und Single-device FIDO **usecasfido**

gegeben Passkeys beispielsweise von mobilen Geräten abzurufen. Dies ist eine Absicherung, falls eine Synchronisierung von Passkeys nicht möglich ist oder Geräte die Nutzung von FIDO2 nicht unterstützen. Dies könnte beispielsweise der Fall sein, wenn Geräte unterschiedlicher Anbieter genutzt werden **usecasfido**.

Eine große Authenticator App, welche bereits eine Nutzung von Passkeys unterstützt ist der Google Password Manager. Dieser ermöglich bereits eine verschlüsselte Synchronisierung der Passkeys. Der benötigte private Schlüssel für die Verschlüsselung wird dabei lediglich auf dem Gerät selbst gespeichert und ist dort selbst zusätzlich verschlüsselt **passkeysgoogle**. Google ermöglicht es zudem die Passkeys inklusive eines gerätegebundenen Schlüssels zu nutzen. Dieser Schlüssel wird auf Android Geräten auf der Trusted Execution Environment (TEE) gespeichert und ist somit zusätzlich auf der Hardware-Ebene geschützt. Die-

ser kann das Gerät nicht verlassen und ist ebenfalls nicht durch Backups o.ä. wiederherstellbar **passkeysgoogle**. Der gerätegebundene Schlüssel sorgt für eine erhöhte Sicherheit, da eine Garantie besteht, dass dieser nicht von anderen Geräten genutzt werden kann **usecasfido**. Dies wird auch in **5.1** dargestellt. Dabei handelt es sich um eine optionale Möglichkeit, welche von den Anbietern der Authentifizierungsgeräte oder Authenticator Apps unterstützt werden muss **usecasfido**.

Auch Apple bietet eine synchronisierte Nutzung von Passkeys an. Diese werden dabei in der iCloud Keychain gespeichert. Wie auch bei Google ist dieser dabei ebenfalls lediglich in verschlüsselter Form synchronisiert. Eine Möglichkeit zur Nutzung von Passkeys inkluse eines device-bound Schlüssels wird aktuell nicht in der Dokumentation von Apple erwähnt **passkeysapple**.

Die Entwicklung von Passkeys ist aktuell noch nicht weitreichend und wird nur wenig in der Fachliteratur betrachtet. Es handelt es sich um eine vielversprechende Technologie, welche eine deutlich höhere Benutzerfreundlichkeit ermöglicht. Die Abhängigkeit an die Anbieter der Authentifizierungsgeräte und Authenticator Apps sollte in wissenschaftlichen Arbeiten allerdings noch weiter betrachtet werden.

Literaturverzeichnis

[1] S. Samonas und D. Coss, „The CIA strikes back: Redefining confidentiality, integrity and availability in security.," *Journal of Information System Security*, Jg. 10, Nr. 3, 2014.

[2] *PONS Worterbuch*. Adresse: `%5Curl%7Bhttps://de.pons.com/%7D` (besucht am 26. 07. 2023).

[3] A. Agarwal und A. Agarwal, „The security risks associated with cloud computing," *International Journal of Computer Applications in Engineering Sciences*, Jg. 1, Nr. Special Issue on, S. 257–259, 2011.

[4] S. Boonkrong, „Security of passwords," *Information Technology Journal*, Jg. 8, Nr. 2, S. 112–117, 2012.

[5] K. Chanda, „Password security: an analysis of password strengths and vulnerabilities," *International Journal of Computer Network and Information Security*, Jg. 8, Nr. 7, S. 23, 2016.

[6] M. Yıldırım und I. Mackie, „Encouraging users to improve password security and memorability," *International Journal of Information Security*, Jg. 18, S. 741–759, 2019.

[7] F. M. Farke, L. Lorenz, T. Schnitzler, P. Markert und M. Dürmuth, „{"You} still use the password after {all"}–Exploring {FIDO2} Security Keys in a Small Company," in *Sixteenth Symposium on Usable Privacy and Security (SOUPS 2020)*, 2020, S. 19–35.

[8] M. Barbosa, A. Boldyreva, S. Chen und B. Warinschi, „Provable security analysis of FIDO2," in *Advances in Cryptology–CRYPTO 2021: 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16–20, 2021, Proceedings, Part III 41*, Springer, 2021, S. 125–156.

[9] Verizon, *2017 data breach investigations report*. Adresse: `%5Curl%7Bhttps://enterprise.verizon.com/%20resources/reports/2017_dbir.pdf%7D` (besucht am 01. 08. 2023).

[10] B. Nahorny, *Email threats 2017. Symantec. Internet Security Threat Report (2017)*. (besucht am 01. 08. 2023).

[11] B. Ives, K. R. Walsh und H. Schneider, „The domino effect of password reuse," *Communications of the ACM*, Jg. 47, Nr. 4, S. 75–78, 2004.

[12] M. Morii, H. Tanioka, K. Ohira et al., „Research on integrated authentication using passwordless authentication method," in *2017 IEEE 41st annual computer software and applications conference (COMPSAC)*, IEEE, Bd. 1, 2017, S. 682–685.

[13] R. S. Chowhan und R. Tanwar, „Password-less authentication: methods for user verification and identification to login securely over remote sites," in *Machine Learning and Cognitive Science Applications in Cyber Security*, IGI global, 2019, S. 190–212.

[14] V. Parmar, H. A. Sanghvi, R. H. Patel und A. S. Pandya, „A comprehensive study on passwordless authentication," in *2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)*, IEEE, 2022, S. 1266–1275.

[15] J. Reynolds, T. Smith, K. Reese, L. Dickinson, S. Ruoti und K. Seamons, „A tale of two studies: The best and worst of yubikey usability," in *2018 IEEE Symposium on Security and Privacy (SP)*, IEEE, 2018, S. 872–888.

[16] yubico, „YubiKey 5 Series,“ 2023.

[17] S. G. Lyastani, M. Schilling, M. Neumayr, M. Backes und S. Bugiel, „Is FIDO2 the kingslayer of user authentication? A comparative usability study of FIDO2 passwordless authentication,“ in *2020 IEEE Symposium on Security and Privacy (SP)*, IEEE, 2020, S. 268–285.

[18] W3C, *About Us*. Adresse: %5Curl%7Bhttps://www.w3.org/about/%7D (besucht am 06. 08. 2023).

[19] N. Bindel, C. Cremers und M. Zhao, „FIDO2, CTAP 2.1, and WebAuthn 2: Provable security and post-quantum instantiation,“ *Cryptology ePrint Archive*, 2022.

[20] Keycloak, *Open Source Identity and Access Management*. Adresse: %5Curl%7Bhttps://www.keycloak.org%7D (besucht am 09. 08. 2023).

[21] *FIDO2-Sicherheitsschlusselanbieter*. Adresse: %5Curl%7Bhttps://learn.microsoft.com/de-de/azure/active-directory/authentication/concept-authentication-passwordless#fido2-security-key-providers%7D (besucht am 20. 07. 2023).

[22] *Durch welche Technologien würden Sie Passworter in Ihrem Alltag gerne ersetzen wollen?* Adresse: %5Curl%7Bhttps://de.statista.com/statistik/daten/studie/1313711/umfrage/alternative-technologien-zum-passwort-in-deutschland/#:~:text=Im%20Rahmen%20der%20Umfrage%20gaben,Passwort%20im%20Alltag%20gerne%20beibehalten.%7D (besucht am 20. 07. 2023).

[23] IBM, *Cost of a Data Breach Report 2023*. (besucht am 26. 07. 2023).