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
PAKE Password Authenticated Key Exchange
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
CDS Corporate Directory Service
AD Active Directory
SSO Single Sign-On
U2F Universal Second Factor
IDS Intrusion Detection System
DDoS Distributed Denial of Service

2FA Two-Factor Authentication

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.

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:

Confidentiality Die Vertraulichkeit gehört zu den wichtigsten Schutzzielen in der Informationssicherheit [1]. Das Wort *Confedentiality* kommt vom lateinischen Wort *confidere* und bedeutet so viel wie *vertrauen* **pons** [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].

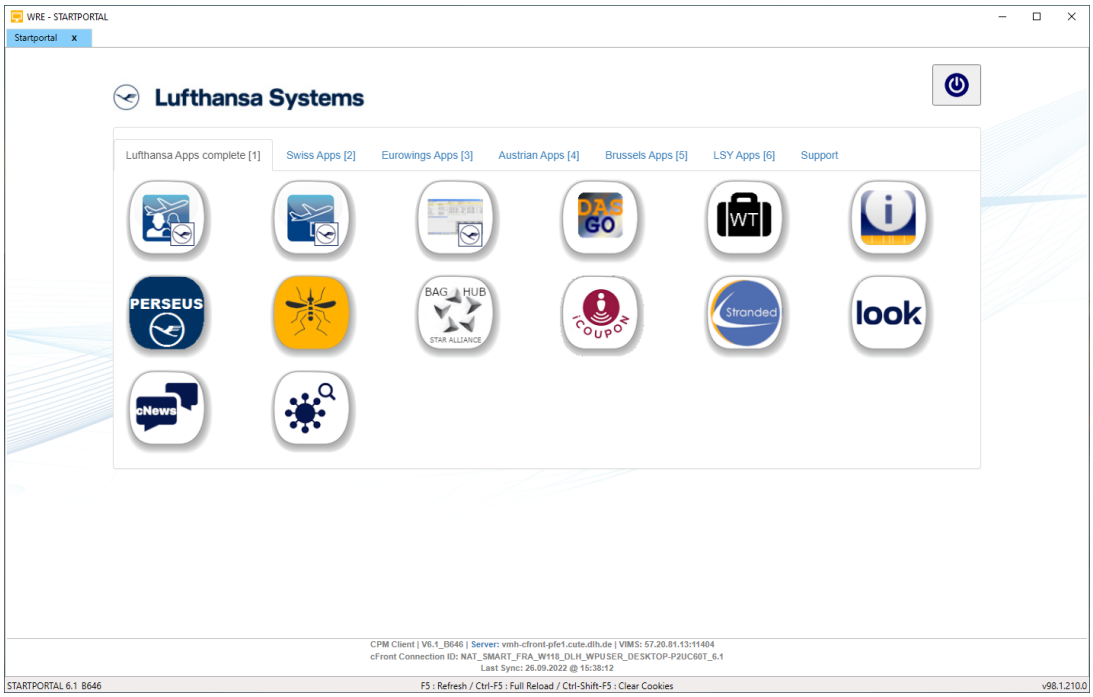

Abbildung 2.1: Umsetzungsmöglichkeit mit Keycloak

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 [2]. 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.

Integrity Das Wort Integrity leitet sich vom lateinischen Wort *tangere* ab und bedeutet so viel wie *berühren* **pons**. 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 [2]. Schickt ein Sender S beispielsweise eine Nachricht an einen

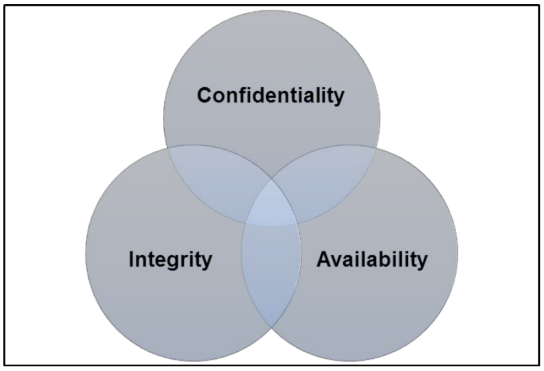

Abbildung 2.2: CIA-Triad

Empfänger E, so soll die Nachricht identisch beim Empfänger E ankommen, wie sie vom Sender S gesendet wurde [2]. Umsetzungsmöglichkeiten die Integrität zu schützen beinhalten Maßnahmen wie beispielsweise das Verwenden einer Firewall, Intrusion Detection System (IDS) oder auch digitale Signaturen [2].

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 [2]. Verfügbarkeit kann beispielsweise durch Netzwerksicherheit (z.B. Schutz vor Distributed Denial of Service (DDoS)) oder Fehlertoleranz (z.B. durch Limitierung von Authentifizierungsversuchen) gewährleistet werden [2].

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

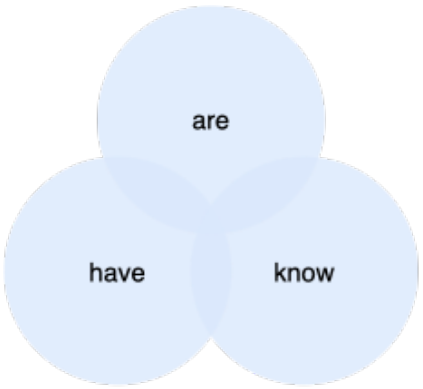

Abbildung 2.4: Faktoren der Authentifizierung

Die Authentifizerung dient häufig als erste Verteidigungslinie von Systemen [3]. 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 [3]. 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 [3]. 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 [3]. 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** [3]. 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 [3].

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 [4] [3] [5]. 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 [4]. Die Sicherheit informationstechnischer Systeme ist somit abhängig von der Sicherheit der genutzten Passwörter [3]. Trotz ihrer weitreichenden Verbreitung gelten Passwörter als eine der größten Sicherheitsrisiken für Systeme, da sie vile Schwachstellen und Angriffsvektoren bieten [5] [6]. Laut einer Studie von *Verizon* basierten 2017 81% der Hackerangriffe auf der Kompromittierung von Passörtern [7] **verizon2017**. Eine weitere Studie zeigt auf, dass 2017 Phishing E-Mails die Angriffsmethode darstellte **Symantec** [7]. 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 [3]. 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 [3]. 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 [9]. Dies liegt insbesondere an der Einfachheit und dem geringen Aufwand, welche die Nutzung von Passwörtern mit sich bringt [5].

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 [4] [8].

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 [4]. 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 län-

gerer 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.

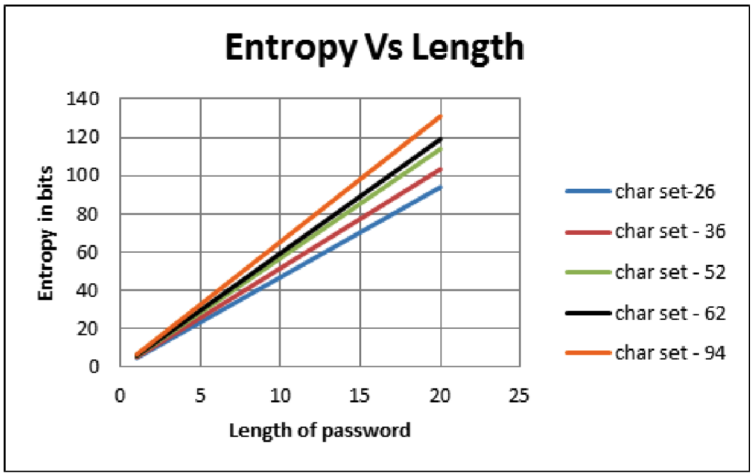

Abbildung 2.5: Entropie in Abhängigkiet der Passwortlänge

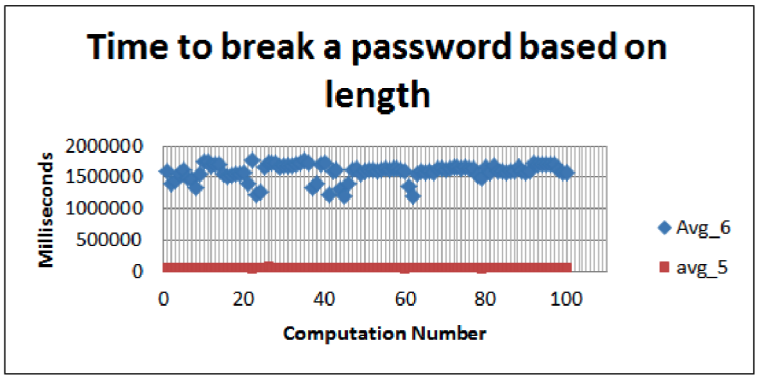

Abbildung 2.6: Zeit, um ein Passwort zu brechen in Abhängigkeit zu der Länge

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 [3]. 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 [4]. 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 [4].

Eine bessere Variante - allerdings weitaus nicht optimale - ist die Verschlüsellung der gespeicherten Passwörter. Der größte Kritikpunkt an dieser Variante ist allerdings, dass Verschlüssellungen 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 [4].

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 [4].

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 [4].

Faktor Mensch:

Neben den aufgezählten technischen Aspekten, stellt der Mensch selbst eine der größten Angriffsvektoren bezogen auf Passwörter dar [9] [5]. 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 [4] [3] [5].

Die genannten Faktoren führen dazu, dass die Anzahl an genutzten Kombinationen für ein Passwort deutlich geringer ist als die gesamte Menge an möglichen

Kombinationen [3]. Das macht von Menschen erstellte Passwörter deutlich anfälliger für Angriffe, da diese einfacher zu erraten sind [4]. 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 [5]. Kontraproduktiv wirken in diesem Zusammenhang auch Policies und Richtlinien zur Erstellung von Passwörtern [5]. 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 [5] [8].

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 [9]. 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 [9] [8].

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 [6].

Passwortlose Verfahren werden dabei als sicherer im Vergleich zur passwortbasierten Alternative angesehen, da viele der in **2.4** aufgeführten Angriffsvektoren passwortlosen Ansätze nicht existieren [10] [11]. 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 [10].

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 [10]. 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:

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 [10] [11]. 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 [10]. Der

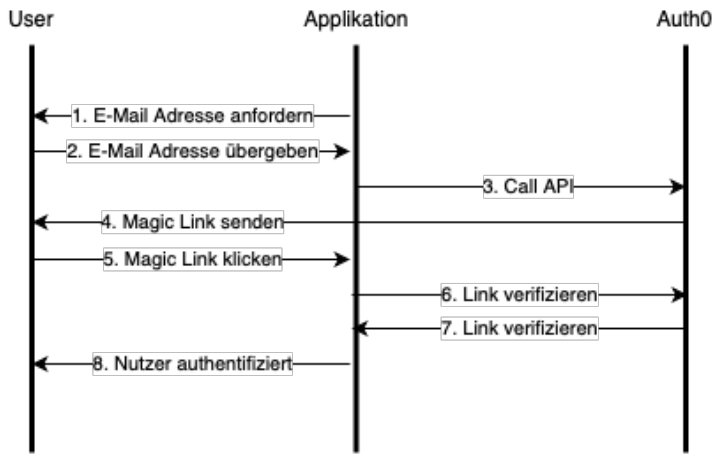

Abbildung 2.7: Beispielhafte Umsetzung eines Magic Links

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 [10].

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 [11]. Auch die Implementierung und die Kosten zur Instandshaltung sind verhältnismäßig gering einzuorden [11].

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 [11]. 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 [10]. Dies kann geschehen, ohne dass der Nutzer dies überhaupt bemerkt [10].

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 [10] [11]. 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 [10]. Häufig werden OTPs allerdings nicht für eine oben beschrieben SFA genutzt, sondern dienen als zusätzlicher Faktor für eine MFA [10]. 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 [10]. 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) [10]. 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 [10]. 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 [10].

Vorteile: Die Nutzung von OTPs ist sehr effektiv für eine MFA, da die Sicherheit im Vergleich zu einer passwortbasierten SFA signifikant erhöht werden kann

[10] [11]. Zudem handelt es sich um eine sehr benutzerfreundliche und einfachz anwendbare Methode, welche sich bereits für die Nutzung von MFA weitreichend etabliert hat [11]. 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 [10] [11].

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 [11]. Dabei werden einzigartige biometrische Merkmale des Nutzers genutzt um seine Identität zu verifizieren. Dazu gehören beispielsweise Fingerabdrücke oder eine Gesichtserkennung [11]. 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 [11]. 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 [11].

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 [11]. Zudem können sich biometrische Daten im Laufe der Zeit verändern. Auch Verletzungen oder Krankheiten können zu Veränderungen der biometrischen Daten beitragen [3]. 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 [11].

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) [12]. 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 **yuibkey2023fido2**.
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*) **yuibkey2023fido2**.
3. Eine passwortlose MFA mit Hilfe einer zusätzlichen PIN für den Security Key **yuibkey2023fido2**.

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 [13].
- Passwortlose Verfahren mit Security Key wurden mehr akzeptiert als tradionelle passwortbasierte Verfahren [13].
- Es handelt sich um eine implizite Garantie, dass sich lediglich Nutzer authentifizieren können, welche auch im Besitz des Security Keys sind [13].
- 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 [13] [6].
- Es wird ein deutlich geringerer kognitiver Aufwand benötigt, da Nutzer keine neuen Passwörter mehr erstellen und merken müssen [13].
- 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 [13] [6].

- Es handelt sich um offene und standardisierte Protokolle. Das verhindert verschieden Lösungsansätze verschiedener Hersteller und führt zu einer unabhängigeren und universellen Lösung [6].

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 [13].
- Da es sich um zusätzliche Hardware handelt kann diese ebenfalls kaputt gehen [6].
- Im Unternehmenskontext, kann die Verwaltung und Verteilung der Authentifizierungsgeräte zu einem Problem werden [6].
- Da es sich um Hardware handelt, können Zugänge nicht an vertraute Personen weitergegeben werden, da der Zugang nur mit dem Authentifizierungsgerät möglich ist [13].
- Ohne das Authentifizierungsgerät sind keine spontanen Logins möglich [13].
- Bereits das aus der Tasche holen des Authentifizierungsgerätes ist für manche Nutzer bereits eine Hürde [6].
- Es wird ein physischer Aufwand benötigt, da das Authentifizierungsgerät mitgeführt werden muss [13].
- Authentifizierungsgeräte sind häufig mit Kosten verbunden, welche vom Nutzer getragen werden müssen [13].
- 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 [13].

- Selbst Nutzern, welchen das Konzept der passwortlosen Authentifizierung gefällt, nutzen in der Praxis häufig weiterhin Passwörter [6].
- Nutzer wollen keine Angewohnheiten verändern, wenn die nicht dazu gezwungen sind [6].
- Nutzer verwenden lieber Passwörter, da sie das Konzept und die Technologie besser verstehen [13].
- Nicht zwangsweise schneller als die Nutzung von Passwortmanagern [6].
- Allgemein fällt das Feeback von Nutzern weniger positiv aus, wenn diese vorher bereits Passwortmanager genutzt haben [6].

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 [13]. 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

- FIDO2 wird von der FIDO und dem World Wide Web Consortium (W3C) entwickelt und bereitgestellt [13] [6].

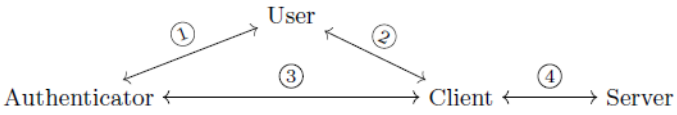

Figure 1: Communication channels

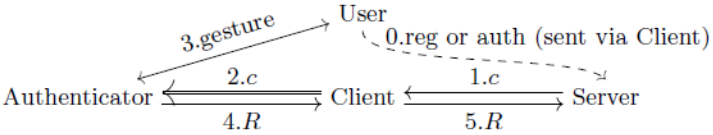

Figure 2: FIDO2 flow (simplified): double arrow = CTAP2 authorized command.

Abbildung 2.9: Umsetzungsmöglichkeit mit Keycloak

- Die FIDO Allianz ist eine Organisation mit weltweit über 250 Mitgliedern. Darunter befinden sich Unternehmen wie Google, Microsoft, Apple Amazon, Facebook, Visa und viele mehr [13] [6].
- Ziel ist es Nutzer zu authentifizieren, ohne, dass diese eine Passwort nutzen müssen [8] [7].
- Basiert auf der Nutzung eines internen oder externen Authentifizierungsgerätes [8] [7].
- Dabei können Authentifizierungsgeräte, ebenfalls mit einer PIN oder einem biometrischen Merkmal, geschützt werden [6].
- Hierbei ist ein PIN allerdings nicht gleichzusetzen mit einem Passwort. Der PIN wird lediglich für das Authentifizierungsgerät genutzt und wird auch nur auf diesem gespeichert [6] [7].
- Es handelt sich dabei also auch nicht um eine MFA, sondern, um einen einzelnen Faktor, welcher lediglich den Zugriff das Gerät selbst authentifiziert [7].

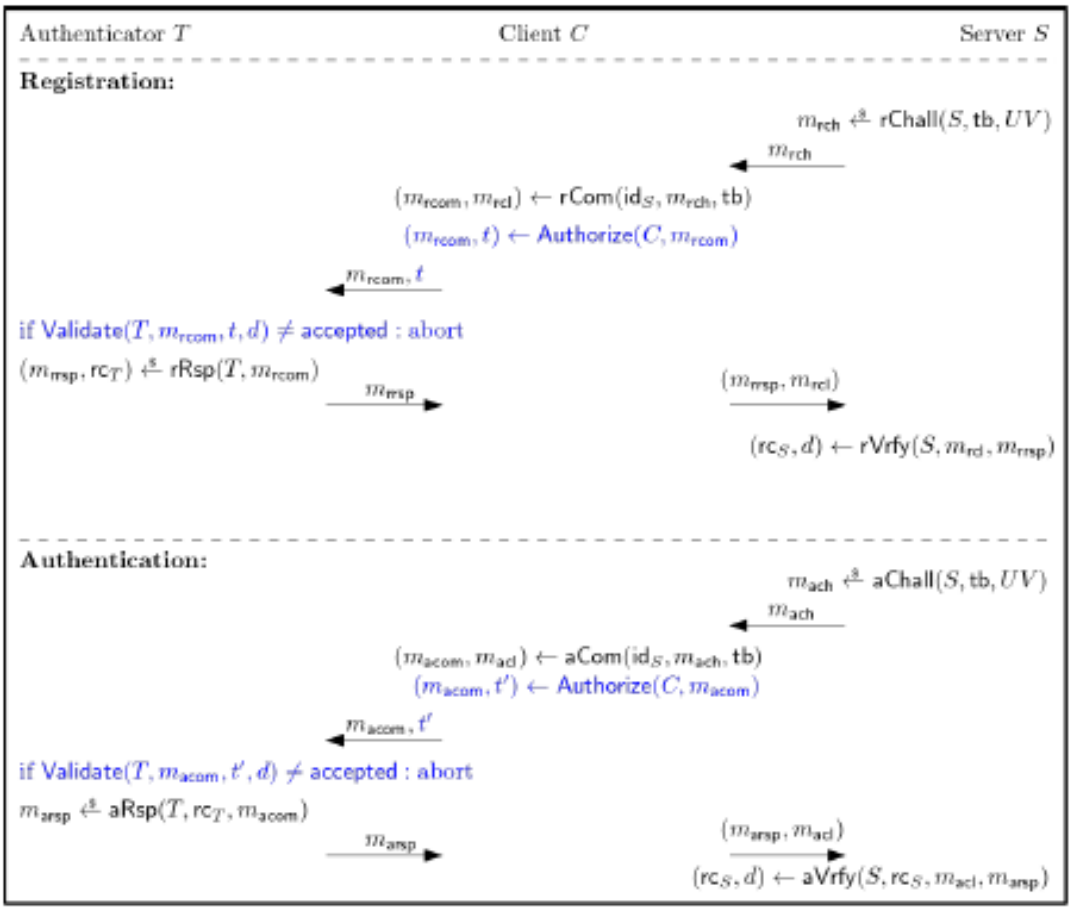

Abbildung 2.10: Umsetzungsmöglichkeit mit Keycloak

- FIDO2 unterstützt sowohl MFA als auch SFA [13] [6].
- Viele Alternativen zur passwortbasierten AUthentifizierung existieren bereits. Diese werden allerdings nur in einem sehr geringen Ausmaß genutzt [6].
- Stellt Zugangsdaten bereit, welche nicht gephisht oder von Datenlecks betroffen sein können [13].
- Das liegt daran, dass keine geteilten Geheimnisse zwischen Nutzer und Dienst existieren, welche auf einem Server gespeichert werden [8].

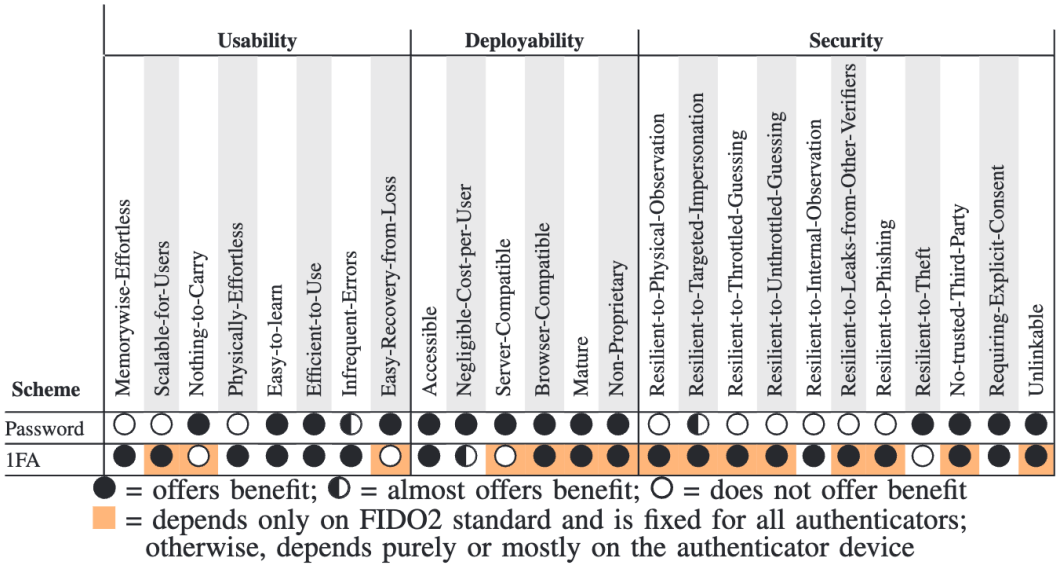

Abbildung 2.11: Umsetzungsmöglichkeit mit Keycloak

- Wird von fast allen Browsern standardmäßig unterstützt [13].
- Viele verfügbare Authentifizierungsgeräte. Z.B. Security Keys oder auch Smartphones. Beispielsweise Apples Touch ID oder Face ID [13].
- Besteht aus zwei Komponenten: CTAP2 für die Kommunikation zwischen Client und Authentifizierungsgerät und WebAuthn für die Kommunikation zwischen Client und Server [6].
- Dabei wird WebAuthn von der W3C spezifiziert und CTAP2 von der FIDO Allianz [6].

2.7.1 Webauthn

- WebAuthn ist ein Standard, welcher von dem W3C entwickelt wird. Das Protokoll erlaubt es Webanwendungen Nutzer zu authentifizieren. Dies

kann dabei auch über Client-to-Authenticator Protocol 2 (CTAP2) erfolgen [13]. ?

- Wurde 2019 ein offizieller Webstandard [6].
- Spezifiziert eine standardiserte, vom Browser unabhängige JavaScript API zur Authentifizierung von Nutzern für Webanwendungen. So können Webanwendungen eine Authentifizierung integrieren, welche resistent gegenüber Phishing, 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 [6].
- Passwortloses Challenge-Response-Verfahren zwischen Client und Server [7].
- WebAuthn unterstützt zwei Opertationen: Registrierung und Anmeldung [7].
- 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 [7].
- Registrierungsphase: Der Server S sendet eine challenge message m_{rch} über den Client C an den Security Key. Diese Challenge beinhaltet eine randomisierte Nonce, Parameter (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 den Security Key T übermittelt. Der Security Key T erzeugt öffentlich-privates Schlüsselpaar, welches an den Server S gebunden ist und diesem ermöglicht eine Verifizierung, während der folgenden AUthentifizierungsphase durchzuführen. Zudem gibt der Security Key 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*. [14]

- Authentifizierungsphase: Der Client empfängt die 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 den Security Key T übermittelt. Der Security Key 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 verifizieren lassen. [14]

- Die Sicherheit von WebAuthn 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 [7].

2.7.2 CTAP2

- 2018 wurde CTAP2 als internationaler Standard der International Telecommunication Union Telecommunication Standardization Sector (ITU-T) anerkannt [7].

- CTAP2 ist 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 wie beispielsweise ein Security Key, welches über USB, Bluetooth oder NFC eine Verbindung mit dem Client aufbaut. Aber auch ein internes Gerät wie beispielsweise ein Fingerabdruckscanner oder ein Trusted Platform Module können als Authentifizierungsgerät genutzt werden [13].

- CTAP2 spezifiziert, die Kommunikation zwischen einem Authentifizierungsgerät und einem Client. 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 erlaubt. Dafür muss der Nutzer beispielsweise einen Knopf am Authentifizierungsgerät drücken und/oder sich mit Hilfe eines PINs oder eines biometrischen Merkmals beim AUthentifizierungsgerät authentifizieren [7].

- Das Ziel ist es somit einen Client an das AUthentifizierungsgerät zu binden. Ist ein CLient nicht an das authentifizerungsgerät gebunden, kann dieser sich nicht authentifizieren [7].

- besteht aus mehreren Phasen. 1. In der Setup Phase initialisiert ein Client C' einen PIN, welcher vom User übergeben wird an den Security Key T. 2. In der Binding Phase tauschen ein Client C (nicht zwangsweise C') und der Security Key T einen gemeinsamen Verbindungsstatus aus, wenn der Client C in der Lage ist, Informationen über die auf dem Security Key T gespeicherte PIN zu liefern. So soll eine einzigartige Verbindung zwischen dem Client C und dem Security Key T hergestellt werden. Schlägt der Client C drei mal in Folge fehl die PIN zu liefern, wird der Security Key T neu gestartet und der Verbindungsstatus wird zurückgesetzt. Schlägt der Client C insgesamt acht mal fehl, wird der Security Key T gesperrt. 3. Ist diese Phase erfolgreich, autorisiert der Client C jeden Befehl, indem eer einen Tag t ausgibt, welcher mit der command message an den Security

Key T übermittelt wird. Der Security Key T fährt lediglich fort, wenn eine *positive decision* d des Users vorliegt (beispielsweise einem Knopfdruck) und validiert darauf hin die command message und den Tag t. [14]

- CTAP2 nutzt unauthentifizierten Diffie-Hellman Schlüsselaustausch [7].
- Dieser kann von Man In The Middle (MITM) Angriffen betroffen sein [7].
- In der Binding Phase sendet das Authentifizierungsgerät dem CLient ein pinToken, welcher beim hochfahren des Authentifizierungsgerätes generiert wird. Dieser pinToken wird lokal auf dem AUthentifizierungsgerät gespeichert und wird von dem verbundenen Client in der Access Channel Phase genutzt, um die nachfolgenden Nachrichten des Clients zu autorisieren [7].
- Jedem AUthentifizierungsgerät wird ein pinToken pro hochfahren zugeordnet. Das bedeutet mehrere CLients erezugen mehrere Access Channels mit dem selbem Authentifizierungsgerät und dem selben pinToken [7].
- Dadurch wird die Sicherheit von CTAP2 limitiert ??

2.7.2.1 CTAP2.1

- Gilt in Verbindung mit WebAuthn 2 als Post-Quantum (PQ) bereit, da ein Operationsmodus ermöglicht wird, der nur kryptographische Primitive, digitale Signaturen und Key Encapsulation Mechanism (KEM) verwendet [14].
- Im Gegensatz zu CTAP2 basiert CTAP2.1 nicht auf unauthentifizierten Diffie-Hellman Schlüsselaustausch, sondern auf einem sogeannten PIN/UV Auth Protocol (puvProtocol), wodurch die PQ-Sicherheit ermöglicht wird [14].

- In CTAP2 wird der Verbindungszustand als *pinToken* definiert, welcher aus mehreren 128 Bit-Blöcken besteht und keine maximale Begrenzung der Länge beseitzt. In CTAP2.1 wird der Verbindungszustand als *pinUvAuthToken* definiert welcher eine feste Länge von 128 oder 256 Bit besitzt [14].
- Der pinToken von CTAP2 wird bis zum nächsten Neustart wiederverwendet. Der pinUvAuthToken von CTAP2.1 wird nach jeder erfolgreichen Authentifizierung neu generiert. Das führt dazu, dass CTAP2.1 eine Strongly Unforgeable (SUF)-t' Sicherheit aufweist und CTAP2 lediglich eine Unforgeable (UF)-t' Sicherheit [14].
- CTAP2 erlaubt es Security Keys und Clients nur den pinUvAuthToken zu teilen, wenn der Nutzer den korrekten PIN eingegeben hat. CTAP2.1 ermöglicht zusätzlich, dass der Nutzer sich mit Hilfe eines biometrischen Merkmals authentifiziert [14].

2.7.3 Sicherheit

- FIDO2 ist eine Erweiterung des FIDO U2F Protokolls und bietet die selbe Sicherheit wie public key Kryptographie [13].
- Es handelt sich um geprüfte assymetrische Kryptographie [6].
- Es handelt sich dabei um ein Challenge-Response-Verfahren mittels Hardware basierten Authentifizierungsgeräten. Dies bietet einige Vorteile gegenüber passwortbasierten Verfahren. Es gibt keine geteilten Geheimnisse zwischen Usern und Diensten, welche durch Phishing oder Datenlecks kompromittiert werden können. Dabei ist das selbe Authentifizierungsgerät für mehrere Dienste nutzbar, ohne, dass sich dabei eine Verknüpfung zurückführen lässt [13] [6].
- lediglich die Session kann kompromittiert werden [8].

- Authentifizierungsgeräte lassen sich mit zusätzlichen PINs oder biometrischen Merkmalen absichern, um sich ebenfalls vor Diebstahl schützen [7].
- Unauthentifizierter Diffie-Hellman Schlüsselaustausch könnte durch ein Password Authenticated Key Exchange (PAKE) Verfahren ersetzt werden [7].
- Das Paper gibt an, dass dieses sicherer und effizienter sein soll [7].
- In folgendem Szenario: 1. Der Nutzer besitzt einen Security Key, welcher mit einem drückbaren Knopf oder ähnlichen ausgestattet ist. 2. Der Security Key ist mit einem geheimen PIN geschützt. 3. Der Nutzer autorisiert vertrauten Clients auf den Security Key zuzugreifen. 4. Der Nutzer verbindet seinen Security Key mit mehreren Clients und nutzt diese um sich bei mehreren Webdiensten zu registrieren/anzumelden. Dann ist versichert, dass: 1. Die Authentifizierung von dem Security Key durchgeführt wurde, welcher die genutzen Zugangsdaten bei dem Webdienst registriert hat. 2. ein autorisierter Befehl auf den Security Key zugegriffen hat. 3. und dieser autorisierte Befehl von einem autorisierten Client beauftragt wurde (sollte der Nutzer den korrekten PIN eingegeben haben). Dies setzt voraus, dass: 1. Der Security Key nicht gestohlen wurde. 2. Der PIN des Security Keys nicht kompromittiert wurde. 3. Der autorisierte Client nicht kompromittiert wurde (korrekte Ausführung von CTAP2 und CLient ist nicht von böswilliger Software betroffen). [7].
- Wird ein Security Key gestohlen, kann dieser nur genutzt werden, wenn ebenfalls der PIN bekannt ist [7].

3 Umsetzung

3.1 Aktueller Stand der LSY

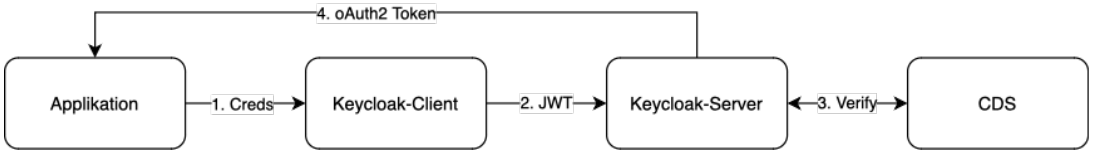

Abbildung 3.1: Aktuelle Umsetzung der Abteilung

- Innerhalb der Abteilung wird eine passwortbasierte Authentifizierung durchgeführt, welche zusätzlich durch MFA geschützt wird.
- Webanwendung nutzt einen eigene Anwendung, welche sich als Keycloak-Client aussgibt. Der Nutzer gibt seine Zugangsdaten an den Keycloak-Client weiter, welche diese verarbeitet. Dieser wandelt die Zugangsdaten in einen validen JWT-Token um und übergibt diesen an den Keycloak-Server. Dieser validiert die Zugangsdaten gegen die Corporate Directory Service (CDS). Ist die Validierung erfolgreich, wird vom Keycloak-Server ein oAuth2-Token erstellt und zurück an die Applikation übergeben.
- Innerhalb der LSY können sich Applikationen allerdings auch gegen das Azure AD authentifizieren lassen.

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 darauf, dass auch andere Hersteller Security Keys anbieten, welche das Fido2-Protokoll unterstützen.

Weitere bekannte Security Keys sind 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 **compWin** 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 Universal Second Factor (U2F). Microsoft ist allerdings nich abwärtskompatibel, was bedeutet, dass der Google Titan nicht für eine passwortlose Authentifizierung gegen das Azure AD genutzt werden kann **seckeytest**.

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.

3.3 Integration eines Yubikeys in die LSY

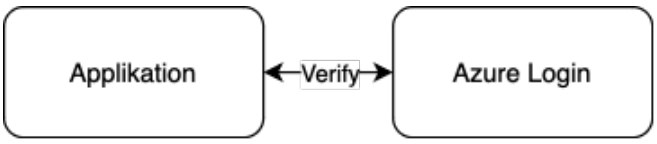

Abbildung 3.2: Umsetzungsmöglichkeit mit Azure AD

- Grundsätzlich gibt es zwei Möglichkeiten in die aktuelle Applikation eine passwortlose Authentifizierung mit Hilfe eines Yubikeys zu integrieren: Die Nutzung der Authentifizierung gegen das Azure AD oder die eine veränderte Nutzung der aktuellen Keycloak-Lösung.

Eine Lufthansa-weite Policy für die Nutzung der Azure AD verbietet allerdings die Nutzung eines Security Keys für eine SFA. Hier kann der Security Key lediglich als zweiter Faktor genutzt werden. Dies kann jeder Nutzer selber verwalten. Registriert ein Nutzer seinen Security Key in seinem Profil, erscheint bei der Anmeldung (nach der Eingabe des Passwortes) ein zusätzliches Feld:

Abbildung 3.3: Umsetzungsmöglichkeit mit Keycloak

Verwendet der Nutzer einen Security Key wird dieser im Folenden aufgefordert die zugehörige PIN einzugeben und den Knopf des Security Keys zu drücken. Grundsätzlich ist eine passwortlose Authentifizierung mit Hilfe eines Security Keys innerhalb der Azure AD möglich. Da eine Änderung dieser Lufthansa Policy notwendig wäre, übersteigt dies allerdings den Rahmen dieser Arbeit.

Da allerdings aktuell eine beschriebene Nutzung von Keycloak stattfindet und Keycloak eine passwortlose Authentifizierung mit Hilfe eines Security Keys unterstützt, wäre eine Umsetzung mit Hilfe von Keycloak möglich. Hierbei wird die aktuelle Lösung verändert und entsprechend angepasst:

Statt bei einer Anmeldung einen Client zu simulieren bietet Keycloak die Möglichkeit eine Anmeldung über eine Nutzeroberfläche zu realisieren. 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.

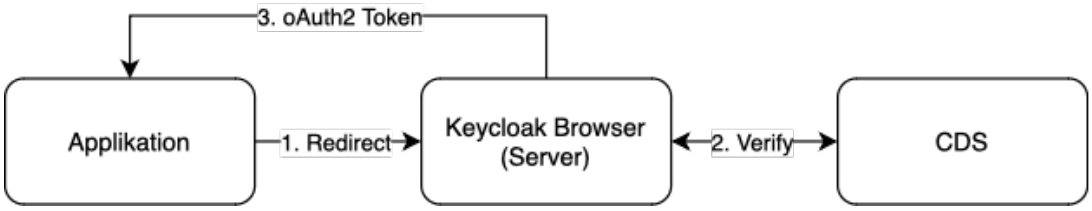

Abbildung 3.4: Veränderter Keycloak-Login

Um eine Passwortlose Authentifizierung in Keycloak zu ermöglichen, muss der Authentication Flow für eine Browser-Anmeldung modifiziert werden. Der angepasste Authentication Flow besteht aus folgenden Schritten:

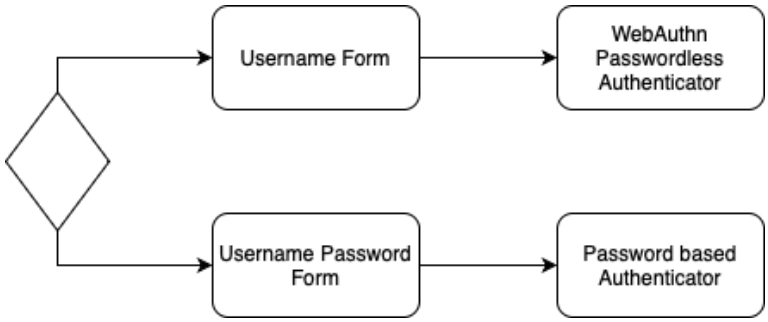

Abbildung 3.5: Authentication Flow

Hierbei wird die untere Hälfte der Grafik weiterhin ermöglicht, da es sich lediglich um einen Test handelt. Grundsätzlich wird diese nicht ermöglicht, da Keycloak eine reine passwortlose SFA unterstützt.

Die obere Hälfte der Grafik entspricht dem für diese Arbeit relevanten Authentication Flow. Dabei wird der User zunächst aufgefordert seinen Nutzernamen einzugeben und anschließend seinen Security Key zu verwenden. Dies ist notwendig, um die Nutzung eines Security Keys für mehrere Zugänge zu ermöglichen. Ermöglicht man lediglich die Nutzung eines Zugangs pro Security Key, so wird die Eingabe des Nutzernamens nicht benötigt. Zusätzlich erfolgt eine Konfiguration des Keycloak-Servers, welche die Registrierung eines Security Keys bei der Registrierung eines neuen Nutzers ermöglicht.

Mit Hilfe dieser Konfiguration werden zwei Abläufe ermöglicht. Eine neue Registrierung:

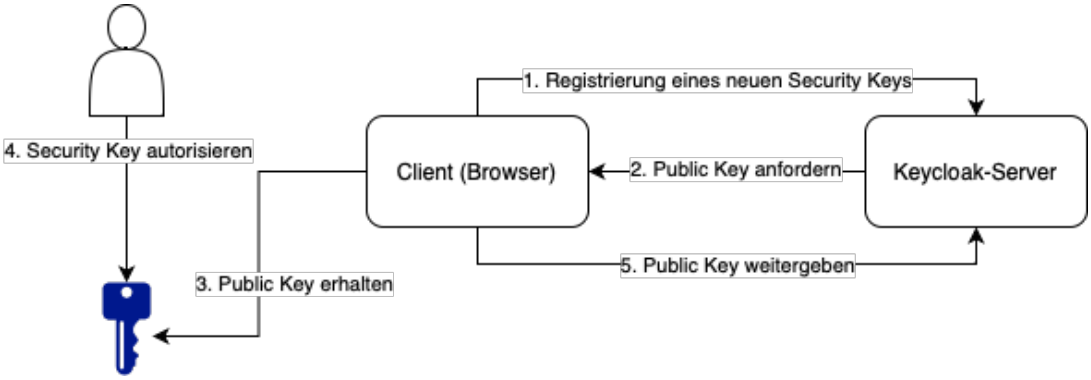

Abbildung 3.6: Registrierung (vereinfacht)

Sowie eine neue Anmeldung:

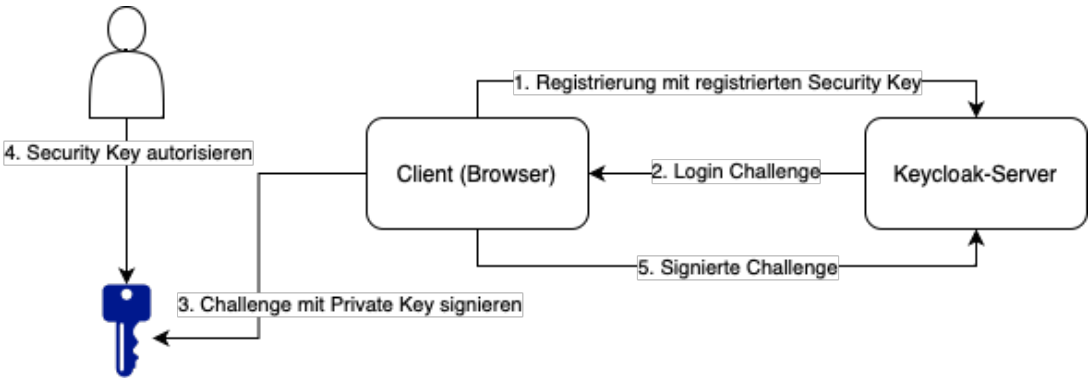

Abbildung 3.7: Anmeldung (vereinfacht)

Die Grafiken stellen den vereinfachten Ablauf der Registrierung und Anmeldung mit Hilfe eines Security Keys dar. Die detailierte Darstellung der Funktionsweise ist in Kapitel 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:

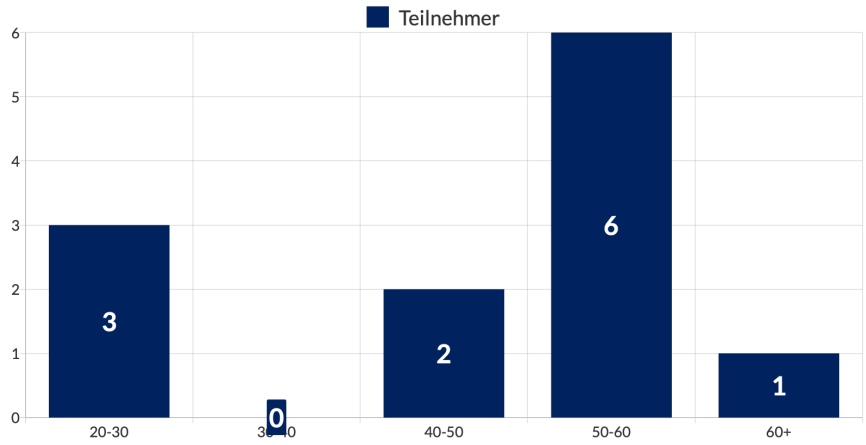

Abbildung 3.8: Alter der Teilnehmer

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. 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.

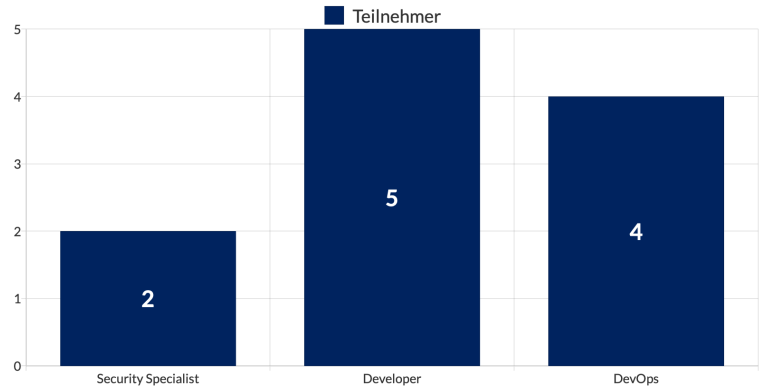

Abbildung 3.9: Alter der Teilnehmer

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

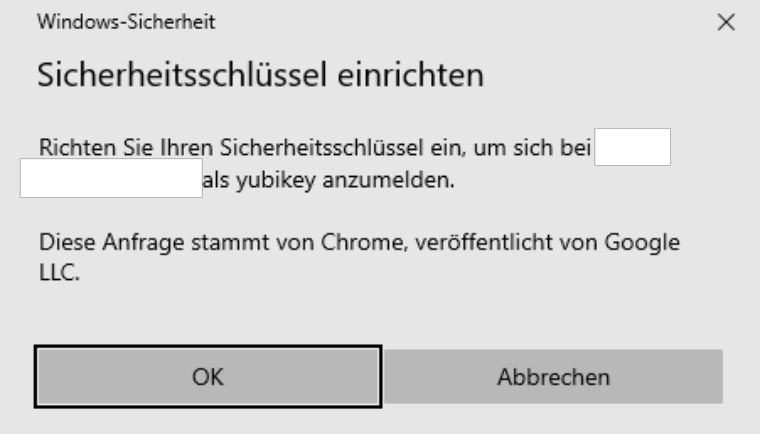

Abbildung 3.11: Veränderter Keycloak-Login

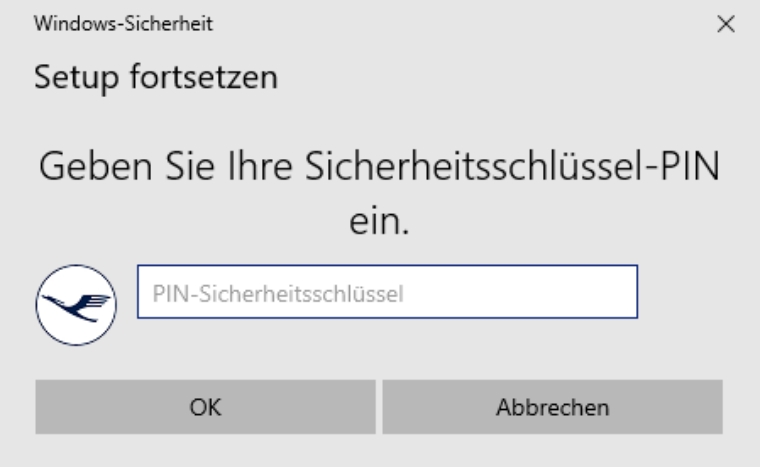

Abbildung 3.12: Veränderter Keycloak-Login

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.

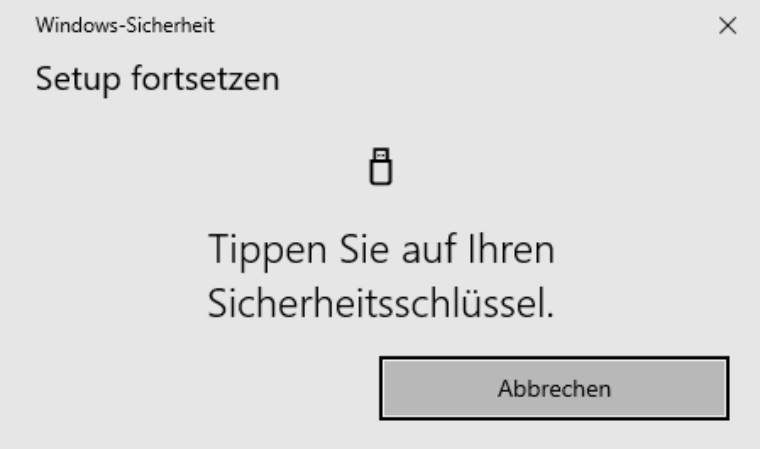

Abbildung 3.13: Veränderter Keycloak-Login

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 [6] 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 **techstat**. 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) [6]. 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 **lyastani2020fido**. 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. 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 **techstat**. 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.

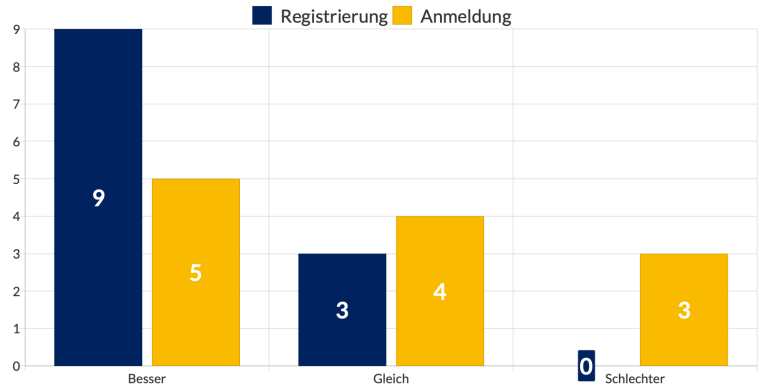

Abbildung 3.15: Veränderter Keycloak-Login

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 **databreach**. 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 **databreach**. 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

In Kapitel xy wurden insbesondere drei Ziele der Arbeit definiert. Diese werden im Folgenden bewertet. Die Ergebnisse der Arbeit machen deutlich, welche Schwachstellen und Angriffsvektoren aufzeigen. Aus diesem Grund existiert jedoch nicht bloß eine Alternative. Es gibt verschiedene passwortlose Ansätze, welche sich für verschiedene Anwendungsfälle eignen. Jeder Ansatz hat dabei seine Vor- und Nachteile. Deutlich wird jedoch, dass das FIDO2-Projekt zu den meist untestützten und am weitesten verbreiteten Ansätzen gehört. Dies liegt auch an der gebotenen Vielfalt, da FIDO2 nicht nur Security Keys, sondern beispielsweise auch Passkeys unterstützt. Im Bezug auf Sicherheit ist das FIDO2-Protokoll eine erhebliche Verbesserung gegenüber der klassischen Passwortauthentifizierung. Da FIDO2 auf öffentliche/private Schlüssel basiert fallen die meisten Angriffsvektoren der klassischen Passwortauthentifizierung weg.

Aus diesen Gründen empfiehlt sich die Integration von FIDO2 als Alternative ebenfalls für den Unternehmenskontext. Für die Nutzung von Security Keys hingegegen lässt sich keine eindeutige Empfehlung auf den gegebenen Kontext der LSY aussprechen. Dies geht vor allem aus den Umsetzungsmöglichkeiten und der erarbeiteten Benutzerfreundlichkeit hervor. Aktuell ist eine FIDO2 Authentifizierung mit Hilfe eines Security Keys noch nicht ausreichend etabliert, um eine gesamte Umstellung vornehmen zu können. Nicht alle Dienste ermöglichen eine FIDO2 Authentifizierung. Zum aktuellen Zeitpunkt eignet sich die Nutzung von Security Keys lediglich für eine MFA. Das Ergebnis dieser Arbeit ist allerdings, dass sich die Nutzung von Security Keys insbesondere für eine SFA eignet.

Dies entspricht nicht den Richtlinien SFA. Solange allerdings keine weitreichende Unterstützung von FIDO2 erfolgt, wird keine Änderung dieser Richtlinie empfohlen. Die Nutzung von Security Keys als zusätzlicher Faktor wird aus wirt-

schaftlichen Gründen nicht empfohlen. In diesem Fall ist die Nutzung einer Authenticator App besser geeignet.

Grundsätzlich ist eine Nutzung von FIDO2 als Alternative zur klassischen Passwortauthentifizierung zu empfehlen. Sobald eine ausschließliche Nutzung ermöglicht wird lassen siche erhebliche Vorteile für die Sicherheit erzielen. Lediglich die Nutzung von Security Keys ist nicht zweifelsfrei zu empfehlen. In dieser Arbeit werden mehrere Kritikpunkte an der Benutzerfreundlichkeit aufgezeigt. Fragwürdig ist, ob diese Kritikpunkte nach einer erweiterten Gewöhnungsphase noch bestehen. Eine Lösung könnte die Nutzung von Passkeys darstellen. Diese sind allerdings noch nicht weitreichend verbreitet.

Daraus folgt die Empfehlung für die LSY nicht direkt auf FIDO2 in Kombination mit Security Keys zu setzen. Stattdessen sollte das Bewusstsein für passwortlose Alternativen erweitert werden. Die Offenheit für Alternativen sollte ebenfalls gefördert. Zusätzlich sollte die Etablierung von FIDO2 weiter beobachtet werden. Sobald eine ausschließliche Nutzung möglich ist, sollte eine Umstellung erfolgen.

5 Ausblick

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] 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.

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

[4] 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.

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

[6] 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.

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

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