





# Privacy by Design Identity Architecture Using Agents and Digital Identities

Kalman C. Toth<sup>1</sup>  , Ann Cavoukian<sup>2</sup>, and Alan Anderson-Priddy<sup>3</sup>

<sup>1</sup> Sovereign Image, Portland Oregon, USA

kalmancloth@gmail.com

<sup>2</sup> Global Privacy and Security by Design Centre, Toronto, Canada

ann.cavoukian@gpsbydesign.com

<sup>3</sup> Portland State University, Portland Oregon, USA

andersonpriddy@protonmail.com

**Abstract.** Today's web is comprised of a patchwork of identity solutions because neither identity nor privacy were designed-in when it was created. This paper proposes an integrative identity architecture that satisfies the principles of privacy by design from inception. Comprised of identity agents and digital identities that are tightly held by their owners, the architecture decentralizes control over identity from providers to users. Owners can manage their digital identities and private data such that liability risks are reduced for service providers without compromising ease-of-use. Identity agents and digital identities enable owners to prove who they are when required, protect their private and identifying data, and securely collaborate. Digital identities are virtualized to look and behave like credentials found in one's wallet thereby facilitating technology adoption and reducing dependency on remote access passwords. A gestalt privacy by design process has been used to discover and validate the architecture's privacy requirements and design elements, systematically reasoning about how the design satisfies the requirements. The process can be applied to organically improve the architecture and create a reference model for open source development. This paper also relates the architecture to W3C's models for verifiable credentials and decentralized identifiers, summarizes the architecture's features, capabilities and benefits, and suggests areas for further study.

**Keywords:** Privacy by design · Identity · Authentication · Verification · Security

## 1 Introduction

The integrative identity architecture described herein addresses the alarming growth in lost privacy, identity theft, impersonation and fraud due to breaches and identity theft over the web. The identity architecture has been motivated by the numerous identity and privacy challenges Internet stakeholders are grappling with every day. The work has also been stimulated by the European Union's General Data Protection Regulation (eugdpr.org) which addresses these and other serious privacy problems.

The described architecture deploys *identity agents* that work on behalf of Internet device owners to tightly control and manage *digital identities* encapsulated therein. Identity agents enable consumers and providers to prove their identities, verify digital identities of other owners, and protect their private data and identifying information. Synthesized from prior art, the proposed architecture elaborates the principles of *Privacy by Design* as articulated by Ann Cavoukian [1] wherein she describes the goals of minimizing the disclosure of private information and data collection; protecting private information; securing transactions end-to-end; enabling express consent to access owner data; and establishing privacy as the system default setting. Our architecture also addresses anonymous and pseudo-anonymous forms of digital identity, ease-of-use, usability, and interoperability. We acknowledge Europe's related legal tradition in the field of data protection by design.

## 2 Challenges Addressed

The Internet's problems with identity, privacy and security have been revealed in recent years by large-scale breaches such as those of Capital One, the Marriott, Sony, Target, JP Morgan, Home Depot, Yahoo, Equifax, Facebook, and Google. Critical root causes include latent vulnerabilities in web services and browsers, and the huge volume of private data collected by providers to sustain their business models and password provisioning. Many providers are honeypots for hacking and malware.

Online web access suffers from many weaknesses. Widely acknowledged, the Internet is far too dependent on passwords. Users create and reuse weakly specified passwords that are easy to crack while validating questions are easy to guess. Identifying information is routinely used to name online accounts thereby helping unscrupulous actors create bogus accounts and impersonate. Biometrics, geo-location schemes and behavioral analytics have marginally elevated identity assurances.

A critical area lacking attention is third-party identity proofing. Banks, license bureaus, passport issuers and enterprises routinely proof the identity of their customers, citizens and employees before issuing attested credentials. To date, little has been written about incorporating proofing and attestation into web-based identity solutions.

Sir Timothy Berners-Lee, inventor of the world-wide-web, has expressed serious concern about privacy, especially as it relates to the Facebook scandal which allowed the private data of millions of users to be misused by political operatives. He has resolved to take back power from the big Internet players by giving users control over their private and identifying information [2].

## 3 Relevant Background

In 2005, Kim Cameron [3] said that the web is comprised of a patchwork of identity schemes. This suggests that inadequately and inconsistently deployed identity provisioning is a critical root cause of many of the Internet's privacy issues.

To tackle the web's privacy problems, Ann Cavoukian [1, 4] advocates Privacy by Design (PbD) where, among other things, privacy is proactive, preventative and embedded in the design; disclosure and data collection are minimized; privacy and

security are simultaneously enhanced; end-to-end security thwarts surveillance; consent to access private data is expressly delegated; visibility, transparency and accountability are paramount; people control their privacy; and privacy is a system's default setting.

George Tomko and Cavoukian [5] propose applying artificial intelligence (AI) and machine learning (ML) to create cognitive agents to secure private data, control disclosure, delegate consent, and de-identify collected information.

Various writers have advocated moving away from server-controlled identity provisioning to single sign-on, federated, and decentralized (user controlled) identity schemes. Christopher Allen [6] and the Sovrin Foundation [7] have proposed leveraging blockchain technology to enable *self-sovereign digital identities* where users control their identities and central authorities have no control over them.

The World Wide Web Consortium (W3C) has been developing [8] an identity data model for specifying “verifiable credentials” (VCs) that can be deployed such that verifiable credentials are cryptographically secure, privacy respecting, and machine verifiable. The model enables multiple methods including digital signatures and zero-knowledge-proofs to verify the integrity and authenticity of verifiable credentials.

The W3C has also been developing a model for decentralized identifiers (DIDs) [9] which are independent of centralized registries, identity providers and certificate authorities. Using distributed ledger (blockchain) technology or other forms of decentralized networks, discrete entities (people, organizations, things) can register and thereby control one or more distinct DIDs across given contexts. Each DID is associated with a DID Document specifying cryptographic material, verification methods (e.g. digital signature), and endpoints used to prove that the entity controls the DID.

In the 2015 timeframe our founders began to develop the identity architecture described herein. Our original design concept was comprised of collaborating identity engines (identity agents) managing e-credentials (self-sovereign digital identities) on behalf of their owners (people and things). The architecture's identity agents decentralize control from central authorities to individual persons by tightly binding them to their digital identities and public/private keys thereby elevating non-repudiation strength and identity assurances associated with their digital identities.

## 4 Decentralizing Digital Identity: Shifting Control to Users

A critical vulnerability of server-centric identity schemes is that service provider repositories become honeypots for identity theft because of the enormous volume of private and identifying information such systems collect. The identity architecture described in this paper addresses this problem by shifting control over digital identity from service providers to users, that is, decentralizing identity.

Decentralization is realized by deploying identity agents that strongly bind owners to digital identities used to identify the owner, interoperate with other identity agents and applications, and proof and attest the identities of other owners.

Rather than agreeing to service provider requests for more private data than necessary, decentralization enables users to create and control digital identities that specify elements of their private and identifying information that they can subsequently use instead of passwords to prove who they are. This reduces how much information providers need

to safeguard mitigating breach risk. Decentralizing identity disperses the attack surface across many users and devices reducing hacking and malware risks.

## 5 The Identity Landscape

Figure 1 positions popular identity solutions across the identity landscape in terms of:

- A. What one knows - e.g. passwords and PINs,
- B. What one holds - a device,
- C. What one is - biometric authenticators,
- D. What one asserts - attributes, claims, images, and
- E. What others attest - in response to identity proofing and knowledge.

Group **A** includes online web services, single sign-on (SSO) systems and federated identity schemes that use remote access passwords and/or PINS to authenticate users. Group **B** depicts messaging solutions where peers use software apps installed on their personal platforms to collaborate securely. Group **C** shows authenticators leveraging biometrics and other schemes to authenticate holders and secure channels between users and web services. Group **D** depicts the World Wide Web Consortium’s emerging Verifiable Claims model (W3C VC) [8] for specifying machine-readable digital identities, and the W3C DID model for specifying decentralized identifiers [9]. Group **E** includes



Fig. 1. Identity landscape

web-based systems where the identities of online users are proofed and attested by third parties using physical documents and web resources. The W3C VC and W3C DID models are covered by both **E** and **D** because third parties can elect to proof and attest verifiable credentials and decentralized identifiers.

None of the solutions depicted in Fig. 1 cover all five dimensions. They leave gaps in the identity landscape and exhibit certain operational shortcomings.

## 5.1 Covering the Gaps in the Identity Landscape

The following scenarios describe how the architecture addresses certain of these gaps:

- a. Most users today specify distinct online user profiles and passwords for the web services they use - manually updating them as circumstances and needs demand. This represents a serious pain-point for most users. The identity agents of the proposed identity architecture addresses this by enabling users to explicitly control, assert, maintain, register and share digital identities that are subsequently employed to prove who they are and secure their private data and transactions. Since digital identities supported by the proposed identity architecture are virtualized physical credentials, maintaining them will be a relatively intuitive task for most users.
- b. Messaging applications such as PGP, Signal and WhatsApp use distinct cryptographic methods and protocols to secure messages. The identity agents of the proposed identity architecture implement an application programming interface (API) that such apps can use to secure messages exchanged between collaborating users.
- c. Selected enterprise systems today reduce or eliminate password usage by deploying personally held authenticators (e.g. FIDO, WebAuthn) employing biometrics and/or other such schemes to cryptographically bind users to designated web services. The proposed architecture leverages native and tethered authenticators to deploy digital identities that can be intuitively selected and used to cryptographically bind the owner to such web services - potentially across multiple sites.
- d. Certain web applications, including social networks, deploy dissimilar password provisioning and reset procedures; implement liberal data collection and information sharing policies; and collect significant volumes of private and identifying data. The proposed architecture deploys identity agents and digital identities on personally held devices that enable owners to assert and control how much identifying and private data they disclose thereby minimizing how much is collected.
- e. Web identities today are typically issued by centralized authorities. In the case of financial institutions, such as the banks, identity proofing is primarily conducted in-person. Online identity proofing and attestation has received little or no attention to date. Identity agents will enable identity proofing and attestation of owners' digital identities by ordinary users as well as designated identity providers.
- f. The World Wide Web Consortium's emerging models for verifiable credentials [8] and decentralized identifiers [9] propose applying digital signature and zero-knowledge proofs to verify credentials and identifiers. However, they do not describe how bindings between users and their VCs or DIDs can be verified. In contrast, identity agents proposed herein are tightly controlled by their owners; encapsulate the

owner’s digital identities and integrated public/private key-pairs; and can cryptographically bind an owner’s identity and attestation to digital identities, consent tokens and other such digital artifacts. Relying parties can cryptographically verify that the attested artifact is controlled by the identified owner (see Sect. 7.1).

## 6 The System Concept

The proposed identity architecture is comprised of *identity agents* and *digital identities* installed on the Internet devices of consumers and providers [11–13]. Identity agents encapsulate the owner’s digital identities and work on behalf of their owners. Each identity agent leverages strong authentication mechanisms, possibly using multiple factors. The owner’s digital identities specify characterizing attributes of the owner and have multiple public/private encryption key-pairs. These keys can be used by the identity agent to encrypt private information of the owner stored outside the context of her identity agent. An identity agent thereby binds the owner to selected private and identifying information specified by the owner’s digital identities as well as private and identifying information stored outside her device or remotely.

Figure 2 illustrates the system concept. Depicted is a user having an identity agent that encapsulates the owner’s authentication data thereby tightly binding the owner to her digital identities, her collaborators’ digital identities, her consent tokens, and other artifacts - possibly by means of multiple authenticators. The owner’s identity agent is

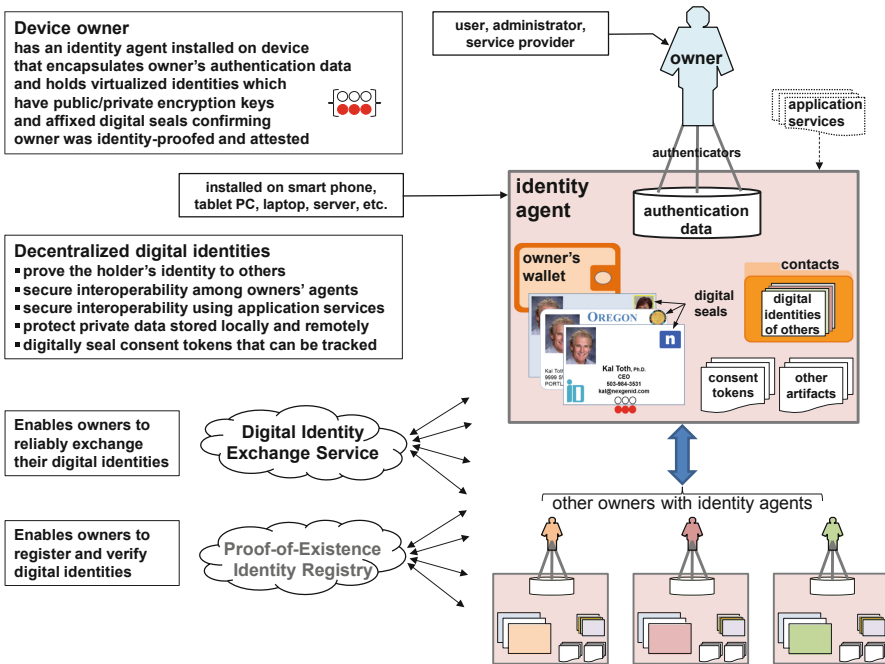


Fig. 2. System concept diagram

the integrative glue giving life to her digital identities, virtualizing them for intuitive ease-of-use such that they mimic physical credentials in her wallet.

An identity agent tightly binds the owner to his/her digital identities; protects each owner's private and identifying information; integrates with locally installed application services; and reliably interoperates with the identity agents of collaborating owners. Identity agents can use a web service to securely exchange digital identities with other parties as well as a proof-of-existence web service to safely register their digital identities such that collaborating owners can verify them.

The identity architecture is designed to encapsulate essential complexity within identity agents including the management of digital identities; protecting "secrets" (PINS, passwords, private keys); supporting local application interfaces; and implementing agent-to-agent protocols. This approach will enable the development<sup>1</sup> of well-behaved identity agents that consistently and correctly handle digital identities while protecting private information. Deploying such identity agents across the web will considerably improve upon the current patchwork of identity solutions and will off-load web applications and services from critical identity and privacy-related tasks.

## 7 Systematized Prior Art

The following prior art has been systematized to implement the identity architecture:

1. Symmetric and public key encryption methods;
2. Digital signature adapted to create digital seals;
3. Proof-of-existence using hashing adapted with digital seals;
4. Diffie-Hellman key agreement method adapted to exchange digital identities, and
5. Proof-of-possession combined with proof-of-custody to verify collaborators.

Three patents and one patent-pending provide additional details regarding items 2-5.<sup>2</sup>

### 7.1 Encryption Keys, Digital Sealing and Non-repudiation

The identity architecture<sup>3</sup> deploys digital identities with integrated private/public keys<sup>4</sup> including embossing/inspecting, signing/verifying, and decrypting/encrypting keys. Identity agents use embossing and inspecting keys to create, affix and verify digital seals<sup>5</sup> and attestations to digital identities, consent tokens, and other artifacts. Digital sealing elevates non-repudiation strength over traditional digital signature because owners tightly control their identity agents, digital identities and private keys used to cryptographically bind their identities and attestations to such digital artifacts.

<sup>1</sup> The founding team has created proof-of-concept and experimental prototypes validating the principle privacy requirements and design elements of the proposed identity architecture.

<sup>2</sup> Founding team intends to issue a license to open source developers similar to RedHat's patent promise to discourage patent aggression <https://www.redhat.com/en/about/patent-promise>.

<sup>3</sup> "Electronic Identity and Credentialing System", US Patent 9,646,150 B2, May 9, 2017.

<sup>4</sup> PGP (Pretty Good Privacy) uses signing/verifying and decrypting/encrypting key-pairs.

<sup>5</sup> "Methods for Using Digital Seals for Non-Repudiation of Attestations", US Patent 9,990,309B2, 2-20-2018; note: "sealing images" are used to render (virtualize) digital seals.

### 8 Methodology: Privacy by Design Process and Validation

Early in the development of critical systems, systems engineers often apply a “gestalt” process to discover requirements and design options, iterating until they converge on a suitable design satisfying the requirements. Figure 3 depicts the gestalt privacy by design process conducted to discover the identity architecture’s privacy requirements and design elements. Privacy Requirements (R) specifies that the system is to enable users to prove who they are, protect their private and identifying information, and collaborate securely. System Design (D) specifies that owners’ have devices with identity agents deploying digital identities used to meet the privacy requirements.

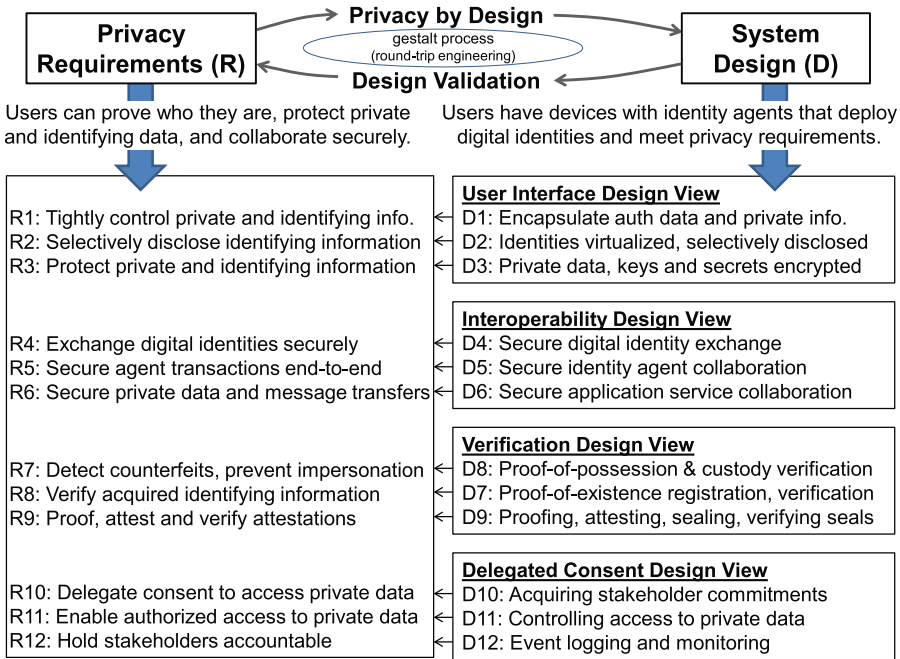


Fig. 3. Privacy by design process

Upon thoroughly iterating over all the requirements and design features, this privacy by design process has yielded mutually validated privacy requirements R1, R2 ... R12 and design elements D1, D2 ... D12 broken down by four design views as shown in Fig. 3. Each iteration organically contributed to the goal of showing that privacy by design principles have been met including minimizing the disclosure and collection of private and identifying information; securing transactions end-to-end; expressly delegating consent for specified purposes; capturing events for accountability purposes; and establishing privacy as the default setting. The privacy by design process can be applied to improve these privacy requirements and design elements.



## 9 Identity Architecture Validation

This section summarizes the results of our gestalt validation process. The annex reasons about how the design elements satisfy the privacy requirements.

### 9.1 User Interface Design View

This design view includes design elements D1 (encapsulate authentication data and private information), D2 (identities virtualized and selectively disclosed), and D3 (private data, keys and secrets encrypted). Figure 4 illustrates this view showing the owner, her device, identity agent, authentication data, digital identities stored in her wallet, and digital identities of others (collaborators) maintained in her contacts list.

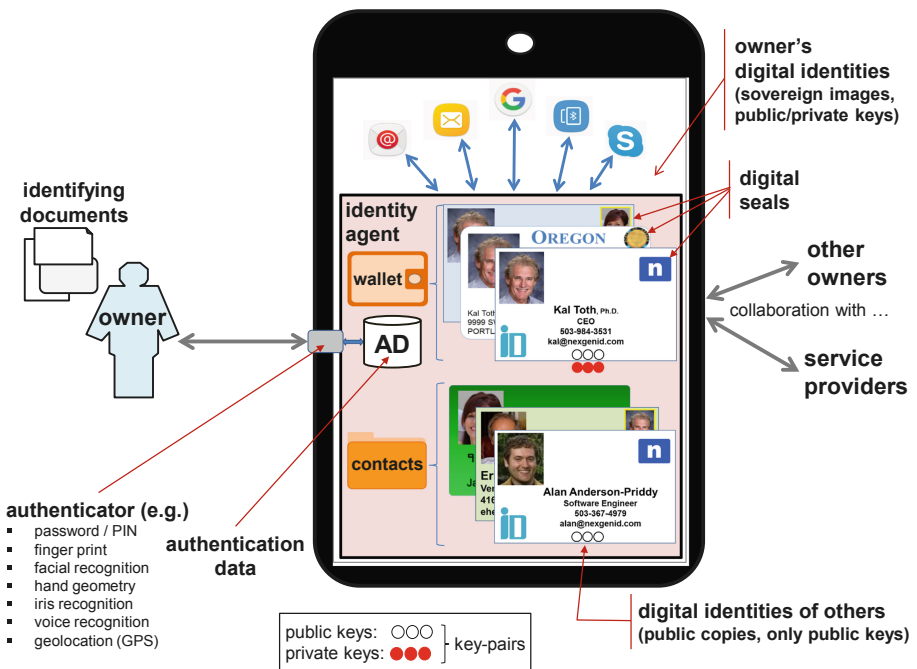


Fig. 4. User interface design view

**Design element D1** (encapsulate authentication data and private information) satisfies privacy requirement R1 (tightly control private and identifying information). To protect against loss, theft and enable custody verification, the identity agent integrates with the device's authenticator(s) via an application programming interface (API) to encapsulate enrolled authentication data and identifying information of the owner.

**Design element D2** (identities virtualized and selectively disclosed) satisfies privacy requirement R2 (selectively disclose identifying information). Owners manage digital identities mimicking the look and behavior of real-world credentials in their physical wallets. Identity agents leverage a common identity data model (e.g. [8]).

**Design element D3** (private data, keys and secrets encrypted) satisfies privacy requirement R3 (protect private and identifying information). Digital identities have public/private keys that can be used by the owner’s identity agent to protect private data and secrets including authentication data, passwords, PINs and digital identities.

**Privacy by Design Default Settings.** When the identity of a remote party is uncertain, the identity agent should use anonymous and pseudo-anonymous identities.

### 9.2 Interoperability Design View

This design view implements design elements D4 (secure digital identity exchange), D5 (secure identity agent collaboration) and D6 (secure application service collaboration). Figure 5 depicts two devices having identity agents capable of interoperating. Each agent offers a common user interface; exposes an API to applications; and leverages the transport layer. Interoperating identity agents thereby implement an identity layer among collaborating consumers and providers.

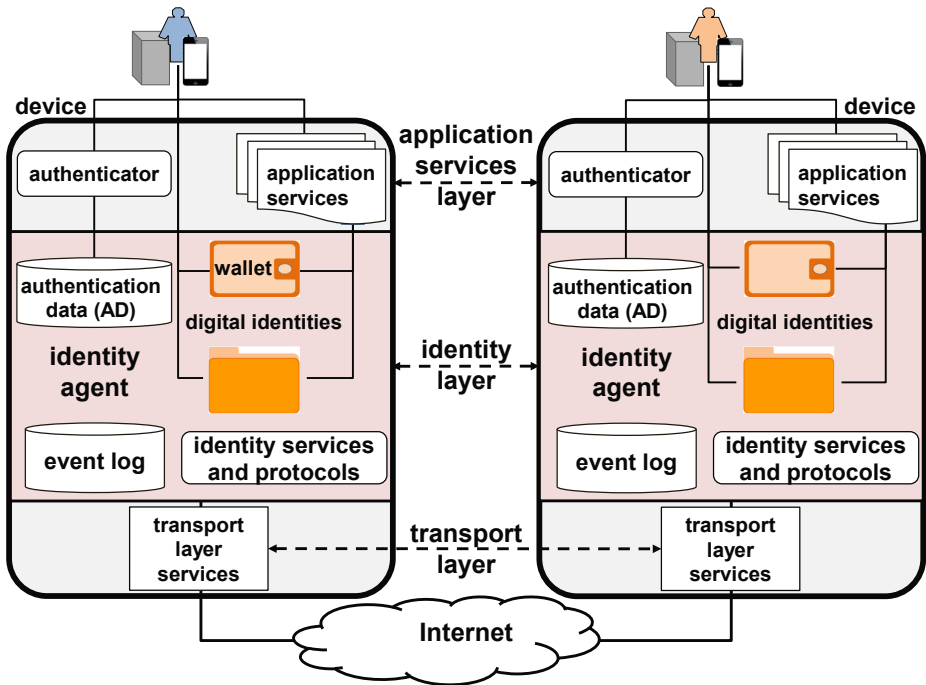
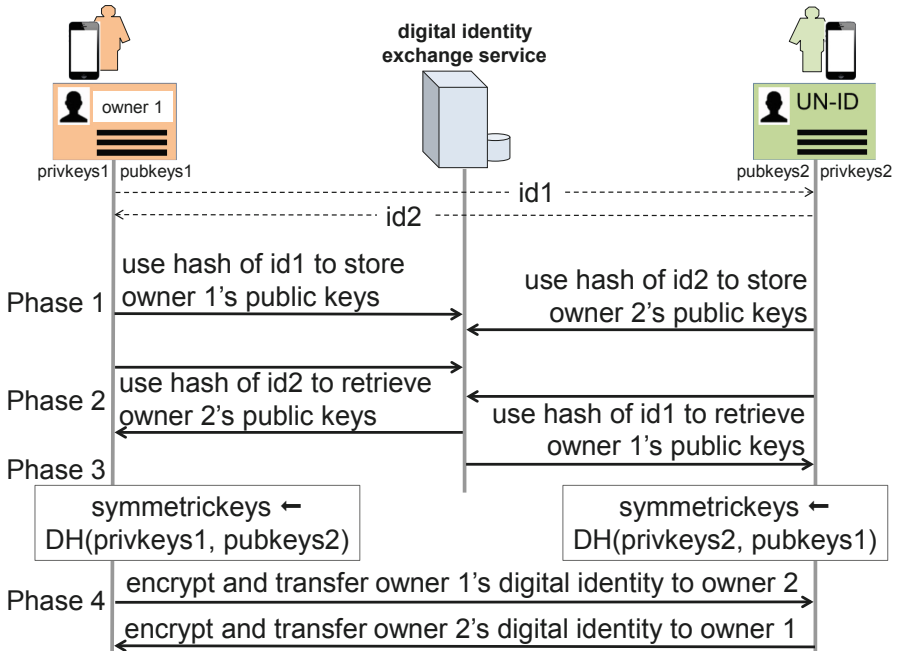


Fig. 5. Interoperability design view

**Design element D4** (secure digital identity exchange) satisfies privacy requirement R4 (exchange digital identities securely). Options include exchanging digital identities in the clear; using one-time passwords; using HTTPS; and transferring them wirelessly. To

mitigate risks, the Diffie-Hellman (DH) key agreement method [14] has been adapted<sup>6</sup>. Figure 6 depicts the protocol sequence where identifiers id1 and id2 are hashed and used to store and retrieve the public keys of each owner’s digital identity, and the DH key agreement method is employed to calculate a shared symmetric key subsequently used to securely transfer the owners’ digital identities. Figure 7 (1) depicts two owners using this exchange service.



**Fig. 6.** Digital identity exchange service

**Design element D5** (secure identity agent collaboration) satisfies privacy requirement R5 (secure agent transactions end-to-end). Once identity agents have reliably transferred digital identities, they can use them to bilaterally secure transactions end-to-end. The sender’s signing key and the recipient’s encrypting key are used when sending, and the recipient’s decrypting and sender’s verifying key when receiving.

**Design element D6** (secure application service collaboration) uses identity agent APIs to satisfy privacy requirement R6 (secure private data and message transfers). Under the control of the sender, the application service acquires the sender’s and recipient’s digital identities, calculates a fingerprint<sup>7</sup>, and then signs, encrypts and sends the message. The receiving application service receives, decrypts, verifies the signature and verifies the fingerprint of the incoming message.

<sup>6</sup> “Architecture and Methods for Self-Sovereign Digital Identity”, US Patent (pending), provisional filed Oct. 8, 2018, utility application filed Nov. 12, 2018.

<sup>7</sup> Digital fingerprints are computed by hashing selected public encryption keys.

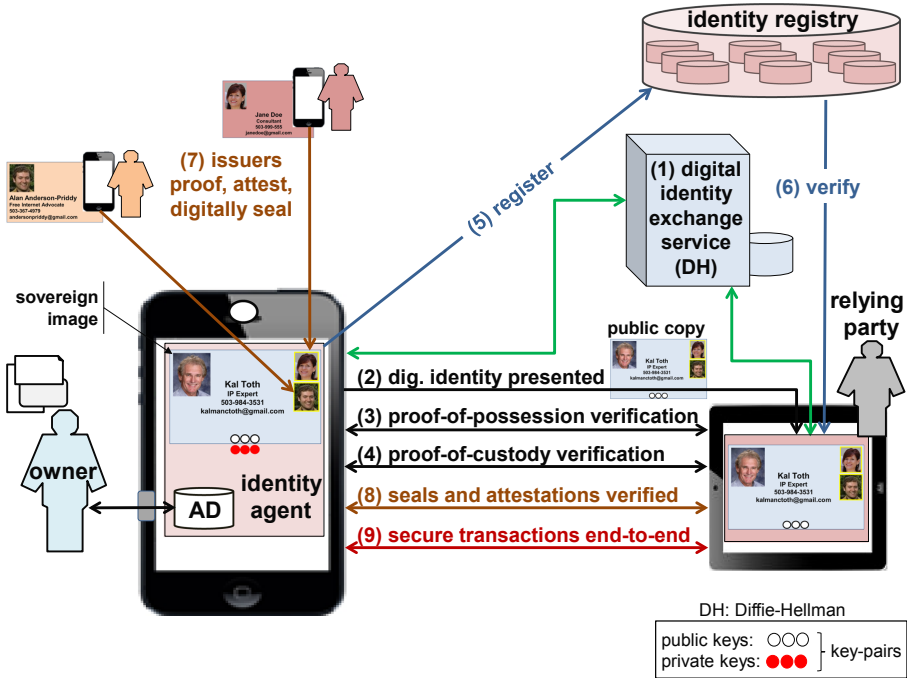


Fig. 7. Verification design view

**Privacy by Design Default Settings.** By default, the adapted Diffie-Hellman should be used to securely exchange digital identities before using them.

### 9.3 Verification Design View

The Verification Design View implements design elements D7 (proof-of-possession and proof-of-custody verification), D8 (proof-of-existence registration and verification), and D9 (proofing, attesting, sealing and verifying seals). Figure 7 depicts synchronous verification (2-4); asynchronous registration and verification (5, 6); issuing and verifying digitally sealed attestations (7, 8), and securing transactions (9).

**Design element D7** (proof-of-possession and proof-of-custody verification) satisfies privacy requirement R7 (detect counterfeits and prevent impersonation). Figure 7 (2–4) shows two parties collaborating synchronously (online). The owner’s identity agent presents a public copy of her digital identity (2) to the relying party’s identity agent which verifies the digital identity by launching a proof-of-possession challenge (3) that only the originator can satisfy using a designated private key of her digital identity [10]. The relying party’s identity agent then sends a proof-of-custody (4) demand to the originator’s identity agent to authenticate the holder. Executed bilaterally, this protocol ensures that digital identities have been securely exchanged enabling subsequent transactions to be secured end-to-end (9).

**Design element D8** (proof-of-existence registration and verification)<sup>8</sup> satisfies privacy requirement R8 (verify acquired identifying information) by combining digital sealing with proof-of-existence popularized by blockchain [15]. Figure 7 (5, 6) shows an owner registering a digital identity and a relying party verifying it. When registering a digital identity (5), the identity registry conducts proof-of-possession and proof-of-custody (D7) challenges to verify the registrant. If verified, the owner’s agent hashes, seals and stores the hashed and sealed identity into the registry. A relying party acquiring a digital identity can hash it and inspect the seal (6) to verify it exists.

**Design element D9** (proofing, attesting, sealing and verifying seals) satisfies privacy requirement R9 (proof, attest, and verify attestations) by proofing [16] and affixing attestations using digital seals. Figure 7 depicts two issuers having proofed, attested and digitally sealed the owner’s digital identity (7) and a relying party verifying the affixed digital seals and attestations (8). Multiple parties can proof and fix attestations that cannot be repudiated using digital seals (see Sect. 7.1).

**Privacy by Design Default Settings.** Default settings should routinely register digital identities whenever they are created and updated and verify proof-of-existence, proof-of-possession and proof-of-custody whenever handling value transactions.

#### 9.4 Delegated Consent Design View

This design view includes design elements D10 (acquiring stakeholder commitments), D11 (controlling access to private data) and D12 (event logging and monitoring). In contrast to server-centric consent models, our consent model enables stakeholders to use their digital identities to create digital seals that cryptographically bind their commitments to consent tokens which they cannot repudiate (see Sect. 7.1). Events are logged to enable mutual accountability.

**Design element D10** (acquiring stakeholder commitments) satisfies privacy requirement R10 (delegate consent to access private data). Figure 8 shows a collaboration sequence wherein the requester, the owner, and the custodian use their identity agents to digitally seal a circulated consent token. These commitments include requesting (1), clearing/approving (2, 3), and granting access (4) to the resources of the owner. The requestor uses the finalized consent token to submit access requests (5) to the custodian to access the owner’s resources (6) until expired or revoked.

**Design element D11** (controlling access to private data) satisfies privacy requirement R11 (enable authorized access to private data). Finalized consent tokens control requester access to the resources of the owner managed by the resource custodian.

**Design element D12** (event logging and monitoring) satisfies privacy requirement R12 (hold stakeholders accountable). Commitments and access events, traceable to stakeholders by way of digital seals and consent tokens, are tracked and reported.

**Privacy by Design Default Settings.** By default, stakeholders should be required to register digital identities and consent tokens in a proof-of-existence registry.

---

<sup>8</sup> “Systems and Methods for Registering and Acquiring E-Credentials using Proof-of-Existence and Digital Seals”, US Pat 10,127,378 B2, issued Nov. 13, 2018.

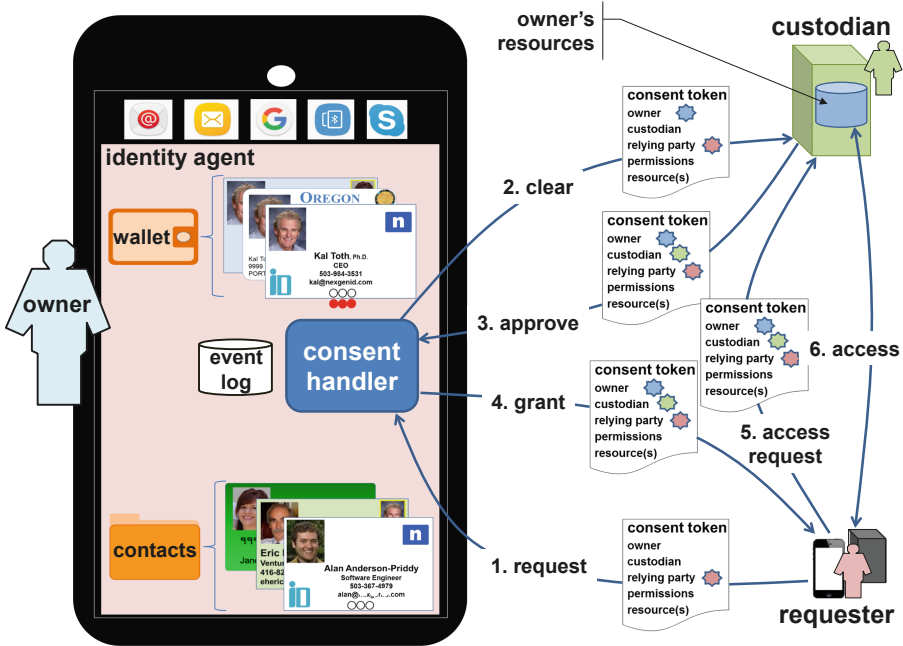


Fig. 8. Delegated consent design view

## 10 Closing Remarks

A privacy by design architecture leveraging software-based agents to decentralize identity has been proposed. The architecture has been validated by applying a gestalt process that can be used to create a reference model for development going forward.

The described architecture uniquely integrates authentication data, cryptographic mechanisms, and virtualization to create identity agents that deliver unanticipated identity and privacy enhancing capabilities. Identity agents enable owners to control what they disclose, protect their private information and transactions, and reliably delegate consent. Shifting control over identity from service providers to users mitigates breach risk because providers need not collect as much private information. Decentralization thwarts hacking and mitigates impersonation risk because the attack surface is more widely dispersed. Meanwhile, broad-based identity proofing and attestation elevates identity assurances and reduces impersonation risks. Combined, these capabilities will significantly reduce dependency on remote access passwords and facilitate technology adoption.

## 11 Areas for Further Study

Quantitative assessment of the risks, liabilities, and trade-offs posed by centralizing identity over decentralizing them; using formal verification methods, trusted platform modules, and trusted execution environments; exploiting artificial intelligence and machine

learning; leveraging W3C's verifiable credentials and decentralized identifier [DID] models; and adopting elements of Signal's messaging protocol [17].

## **Annex: Identity Architecture Validation**

A gestalt privacy by design process has been used to iteratively identify and validate the privacy requirements and design elements of the identity architecture. This annex explains how each design element satisfies the privacy requirements thereby mutually validating the design elements and the privacy requirements. The paragraphs below reference Fig. 4, Fig. 5, Fig. 6, Fig. 7 and Fig. 8 in the body of this paper.

### **A. User Interface Design View**

The User Interface Design View shown in Fig. 4 encompasses design elements D1, D2 and D3 implementing privacy requirements R1, R2 and R3. The proposed privacy by design default settings for the User Interface Design View are also identified.

#### **R1: Tightly Control Private and Identifying Information.**

##### **D1: Encapsulate Authentication Data and Private Information.**

Design element D1 combined with physical custody of the owner's Internet device satisfies R1 by providing strong assurances that the owner, whether an end-user or a system administrator, controls her Internet device, installed identity agent, and encapsulated digital identities.

Although physical custody of her device enables the owner to maintain control over her identity agent, this is not enough to guarantee that her device has not been stolen or lost. Relying parties need objective proof that the owner of a remotely held device has control over it. This design element satisfies this need by exploiting native mechanisms built into Internet devices (smart phones, tablet PCs, laptops) including password, PIN, biometric, and geo-location authentication mechanisms.

Design element D1 significantly reduces the risk of theft and loss of the owner's device by encapsulating the owner's authentication data while providing the device's authenticator(s) access to this data by way of an application programming interface (API). When the owner's device is first used, her native authenticator enrolls her authentication data by writing this data via the API to the identity agent. Subsequently, the authentication data can be accessed via the API to support the authenticator's mechanisms for verifying the device owner. Once a positive authentication indication is detected, the identity agent's digital identities, interfaces and other data are made available for use. The authentication data is not revealed by the identity agent.

Design element D1 thereby provides strong assurances that the owner tightly controls his/her digital identities including private and identifying information, secrets and crypto keys. When digital identities are created, the identity agent of an owner allocates public/private encryption key-pairs to each and vaults its so-called *sovereign image*. The

private keys of the digital identity are not disclosed by the identity agent. However, the identity agent can distribute public copies<sup>9</sup> to other parties.

**R2: Selectively Disclose Identifying Information.**  
**D2: Identities Virtualized and Selectively Disclosed.**

Identity agents use an identity data model (e.g. [8]) to support the specification of digital identities that characterize the owner (e.g. claims, attributes and images). Design element D2 satisfies R2 by addressing usability and ease-of-use thereby enabling owners to create and select digital identities that have the appearance and behavior (“look and feel”) of identity credentials used in the real world.

Figure 4 depicts the owner holding multiple digital identities in her wallet such as a digital business card, digital driver’s license, e-health card, digital credit card, and/or electronic membership card. Instead of using remote access passwords, owners intuitively select digital identities that disclose only the private and identifying information necessary to satisfy the needs and purposes of the collaborating service provider or consumer. To facilitate disclosure, design element D2 enables owners to specify “anonymous identities” where the attributes are known only to the owner; “pseudo-anonymous identities” where the attributes are disclosed only to trusted collaborators; and “civil identities” where identifiers and attributes partially or fully elaborate identifying information.

**R3: Protect Private and Identifying Information.**  
**D3: Private Data, Keys and Secrets Encrypted.**

Digital identities have multiple public/private encryption key-pairs that can be used to protect private data and secrets of the identity agent owner. Design element D3 satisfies R3 by applying these encryption keys to prevent hackers and malware from maliciously accessing such data including authentication data, digital identities, encryption keys, passwords, and PINs encapsulated by the identity agent. A designated public key of a digital identity of the owner can be used to encrypt her private data and secrets. Only the owner can decrypt the data using the paired private key.

**Privacy by Design Default Settings of User Interface Design View**

When the identity of an originating party is unknown or uncertain, the identity agent should not use digital identities that include identifying, private or secret information. Anonymous and pseudo-anonymous identities could be used as defaults when first establishing an online session or when exchanging digital identities with newly introduced or unvetted parties at meetings or public gatherings. Wireless mechanisms like QR codes, NFC, WiFi and Bluetooth could be used to securely transfer digital identities and private data when collaborators meet in-person.

**B. Interoperability Design View**

The Interoperability Design View shown in Fig. 5 encompasses design elements D4, D5 and D6 implementing privacy requirements R4, R5 and R6. Figure 6 supports the

<sup>9</sup> Public copies of a digital identity disclose only the public keys (private keys not revealed).



reasoning behind how design element D4 satisfies privacy requirement R4. The proposed privacy by design default settings for this design view are also identified.

#### **R4: Exchange Digital Identities Securely.**

##### **D4: Secure Digital Identity Exchange.**

Design element D4 satisfies R4 by enabling digital identities to be reliably and securely exchanged, the aim being to prevent man-in-the-middle attacks akin to robocalls and wiretaps on telephone networks. Options for exchanging digital identities include exchanging them in the clear; exchanging one-time passwords (OTPs) out-of-band; transferring them over HTTPS once logged in using an online password; and meeting in-person to transfer digital identities wirelessly.

Although the above methods carry various risks, they are safe enough to use in many contexts. One risk is that of digital identities being highjacked. Such risks can be mitigated by using an online service and combining hashing with Diffie-Hellman's key agreement method (DH) [14]. Our adapted Diffie-Hellman protocol<sup>10</sup> is depicted in Fig. 6 as well as in Fig. 7 (1) where it is shown in the context of verification.

Figure 6 illustrates how two owners can securely exchange public copies of their digital identities. Owners 1 and 2 have digital identities they wish to exchange respectively using identifiers id1 and id2. They first use their identity agents to store the public keys of their digital identities in the exchange service's repository at locations computed by respectively hashing id1 and id2 (e.g. using SHA256). Subsequently exchanging id1 and id2 by alternate means (e.g. physical transfer or out-of-band), they use their identity agents to hash the opposite owner's identifier to locate and retrieve the public keys of the other owner. The DH key agreement method is then applied by both owner's identity agents to combine the private keys of the owner with the retrieved public keys of the other owner thereby generating the same symmetric encryption key (or keys) for both owners. Finally, the symmetric keys are applied to encrypt and thereby securely exchange public copies of their digital identities.

To overtake the owners' digital identities, a malicious high-jacker would be obliged to successfully intercept id1 and id2, breach the digital identity exchange service, and discover the private keys from the captured public keys. Alternatively, the high-jacker could attempt to breach both owners' devices and identity agents.

#### **R5: Secure Agent Transactions End-to-End.**

##### **D5: Secure Identity Agent Collaboration.**

Design element D5 satisfies R5 by enabling collaborating identity agents to secure transactions end-to-end in order to thwart surveillance and tampering. Once identity agents have reliably transferred digital identities, they can use them to securely collaborate bilaterally by selecting designated keys bound to their digital identities. When sending a payload, the sender's private signing key and the recipient's public encrypting

---

<sup>10</sup> "Architecture and Methods for Self-Sovereign Digital Identity", US Patent (pending), provisional filed Oct. 8, 2018, utility application filed Nov. 12, 2018.

key are applied. When receiving, the recipient's private decrypting key and the sender's public verifying key are applied.

**R6: Secure Private Data and Message Transfers.**

**D6: Secure Application Service Collaboration.**

Design element D6 enables collaborative services (e.g. web messaging apps) to implement R6 by leveraging already exchanged digital identities held in owners' wallets and contact lists to secure messages end-to-end. The owners' identity agents expose an API to the service at each endpoint. By means of these APIs, the sending owner directs the service to select a digital identity from her wallet and a digital identity of the recipient from her contacts list. The sender's identity agent uses the private signing key of her digital identity and the public encrypting key of the recipient's digital identity to respectively sign and encrypt the message and send this message together with digital fingerprints<sup>11</sup> of the sender's and recipient's digital identities to the recipient. The service at the recipient's endpoint uses the identity agent's API to select the digital identities from the recipient's wallet and contacts list, verify the received digital fingerprints, decrypt the message, and verify the digital signature.

**Privacy by Design Default Settings of Interoperability Design View**

By default the interoperability design view could be configured to invoke the adapted Diffie-Hellman exchange protocol to reliably transfer collaborators' digital identities. Identity agents and applications could default to using digital identities to secure all transactions end-to-end. Owners should be warned of the risks of choosing to exchange digital identities and transactions when using less reliable transfer methods.

**C. Verification Design View**

The Verification Design View shown in Fig. 7 encompasses design elements D7, D8 and D9 implementing privacy requirements R7, R8 and R9. The proposed privacy by design default settings for the Verification Design View are also identified.

**R7: Detect Counterfeits and Prevent Impersonation.**

**D7: Proof-of-Possession and Proof-of-Custody Verification.**

Design element D7 combines prior art proof-of-possession [10] with proof-of-custody (remote authentication-on-demand) to satisfy R7 by verifying that a remotely located owner controls her digital identities and device. This adaptation relies on the identity agent encapsulating the owner's authentication data and digital identities.

Figure 7 (2, 3, 4) shows two parties synchronously collaborating online. The identity agent of the depicted owner presents a public copy of her digital identity to the identity agent of the relying party (2). The identity agent of the relying party verifies the presented identity and obtains proof that the originator controls the associated digital identity. To accomplish this, the relying party's identity agent uses a public key of the presented

<sup>11</sup> Digital fingerprints are computed by hashing selected public encryption keys.

digital identity to launch a proof-of-possession challenge (3) to verify the presented digital identity. Such challenges can only be satisfied by using the paired private key of the digital identity controlled by the originator's identity agent. Executed bilaterally, both parties can determine whether the identity agent of the collaborator controls the presented digital identity thereby detecting counterfeits.

Once possession of the private key has been verified, the identity agent of a relying party can send a proof-of-custody demand (4) to the originating identity agent to verify custody by the enrolled holder using design element D1. An affirmative proof-of-custody indication is returned if the originator is successfully authenticated. This element of the protocol can also be conducted bilaterally.

Used together, collaborating identity agents can execute proof-of-possession and proof-of-custody challenges to prove that the corresponding party controls the presented digital identity thereby detecting impersonation and ensuring that subsequent transactions can be secured end-to-end (9).

### **R8: Verify Acquired Identifying Information.**

#### **D8: Proof-of-Existence Registration and Verification.**

To satisfy R8, design element D8 has adapted<sup>12</sup> a "proof-of-existence" hashing method popularized by blockchain [15] with our digital sealing method. This method enables owners to verify each other's digital identities when collaborating asynchronously (e.g. email and messaging). Figure 7 illustrates the owner registering a digital identity (5) in a proof-of-existence identity registry and a relying party verifying it (6).

When registering a digital identity in the proof-of-existence identity registry (5), the registry's identity agent first uses design element D7 to execute proof-of-possession and proof-of-custody challenges to verify the identity of the registrant. If successfully verified, the owner's identity agent hashes the digital identity, digitally seals the hashed digital identity, and uses the registry's identity agent to store the hash and digital seal in the registry. A relying party, having acquired a registered digital identity from the owner, verifies the acquired digital (6) by having her identity agent hash the digital identity, use the hash to verify that the record exists in the identity registry, retrieve the digital seal, and use a designated public key of the acquired digital identity to verify the affixed digital seal. These steps enable a relying party to determine whether an acquired digital identity exists and was registered by the originating owner. A breach of the registry will not reveal private data of registered digital identities (attributes, images, keys, etc.) because they are hashed.

### **R9: Proof, Attest and Verify Attestations.**

#### **D9: Proofing, Attesting, Sealing and Verifying Seals.**

Design element D9 satisfies R9 by implementing procedures and methods for proofing, attesting, creating and verifying digital seals and attestations affixed to digital identities. NIST provides identity proofing guidance [16].

---

<sup>12</sup> "Systems and Methods for Registering and Acquiring E-Credentials using Proof-of-Existence and Digital Seals", US Pat 10,127,378 B2, issued Nov. 13, 2018.

Figure 7 depicts two issuers having proofed, attested and affixed digital seals to the owner's digital identity. The process begins with an issuing owner meeting the requesting owner to validate her identity by inspecting presented identifying information. If successfully identity-proofed, the issuer uses his identity agent to affix an attestation (e.g. "proofed") to her digital identity using a digital seal (7). A digital seal is created by using a pre-determined sealing image and the private embossing key of a selected digital identity of the issuer (see 7.1). A relying party can verify such digital seals and attestations (8) by using the public inspection key of the public copy of issuer's digital identity.

As illustrated, multiple parties (users and providers) can proof, attest and digitally seal a given digital identity. Digital identities affixed with multiple digital seals arguably elevate identity assurances over digital identities having a single digital seal or no seals at all. Digital seals can be used to affix attestations to other electronic documents including consent tokens as discussed below.

### **Privacy by Design Default Settings of Verification Design View**

Default settings could include routinely registering digital identities whenever they are created and updated. Proof-of-existence, proof-of-possession and proof-of-custody methods should be employed by default whenever executing high risk or high value transactions (e.g. for banking and critical infrastructures).

### **D. Delegated Consent Design View**

The Delegated Consent Design View shown in Fig. 8 encompasses design elements D10, D11 and D12 implementing privacy requirements R10, R11 and R12. The privacy by design default settings for this design view are identified below.

Today's consent models are managed by service providers. For example, OpenID Connect (openid.net) is a server-centric consent model where user authentication and access tokens are wholly controlled by service providers. In contrast, the model described herein decentralizes consent by using digital seals to cryptographically bind stakeholder commitments and identities to consent tokens which can be expired, revoked, registered and tracked. Since commitments are affixed using digital seals, they cannot be repudiated (see Sect.7.1). Figure 8 depicts the consent delegation process. Each consent token identifies the resource owner, resource custodian, requester, the owner's private resources, access permissions including purposes, and expiry date/time. Given digital seals have sealing images, consent tokens rendered by identity agents visualize commitments for users. The event logger enables accountability.

#### **R10: Delegate Consent to Access Private Data.**

#### **D10: Acquiring Stakeholder Commitments.**

Design element D10 satisfies R10 by enabling resource owners to use their identity agents to delegate consent to other parties requesting access to their private resources. Stakeholders use their identity agents to digitally seal a circulated consent token.

Figure 8 shows a consent token collaboration sequence (1–6) among stakeholders. They use their identity agents and selected digital identities to create digital seals affixing

their commitments to the consent token. The resource owner first requires the requesting owner to affix requested permissions and intended purpose to the consent token with a digital seal, and then the resource custodian's approval to provide access by affixing a digital seal. Once satisfied, the resource owner grants access by digitally sealing the access token and issuing it to the requester who can present the token to the resource custodian whenever requesting access.

**R11: Enable Authorized Access to Private Data.**

**D11: Controlling Access to Private Data.**

Design element D11 satisfies R11 by using finalized consent tokens to provide authorized access to owner resources controlled by resource custodians according to commitments and approvals affixed to the consent token by stakeholders.

**R12: Hold Stakeholders Accountable.**

**D12: Event Logging and Monitoring.**

Design element D12 implements R12 by tracking and reporting digitally sealed stakeholder commitments as well as access, expiry and revocation events.

**Privacy by Design Default Settings of Delegated Consent Design View**

Identity agents should register consent tokens when digitally sealed into a proof-of-existence registry to enable expiry and revocation checking by stakeholders.

**E. Summary**

Our privacy by design validation process has confirmed that the proposed identity architecture is capable of reliably and securely supporting the following:

- Identity agents will be able to control owners' authentication data and deploy digital identities thereby enabling them to control what they disclose;
- Digital identities managed by identity agents will enable owners to secure their transactions end-to-end and protect their private data stored locally or remotely;
- Owners will be able to use their identity agents to proof and attest the digital identities of other parties thereby elevating identity assurances and privacy protection;
- Owners will be able to use their identity agents and digital identities to create digital seals enabling express delegated consent among stakeholders;
- The identified privacy default settings will minimize how much private and identifying information is disclosed by owners, and how much is collected by service providers and collaborating peers.
- Visibility and transparency into the design by way of the privacy by design process will enable third-party validation and improvement benefiting users and providers.

## References

1. Cavoukian, A.: Privacy by Design, The 7 Foundational Principles. <https://www.ipc.on.ca/wp-content/uploads/Resources/7foundationalprinciples.pdf>
2. Brooker, K.: Tim Berners-Lee tells us his radical new plan to upend the World Wide Web, FastCompany, 29 September 2018
3. Cameron, K.: The Laws of Identity, May 2005. <http://myinstantid.com/laws.pdf>
4. Cavoukian, A.: Consumers bear the cost of their privacy protection, Globe and Mail, 7 September 2018
5. Jones, H.: Accelerating the future of privacy through smartdata agents, Cognitive World, AI & Big Data, 3 November 2018
6. Allen, C.: The path to self-sovereign identity, 27 April 2016. <http://coindesk.com>
7. Sovrin Foundation, Sovrin: A Protocol and Token for Self-Sovereign Identity and Decentralized Trust, Version 1, January 2018. <https://sovrin.org>
8. World Wide Web Consortium (W3C), verifiable credentials data model 1.0: expressing verifiable information on the Web, W3C recommendation, 19 November 2019
9. World Wide Web Consortium (W3C), Decentralized Identifiers (DIDs) v1.0: Core Data Model and Syntaxes, WC3 Working Draft 09 December 2019
10. Asokan, N., Niemi, V., Laitinen, P.: On the usefulness of proof of possession. In: 2nd Annual PKI Workshop, 28–29 April 2003, pp. 136–141 (2003)
11. Toth, K.C., Anderson-Priddy, A.: Architecture for self-sovereign digital identity. Computer Applications for Industry and Engineering, New Orleans, LA, 8–10 October 2018
12. Toth, K.C., Anderson-Priddy, A.: Self-sovereign digital identity: a paradigm shift for identity. *IEEE Secur. Priv.* **17**(3), 17–27 (2019)
13. Toth, K.C., Anderson-Priddy, A.: Privacy by design using agents and sovereign identities. In: Information Security and Privacy Protection Conference (IFIP-SEC), Work in Progress and Emerging Technology Track, Lisbon, Portugal, 25–27 June 2019 (2019)
14. Rescorla, E.: Diffie-Hellman key agreement method, RTFM Inc., June 1999
15. Robles, K.: BlockchainMe, tool for creating verifiable IDs on the blockchain, 2 December 2016. <https://github.com/kiarafrobes/blockchainMe>
16. NIST Special Publication 800–63A, Digital Identity Guidelines, Enrollment and Identity Proofing, January 2017. <https://doi.org/10.6028/NIST.SP.800-63a>
17. Cohn-Gordon, K., et al.: A formal analysis of the signal messaging protocol, November 2017. <https://eprint.iacr.org/2016/1013.pdf>