TECHNISCHE HOCHSCHULE NÜRNBERG
GEORG SIMON OHM

Fakultät Informatik

Bachelorarbeit

Untersuchung von Decentralized Identifiern, deren Auswirkung auf den Datenschutz sowie ihre Anwendung in einem Prototyp eines digitalen Geldbeutels

Vorgelegt von:	Veronika Sedlackova
Matrikelnummer:	3004513
Erstprüfer:	Prof. Dr. Ronald Petrlic
Zweitprüfer:	Prof. Dr. Jens Albrecht
Betreuer:	Francis Pouatcha

Kurzdarstellung

Klassische Authentifizierungsverfahren wie z.B. Single Sign-On, speichern Daten des Benutzers, beispielsweise den Benutzername und das Passwort, in der Regel auf firmeninternen Servern ab. Dieses Konzept entzieht dem eigentlichen Dateninhaber allerdings die volle Kontrolle über seine Daten. Beispielsweise wären mit einem Hackerangriff die Daten gefährdet oder sogar weitere Daten des Users mit einem Löschen der Webseite nicht mehr verfügbar. Im Rahmen dieser Bachelorarbeit wird ein dezentralisiertes Verfahren vorgestellt, das keine Speicherung von Identitätsdaten, z.B. für eine Anmeldung, beim Webseitenbetreiber benötigt. Des Weiteren wird eine Möglichkeit vorgestellt, wie physikalische Identitätsnachweise, wie z.B. ein Personalausweis, durch ein digitales Pendant ersetzen werden können. Darauf aufbauend werden Strategien vorgestellt, die es gestatten den Informationsaustausch mit Dritten auf ein Minimum zu beschränken und so nur ausgewählte Daten zu teilen. Durch eine lokale Speicherung dieser Identitätsnachweise in einem digitalen Wallet, kann der Benutzer künftig selber entscheiden was und wieviel von seiner Identität preisgegeben werden kann.

Abkürzungsverzeichnis

API	Application Programming Interface
CSS	Cascading Style Sheets
DID	Decentralized Identifier
DLT	Distributed-Ledger-Technologie
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
ID	Identifier
IETF	Internet Engineering Task Force
JSON	JavaScript Object Notation
JSON-LD	JavaScript Object Notation for Linked Data
JWT	JSON Web Token
OASIS	Organization for the Advancement of Structured Information Standards
OIDC	OpenID Connect
PII	Personally Identifiable Information
PKI	Public Key Infrastructure
QR	Quick Response
SAML	Security Assertion Markup Language
SSID	Service Set Identifier
SSO	Single Sign-On
TLS	Transport Layer Security
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
VC	Verifiable Credentials
VP	Verifiable Presentation
VPN	Virtual Private Network
WWW	World Wide Web
W3C	World Wide Web Consortium
XML	Extensible Markup Language

Abbildungsverzeichnis

2.1 Föderierte Identität 6
2.2 Zentrale Identität 6

3.1 Selbstbestimmte Identität mit einer DLT 12

4.1 DID Architektur[1] 13
4.2 DID Syntax 15
4.3 Das aus Abbildung 4.2 aufgelöste DID Dokument 16
4.4 Modell einer Bitcoin Transaktion 18
4.5 Aufbau des Universal Resolvers[2] 19
4.6 Austausch von Identitätsdaten[3] (näheres in Kapitel 4.6.4) 20
4.7 Struktur eines Verifiable Credentials 21
4.8 Struktur einer Verifiable Presentation 22
4.9 Beispiel einer VP 23
4.10 Architektur einer DID Authentifizierung 25
4.11 Architektur eines Austauschs von Verifiable Credential 27

7.1 Ablauf eines Austausch von einem VC 38

Inhaltsverzeichnis

Abbildungsverzeichnis . . . . . **iv**

1 Einleitung **1**
1.1 Ausgangssituation 1
1.2 Ziel der Arbeit 1
1.3 Aufbau der Arbeit 2
1.4 Projektträger 2

2 Zugrundeliegende Technologien und Vorgeschichte **4**
2.1 Public Key Cryptography 4
2.2 Public Key Infrastructure 4
2.3 Blockchain 4
2.4 Authentifizierungs- und Autorisierungsverfahren 5

3 Self-Sovereign Identity **10**
3.1 Modell 10
3.2 Funktionsweise 12

4 Decentralized Identifier **13**
4.1 Architektur und Begriffe 13
4.2 Aufbau 15
4.3 DID Methode 17
4.4 DID Resolution 19
4.5 DID Integration in Verifiable Credentials 20
4.6 Anwendungsbeispiele 24

5 Auswirkungen auf den Datenschutz **29**
5.1 Datenschutz-Grundverordnung für die Blockchain 29
5.2 Reduzierung der Datenoffenlegung 30

6 Aktueller Entwicklungsstand **34**
6.1 Dezentrales Netzwerk „IDunion“ 34
6.2 DID Authentifizierung mit SelfKey 34
6.3 Wallet für digitale Identitäten 35

7 Prototyp 36
7.1 Konzept eines Wallets 36
7.2 Konzeptionelle Modellierung 37
7.3 Implementierung 38

8 Schlussbetrachtung 41
8.1 Zusammenfassung 41
8.2 Ausblick 42

Literatur 43

1 Einleitung

1.1 Ausgangssituation

Der Besitz eines Identifikationsnachweises gewährt jeder Person den Zugang zum Gesundheitswesen, Bildung oder fundamentale Rechte wie beispielsweise das Wahlrecht. In der realen Welt werden dafür physische Nachweise wie der Reisepass, Führerschein oder der Personalausweis von staatlichen Behörden ausgestellt [1]. Diese, sowie nicht staatlich ausgestellte Zertifikate, sind nach der Ausstellung im Besitz des Eigentümers und können ohne die Erlaubnis des Ausstellers bei jeglichen Stellen vorgezeigt werden.

Doch nicht nur materielle Belege, sondern auch im World Wide Web (WWW) erstellte Identifikatoren, wie E-Mail-Adressen oder Benutzernamen, können eine Person identifizieren [1]. Der Webseitenbetreiber ist in diesem Fall die zentrale Stelle, der die zur Verfügung gestellten Daten des Benutzers sicher aufzubewahren hat. Solche Datenpools waren und sind ein beliebtes Ziel für Hackerangriffe. Ein Beispiel für einen Angriff waren die im März 2020 gestohlenen Anmeldedaten der Mitarbeiter der World Health Organization [2]. Der Großteil dieser immateriellen Identifikatoren hat gemeinsam, dass der Identitätsträger nicht die volle und alleinige Kontrolle seiner Daten besitzt, sondern bei zentralen Stellen eine digitale Identität anmietet. Um dem Einzelnen, wie im realen Kontext, ein Recht auf Selbstbestimmung seiner Daten zu gewährleisten, wurde das Konzept der Self-Sovereign Identity entwickelt. Hierbei kann eine Person oder eine Organisation eine dezentralisierte Identität erzeugen, über die sie die alleinige Kontrolle besitzt.

1.2 Ziel der Arbeit

Um ein Verständnis für die Funktionsweise von Decentralized Identifiern entwickeln zu können, soll das Konzept aus theoretischer Sicht untersucht werden. Anschließend soll deren Integration in Verifiable Credentials betrachtet werden. Ebenso wird analysiert inwieweit Daten auf einer öffentlichen Blockchain ein Problem für die Privatsphäre des Benutzers darstellen können. Des Weiteren werden datenschutzfreundliche Alternativen vorgestellt, die es ermöglichen sollen, nur ein Minimum an erforderlichen Daten für einen bestimmten Anwendungsfall an Dritte zu übermitteln. Das Gesamtbild wird durch einen

minimalistischen Prototyp eines digitalen Wallets abgerundet, der einen Anwendungsfall visualisiert.

1.3 Aufbau der Arbeit

Das erste inhaltliche Kapitel führt zugrundeliegende Technologien ein, die zum Verständnis in den nachfolgenden Kapiteln benötigt werden. Zusätzlich werden bereits standardisierte Authentifizierungs- und Autorisierungsverfahren vorgestellt.

Der Begriff der Self-Sovereign Identity sowie dessen Konzept wird im dritten Kapitel mit zentralen Verfahren verglichen. Zum Schluss des Kapitels werden Decentralized Identifier und Verifiable Credentials als mögliche Formen der Self-Sovereign Identity eingeführt.

Im vierten Kapitel werden die Grundlagen zu Dencentralized Identifiern, sowie deren Integration in Verifiable Credentials vermittelt. Zur Verdeutlichung werden die beiden genannten Themen anschließend durch je ein Anwendungsbeispiel ergänzt.

Inwiefern Daten auf einer öffentlichen Blockchain ein Problem für die Privatsphäre des Benutzers darstellen, wird im folgenden Kapitel erörtert. Ergänzend dazu werden datenschutzfreundliche Alternativen aufgezeigt, die ein Offenlegen nicht benötigter Daten gegenüber Dritten verhindern können.

Das sechste Kapitel gewährt einen Einblick in den aktuellen Entwicklungsstand der Self-Sovereign Identity.

Die in der Theorie vorgestellten Konzepte und Grundlagen werden im nun folgenden Kapitel durch den Prototypen eines digitalen Wallets in die Praxis umgesetzt. Der Prototyp wird zunächst den generellen Anforderungen eines Wallets gegenübergestellt. Anschließend werden die Anforderungen ausgewählt, die für die grundlegende Funktionsweise eines Wallets benötigt werden und mit Hilfe des Prototyps veranschaulicht.

Im abschließenden Kapitel wird die Arbeit in ihren Kernpunkten zusammengefasst sowie ein möglicher Ausblick, auf weiterführende Themen für folgende Arbeiten, gegeben.

1.4 Projektträger

Diese Arbeit wurde in Kooperation mit der adorsys GmbH & Co. KG verfasst. Adorsys ist ein mittelständisches IT-Unternehmen, das 2006 in Nürnberg gegründet wurde. Der Hauptfokus richtet sich auf die Erstellung individueller Software und das Angebot von

Dienstleistungen für Geldinstitute, Versicherungen sowie Drittanbieter der Finanzbranche. Zu den Kunden gehören Unternehmen wie die DATEV eG, Bausparkasse Schwäbisch Hall AG oder die Deutsche Leasing AG. [3]

2 Zugrundeliegende Technologien und Vorgeschichte

2.1 Public Key Cryptography

Public Key Cryptography (Deutsch: Public-Key-Verschlüsselung) ist ein asymmetrisches Verschlüsselungsverfahren, bei dem jeder Benutzer ein Schlüsselpaar besitzt. Das Schlüsselpaar besteht aus einem öffentlichen sowieso einem dazugehörigen privaten Schlüssel.

Privater Schlüssel: Ist geheim und nur dem Besitzer bekannt. Er wird verwendet, um Daten zu signieren oder um mit dem öffentlichen Schlüssel verschlüsselte Nachrichten zu entschlüsseln.

Öffentlicher Schlüssel: Ist offen bekannt und kann eine digitale Signatur verifizieren oder Daten für den Besitzer des privaten Schlüssels verschlüsseln.

2.2 Public Key Infrastructure

Public Key Infrastructure (Deutsch: Public-Key-Infrastruktur) ermöglicht eine sichere Kommunikation im Internet durch die Verwendung von Public Key Cryptography. Eine sogenannte Registration Authority (Deutsch: Registierungsstelle) überprüft im Auftrag einer Certification Authority (Deutsch: Zertifizierungsstelle) die Authentizität des Antragstellers. Die Certification Authority stellt im Falle einer erfolgreichen Überprüfung jedem Antragssteller ein individuelles Zertifikat aus. Mit Hilfe des Zertifikats können Webseite-Besucher überprüfen, ob die Kommunikation mit der Webseite sicher ist [4].

2.3 Blockchain

Eine Blockchain kann als eine Kette von Blöcken, mit darin enthaltenen Datensätzen (z.B. Transaktionen), definiert werden. Jeder Block enthält eine aus dem vorhergehenden Block berechnete eindeutige Kennung (Hash), sodass eine chronologische Verkettung entsteht. Ein bedeutendes Merkmal der Blockchain ist ihre Manipulationssicherheit. Mit

jeder Änderung ändern sich alle nachfolgenden Hashwerte, wodurch eine Täuschung erkannt werden würde. Zusätzlich schafft sie durch ihre Dezentralität das typische Client-Server-Modell ab. Eine Kopie des kompletten Datensatzes ist dafür auf mehreren unabhängigen Rechner verteilt. Das Konzept der Blockchain ist vor allem durch ihren Einsatz bei digitalen Währungen (z.B. Bitcoins) bekannt, wird allerdings auch vermehrt für andere Innovationen verwendet (z.B. Smart Contracts, Decentralized Identifier).

Im Rahmen dieser Bachelorarbeit bezieht sich die Verwendung des Begriffs „Blockchain“ auf eine öffentliche Blockchain. Diese kennzeichnet sich dadurch, dass sie zusätzlich zu den bereits genannten Merkmalen ohne Einschränkungen zugänglich ist. Zudem besitzt sie eine frei einsehbare Transaktionshistorie, wodurch jeder die dort hinterlegten Daten prüfen kann [5].

2.4 Authentifizierungs- und Autorisierungsverfahren

Authentifizierung: Nachweis der Trägerschaft einer Identität

Autorisierung: Gewährleistung eines speziellen Rechtes

Eine wissensbasierte Authentifizierung mittels Passwortes oder einer persönlichen Identifikationsnummer ist ein häufig genutzter Mechanismus, um Zugang zu einem System oder auf dessen Ressourcen zu erhalten. Laut einer Studie der Forschungsabteilung der Microsoft Corporation verwendet ein Benutzer 8,1 Passwörter pro Tag [6]. Ein damit einhergehendes Problem ist das menschliche Erinnerungsvermögen, denn oft werden nur kurze und einprägsame oder sich wiederholende Passwörter verwendet [7]. Die aus kurzen Passwörtern resultierende geringe Passwortstärke kann eine Schwachstelle für beispielsweise einen Brute-Force-Angriff darstellen [6].

In den nachfolgenden Kapiteln werden die Anfänge der Authentifizierungs- und Autorisierungsverfahren vorgestellt, welche die wissensbasierte Authentifizierung integrieren, aber ein höheres Maß an Benutzerfreundlichkeit bieten.

2.4.1 Single Sign-On

Ein Verfahren, um einem Benutzer einen schnellen Zugriff auf verschiedene Anwendungen des gleichen Anbieters zu gewährleisten, ist das Single Sign-On Verfahren (SSO). Dieses ermöglicht den Zugang zu mehreren Diensten durch eine einmalige Anmeldung mittels Benutzernamen und Passwort durch das Erstellen von Browser Cookies [8]. Ein bekanntes Unternehmen, das diesen Authentifizierungsmechanismus verwendet, ist Google LLC [9] mit Diensten wie YouTube, Gmail und Google Drive [10].

Der Vorteil des Verzichts auf Anmeldung bei jedem einzelnen Dienst ist gleichzeitig ein Nachteil des SSOs. Der Angreifer erhält durch Kompromittierung eines Kontos in einem SSO-System Zugang zu allen entsprechenden Diensten des Opfers [8].

2.4.2 Technologien im Federated Identity Management

Unter „Federated“ (Föderation) versteht man hier einen Zusammenschluss aus drei miteinander kommunizierenden Parteien (Abbildung 2.1[1]). Anders als bei der zentralen Identität wie in Abbildung 2.2[2] veranschaulicht, besteht das Konstrukt aus dem Benutzer und jeweils einer Anwendungen, die entweder die Identität des Benutzers bestätigt oder bestimmte Dienste anbietet [11]. Basierend auf dieser Kooperationsbeziehung können domänenübergreifend authentifizierte Nutzer untereinander ausgetauscht oder der Zugriff auf Ressourcen weitergeleitet werden. Dies erfordert allerdings von vornherein ein beständiges Vertrauen zwischen den Parteien [12].

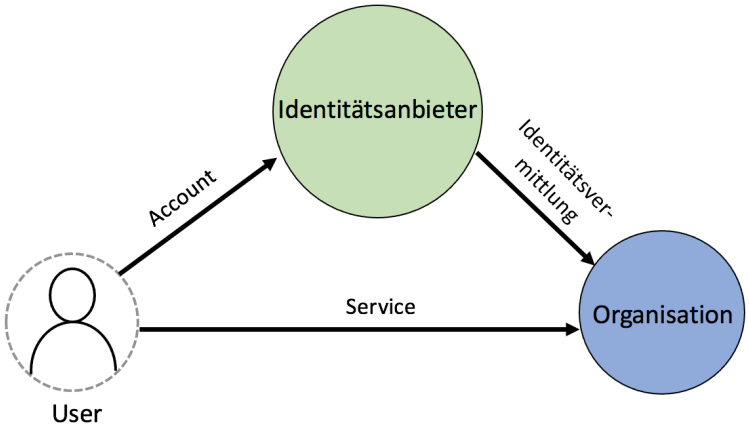

Abbildung 2.1: Föderierte Identität

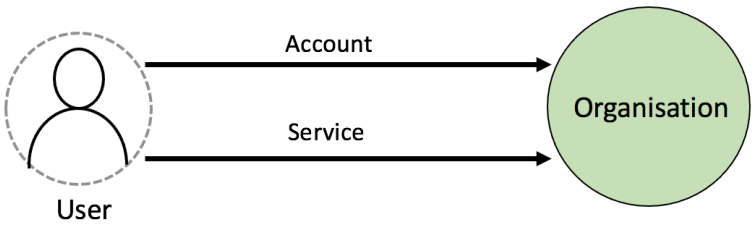

Abbildung 2.2: Zentrale Identität

[1] Eigene Darstellung
[2] Eigene Darstellung

Im Folgenden werden drei Technologien für das Federated Identity Management vorgestellt.

2.4.2.1 SAML

SAML ist ein offener XML-Standard mit dem ein sicherer Austausch von Identitätsinformationen über Domänengrenzen hinweg ermöglicht wird. Der Standard wurde 2005 durch das Konsortium OASIS mit der Version 2.0 finalisiert [13]. Eine Hauptkomponente ist die dort eingesetzte SSO-Funktionalität, die durch eine Interoperabilität zwischen verschiedenen Webservern erweitert wird. Dadurch kann ein bereits registrierter Benutzer einer Webseite seine dort hinterlegten Zugangsdaten für eine andere Organisation zur Authentifizierung oder Autorisierung verwenden. Dieser Prozess kann durch die Verwendung von Pseudonymen, statt direkter Identitätsdaten wie dem Benutzernamen, eine datenschutzfreundliche Anonymität liefern [14]. Ein Nachteil dieses Standards ist jedoch, dass der Benutzer nicht kontrollieren kann, welche und wie viele seiner Daten zwischen den jeweiligen Parteien übertragen werden [15]. Ebenso ist SAML nur für Anwendungen im WWW ausgelegt und nicht für mobile oder native Applikationen gedacht [16].

2.4.2.2 OpenID

Eine weitere Alternative im Federated Identity Management ist das, ebenfalls im Jahre 2005 veröffentlichte, Protokoll OpenID. Dessen Idee zielt darauf ab, jedem Benutzer eine digitale Identität bereitzustellen. Der Ansatz basiert auf einer dezentralen Identität und überlässt dem Identitätshalter die Kontrolle, welche Daten er mit anderen teilen möchte [17]. Jede Identität wird durch eine URL repräsentiert, die auf die vom Benutzer gewählten Identitätsinformationen verweist. Diese Webadresse zeigt entweder auf einen eigenen Server des Identitätshalters oder kann durch die Nutzung eines Identitätsanbieters realisiert werden [18]. Somit ist der Halter im Besitz eines Identifikators und kann sich mit diesem bei vielen verschiedenen Diensten authentifizieren. Als Grundvoraussetzung dient hierbei, dass die Dienste selber eine Anmeldung mittels OpenID anbieten müssen.

Trotz der damals zukunftsweisenden Perspektive der dezentralen Identität, gelang OpenID und auch die zwei Jahre später entwickelten Version OpenID 2.0 [18], kein Durchbruch. Vor allem die fehlende Kompatibilität OpenIDs mit nativen Applikationen, das benutzerdefinierte Signaturschema und ein Datenaustausch ausschließlich mittels XML führte zu einer mangelnden Akzeptanz [16].

2.4.2.3 OAuth

Anders als OpenID definiert OAuth 1.0 nicht nur die Authentifizierung als solches, sondern ergänzend dazu auch einen Mechanismus zur Autorisierung. Das 2007 veröffentlichte Informationsdokument [19] beschreibt, wie eine Anwendung Zugriff auf die Daten des Benutzers erhält, wobei diese Daten von einer anderen Applikation bereitgestellt werden. Dieses Konzept ist vor allem dadurch vorteilhaft, dass sensitive Zugangsdaten des Benutzers in Form eines Tokens[3] zwischen den Parteien versendet wird. Demzufolge muss der Benutzer sein Passwort nur mit der Anwendung, die seine Daten hält, abgleichen. Bei einer erfolgreichen Authentifizierung erfolgt ein Austausch mittels Zugriffstoken und der Verwendung eines vorher vereinbarten Schlüssels zum Schutz der Kommunikation [19].

Die zwei Jahre später überarbeitete Version „OAuth Core 1.0 Revision A“ [21] wird heute noch vom bekannten Mikroblogging Dienst Twitter Inc [22] zum delegationsbasierten Austausch verwendet.

2.4.3 OAuth 2.0

Trotz gemeinsamer Grundsätze unterscheiden sich die Versionen OAuth 1.0 und 2.0 in wesentlichen Punkten. Zumal ist das 2012 veröffentlichte Framework ein offener Internetstandard für die Autorisierung und fällt nicht wie OAuth 1.0 in die Kategorie der informellen Dokumente [20]. Die Komplexität der API-Aufrufe wurde durch den Einsatz von Bearer Token reduziert. Diese werden über eine HTTPS-Verbindung versendet, ohne jede Anfrage einzeln zu signieren, wohingegen OAuth 1.0 jede Anfrage mit komplexen Signaturverfahren erweitert [23]. Des Weiteren wurde das Problem behoben, dass Nutzer die einer Anwendung erteilten Zugriffsrechte nur schwer widerrufen konnten, da unbegrenzt langlebige Zugriffstoken verwendet wurden. Durch zeitlich begrenzte Zugriffstoken und einem dauerhaften Aktualisierungstoken kann der Benutzer ohne Probleme seine Angaben rückgängig machen [24].

Die obsolete Version soll lediglich von bereits existierenden Anwendungen unterstützt werden, neuen Implementierung empfiehlt die Internet Engineering Task Force (IETF) OAuth 2.0 [20].

[3] Zeichenfolge mit in der Regel begrenzter Lebensdauer [20]

2.4.4 OpenID Connect

OpenID Connect (OIDC) ist ein am 26. Februar 2014 veröffentlichter offener Standard. Die Idee des Internetstandards ist, den gesamten Authentifizierungsprozess von der eigentlichen Anwendung zu entkoppeln und die Funktionalität und Sicherheit der Authentifizierung anderen Fachexperten zu überlassen. OIDC wurde von unabhängigen, aber auch unternehmensorientierten Spezialistengruppen wie der Microsoft Corporation und Google LLC entwickelt, die eine führende Rolle in der Entwicklung der progressiven Authentifizierungstechnologie spielen. Die Grundlage bildet das Autorisierungsprotokoll OAuth 2.0 und das Verschlüsselungsprotokoll TLS. Durch die Vereinigung dieser Internetstandards wird eine sichere Infrastruktur für das Authentifizieren von Benutzern und den Zugriff auf externe Systemressourcen gewährt [25].

OIDC hat im Vergleich zur zweiten OpenID-Technologie OpenID 2.0 nur hinsichtlich der architektonischen Konzeption eine Ähnlichkeit. Aus technischer Sicht werden beispielsweise zur Signatur von Token sogenannte JSON Web Token (JWT) eingesetzt. Anders als die benutzerdefinierte Signatur-Funktionalität aus OpenID 2.0, sind JWT einfacher zu implementieren und interoperabler mit anderen Anwendungen. OIDC ist zusätzlich mit nativen und mobilen Anwendungen kompatibel und fokussiert sich nicht mehr ausschließlich auf den webbasierten Datenaustausch wie SAML und OpenID 2.0 [16].

3 Self-Sovereign Identity

Der Begriff „*Identität*“ wird in der Psychologie als das innere Selbst einer Person beschrieben, beeinflusst durch ständige Geschehnisse in der Außenwelt [26]. Laut Christopher Allen, einem Experten in den Bereichen Onlinesicherheit und Blockchain-Technologien und Co-Vorsitzender der W3C Credentials Community Group [27], ist die Identität ein fundamentaler Bestandteil des Menschen. Mit der Betonung auf „*I*“ (Deutsch: Ich) in „*Identity*“ (Deutsch: Identität) soll deutlich gemacht werden, dass einer Person das Bewusstsein um seiner selbst nicht genommen werden kann. In der realen sowie digitalen Welt musste sich der Begriff der Identität allerdings materiellen (z.B. Personalausweis) und immateriellen (z.B. E-Mail-Adresse) Identitätsmerkmalen fügen. Diese können dem Identitätshalter, zum Beispiel durch den Widerruf eines Ausweises von offiziellen Stellen, entzogen werden. Somit hat der Halter keine volle und alleinige Kontrolle über seine Identität, sondern ist auf zentrale Stellen angewiesen [28].

3.1 Modell

Mit dem Konzept der Self-Sovereign Identity (SSI, Deutsch: Selbstbestimmte Identität) erlangt der Benutzer das Eigentum und die Kontrolle über seine Identitätsdaten zurück, ohne dass die Regierung oder eine andere Organisation sie ihm wieder entziehen kann. Dabei spielt der dezentralisierte Ansatz eine zentrale Rolle, denn anders als bei konventionellen Verfahren ist der Benutzer für die Verwaltung seiner Daten zuständig. Somit entfällt die Abhängigkeit von der Verwahrung der Daten bei Drittparteien und verbessert außerdem die Privatsphäre, da die Freigabe von Daten nur für diejenigen erfolgt, die sie verifizieren müssen [29].

Bereits im Jahr 2005 kündigte Kim Cameron, der damalige Architekt für Identitätsprozesse der Microsoft Corporation, zum ersten Mal das Konzept einer selbstbestimmten Identität an und rundete dieses mit 7 Prinzipien, den sogenannten „*Laws of Identity*“, ab [30]:

1. **Benutzerkontrolle und Zustimmung:** Identitätssysteme müssen vertrauenswürdig sein, vor Täuschungen schützen und dürfen nur mit Einverständnis des Benutzers Daten offenlegen.

2. **Minimale Offenlegung für eine eingeschränkte Verwendung:** Das Offenlegen von identifizierbaren Informationen muss auf ein Minimum reduziert und so weit wie möglich anonymisiert werden (z.B. „ist älter als 18 Jahre“ statt dem Geburtsdatum).

3. **Berechtigte Parteien:** Nur Parteien mit berechtigtem und nachvollziehbarem Interesse dürfen Daten erhalten.

4. **Gezielte Identität:** Sowohl omnidirektionale Beziehungen in der Öffentlichkeit als auch direkte Beziehungen zu Konsumenten müssen unterstützt werden. Zum Beispiel muss ein SSID-Broadcast beim WLAN für jeden nach außen hin sichtbar sein, aber auch eine 1:1-Verbindung unterstützen, z.B. bei VPN.

5. **Unterstützung mehrere Betreiber und Technologien:** Das System muss polymorph und polyzentrisch[1] sein, um einen Informationsaustausch zwischen unterschiedlichsten Systemen zu ermöglichen.

6. **Menschliche Integration:** Der Mensch muss als Teil des Systems betrachtet und durch eine eindeutige und möglichst einfache Kommunikation eingebunden werden, um ihn so vor möglichen Angriffen auf seine Identität zu schützen.

7. **Konsistente Erfahrung in verschiedenen Kontexten:** Die Benutzererfahrung muss einfach und konsistent, aber dennoch anpassungsfähig an verschiedene Kontexte sein (z.B. verschiedene Identitäten für den Arbeitsplatz, Webshops, oder andere Aktivitäten).

[1] Mehrere Zentren aufweisend, zu mehreren Zentren gehörend [31]

3.2 Funktionsweise

Selbstbestimmte Identitäten werden häufig in Form von Decentralized Identifiern (DID) und Verifiable Credentials (VC) verwendet. Ein Decentralized Identifer ist ein eindeutiger sowie anonymer Bezeichner für eine Person oder Organisation und ist mit der klassischen Kombination aus Benutzername und Passwort zu vergleichen. Verifiable Credentials wiederum repräsentieren einen bestimmten Identitätsnachweis und enthalten zusätzlich einen DID um den Identitätsnachweis jemanden zuordenen zu können. Sie bilden das Gegenbild eines physischen Berechtigungsnachweises, wie z.B. einem Studentenausweis. Mit der Verbindung dieser beiden Konstrukte kann ein digitaler Identitätsnachweis genau einem Identitätshalter zugeordnet werden.

Die SSI basiert in der Regel auf einer Blockchain oder einer anderen Distributed-Ledger Technologie (DLT). Wie in Abbildung 3.1[2] zu sehen ist, sendet der Nutzer den digital signierten Identifier *DID:1234* einer Organisation zu, um Zugang zu einem bestimmten Service zu erhalten. Um zu verifizieren, dass jener DID dem serviceanfragenden Nutzer gehört, ermittelt die Organisation den zum Identifier *DID:1234* gehörenden öffentlichen Schlüssel aus der Blockchain. Bei einer erfolgreichen Verifizierung kann der Nutzer den entsprechenden Service nutzen.

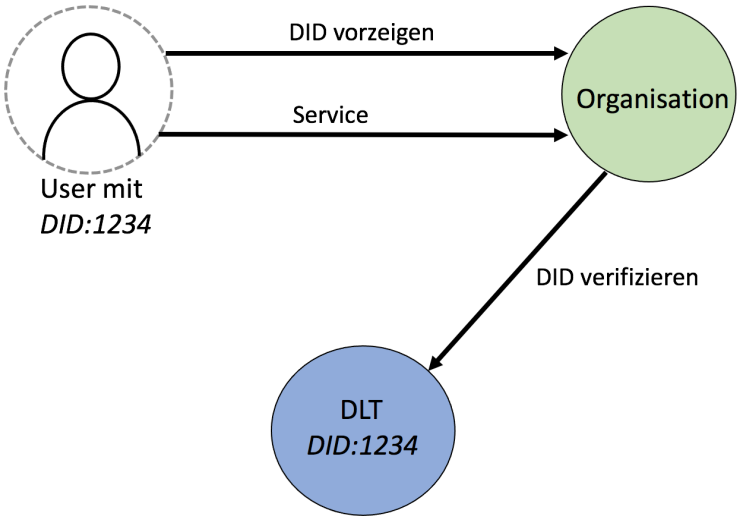

Abbildung 3.1: Selbstbestimmte Identität mit einer DLT

Im Folgenden werden Decentralized Identifier (DID) und ihre Integration in Verifiable Credentials näher vorgestellt.

[2]Eigene Darstellung

4 Decentralized Identifier

Das World Wide Web Consortium (W3C) ist eine internationale Organisation, die unter anderem von Tim Berners-Lee, dem Erfinder des WWW, geleitet wird [32]. Das Ziel des Konsortiums ist das gemeinsame Entwickeln von Webstandards, um die Entwicklung des WWW kontinuierlich zu fördern [33]. Am 07. November 2019 wurde die erste Version des W3C Arbeitsentwurfs bezüglich Decentralized Identifier veröffentlicht [34]. Mit der aktuellen Version vom 02.03.2021 wird darauf hingearbeitet, dass die Phase der „Candidate Recommendation“ (Deutsch: Kandidat-Empfehlung) [35] beendet wird, um mindestens eine von der W3C empfohlene Spezifikation zu werden. In dem Entwurf wird die Kernarchitektur und damit in Verbindung stehende Verfahren vorgestellt, um u.a. DID in Systeme integrieren zu können [36].

4.1 Architektur und Begriffe

Folgend wird ein Überblick der architektonischen Beziehungen zwischen den jeweiligen Bausteinen sowie eine kurze Erklärung dieser gegeben.

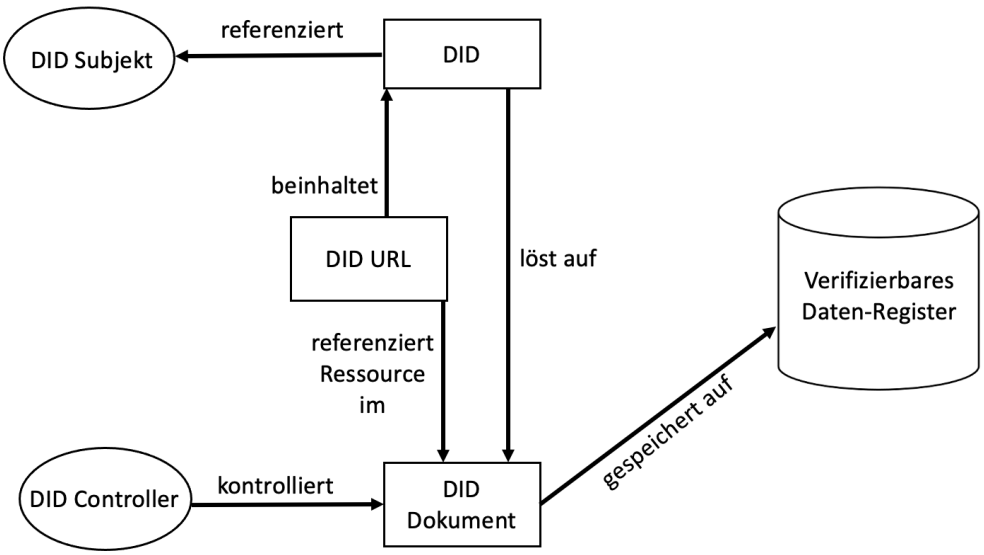

Abbildung 4.1: DID Architektur[1]

DID

Ein Decentralized Identifier ist ein Uniform Resource Identifier (URI), der genau ein DID Subjekt mit einem DID Dokument verbindet.

DID Subjekt

Ein DID Subjekt besitzt keine Eingrenzung bezüglich seiner Identität, sondern kann jegliche Form annehmen (Person, Organisation, Gegenstand, usw.). Ein DID Subjekt wird durch einen DID repräsentiert. Ein Subjekt kann dabei mehrere DID haben, ein DID kann aber nur einem DID Subjekt zugeordnet werden.

DID Dokument

Das DID Dokument enthält weitere Informationen über einen DID, wie z.B. den öffentlichen Schlüssel für Verifizierungszwecke.

DID Controller

Hat die Befugnis Änderungen an einem DID Dokument vorzunehmen. Ein DID Controller kann das zum Dokument gehörende DID Subjekt oder ein anderes berechtigtes Subjekt, wie z.B. eine Organisation sein. Beispielsweise hat ein Elternteil in der Rolle des DID Controllers die Bevollmächtigung, Änderungen an dem DID Dokument des DID Subjekts (des Kindes) vorzunehmen.

DID URL

Ermöglicht durch eine syntaktische Erweiterung des DIDs, den Zugriff auf interne und externe Ressourcen im DID Dokument.

Verifizierbares Daten-Register

Speichert DID und deren DID Dokument in einem Netzwerk, wie z.B. einer Blockchain oder einem System, wie z.B. einer Datenbank.

[1]Original geändert und übersetzt [1][37]

4.2 Aufbau

4.2.1 DID

Ein DID besteht, wie in Abbildung 4.2[2] dargestellt, aus folgenden drei Teilen:

1. Dem zugrunde liegenden Schema (bei Decentralized Identifier nennt sich das Schema „did“).
2. Der DID Methode, die beschreibt, wie z.B. ein DID erstellt oder ein DID Dokument aktualisiert wird.
3. Dem, in der jeweiligen DID Methode, einzigartigen Bezeichner.

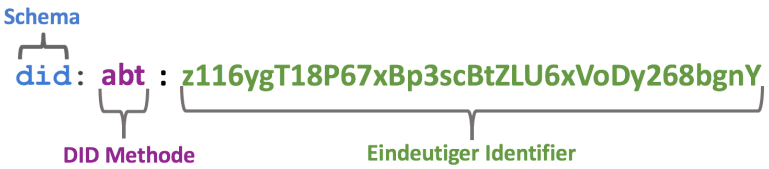

Abbildung 4.2: DID Syntax

4.2.2 DID Dokument

Mit dem Auflösen (Resolution) eines DIDs in ein DID Dokument (näheres in Kapitel 4.4) werden die eigentlichen Daten freigelegt. Ein DID Dokument ist ein textbasiertes Datenformat wie JSON, JSON-LD oder XML. Die Daten in einem DID Dokument beschreiben dabei das korrespondierende DID Subjekt. Durch eine Speicherung des Dokuments auf einer DLT wird zudem eine Interaktion mit dem Subjekt ermöglicht [1]. Es wird vom W3C dringend empfohlen, keine personenbezogenen Daten (PII), z.B. als Query in einer URI, in das DID Dokument einzufügen.

Im Folgenden wird nun das Ergebnis einer DID Resolution vorgestellt. Das DID Dokument in Abbildung 4.3[3] wurde hierbei aus dem DID (Abbildung 4.2) aufgelöst und enthält folgenden Eigenschaften [38]:

- **@context:**
  Festgelegtes Datenaustauschformat, das beide Parteien unterstützen. Bei DID ist die Angabe mindestens einer URI mit dem Wert „https://www.w3.org/ns/did/v1“ *verpflichtend*.

[2]Original geändert und übersetzt
[3]https://dev.uniresolver.io/ mit der Eingabe did:abt:z116ygT18P67xBp3scBtZLU6xVoDy268bgnY

- **id:**
  Um das DID Dokument mit einem DID Subjekt verlinken zu können, ist die Nennung des jeweiligen DIDs *verpflichtend* (DID ist analog zur Abbildung 4.2).

- **service:**
  Service-Endpunkte ermöglichen eine Verknüpfung zwischen dem Subjekt und weiteren Diensten, wie z.B. Datenspeicherdiensten oder sozialen Netzwerken. So kann mittels der Endpunkte, z.B. auf externe personenbezogene Daten verwiesen werden, ohne diese im Dokument veröffentlichen zu müssen. *(optional)*

- **authentication:**
  Mit der Angabe dieser Eigenschaft kann das DID Subjekt beweisen, dass es Träger dieses DIDs ist. *(optional)*

- **publicKey:**
  Ein öffentlicher Schlüssel kann z.B. kryptografisch beweisen, dass dem DID Subjekt ein DID gehört oder dass eine sichere Kommunikation mit Service-Endpunkten gewährleistet werden kann. *(optional, aber empfohlen)*


```
{
  "@context": "https://www.w3.org/2019/did/v1",
  "id": "did:abt:z116ygT18P67xBp3scBtZLU6xVoDy268bgnY",
  "service": [
    {
      "type": "DIDResolver",
      "serviceEndpoint": "https://did.abtnetwork.io"
    },
    {
      "type": "BlockExplorer",
      "serviceEndpoint": "https://explorer.abtnetwork.io"
    }
  ],
  "authentication": [
    {
      "type": "Ed25519SignatureAuthentication2018",
      "publicKey": [
        "did:abt:z116ygT18P67xBp3scBtZLU6xVoDy268bgnY#owner"
      ]
    }
  ],
  "publicKey": [
    {
      "id": "did:abt:z116ygT18P67xBp3scBtZLU6xVoDy268bgnY#owner",
      "type": "Ed25519VerificationKey2018",
      "owner": "did:abt:z116ygT18P67xBp3scBtZLU6xVoDy268bgnY"
    }
  ]
}
```


Abbildung 4.3: Das aus Abbildung 4.2 aufgelöste DID Dokument

4.3 DID Methode

Identifikatoren, die DID Methoden repräsentieren, kennzeichnen den zweiten Teil in einem DID, wie in Abbildung 4.2 veranschaulicht. Jede DID Methode entspricht dabei einer Spezifikation, die bestimmte Anforderungen der DID-Spezifikation des W3Cs erfüllen muss. Folgende Anwendungsszenarien sind, neben weiteren Anforderungen, in den bereits über 80 experimentellen DID Methoden verpflichtend [39][40]:

- DID Controller:
  - Erstellt Decentralized Identifier und deren DID Dokumente
  - Aktualisiert DID Dokumente
  - Deaktiviert Decentralized Identifier

- DID Resolution:
  - Löst ein DID in ein DID Dokument auf

Jede Methode baut unter Umständen auf verschiedenen DLT oder Netzwerken auf, weshalb sich die Methoden in ihrer Implementierung sehr unterscheiden können. Beispielhafte DLT, die in bestimmten DID Methoden verwendet werden, sind die Bitcoin- oder Ethereum Blockchain, IOTA Tangle und Tezos [39].

Im nächsten Kapitel wird eine dieser DID Methoden genauer beschrieben [41][42].

4.3.1 Bitcoin Reference

Die sogenannte „Bitcoin Reference DID Methode“ ist eine der bekanntesten DID Methoden, nicht zuletzt durch die Kryptowährung Bitcoin. Die Methode basiert auf einer öffentlichen Bitcoin Blockchain und wird durch den Namen „btcr“ in dem DID gekennzeichnet (z.B. did:btcr:1234).

4.3.1.1 Aufbau eines DIDs

Der eindeutige Identifier, der den dritten Teil des DIDs darstellt, ist bei dieser Methode eine Referenz auf die exakte Speicherstelle der Bitcoin Transaktion in der Blockchain. Der Identifier enthält dabei Informationen zur Lokalisierung der Transaktion, wie die Art der Blockchain[4], die korrespondierende Blockhöhe und den Transaktionsindex.

[4]Testnet: Blockchain zu Testzwecken, Mainnet: Hauptnetzwerk für echte Transaktionen

4.3.1.2 Erstellen eines DIDs

Eine BTCR DID wird erstellt, indem eine Bitcoin Transaktion getätigt wird. In Abbildung 4.4[5] ist das Modell einer Bitcoin Transaktion zu sehen. Der Input ist die Transaktion selber, die von einer Bitcoin Adresse zu einer anderen übertragen wird. Diese Transaktion kann zudem den optionalen Parameter „OP_RETURN“ enthalten, der anschließend mittels einer HTTP URL auf weitere Daten wie, z.B. ein DID Dokument verweist.

Der genaue DID ist allerdings erst bekannt, nachdem der Block mit der Transaktion bestätigt wird, d.h. an die eigentliche Blockchain-Kette gehängt wird.

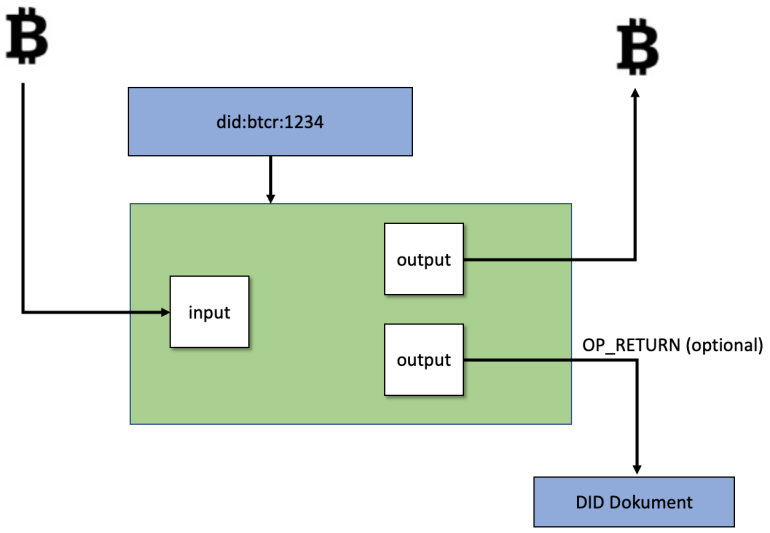

Abbildung 4.4: Modell einer Bitcoin Transaktion

[5]Original minimal geändert [41]

4.4 DID Resolution

Wie bereits in vorherigen Kapiteln erwähnt, wird durch eine DID Resolution ein DID in ein DID Dokument aufgelöst. Dieser Prozess wird durch einen DID Resolver realisiert [1]. Ein DID Resolver kann dabei speziell auf einer DID Methode basieren oder aber durch den „Universal Resolver“ vertreten werden. Der Universal Resolver bietet Schnittstellen zu verschiedenen DID Methoden an und ermöglicht dadurch eine Interoperabilität.

Im Folgenden wird das Konzept des Universal Resolvers der Decentralized Identity Foundation (DIF)[6] kurz vorgestellt.

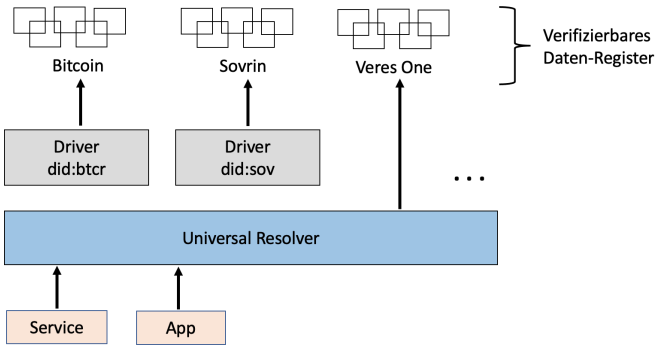

Abbildung 4.5: Aufbau des Universal Resolvers[7]

Erfolgt die Speicherung eines DID Dokuments direkt auf einem verifizierbaren Daten-Register (DLT/Netzwerk), kann der Resolver dieses einfach herunterladen (Vergleich „Veres One“ in Abbildung 4.5). Wenn ein DID Dokument allerdings auf verschiedene Komponenten verweist (wie z.B. bei der BTCR-Methode), muss ein Driver (Deutsch: Treiber) der jeweiligen Methode das Dokument dynamisch erzeugen. Für Services oder Applikationen, die mit Decentralized Identifiern arbeiten, bietet der Universal Resolver eine API an, um diesen Dienst nutzen zu können [43].

Der Universal Resolver befindet sich aktuell in der Entwicklung und unterstützt derzeit durch seine Interoperabilität 42 verschiedene DID Methoden [44][45].

[6] Zu finden unter: https://dev.uniresolver.io/
[7] Abgeänderte Folie der Präsentation: https://ssimeetup.org/did-resolution-given-did-how-do-retrieve-document-markus-sabadello-webinar-13/

4.5 DID Integration in Verifiable Credentials

Der zweite wichtige Baustein der SSI ist ein sogenannter Verifiable Credential (Deutsch: Verifizierbarer Berechtigungsnachweis). Im Gegensatz zu DID wird nicht die Authentizität des Subjekts identifiziert, sondern dessen Merkmale (z.B. das Geburtsdatum, Nationalität, usw.). Ein VC hat genauso wie ein DID ein textuelles Datenformat wie z.B. JSON oder XML. Allerdings wird ein VC zusätzlich, um den zum VC gehörenden DID erweitert, damit dem VC genau ein DID (analog ein Subjekt) zugeordnet werden kann. Mit der Kombination aus DID und VC sowie zusätzlichen technologischen Verfahren kann die Verwendung von digitalen Nachweisen sogar als fälschungssicherer und somit vertrauenswürdiger als ihr physisches Gegenstück eingestuft werden.

Die Spezifikation der W3C zu Verifiable Credentials ist vor allem auf die Privatsphäre der Benutzer ausgerichtet und enthält zusätzliche Empfehlungen diese durch geeignete Verhaltens- und Funktionsweisen weiter zu erhöhen [46].

4.5.1 Architektur und Begriffe

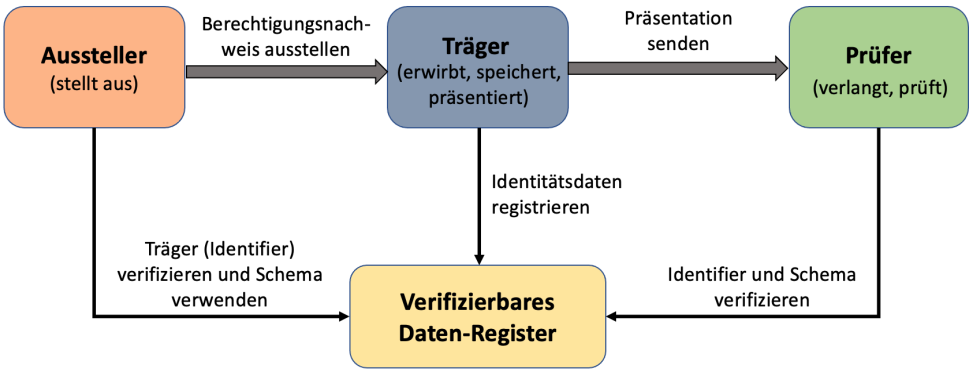

Abbildung 4.6: Austausch von Identitätsdaten[8] (näheres in Kapitel 4.6.4)

Subjekt

Über ein Subjekt werden gewisse Behauptungen gemacht, wie z.B.: „Ist ein Student“. Ein Subjekt kann jegliche Form annehmen (Person, Organisation, Gegenstand, usw.).

Träger

Besitzt ein oder mehrere Berechtigungsnachweise und kann daraus eine Präsentation erstellen, die dann wiederum an einen Prüfer versendet wird. Ein Träger kann das Subjekt selber sein oder eine übergeordnete Rolle besitzen, z.B. Träger ist die Mutter, Subjekt das Kind.

[8] Original geändert und übersetzt [47][37]

Aussteller

Hält gewisse Informationen über das Subjekt und kann sie ihm in einem Berechtigungsnachweis ausstellen. Beispiele für Arten von Ausstellern sind Organisationen, Einzelpersonen, Regierungen, usw.

Prüfer

Verifiziert ein oder mehrere Berechtigungsnachweise, die von einem Träger an ihn übermittelt werden. Es besteht eine unidirektionale Vertrauensbeziehung vom Prüfer zum Aussteller.

Verifizierbares Datenregister

Ein Distributed Ledger, eine Datenbank, oder ein Verbund aus mehreren davon, der bestimmte Informationen speichert, wie beispielsweise öffentliche Schlüssel zur Verifizierung von Berechtigungsnachweisen.

Schema

Ein bestimmter Aufbau (Schablone) des Credentials, z.B. ein Personalausweis muss bestimmte Informationen enthalten (Name, Geburtsort, usw.).

4.5.2 Verifiable Credential

Ein Credential (Deutsch: Berechtigungsnachweis) identifiziert ein oder mehrere Merkmale eines Subjekts und ist genau auf einen Träger zurückzuführen. Die Abbildung 4.7[9] stellt den Aufbau eines Verifiable Credentials dar [48].

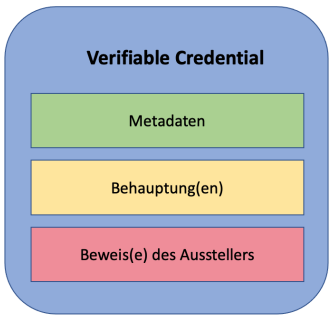

Abbildung 4.7: Struktur eines Verifiable Credentials

- **Metadaten:** Eigenschaften des Credentials, wie z.B. das Ausstellungsdatum oder eine Referenz auf den Aussteller. (beide verpflichtend)
- **Behauptung(en):** Über das Subjekt gemachte Aussagen, wie z.B. „Alice ist Studentin an der Universität X“.

[9] Original geändert und übersetzt [48][37]

- **Beweis(e):**
  - Wert des eigentlichen kryptografischen Beweises vom Aussteller
  - Metadaten über den kryptografischen Beweis (bei einer digitalen Signatur wären dies z.B. Art der Signatur, Erstellungsdatum, usw.)

4.5.2.1 Widerruf eines VC

Eine Person besitzt einen VC mit der Behauptung, ein Student an einer Universität zu sein. Mit Abschluss des Studiums ist diese Behauptung allerdings nicht mehr gültig. Für solche Fälle existieren deshalb folgende Möglichkeiten, einen VC zu widerrufen:

- Angabe eines Verfalldatums im DID Dokument des VCs.
- Speicherung eines festgelegten Widerrufsformulars auf der Blockchain, das die eindeutige ID des VCs enthält (Verifizierer ist verpflichtet, dies bei der Verifizierung des VC mit einzubeziehen.).

4.5.3 Verifiable Presentation

In einer Verifiable Presentation (VP) wie in Abbildung 4.8[10] dargestellt, werden Daten einer oder mehrerer VCs oder Teile davon zusammengefasst. Dies ermöglicht dem Halter alle relevanten Daten zusammenzustellen, z.B. für eine berufliche VP, ohne dabei weitere Daten preisgeben zu müssen. Die einzelnen VCs in einer VP beziehen sich dabei auf das gleiche Subjekt, können aber von unterschiedlichen Ausstellern erstellt worden sein.

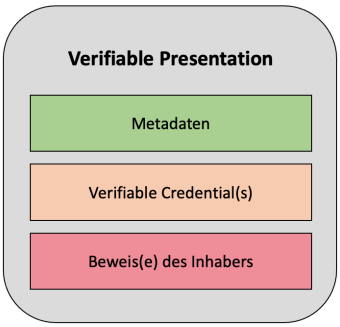

Abbildung 4.8: Struktur einer Verifiable Presentation

Eine VP wird immer dann erstellt, wenn der Träger diese an einen Verifizierer versenden möchte. In Abbildung 4.9[11] ist eine vom Träger signierte VP zu sehen, die einen VC mit

[10] Original geändert und übersetzt [49][37]
[11] Original minimal geändert [50]

einem Studentenstatus enthält. Die digitale Signatur des Trägers beweist, dass er der Besitzer der VP ist und schützt außerdem vor einem Replay-Angriff[12][49].

```
{  // Metadaten
   "@context": [
      "https://www.w3.org/2018/credentials/v1",
      "https://www.w3.org/2018/credentials/examples/v1"
   ],
   "type": "VerifiablePresentation",
   // Verifiable Credential
   "verifiableCredential": [{
      "@context": [
         "https://www.w3.org/2018/credentials/v1",
         "https://www.w3.org/2018/credentials/examples/v1"
      ],
      // eindeutige ID des VCs
      "id": "http://example/credentials/1872",
      "type": ["VerifiableCredential", "AlumniCredential"],
      "issuer": "did:example:university",
      "issuanceDate": "2021-01-01T19:73:24Z",
      "credentialSubject": {
         "id": "did:example:subject",
         "alumniOf": {
            "id": "did:example:university",
            "name": "Example University",
         }
      },
      // Digitale Signatur der Universität
      "proof": {
         "type": "RsaSignature2018",
         "verificationMethod": "did:example:university",
         "jws": "eyJhbGciOiiOlsiYjY0Il19..TCYt5X sITJX1CxPCT8yAV-T"
      }
   }],
   // Digitale Signatur des Inhabers
   "proof": {
      "type": "RsaSignature2018",
      "verificationMethod": "did:example:subject",
      "jws": "eyJhbGciOiJNyaXQiOlsiYjY0Il19..kTCYt5XsITJX1CxPCT8yAV-T78"
   }
}
```

Abbildung 4.9: Beispiel einer VP

[12] Vorher abgegriffene Daten des Opfers werden vom Angreifer genutzt, um eine fremde Identität vorzutäuschen

4.6 Anwendungsbeispiele

Im Folgenden werden die in den Anwendungsbeispielen verwendeten Begriffe genauer erklärt sowie bereits im Vorfeld gegebene Voraussetzungen für die Beispiele vorgestellt.

4.6.1 Begrifflichkeiten

Identity Owner

Besitzt die volle Kontrolle über den DID oder den VC. Kann eine Person, DID Subjekt, Organisation, usw. sein. (analog Träger)

Relying Party

Eine Person, Organisation oder Sache die den Identity Owner mittels eines DIDs authentifiziert bzw. seine Identitätsdaten verifiziert (analog Prüfer).

Challenge

Beliebiges Format einer Aufgabe, die der Identity Owner nachkommen muss, um etwas zu beweisen, z.B. in Form eines Buttons mit „Sign in mit DID Auth“, QR Codes, usw.

digitaler Wallet

Hier eine App auf dem Smartphone, die den privaten Schlüssel des Identity Owners, sowie dessen DID und VCs speichert. Eine Relying Party kann mittels Challenge mit der App kommunizieren.

4.6.2 Voraussetzungen

Folgende Punkte gelten für die kommenden Anwendungsbeispiele. Es wurde bereits ein DID mit einer beliebigen DID Methode erstellt. Zudem ist ein digitaler Wallet auf einem Smartphone installiert. In dieser App befindet sich der erzeugte DID und der dazugehörige private Schlüssel. Der dazugehörende öffentlichen Schlüssel ist im DID Dokument auf der Blockchain gespeichert.

4.6.3 Authentifizieren mit einem DID

Die im vorherigen Kapitel dargestellten Grundlagen zu Decentralized Identifiern werden nun in einem Fallbeispiel veranschaulicht. Es handelt sich bei diesem Beispiel um eine Authentifizierung mit einem DID. Eine DID Authentifizierung (DID Auth) basiert auf einem Challenge-Response-Verfahren und wird derzeit von 10 verschiedenen Architekturen unterstützt. Diese Architekturen sind dabei für unterschiedliche Anwendungsbeispiele (z.B. verschiedene Endgeräte) konzipiert. Beispiele hierfür sind:

- **Architektur 1:** Webseite und mobile App
- **Architektur 2:** Mobile Webseite und mobile App
- **Architektur 9:** HTTP Signaturen

Die **schwarzen Pfeile** in Abbildung 4.10[13] bilden die Basis DID Authentifizierung und werden in dem Beispiel um die orangenen Pfeile, die *Architektur 1* darstellen, erweitert. In der hier vorgestellten Architektur nutzt die Relying Party einen QR-Code, der alle relevanten Informationen enthält, um eine Challenge (oder alternativ Authentifizierung) mit dem Identity Owner zu initiieren. Dieser übermittelt seine Antwort anhand eines HTTP POST Requests zurück an die Relying Party [51].

Mittels dieser Architektur wird nun ein Szenario zwischen einer Pseudo-Bank Webseite und einem Benutzer (im Folgenden „Bob“ genannt) und dessen digitalen Wallet vorgestellt. Der Vorgang beginnt mit dem Aufrufen der Bankseite durch Bob im Browser. Es erscheint ein Button zur Authentifizierung, auf den Bob anschließend klickt.

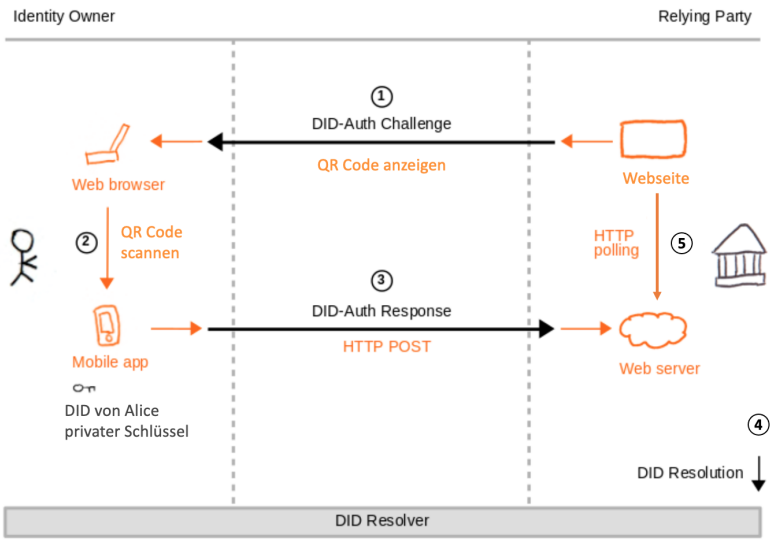

Abbildung 4.10: Architektur einer DID Authentifizierung

[13] Original geändert und übersetzt [51][52]

1. *Die Bankseite sendet eine Challenge an Bob*
    - Die Challenge erscheint in Form eines QR Codes auf der Webseite.
2. *QR Code scannen*
    - Bob scannt mit seinem digitalen Wallet den QR Code und erhält eine Anfrage, ob seine Daten von der App verwendet werden dürfen (DID, dazugehöriger privater Schlüssel).
3. *App sendet Antwort an Bank*
    - Die Challenge wird mit dem zum DID gehörenden privaten Schlüssel signiert und mit dem DID an die Bank mittels HTTP POST zurückgesendet.
4. *Bank verifiziert die digitale Signatur*
    - Durch die Antwort von Bob hat die Bank die signierte Challenge sowie den DID.
    - Die Bank führt eine DID Resolution mit dem DID durch.
    - Nach Erhalt des öffentlichen Schlüssels aus dem DID Dokument kann die digitale Signatur von Bob mit diesem verifiziert werden.
5. *HTTP Polling durch die Bankseite*
    - Die Bankseite sendet in regelmäßigen Zeitabständen einen Request an den Web Server, um zu erfahren, ob Bob erfolgreich verifiziert wurde.
    - Sobald dies der Fall ist, ist Bob erfolgreich mit seinem DID authentifiziert und erhält Zugang zu der Bankseite.

4.6.4 Austausch eines Verifiable Credentials

Das zweite Fallbeispiel ist eine Erweiterung des Decentralized Identifiers um einen Verifiable Credential. In der zugrundeliegenden Abbildung 4.11[14] möchte der User (im Folgenden „Alice“ genannt) einen Rabatt bei einer Online-Buchhandlung erhalten. Dieser Rabatt ist allerdings nur für Studenten vorgesehen, weswegen sich Alice einen Verifiable Credential mit dieser Anforderung bei ihrer Universität ausstellen lassen muss. Der öffentliche Schlüssel der Universität ist bereits auf der Blockchain in einem DID Dokument gespeichert.

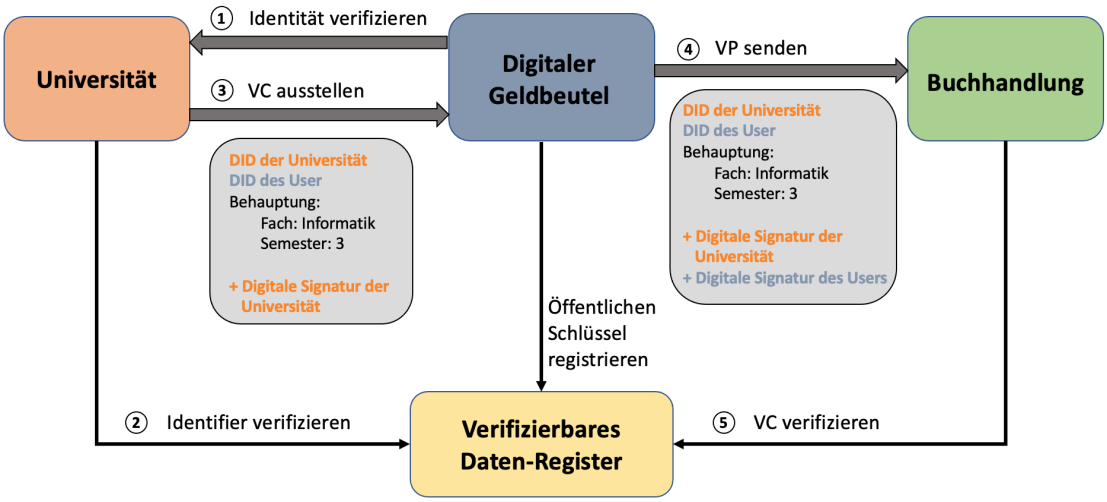

Abbildung 4.11: Architektur eines Austauschs von Verifiable Credential

1. *Identität verifizieren*
    - Alice besucht ihre Universitätswebseite und authentifiziert sich bei dieser (Login).
    - Danach scannt sie mit dem digitalen Wallet eine Challenge in Form eines QR Codes und übermittelt der Universität die signierte Challenge sowie den DID, auf welchen der VC ausgestellt werden soll.
2. *Identifier verifizieren*
    - Die Universität verifiziert den DID von Alice.
3. *Verifiable Credential ausstellen*

[14] Eigene Darstellung

- Die Universität stellt Alice den VC aus und übermittelt diesen in Form eines QR Codes an sie.
- Der VC enthält alle notwendigen Informationen, die die Relying Party (Buchhandlung) für die Verifizierung benötigt.

4. *VC senden*
   - Alice signiert den VC mit ihrem privaten Schlüssel und übermittelt die daraus resultierende VP an die Buchhandlung.
5. *VC verifizieren*
   - Die Buchhandlung muss nun jeweils die digitale Signatur von Alice und von der Universität verifizieren.
   - Sobald die Verifizierung erfolgt ist, erhält Alice ihren Rabatt.

5 Auswirkungen auf den Datenschutz

Das Speichern von Daten auf der Blockchain sowie der Austausch von VPs, unterliegen bei einer nicht sachgerechten Entwicklung gewissen Einschränkungen der Privatsphäre. Im Folgenden werden die damit verbundenen Probleme, sowie mögliche Lösungsmöglichkeiten näher erläutert.

5.1 Datenschutz-Grundverordnung für die Blockchain

Die Datenschutz-Grundverordnung (DSGVO) gewährt natürlichen Personen in Form eines Grundrechts eine Mitbestimmung bei der Verarbeitung ihrer personenbezogenen Daten [53]. Unter personenbezogenen Daten versteht m an all jene Merkmale, die eine natürliche Person[1] direkt (z.B. durch den Name oder Telefonnummer) oder indirekt (z.B. durch Standortdaten, Arbeitszeiten) identifizieren können [54]. Aufgrund gewisser Eigenschaften der Blockchain ist die Einhaltung der Vorgaben der DSGVO mit Schwierigkeiten verbunden. So hat laut Artikel 17 jede natürliche Person das „Recht auf Vergessenwerden“. Einhergehend mit der Eigenschaft der Unveränderlichkeit der Daten auf einer Blockchain, ist die Umsetzung dieses Rechts problematisch [53]. Zusätzlich sorgt die Transparenz einer öffentlichen Blockchain, wie z.B. der Bitcoin Blockchain dafür, dass jede Person Daten einsehen kann. In einem DID Dokument auf der Blockchain werden keine persönlichen Daten (wie z.B. ein VC) gespeichert, nichtsdestotrotz können unter Umständen personenbezogene Daten verarbeitet werden [55].

Im Folgenden werden Prinzipien beim Umgang mit Decentralized Identifiern erläutert, um einen Personenbezug zu verhindern.

5.1.1 Prinzipien der Privacy by Design

Mit „Privacy by Design“ werden Datenschutz-Prinzipien bereits bei der Entwicklung von Systemen integriert [56]. Die Spezifikation der W3C zu Decentralized Identifiern empfiehlt jedem Entwickler, diese Prinzipien anzuwenden, um für eine datenschutzkonforme Architektur zu sorgen. Als nächstes erfolgt ein Ausschnitt dieser Prinzipien beim Umgang mit personenbezogenen Daten [57]:

[1] Natürliche Person: Träger von Rechten und Pflichten

- **Persönliche Daten privat halten:** Vor allem bei einer öffentlichen Blockchain sollten keine personenbezogenen Daten in einem DID Dokument vorhanden sein.
  - Verschlüsselung von Daten auch nicht sinnvoll, da sie möglicherweise durch neue Technologien (z.B. Quantencomputer) gehackt werden können (Entfernen der Daten anschließend nicht möglich).
- **Korrelationsrisiko für DID minimieren:** Für jeden VC-Austausch sollte ein anderer DID verwendet werden, um so einen Zusammenhang zum DID Subjekt vermeiden zu können.
- **Korrelationsrisiko innerhalb der DID Dokumente minimieren:** Zusätzlich zur Verwendung verschiedener DID sollten die Werte bestimmter Eigenschaften, z.B. öffentliche Schlüssel oder Service-Endpunkte, unterschiedlich sein, um eine Korrelation zu vermeiden.

5.2 Reduzierung der Datenoffenlegung

Die in Kapitel 5.1.1 genannten Korrelationsrisiken, zeigen ein weiteres Problem beim Umgang mit personenbezogenen Daten. Neben den auf der Blockchain liegenden anonymen Daten, werden sensible Daten (wie z.B. „ist Student an der Universität X“) zudem auch mit Relying Parties geteilt. Um dem Endnutzer auch in diesem Rahmen eine Verbesserung der Privatsphäre zu ermöglichen, werden nun drei Strategien vorgestellt. Diese werden durch alltagsnahe Beispiele von den Personen „Diego“, „Selena“ und „Proctor“ veranschaulicht.

5.2.1 Data Minimization

Der Benutzer „Diego“ möchte einen Dienst auf seinem Smartphone nutzen. Um diesen Dienst verwenden zu können, ist die Freigabe seines Standorts notwendig. Diego kommen Zweifel bezüglich der Verwendung seines Standorts: *Was wird mit seinen Daten gemacht? Werden sie mit Dritten, ohne seine Zustimmung, geteilt? Gibt es eine Möglichkeit, seinen Standort nur einmal zu teilen oder nur einen ungefähren Ort anzugeben?*

Mittels Data Minimization (Datenminimierung) werden Daten auf das Nötigste reduziert. Dabei werden nur Daten geteilt, welche für eine bestimmte Anwendung erforderlich sind. Eine Speicherung der Daten beim Verifizierer soll außerdem nur solange wie notwendig stattfinden [58]. Bezüglich Verifiable Credentials bedeutet Datenminimierung folgendes [59][58]:

- **Ein Aussteller** stellt nur die Informationen in einem VC bereit, welche vom Verifizierer benötigt werden. Diese Daten lassen sich dabei noch weiter reduzieren, indem der Aussteller:
  - Abstrakte Attribute verwendet (z.B. „InDeutschland“)
  - Die Daten auf einzelne VCs aufteilt (z.B. je ein VC für Wohnort und Land)
  - Signatur-Schemata einsetzt, um Selective Disclosure zu ermöglichen (näheres in Kapitel „Selective Disclosure“)
- **Ein Verifizierer** legt beim Umgang mit Daten beispielsweise das Höchstmaß an benötigten Daten sowie die Dauer ihrer Verwahrung fest. Dies erhöht die Privatsphäre des Benutzers und minimiert die Haftung des Verifizierers bei sensiblen Daten.

5.2.2 Progressive Trust

In einem Gespräch mit einem Immobilienmakler möchte „Proctor“ ein Haus erwerben. Im Zuge der Verhandlung muss Proctor sein Kreditlimit offenlegen. Proctor würde gerne nur einen ungefähren Betrag nennen, besitzt allerdings nur einen Bescheid mit seinem genauen Kreditlimit. Der Makler besteht darauf, die Authentizität des Briefes zu überprüfen und nimmt den Brief entgegen. *Proctor wäre es lieber gewesen, wenn der Makler bei der Prüfung der Authentizität bei seiner Bank nicht auch sein Kreditlimit erfahren hätte.*

Beim Progressive Trust (Schrittweises Vertrauen) wächst die Menge der offengelegten Daten mit dem Vertrauen in den Verifizierer. Im Falle von Proctor wäre es sinnvoller gewesen, in einem ersten Schritt prüfen zu können, ob ihm die Bank überhaupt einen Kredit gewährt. Bei einem weiteren positiven Verlauf des Gesprächs hätte sich Proctor entscheiden können, weitere Daten, wie sein Kreditlimit, preiszugeben. Bezogen auf VCs bedeutet dies [58]:

- **Ein Aussteller** kann ein VC so gestalten, dass schrittweises Vertrauen möglich ist. Eine Option wäre zum Beispiel das Ausstellen von mehreren einzelnen VCs. Diese werden anschließend so zu einer Verifiable Presentation kombiniert, dass die optimale Datenmenge preisgegeben wird.
- **Ein Verifizierer** muss ein Vertrauensverhältnis schaffen, indem er nur jene Daten abfragt, die für einen bestimmten Schritt unbedingt nötig sind.

5.2.3 Selective Disclosure

Vor einer Diskothek überprüft ein Türsteher den Führerschein von „Selena“, um ihr Alter zu überprüfen. Selena hat das Gefühl, dass er dabei nicht nur ihr Alter, sondern auch ihre Adresse mustert. *Gibt es eine andere Möglichkeit zu beweisen, dass sie über 18 Jahre alt ist, ohne dass weitere Daten preisgegeben werden müssen?*

Selective Disclosure (Selektive Offenlegung) ist ein weiteres Mittel, um eine Datenminimierung zu erreichen. Hierbei entscheidet ein Individuum durch fein-granulares Aufteilen seiner Daten darüber, welche Daten er mit jemandem teilen möchte. Realisiert wird dies im VC-Umfeld wie folgt [58]:

- **Ein Aussteller** stellt sicher, dass ein VC eine bestimmte Struktur hat, damit eine selektive Offenlegung gewährleistet werden kann. Hier können, wie bei der Datenminimierung, einzelne VCs mit jeweils einem Attribut ausgestellt werden. Zusätzlich kann der VC so angepasst werden, dass der Zugriff auf Daten durch den im nächsten Kapitel erläuterte „Zero-Knowledge Proof“, weiter eingeschränkt werden kann.
- **Ein Verifizierer** formuliert die Anfrage so, dass eine selektive Offenlegung unter der Verwendung von kryptografischen Verfahren unterstützt wird.

Die selektive Offenlegung ist ein Bestandteil des im nächsten Kapitel vorgestellen Zero-Knowledge Proofs.

5.2.3.1 Datenanonymisierung durch Zero-Knowledge Proof

Der Zero-Knowledge Proof (Deutsch: Kenntnisfreier Beweis) ist ein kryptografisches Verfahren, wodurch ein Verifizierer beweisen kann, dass er ein Geheimnis, nicht aber dessen eigentlichen Wert kennt. Der mathematische Hintergrund des Zero-Knowledge Proofs wird im Rahmen dieser Arbeit nicht weiter behandelt, weshalb nur Beispiele erläutert werden. Einem Identitätsträger werden dabei folgende datenschutzfreundliche Möglichkeiten durch den Zero-Knowledge Proof ermöglicht:

1. Kombination verschiedener VC zu einer einzigen VP, ohne die Identität des Ausstellers preiszugeben.

   → Verhindert manipulative Absprachen zwischen Aussteller und Verifizierer

2. Selektive Maskierung einzelner Attribute innerhalb eines VCs.

   → Vermeidet die Ausstellung mehrerer VCs mit jeweils einzelnen Attributen (Selective Disclosure)

3. Erzeugen eines VCs nach dem Datenschema (für eine Ableitung) des Verifizierers und nicht des Ausstellers. Dabei werden Abfragen beim Verifierer unterstützt, die nicht konkrete Daten preisgeben, sondern abgeleitete wie z.B. anstelle des Geburtstags, „ist älter als 18 Jahre“.

   → Ein Aussteller muss nach der Erstellung eines ableitungsfähigen VC nicht mehr mit einbezogen werden

Im Falle von Selena kann statt des Vorzeigens des gesamten Führerscheins eine VP verwendet werden die beweist, dass sie älter als 18 Jahre ist (analog Punkt 3.) [47].

6 Aktueller Entwicklungsstand

Dieses Kapitel gibt einen Überblick darüber, welche Produkte für die hier vorgestellten Themen bereits in Entwicklung sind. Zunächst wird eine DLT vorgestellt, die von einem Konsortium renommierter Unternehmen und Institutionen aus dem öffentlichen und privaten Sektor entwickelt wird. Im Anschluss werden zwei Wallets eingeführt, die jeweils eine DID Authentifizierung oder den Austausch von VCs ermöglichen.

6.1 Dezentrales Netzwerk „IDunion“

Das Projekt „IDunion“, mit dem ursprünglichen Namen „SSI für Deutschland“ [60], ist ein Wegbereiter der SSI auf dem europäischen Markt. Das Projekt wird durch ein Konsortium von öffentlichen sowie privaten Unternehmen, wie der Bundesdruckerei GmbH, Siemens AG und der Commerzbank AG, entwickelt. Das Ziel ist der Aufbau eines blockchain-basierten selbstbestimmten Identitätssystem, welches es Nutzern ermöglicht über ihre Identitätsdaten selbst zu bestimmen. Die DLT besteht derzeit aus 15 Knoten und unterstützt die in dieser Bachelorarbeit vorgestellten Standards des W3Cs. Im Jahr 2023 soll dieses Projekt soweit fortgeschritten sein, dass eine Adaption im europäischen Markt stattfinden kann [61].

6.2 DID Authentifizierung mit SelfKey

Das Startup „SelfKey“ entwickelt Identitätssysteme und hat es sich zum Ziel gesetzt, diese mit Fokus auf die SSI und der Blockchain zu verbessern [62]. Mit Hilfe des digitalen Wallets von SelfKey wird eine Authentifizierung mit Ethereum-basierten DID bei Webseiten ermöglicht. Damit eine DID-Kommunikation zwischen Webseiten und mobilen Endgeräten gewährleistet werden kann, bietet das Startup eine Dokumentation an, um den Authentifizierungsmechanismus in Webseiten zu integrieren. Der Wallet selber speichert DID, kann allerdings auch die digitale Währung Ether verwalten. Aktuell ist der Wallet nur als Desktop-Anwendung verfügbar, soll allerdings im nächsten Schritt auch auf mobilen Endgeräten bereitgestellt werden [63].

6.3 Wallet für digitale Identitäten

In einer Kooperation mit dem in Kapitel 6.1 vorgestellt Projekt IDunion und dem Software-Anbieter „lissi“ wurde das gleichnamige Forschungsprojekt lissi im Juni 2019 gegründet. Das Projekt fokussiert sich auf zwei SSI-Innovationen: Einem Wallet und einer Anwendung für das Ausstellen und Verifizieren von Identitätsdaten. Der Wallet unterstützt die Konzepte der Datenminimierung und -anonymisierung durch die Umsetzung der selektiven Offenlegung und des Zero-Knowledge Proofs. Aktuell ist der Wallet als Testversion auf mobilen Endgeräten verfügbar, soll allerdings im Jahr 2021 als fertiges Produkt erscheinen. Die Anwendung, das zweite Produkt des Forschungsprojekts, ist eine Kombination aus den in dieser Arbeit vorgestellten Rollen des Ausstellers und Verifizierers. Sie wird als eine White-Label-Lösung[1] für Unternehmen und Institute vorgestellt [64][65].

Als Basis nutzen beide Produkte die DLT des IDunion und das im Rahmen dieser Arbeit nicht weiter vorgestellte Sovrin-Netzwerk [66][67].

[1] Produkte/Dienstleistung eines Herstellers werden unter verschiedenen Namen vermarktet, der Hersteller tritt dabei aber nicht in Erscheinung

7 Prototyp

Dieses Kapitel beschreibt die prototypische Umsetzung eines digitalen Wallets. Da es sich hier nur um einen minimalistischen Prototyp handelt, soll lediglich ein Fallbeispiel den Austausch mit einem Verifiable Credential veranschaulichen. Dafür werden die Teilnehmer des in Kapitel 4.6.4 vorgestellten Anwendungsbeispiels verwendet. Bevor auf den konkreten Prototypen näher eingegangen wird, wird das grundlegende Konzept eines digitalen Wallets vorgestellt.

7.1 Konzept eines Wallets

Analog zu einem physischen Kartenetui oder Portemonnaie speichert ein digitaler Wallet Identitätskarten (wie z.B. den Führerschein) in Form von Datenstrukturen auf einem Endgerät. Ein digitaler Wallet unterliegt dabei gewissen Anforderungen [68]:

- **Anforderung 1:** Der Wallet soll ein Schlüsselpaar[1] sicher aufbewahren können, um kryptografische Beweise durchführen zu können (z.B. digitale Signaturen).
- **Anforderung 2:** Der Wallet soll eine VC signieren können, um dem Verifizierer zu beweisen, dass der Träger auch Eigentümer der VP ist.
- **Anforderung 3:** Der Wallet soll nur für den Träger zugänglich sein, um zu vermeiden dass sensible Daten in falsche Hände geraten.
- **Anforderung 4:** Der Wallet soll VCs empfangen können, nachdem der Aussteller diese erstellt hat.
- **Anforderung 5:** Der Wallet soll VCs speichern können, um mehrere VCs in einer gekapselten Anwendung zu sichern.
- **Anforderung 6:** Der Wallet soll in der Lage sein, VCs mit einem Verifizierer zu teilen, um z.B. die jeweilige Berechtigung mit diesem zu erhalten.
- **Anforderung 7:** Der Wallet soll fähig sein, eine Selective Disclosure für VCs zu ermöglichen, um die Offenlegung weiterer Identitätsdaten zu verringern.

[1] privater und öffentlicher Schlüssel

Aus den hier genannten Anforderungen wurden für die Implementierung des Prototyps nur die **Anforderungen**: **4**, **5** und **6** realisiert. Das Ziel des Prototyps besteht darin, das Grundprinzip eines digitalen Geldbeutels und den damit zusammenhängenden Parteien, wie dem Aussteller und dem Verifizierer, zu verdeutlichen.

7.2 Konzeptionelle Modellierung

In diesem Kapitel werden die im Prototyp vorhandenen Rollen sowie der eigentliche Anwendungsfall anhand eines Sequenzdiagramms erläutert.

7.2.1 Rollen

Wie bereits erwähnt spiegelt der Prototyp die Kommunikationspartner im Anwendungsbeispiel in Kapitel 4.6.4 wider. Diese Parteien werden im Folgenden mit ihrer Funktion im Prototyp vorgestellt:

- **Der Aussteller** ist eine Universität, die einem ihrer Studierenden einen VC ausstellt, der deren Immatrikulation bestätigt.
- **Der digitale Wallet** speichert den VC der Universität.
- **Der Verifizierer** ist eine Buchhandlung die einen Rabatt für Studierende anbietet.

7.2.2 Anwendungsfall

Alice Doe möchte in einer Buchhandlung einen Rabatt von 30% erhalten. Dieser Rabatt ist aber nur für Studierende verfügbar. Um zu beweisen, dass sie Studentin ist, besorgt sie sich einen VC von ihrer Universität mit der Behauptung[2], dort Studentin zu sein.

Der Kommunikationsaustausch der in Kapitel 7.2.1 vorgestellten Parteien, kann den jeweiligen Parteien im Anwendungfall folgend zugeordnet werden: Aussteller (Virtuohm als Universitätsseite), digitaler Wallet (Alice als Nutzerin), Verifizierer (Buchhandlung). Abbildung 7.1[3] stellt eine schrittweise Kommunikation des Ablaufs dar und modelliert zugleich eine mögliche Test-Reihenfolge des Prototyps:

[2] siehe Kapitel 4.5.2 zu dem Begriff „Behauptung“
[3] Eigene Darstellung

1. Alice meldet sich bei ihrem digitalen Wallet an, um eine Session für den Austausch eines VCs zu starten.
2. Sie besucht ihre Universitätsseite und lässt sich einen VC ausstellen, der sie als eingeschriebene Studentin ausweist.
3. Ein von der Universität signierter VC wird an ihren Wallet übermittelt.
4. Optionale Möglichkeit: Wallet neu laden, um den VC einzusehen.

   → Das zugehörige DID Dokument kann durch einen Klick auf den VC angezeigt werden
5. Alice besucht die Seite der Buchhandlung und verifiziert sich mit ihrer signierten VP.
6. Sie erhält den Rabatt von 30%.

   → Die signierte VP kann im unteren Bereich der Seite eingesehen werden

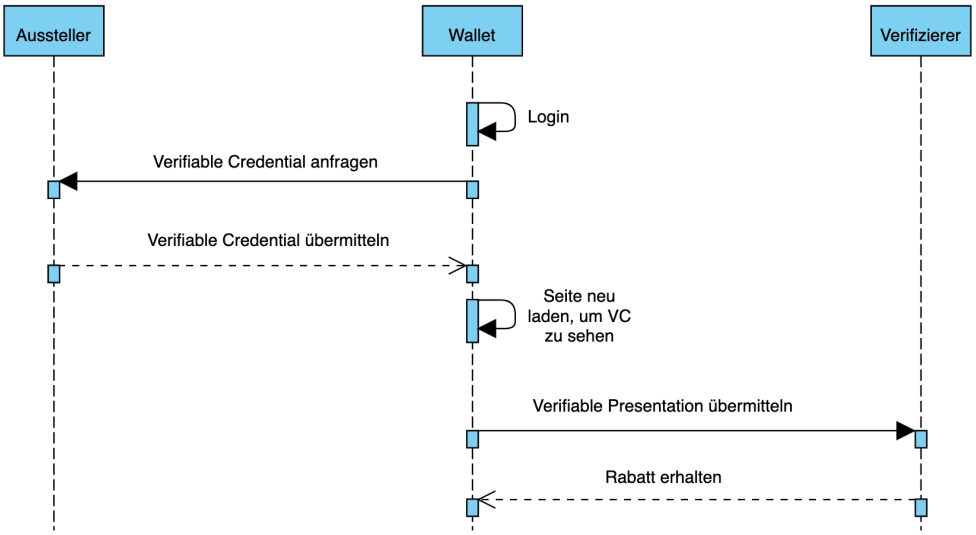

Abbildung 7.1: Ablauf eines Austausch von einem VC

7.3 Implementierung

Im Folgenden wird die Technologie für die Entwicklung und die Funktionsweise des Prototyps erklärt. Im Anschluss daran folgt eine Auflistung der vorgenommenen Änderungen am bereits bestehenden Projekt der Firma Digital Bazaar.

7.3.1 Technologien

Die Funktionalität des Prototyps baut auf der Skriptsprache JavaScript auf und wird durch HTML und CSS veranschaulicht. Als Webserver dient die JavaScript-Laufzeitumgebung „Node.js“.

7.3.2 Funktionsweise

Der Prototyp basiert auf einem Open Source Projekt der Firma Digital Bazaar und enthält bereits die drei Kommunikationspartner aus Kapitel 7.2.1. Die Funktionalitäten, welche für die **Anforderungen 4**, **5** und **6** in Kapitel 7.1 erforderlich sind, sind bereits vorhanden.

Das Zusammenspiel der Kommunikationspartner im Prototyp soll einen rudimentären Austausch eines VCs simulieren, um lediglich das Konzept zu vermitteln. Die vom W3C propagierten Empfehlungen bezüglich Sicherheit und Datenschutz haben für diesen Prototypen keine Relevanz, da es nur um das grundlegende Konzept geht. Die VCs werden im Hintergrund durch keinen Signaturmechanismus signiert, sondern durch einen Pseudo-Wert im DID Dokument vertreten. Das DID Dokument ist im Vorfeld erstellt worden und wird dementsprechend durch keinen DID Resolver generiert. Eine tatsächliche Verifizierung der Signatur mit dem öffentlichen Schlüssel auf einer Blockchain findet in diesem Prototyp nicht statt.

7.3.2.1 Änderungen

Der Aussteller, zu Beginn eine eher rudimentäre Seite, wurde durch eine realitätsnahe Webseite ersetzt, um den Vorgang ansprechender und nachvollziehbarer zu gestalten. Als Universitätsbeispiel dient hier die Technische Hochschule Nürnberg und deren VirtuOhm-Seite. Zusätzlich wurde ein zur Webseite passendes Feld eingebettet, um einen VC anfordern zu können. Innerhalb des VCs wurden Daten vereinfacht um einen schnelleres Verständnis zu ermöglichen (z.B. „did:example:subject“ statt „did:example:ebfeb1f-712ebc6f1c276e12ec21“). Neben der Vereinfachung des VCs mussten auch Daten aus dem VC entfernt werden, da diese nicht korrekt waren. Beispielsweise wurde der Typ des VCs als eine VP definiert, was in diesem Schritt der Kommunikation allerdings nicht korrekt ist.

Im Wallet kann das zum VC gehörige DID Dokument, durch einen Klick auf den VC, angezeigt werden. Vorher wurde der VC auf der Austeller-Seite im unteren Bereich angezeigt.

Der Verifizierer wurde aus den gleichen Gründen wie der Aussteller, durch eine realitätsnahe Webseite ersetzt. Die Buchhandlung orientiert sich dabei am Design einer großen Buchhandelskette. In dieses wurde zusätzlich eine Anzeige für den Rabatt, sowie ein Feld um diesen zu beantragen, eingefügt. Analog zum VC wurde die VP hier ebenso vereinfacht und ausgebessert wie, z.B. das Hinzufügen einer Signatur des Trägers.

8 Schlussbetrachtung

In der folgenden Schlussbetrachtung werden die wesentlichen Kernpunkte der Arbeit zusammengefasst. Als Abschluss dient ein Ausblick auf mögliche Themen, die in folgenden Arbeiten untersucht werden könnten.

8.1 Zusammenfassung

Die vorliegende Arbeit thematisiert zwei dezentrale Verfahren, um den Umgang mit Identitätsdaten in der digitalen Welt zu vereinfachen und dem Nutzer mehr Privatsphäre zu ermöglichen. Ziel dieser Arbeit ist es zu zeigen, dass das Konzept der Decentralized Identifier dem Benutzer, z.B. bei einer Anmeldung an einer Webseite mehr Privatsphäre gewährt als zentrale Authentifizierungsverfahren wie zum Beispiel dem SSO. Dies kann erreicht werden, indem keine sensiblen Daten des Benutzers auf den Servern einer Webseite gespeichert werden, sondern lokal in einem Wallet des Benutzers. Des Weiteren wurde die Integration von DID in Verifiable Credentials vorgestellt. Mit einer Kombination dieser beiden dezentralen Konzepte können physische Nachweise wie z.B. ein Führerschein, digitalisiert werden.

Dies ermöglicht dem Benutzer, seine Identitätsnachweise sicher in seinem Wallet, analog einem physischen Kartenetui, aufzubewahren. Ebenso wurde untersucht inwieweit das Speichern von Daten eines Benutzers auf einer öffentlichen Blockchain, um beispielsweise eine VC zu verifizieren, negative Auswirkungen auf den Datenschutz haben könnte. Dafür wurden Prinzipien aufgestellt wie der Empfehlung, keine persönlichen Daten auf einer Blockchain zu speichern. Neben diesen Möglichkeiten einen Personenbezug durch die Blockchain zu vermeiden, wurden weitere Strategien vorgestellt, die es gestatten den Informationsaustausch mit Dritten auf ein Minimum zu beschränken. Eine mögliche Strategie ist hierbei die Datenminimierung, welche z.B. durch den Einsatz von Selective Disclosure (nur ausgewählte Daten werden an Relying Parties übermittelt), erreicht wird. Eine weitere Strategie ist die Verwendung des Zero-Knowledge Proofs, der es ermöglicht eine Daten-Abstraktion durchzuführen, wodurch einer Relying Parties, z.B. nicht das genaue Geburtstagsdatum, sondern beispielsweise nur ein „älter als“ mitgeteilt wird. Unter Verwendung dieser Strategien kann der Benutzer selbst entscheiden was und wieviel von seiner Identität preisgegeben werden soll. Im Anschluss daran wurde

ein dezentrales Netzwerk und zwei digitale Wallets vorgestellt, die sich derzeit noch in Entwicklung befinden und dabei den aktuellen Entwicklungsstand der in dieser Arbeit vorgestellten Themen widerspiegeln. Zuletzt wurden Anforderungen für einen digitalen Wallet definiert und drei davon in einem Prototypen umgesetzt. Diese Anforderungen konnten insofern realisiert werden, als dass die Ausstellung eines VCs, das Speichern in einem Wallet und die Freigabe mit Relying Parties mit diesem VC ermöglicht wurde.

8.2 Ausblick

Der Fokus der Arbeit lag darin, das grundlegende Konzept dieser neuen dezentralisierten Verfahren vorzustellen. Da noch keine Standardisierung der einzelnen Komponenten wie z.B. der DID Authentifizierung, DID Resolution oder den jeweiligen DID Methoden vorliegt, wurde nur der jetzige Stand dieser Entwicklung dargestellt. Darauf aufbauend könnte in folgenden Arbeiten tiefer auf die jeweiligen Komponenten eingegangen werden. Vor allem die Verwendung verschiedener DLT in den DID Methoden, erweitert den Anwendungsbereich der Blockchain-Technologie nicht nur auf Kryptowährungen, sondern auch auf digitale Identitäten. Ein Vergleich verschiedener DID Methoden, könnte deshalb auch weitere Einblicke in das Zusammenspiel unterschiedlicher DLT und Identitätsnachweise liefern. Neben einer möglichen Vertiefung bestimmter Kapitel dieser Arbeit, könnten auch Anwendungsfälle wie der Verlust des privaten Schlüssels untersucht werden. Dieses Szenario kann im realen Leben durchaus vorkommen und würde dem Identitätsträger die Kontrolle über seinen DID nehmen.

Als Abschluss ist zu sagen, dass vor allem in der heutigen Zeit das Thema der digitalen Identitäten immer mehr in den Fokus rückt. Physische Nachweise haben zwar ihre Vorteile, sind den digitalen Nachweisen allerdings aufgrund deren kryptografischen Eigenschaften teilweise unterlegen. Nicht zuletzt wurde durch das Gesundheitsministerium die Entwicklung eines digitalen Impfnachweises durch Unternehmen wie IBM und dem Startup Ubirch, verkündet. Dieser bringt dezentrale Aspekte der SSI mit und setzt wie DID und VCs auf eine Verifizierung mittels DLT. Nichtsdestotrotz enthält dieses Projekt zwar dezentrale Komponenten, entspricht jedoch nicht den hier vorgestellten Empfehlungen der W3C. Die Verifizierung des Impfnachweises erfolgt beispielsweise nicht über eine digitale Signatur des Trägers, sondern verlangt weiterhin einen physischen Nachweis, wie z.B. einen Personalausweis. Dennoch ist ein Schritt in Richtung SSI zu erkennen und kann möglicherweise in Zukunft auch in anderen Bereichen, wie z.B. in einen Wahlvorgang bei der Bundestagswahl, integriert werden [69].

Literatur

[1] *Decentralized Identifiers (DIDs) v1.0.* https://www.w3.org/TR/did-core/. [Eingesehen am 10.11.2020].

[2] *Beware of criminals pretending to be WHO.* https://www.who.int/about/communications/cyber-security. [Eingesehen am 10.11.2020].

[3] *adorsys - Unternehmen.* https://adorsys.com/de/ueber-uns/unternehmen. [Eingesehen am 27.12.2020].

[4] *Internet X.509 Public Key Infrastructure.* https://tools.ietf.org/html/rfc2510#section-1. [Eingesehen am 08.04.2021].

[5] *Die Blockchain-Technologie - Potenziale und Herausforderungen in den Netzsektoren Energie und Telekommunikation.* https://www.bundesnetzagentur.de/SharedDocs/Downloads/DE/Allgemeines/Bundesnetzagentur/Publikationen/Berichte/2019/DiskussionspapierBlockchain.pdf. [Eingesehen am 09.04.2021].

[6] D. Florencio und C. Herley. „A Large-Scale Study of Web Password Habits“. In: *Proceedings of the 16th International Conference on World Wide Web.* [Eingesehen am 02.01.2021].

[7] C. Katsini, M. Belk, C. Fidas, N. Avouris und G. Samaras. „Security and Usability in Knowledge-Based User Authentication: A Review“. In: *Proceedings of the 20th Pan-Hellenic Conference on Informatics.* [Eingesehen am 30.12.2020].

[8] Y. Wilson und A. Hingnikar. „Single Sign-On“. In: *Solving Identity Management in Modern Applications: Demystifying OAuth 2.0, OpenID Connect, and SAML 2.0.* [Eingesehen am 11.02.2021].

[9] *Einmalanmeldung (SSO) mit Google als Identitätsanbieter einrichten.* https://support.google.com/a/topic/7556794?hl=de&ref_topic=7556686. [Eingesehen am 11.02.2021].

[10] *Unsere Produkte.* https://about.google/intl/ALL_de/products/. [Eingesehen am 11.02.2021].

[11] *“Baseline capabilities for enhanced global identity management trust and interoperability”. Draft New Recommendation ITU-T X.1250.* https://www.itu.int/rec/T-REC-X.1250-200909-I. [Eingesehen am 19.02.2021].

[12] S. K. Heribert M. Anzinger Kay Hamacher. „2.1 4-Parteien Modell sowie Terminologie“. In: *Schutz genetischer, medizinischer und sozialer Daten als multidisziplinäre Aufgabe*. [Eingesehen am 19.02.2021].

[13] *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0*. `https://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf`. [Eingesehen am 12.02.2021].

[14] *Security Assertion Markup Language (SAML) V2.0 Technical Overview*. `https://www.oasis-open.org/committees/download.php/27819/sstc-saml-tech-overview-2.0-cd-02.pdf`. [Eingesehen am 12.02.2021].

[15] S. K. Heribert M. Anzinger Kay Hamacher. „2.2 SAML – Das Sicherheits-HTML“. In: *Schutz genetischer, medizinischer und sozialer Daten als multidisziplinäre Aufgabe*. [Eingesehen am 13.02.2021].

[16] *What is the history of OpenID?* `https://openid.net/connect/faq/`. [Eingesehen am 14.02.2021].

[17] *What is OpenID?* `https://openid.net/what-is-openid/`. [Eingesehen am 13.02.2021].

[18] S. K. Heribert M. Anzinger Kay Hamacher. „3.1 OpenID“. In: *Schutz genetischer, medizinischer und sozialer Daten als multidisziplinäre Aufgabe*. [Eingesehen am 14.02.2021].

[19] *The OAuth 1.0 Protocol*. `https://tools.ietf.org/html/rfc5849`. [Eingesehem am 19.02.2021].

[20] *The OAuth 2.0 Authorization Framework*. `https://tools.ietf.org/html/rfc6749`. [Eingesehen am 19.02.2021].

[21] *OAuth Core 1.0 Revision A*. `https://oauth.net/core/1.0a/`. [Eingesehen am 19.02.2021].

[22] *Twitter Authentication*. `https://developer.twitter.com/en/docs/authentication/oauth-1-0a`. [Eingesehen am 19.02.2021].

[23] *Authentication and Signatures*. `https://www.oauth.com/oauth2-servers/differences-between-oauth-1-2/authentication-and-signatures/`. [Eingesehen am 19.02.2021].

[24] *Short-lived tokens with Long-lived authorizations*. `https://www.oauth.com/oauth2-servers/differences-between-oauth-1-2/short-lived-tokens-long-lived-authorizations/`. [Eingesehen am 20.02.2021].

[25] *The OpenID Foundation Launches the OpenID Connect Standard*. `https://openid.net/2014/02/26/the-openid-foundation-launches-the-openid-connect-standard/`. [Eingesehen am 20.02.2021].

[26] *Identität*. https://dorsch.hogrefe.com/stichwort/identitaet. [Eingesehen am 26.02.2021].

[27] *Christopher Allen*. https://www.linkedin.com/in/christophera. [Eingesehen am 26.02.2021].

[28] *The Path to Self-Sovereign Identity*. http://www.lifewithalacrity.com/2016/04/the-path-to-self-soverereign-identity.html. [Eingesehen am 26.02.2021].

[29] A. Satybaldy, M. Nowostawski und J. Ellingsen. „Self-Sovereign Identity Systems". In: *Privacy and Identity Management. Data for Better Living: AI and Privacy: 14th IFIP WG 9.2, 9.6/11.7, 11.6/SIG 9.2.2 International Summer School, Windisch, Switzerland, August 19–23, 2019, Revised Selected Papers*. [Eingesehen am 27.02.2021].

[30] *The Laws of Identity*. https://www.identityblog.com/stories/2005/05/13/TheLawsOfIdentity.pdf. [Eingesehen am 27.02.2021].

[31] *Duden - polyzentrisch*. https://www.duden.de/rechtschreibung/polyzentrisch. [Eingesehen am 26.04.2021].

[32] *FACTS ABOUT W3C*. https://www.w3.org/Consortium/facts. [Eingesehen am 04.03.2021].

[33] *W3C MISSION*. https://www.w3.org/Consortium/mission. [Eingesehen am 04.03.2021].

[34] *W3C First Public Working Draft 07 November 2019*. https://www.w3.org/TR/2019/WD-did-core-20191107/. [Eingesehen am 04.03.2021].

[35] *W3C Working Draft 02 March 2021*. https://www.w3.org/TR/2021/WD-did-core-20210302/#proving. [Eingesehen am 05.03.2021].

[36] *7 W3C Recommendation Track Process*. https://www.w3.org/2004/02/Process-20040205/tr.html. [Eingesehen am 05.03.2021].

[37] *W3C DOCUMENT LICENSE*. https://www.w3.org/Consortium/Legal/2015/doc-license. [Eingesehen am 04.04.2021].

[38] *5. DID Documents*. https://www.w3.org/TR/2019/WD-did-core-20191107/#did-documents. [Eingesehen am 12.03.2021].

[39] *8.2 Method Operations*. https://www.w3.org/TR/did-core/#method-operations. [Eingesehen am 17.03.2021].

[40] *12. DID Methods*. https://w3c.github.io/did-spec-registries/#did-methods. [Eingesehen am 17.03.2021].

[41] *DIDs Demystified*. https://ssimeetup.org/dids-demystified-hands-intro-dids-btcr-did-method-kim-hamilton-duffy-webinar-5/. [Eingesehen am 23.04.2021].

[42] *BTCR DID Method*. https://w3c-ccg.github.io/didm-btcr/. [Eingesehen am 17.03.2021].

[43] *DID Resolution: Given a DID how do I retrieve its document?* https://ssimeetup.org/did-resolution-given-did-how-do-retrieve-document-markus-sabadello-webinar-13/. [Eingesehen am 19.03.2021].

[44] *Universal Resolver*. https://github.com/decentralized-identity/universal-resolver. [Eingesehen am 19.03.2021].

[45] *Universal Resolver Warning*. https://dev.uniresolver.io/. [Eingesehen am 19.03.2021].

[46] *Verifiable Credentials Data Model 1.0*. https://www.w3.org/TR/vc-data-model. [Eingesehen am 21.03.2021].

[47] *Verifiable Credentials Data Model 1.0*. https://www.w3.org/TR/vc-data-model/. [Eingesehen am 04.04.2021].

[48] *3. Core Data Model*. https://www.w3.org/TR/vc-data-model/#core-data-model. [Eingesehen am 24.03.2021].

[49] *3.3 Presentations*. https://www.w3.org/TR/vc-data-model/#presentations. [Eingesehen am 25.03.2021].

[50] *A simple example of a verifiable presentation*. https://www.w3.org/TR/vc-data-model/#example-2-a-simple-example-of-a-verifiable-presentation. [Eingesehen am 26.04.2021].

[51] *Introduction to DID Auth*. https://github.com/WebOfTrustInfo/rwot6-santabarbara/blob/238be6d91a8929696bba90cefa7af1a67a1a3bbd/final-documents/did-auth.md. [Eingesehen am 26.03.2021].

[52] *LICENSE-CC-BY-4.0.md*. https://github.com/WebOfTrustInfo/rwot1-sf/blob/41ce5f49107629bb2db1899e0da8aa5929985d14/final-documents/LICENSE-CC-BY-4.0.md. [Eingesehen am 04.04.2021].

[53] *VERORDNUNG (EU) 2016/679 DES EUROPÄISCHEN PARLAMENTS UND DES RATES*. https://eur-lex.europa.eu/legal-content/DE/TXT/HTML/?uri=CELEX:32016R0679. [Eingesehen am 01.04.2021].

[54] *Artikel 4, EU DS-GVO, "Begriffsbestimmungen"*. https://www.privacy-regulation.eu/de/artikel-4-begriffsbestimmungen-DS-GVO.htm. [Eingesehen am 1.04.2021].

[55] *Blockchain und Datenschutz - Faktenpapier*. https://www.bitkom.org/sites/default/files/file/import/180502-Faktenpapier-Blockchain-und-Datenschutz.pdf. [Eingesehen am 01.02.2021].

[56] *Privacy by Design - The 7 Foundational Principles*. https://iapp.org/media/pdf/resource_center/pbd_implement_7found_principles.pdf. [Eingesehen am 01.04.2021].

[57] *10. Privacy Considerations*. https://www.w3.org/TR/did-core/#privacy-considerations. [Eingesehen am 01.04.2021].

[58] *Engineering Privacy for Verified Credentials*. https://github.com/WebOfTrustInfo/rwot5-boston/blob/fcbc33835c1b76b7526a8a82cd9cac9c23828711/final-documents/data-minimization-sd.pdf. [Eingesehen am 03.04.2021].

[59] *7.8 The Principle of Data Minimization*. https://www.w3.org/TR/vc-data-model/#the-principle-of-data-minimization. [Eingesehen am 03.04.2021].

[60] *"SSI für Deutschland" Konsortium startet dezentrales Identitäts-Netzwerk*. https://idunion.org/2020/08/31/ssi-fuer-deutschland-konsortium-startet-dezentrales-identitaets-netzwerk-2/. [Eingesehen am 07.04.2021].

[61] *An ecosystem for trusted identities*. https://www.hslu.ch/-/media/campus/common/files/dokumente/i/community-im-gespraech/idunion.pdf. [Eingesehen am 07.04.2021].

[62] *We Are SelfKey*. https://selfkey.org/about-us/. [Eingesehen am 23.04.2021].

[63] *Welcome to the SelfKey API*. https://selfkeyfoundation.github.io/selfkey-integration-docs. [Eingesehen am 23.04.2021].

[64] *FAQ*. https://lissi.id/faq. [Eingesehen am 24.04.2021].

[65] *Aktuell unterstützen zwei Wallets das IDUnion Netzwerk*. https://idunion.org/projekt/. [Eingesehen am 24.04.2021].

[66] *Lissi Wallet - Beta*. https://play.google.com/store/apps/details?id=io.lissi.mobile.android.beta. [Eingesehen am 24.04.2021].

[67] *Lissi INSTITUTIONAL AGENT*. https://lissi.id/institutions. [Eingesehen am 24.04.2021].

[68] *Digital Credential Wallets*. https://github.com/WebOfTrustInfo/rwot7-toronto/blob/d2b9833b562d62edf0f6dd94792ce165133776c0/draft-documents/Digital%20Credential%20Wallet.md. [Eingesehen am 19.04.2021].

[69] *Coronavirus - Fragen und Antworten zum digitalen Impfnachweis*. https://www.bundesgesundheitsministerium.de/coronavirus/faq-covid-19-impfung/faq-digitaler-impfnachweis.html. [Eingesehen am 25.04.2021].