(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
21 February 2008 (21.02.2008)

PCT

(10) International Publication Number
WO 2008/020991 A2

(51) International Patent Classification:
*G06Q 30/00* (2006.01)

(21) International Application Number:
PCT/US2007/017047

(22) International Filing Date: 30 July 2007 (30.07.2007)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/833,983 28 July 2006 (28.07.2006) US

(71) Applicant *(for all designated States except US)*: **BROWN UNIVERSITY** [US/US]; One Prospect Street, P.o. Box 1949, Providence, RI 02912 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **GOODRICH, Michael, T.** [US/US]; 11 Twain Street, Irvine, CA 92617 (US). **YAO, Danfeng** [CN/US]; 3133 Edgerton Street, West Lafayette, IN 47906 (US). **TAMASSIA, Roberto** [IT/US]; 44 Alumni Avenue, Providence, RI 02906 (US).

(74) Agents: **HARRINGTON, Mark, F.** et al.; Harrington & Smith, Pc, 4 Research Drive, Shelton, CT 06484-6212 (US).

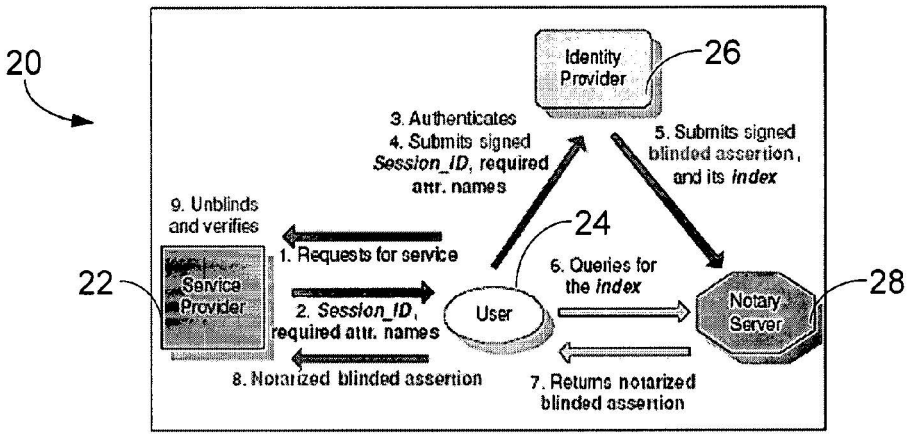
(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

**Published:**

— *without international search report and to be republished upon receipt of that report*

(54) **Title:** NOTARIZED FEDERATED IDENTITY MANAGEMENT




(57) **Abstract:** The exemplary embodiments of this invention describe methods, computer programs, apparatus and systems that provide notarized federated identity management. The exemplary embodiments may be useful, for example, for supporting efficient user authentication when providers are unknown to each other and/or for avoiding direct communication between identity providers and service providers, which provides improved privacy protection for users. In one non-limiting, exemplary embodiment, a method includes: receiving through a data communication network an assertion generated by a first entity; notarizing the assertion to obtain a corresponding notarized assertion; and in response to receiving from a second entity via the same or a different data communication network a query corresponding to the assertion, returning the corresponding notarized assertion. In a further exemplary embodiment, the method further includes: determining a user private key for a corresponding user public key consisting of at least one piece of user identity information; and returning the user private key to the user as data to be stored on a storage medium.

WO 2008/020991 A2

# NOTARIZED FEDERATED IDENTITY MANAGEMENT

TECHNICAL FIELD:

**[0001]** The teachings in accordance with the exemplary embodiments of this invention relate generally to identity management and, more specifically, relate to managing user authentication in a federation of providers.

BACKGROUND:

**[0002]** Digital identity management is becoming an integral part of our lives, as consumers and businesses rely more and more on online transactions for daily tasks, such as banking, shopping, and bill payment. These transactions crucially depend on networked computer systems to communicate sensitive identity data across personal, company, and enterprise boundaries.

**[0003]** Unfortunately, the overuse of personal information in online transactions opens the door to identity theft, which poses a serious threat to personal finances and credit ratings of users and creates liability for corporations. Moreover, the increasing dangers of identity theft are negatively affecting people's collective confidence in the digital world for online financial transactions. Thus, effective solutions for managing digital identity, on both the individual and enterprise levels, are urgently needed.

**[0004]** Additionally, end users are challenged with an increasing number of websites that require access control and authentication. Studies show that users resort to using weak passwords or writing them down to alleviate the burden of memorizing multiple passwords. One well-known identity management solution that deals with this issue is the *single sign-on* (*SSO*) technique, which requires the user to authenticate only once to a website, and then automatically authenticates the user to other websites from then on, within a session. There are two primary approaches to single sign-on. One approach is based on browser cookies and redirects, and the other is based on cryptographic-enabled assertions.

**[0005]** The cookies-based approach is simple to implement, yet has several limitations. This approach is implemented by using browser cookies to maintain the state



APPL-1052 / IPR2024-00331 / Page 2 of 44
Apple Inc. v. Carbyne Biometrics, LLC

of the browser, so that re-authentication at a secure web site is not required. Because browser cookies are not transferred between different administrative domains, however, cookies obtained from one site are not sent (in any HTTP messages) to other sites where authentications are required. Thus, the cookies-based approach is only useful inside a single administrative domain. In fact, this problem also exists in single organizations that maintain separate divisional domains.

**[0006]** The approach based on cryptographic-enabled assertions is embodied by the *Security Assertion Markup Language* (*SAML*). See S. Cantor, F. Hirsch, J. Kemp, R. Philpott, E. Maler, J. Hughes, J. Hodges, P. Mishra, and J. Moreh. Security Assertion Markup Language (SAML) V2.0. Version 2.0. OASIS Standards. SAML 2.0 is generally believed to support general cross-domain authentication and SAML is quickly becoming the de-facto means for exchanging user credentials between trusted environments. The identity federation architecture of *Liberty Alliance* is compliant with the SAML 2.0 standard. Indeed, SAML is specifically designed to support cross domain single sign-on, which is illustrated in the following example.

**[0007]** Assume a user has a secure logon session to a website, (e.g., Airline.com), and is accessing resources on that site. Airline.com serves as the *identity provider* site. At some point in the session, the user is directed to another web site in a different DNS domain for a related service, and this outside domain is called the *service provider* site (e.g., CarRental.com). The identity provider (Airline.com) asserts to the service provider (CarRental.com) that the user is known to the identity provider and gives to the service provider the user's name and session attributes (e.g., Gold member). Since the service provider trusts the assertions generated by the identity provider, it creates a session for the user based on the information received. The user is not required to authenticate again when directed to the service provider site. Hence, single sign-on is achieved.

**[0008]** The *identity provider* (*IdP*) in SAML is defined as the system, or administrative domain, that asserts information about a subject. An identity provider asserts that a user has been authenticated and has certain attributes. The *service provider* (*SP*) is defined as the system, or administrative domain, that relies on the information supplied to it by the identity provider.



APPL-1052 / IPR2024-00331 / Page 3 of 44
Apple Inc. v. Carbyne Biometrics, LLC

**[0009]** Anonymous credential systems (a.k.a. pseudonym systems) allow anonymous yet authenticated and accountable transactions between users and service providers. One of the main design goals of these systems is to achieve unlinkability of multiple showing of credentials. The Identity mix (idemix) project is an anonymous credential system using presented cryptographic protocols. Such a system includes users and organizations. Organizations know the users only by pseudonyms. Different pseudonyms of the same user cannot be linked. An organization can issue a credential to a pseudonym, and the corresponding user can prove the possession of this credential to another organization (who knows the user by a different pseudonym), without revealing anything more than the fact that the user owns such a credential.

**[0010]** In the past decade, the European Union and its member states have implemented a legal framework to provide guidance on processing of personal data with the specific aim to restore citizens' control over their data. To complement the legal framework, Camenisch *et al.* presented the architecture of PRIME (Privacy and Identity Management for Europe), which implements a technical framework for processing personal data. See J. Camenisch, abhi shelat, D. Sommer, S. Fischer-H¨ubner, M. Hansen, H. Krasemann, G. Lacoste, R. Leenes, and J. Tseng. Privacy and identity management for everyone. Proceedings of the 2005 ACM Workshop on Digital Identity Management, pages 20–27, November 2005. PRIME focuses on enabling users to actively manage and control the release of their private information. Thus, the PRIME system places a significant burden on users.

**[0011]** One previously proposed federated identity management solution emphasizes the need for proving the knowledge of personal information without actually revealing it, in order to help prevent identity theft. In that solution, personal data such as a social security number is never transmitted in the clear. Commitment schemes and zero-knowledge proofs are used to commit data and prove the knowledge of the data.

**[0012]** BBAE is the federated identity-management protocol proposed by Pfitzmann and Waidner. See B. Pfitzmann and M. Waidner. Federated identity-management protocols. Security Protocols Workshop, pages 153–174, 2003. They give a concrete browser-based single sign-on protocol that aims at the security of communications and the privacy of user's attributes. Their protocol is based on a standard



APPL-1052 / IPR2024-00331 / Page 4 of 44
Apple Inc. v. Carbyne Biometrics, LLC

browser, and therefore does not require the user to install any program.

**[0013]** In the access control area, one approach is the framework for regulating service access and release of private information in web-services by Bonatti and Samarati. See P. A. Bonatti and P. Samarati. A uniform framework for regulating service access and information release on the web. Journal of Computer Security, 10(3):241–272, 2002. They study the information disclosure using a language and policy approach.

**[0014]** A counter measure for identity theft through location cross-checking and information filtering was proposed. This addresses the identity cloning problem, and proposes to use personal location devices, such as GPS and central monitoring systems, to ensure the uniqueness of identities. However, the central monitoring system in their solution is likely to be a performance bottleneck. Moreover, because identity thieves are geographically dispersed, distributing the monitoring task into several locations is not feasible.

SUMMARY:

**[0015]** In an exemplary aspect of the invention, a method includes: receiving through a data communication network an assertion generated by a first entity; notarizing the assertion to obtain a corresponding notarized assertion; and in response to receiving from a second entity via the same or a different data communication network a query corresponding to the assertion, returning the corresponding notarized assertion.

**[0016]** In another exemplary aspect of the invention, a computer program product includes program instructions embodied on a tangible computer-readable medium, execution of the program instructions resulting in operations including: receiving through a data communication network an assertion generated by a first entity; notarizing the assertion to obtain a corresponding notarized assertion; and in response to receiving from a second entity via the same or a different data communication network a query corresponding to the assertion, returning the corresponding notarized assertion.

**[0017]** In a further exemplary aspect of the invention, a system includes: a notary component configured to receive, via a data communication network, an assertion generated by a first entity and to notarize the assertion to obtain a corresponding



APPL-1052 / IPR2024-00331 / Page 5 of 44
Apple Inc. v. Carbyne Biometrics, LLC

notarized assertion; and a responder component configured to receive the notarized assertion from the notary component and, in response to receiving from a second entity via the same or a different communication network a query corresponding to the assertion, to return the corresponding notarized assertion.

BRIEF DESCRIPTION OF THE DRAWINGS:

**[0018]** The foregoing and other aspects of embodiments of this invention are made more evident in the following Detailed Description, when read in conjunction with the attached Drawing Figures, wherein:

**[0019]** FIG. 1 illustrates an exemplary system based on a notarized federated identity management model;

**[0020]** FIG. 2 shows an exemplary high-level description of the Secure Transaction Management System (STMS) parties and protocol;

**[0021]** FIG. 3 shows an exemplary schematic drawing of the STMS-implemented notarized federated identity management protocol;

**[0022]** FIG. 4 shows a diagram of an exemplary identity-based encryption (IBE) based authentication protocol; and

**[0023]** FIG. 5 depicts a flowchart illustrating one non-limiting example of a method for practicing the exemplary embodiments of this invention.

DETAILED DESCRIPTION:

**[0024]** A notarized federated identity management model is proposed that supports efficient user authentication when providers are unknown to each other. This model introduces a notary service, owned by a trusted third-party, to dynamically notarize assertions generated by identity providers. An additional feature of this model is the avoidance of direct communications between identity providers and service providers, which provides improved privacy protection for users. An efficient implementation of this notarized federated identity management model is presented based on the secure transaction management system (STMS). A practical solution is also given for mitigating aspects of the identity theft problem and its use is discussed in the notarized federated



APPL-1052 / IPR2024-00331 / Page 6 of 44
Apple Inc. v. Carbyne Biometrics, LLC

identity management model. The unique feature of the disclosed cryptographic solution is that it enables one to proactively prevent the leaking of secret identity information.

1.1 Motivation for Notarized ID Federation

**[0025]** In existing federated identity management systems that support SAML, such as the *Liberty Identity Federation Framework* (*ID-FF*) and *WS-Federation*, it is up to the service provider to decide whether it trusts the assertions provided to it. Service providers in SAML are also known as *relying parties* due to the fact that they rely on the information provided by an identity provider. This reliance implies that websites of different administrative domains need to trust each other's access control verdicts on end users. In fact, SAML single sign-on relies on the concept of *identity federation* in order for it to work at all. An identity federation is said to exist between an identity provider and a service provider, when the service provider accepts assertions regarding a user from the identity provider.

**[0026]** Efficiently maintaining a federated identity infrastructure is important, especially when member domains in the federation dynamically join or leave. Methods for effectively disseminating member domains in the current federation is crucial, as service providers rely on this information for making access control decisions. These access control decisions directly protect the security of the resources of the service provider and have to be made with high assurance. Because the role of service providers and identity providers are sometimes inter-changeable in web services, all participating domains in federated identity management systems must face the trust decisions implied by all possible cross-domain interactions.

**[0027]** In fact, most existing SSO solutions assume a preexisting trust relationship among providers and do not provide concrete mechanisms for the trust establishment between providers. The WS-Federation specification discusses several trust relationships between identity providers and service providers, including directed trust, indirected brokered trust, and chained trust. However, details on how the trust relationships and identity brokers can be instantiated are not given. This limitation hinders the wide deployment of SSO in web-service environments, because providers may be unknown to each other. Therefore, flexible, reliable, and secure trust establishment mechanisms need



APPL-1052 / IPR2024-00331 / Page 7 of 44
Apple Inc. v. Carbyne Biometrics, LLC

to be provided for federated identity management.

1.2 Contributions

**[0028]** Herein, a notarized federated identity management protocol is presented that supports flexible and efficient authentication of assertions, and enables a service provider to proactively obtain the trustworthiness information of unknown identity providers.

**[0029]** Important aspects of the problem of large-scale identity theft are also addressed. In June 2005, CardSystems® Solutions, a large credit card payment processor in Tucson, Arizona announced that forty million credit card numbers may have been stolen by computer hackers. The theft was a direct result of the company's illegal practice of retaining transaction records. It is insufficient to simply trust financial institutions' abilities and intentions for secure data management. Thus, rather than putting faith in the data management of financial institutions, a proactive solution is provided for protecting the disclosure of users' sensitive personal data with a cryptographic approach.

**[0030]** 1. A notarized federated identity management model is proposed that supports automatic user authentications when the providers are unknown to each other. This model introduces a *notary server*, which is owned by a trusted third-party to dynamically notarize assertions generated by identity providers. Assertions are generated by identity providers and registered with a trusted notary server. When a service provider needs to verify an assertion, it queries the notary server to get a notarized assertion. The notary information shows that the identity provider is trusted by the notary server, and proves the trustworthiness of the identity provider that generates the assertion. As an extra feature provided by the notary server, the presented federated identity management model reduces possible collusions between identity providers and service providers, and gives improved privacy protections for users.

**[0031]** 2. An efficient implementation of the federated identity management protocol is described with the existing *Secure Transaction Management System* (*STMS*). The notary server caches the assertions at a collection of responders deployed in the network. Even when the responders are located in insecure, untrusted locations, a service provider can easily identify a forged or tampered assertion so that the integrity of an



APPL-1052 / IPR2024-00331 / Page 8 of 44
Apple Inc. v. Carbyne Biometrics, LLC

assertions is maintained. The disclosed protocol is a concrete solution for a trust broker model proposed by existing federated identity management systems. Besides brokering trust, the solution offers additional features. Accountability may be supported by archiving signatures on requests and assertions. User privacy may be achieved by encrypting assertions stored by the notary server. Verification efficiency may be achieved, for example, by using the authenticated-dictionary technique implemented in STMS.

**[0032]** 3. A practical solution is also given for mitigating aspects of the identity theft problem. How it is used in the presented federated identity management protocol is also discussed. The cryptographic solution is based on the *Identity-Based Encryption* (*IBE*) scheme. The main feature of the cryptographic solution is that it enables one to proactively prevent the leaking of secret identity information.

### 1.3 Organization

**[0033]** The presented model for notarized federated identity management is described in Section 2. The STMS implementation of the notarized federated identity management protocol is presented in Section 3. In Section 4, an IBE-based authentication protocol is given. The security of the federated identity management protocol and the IBE-based authentication protocol is analyzed in Section 5. Related work is given in Section 6.

## 2 Notarized Federated Identity Management

**[0034]** The presented notarized federated identity management model introduces a *notary server*, a trusted third-party that dynamically maintains assertions generated by identity providers. Assertions are generated by identity providers and stored by the notary server. When a service provider needs to verify an assertion, it queries the notary server for a *notarized assertion* that shows the trustworthiness of the identity provider generating the assertion.

### 2.1 Notary Server and Notarized Assertion

**[0035]** In a notarized ID federation, a notary server is trusted by both identity providers and service providers. Identity providers that have good internet behavior and reputation are allowed to register with the notary server, and thus are trusted. The notary



APPL-1052 / IPR2024-00331 / Page 9 of 44
Apple Inc. v. Carbyne Biometrics, LLC

server stores the assertions generated by registered identity providers. A notary server supports at least two operations: SUBMIT and QUERY.

**[0036]** – SUBMIT($id, S_{id}, sig$): a registered identity provider *IdP* authenticates itself to the notary server, and submits via a secure channel the tuple (id, assertion, signature), denoted by ($id, S_{id}, sig$), to the notary server. The assertion $S_{id}$ states the attributes of an identity *id*, and the signature *sig* is signed by *IdP* on the assertion $S_{id}$. The notary server stores the tuple.

**[0037]** – QUERY(*id*): a service provider *SP* queries in a public (insecure) channel the notary server for assertions associated with identity *id*, and the notary server returns the *notarized assertion*(*s*).

**[0038]** A notarized assertion has a proof showing that the assertion is indeed stored by the notary server, which implies that the identity provider that generates the assertion is trustworthy. The reason for not using a secure channel in QUERY is for higher efficiency and scalability in a distributed environment. The challenge, thus, becomes how to efficiently generate and verify the notarized assertion, even when it is transmitted in an insecure channel. The presented solution is based on the authenticated dictionary technique, which is generally more scalable than using a signature scheme.

**[0039]** The notary server provides the assurance of the trustworthiness of assertions when identity providers are unknown to the service providers. The notary server acts as a bridge of trust between providers in web-service transactions. Another advantage of storing assertions on the notary server is the prevention of direct contact between identity providers and service providers. A notarized assertion does not contain the name of the identity provider. This further increases the difficulty of collusions among providers to discover private user information.

**[0040]** It is assumed that the notary server is trustworthy, and is trusted by all entities (e.g. users, identity providers, service providers). The security properties of the presented notarized federated identity management protocol are summarized below and are analyzed in Section 5.

**[0041]** – *Security* is defined such that no polynomial-time adversary can forge a



APPL-1052 / IPR2024-00331 / Page 10 of 44
Apple Inc. v. Carbyne Biometrics, LLC

notarized assertion that can be accepted by a service provider.

**[0042]** – *Secrecy* is defined intuitively such that the protocol does not leak any information about a notarized assertion to a (polynomial-time) adversary. This property provides privacy protection to the users.

**[0043]** – *Accountability* is defined such that identity providers should be held accountable for the assertions generated, and for any unauthorized information disclosure about the users.

**[0044]** Note that the notary server only certifies that the source of an assertion is trustworthy; it is not required to examine and certify the content of an assertion. In fact, the protocol, which is described next, deliberately avoids disclosing assertion contents to the notary server by encrypting the assertions. This feature is for the purpose of user privacy, and prevents the notary server from gaining knowledge of private user information.

2.2 Protocol

**[0045]** In this section, the protocol for the notarized federated identity management model is presented. At least the following entities participate in the protocol: a user, an identity provider, a service provider, and a notary server. The protocol gives an instantiation of operations SUBMIT and QUERY. Note that the roles of identity provider and service provider are interchangeable. For example, a bank can be the identity provider in one scenario and the service provider in another scenario.

**[0046]** Assume that the notary server knows the public keys of registered identity providers. In addition, the public key of the notary server is known by all of the providers. An exemplary schematic drawing of the protocol is shown in FIG. 1.

**[0047]** FIG. 1 illustrates an exemplary system 20 based on a notarized federated identity management model. The system 20 includes a service provider 22, a user 24, an identity provider 26 and a notary server 28. In further exemplary embodiments, the system 20 may comprise more than one service provider, user, identity provider and/or notary server. In the exemplary system 20 of FIG. 1, the four entities are configured to



APPL-1052 / IPR2024-00331 / Page 11 of 44
Apple Inc. v. Carbyne Biometrics, LLC

communicate as further described below.

**[0048]** Upon receiving a service request (1), a service provider 22 opens a secure channel with the user 24. The service provider 22 and the user 24 jointly generate a random session_ID (2) for the user's request. The identity provider 26 is given the session_ID by the user 24 in a secure channel (4), after successful authentication (3). The identity provider 26 computes the hashed session_ID (e.g. using a collision-resistant one-way hash function). The assertion associated with the user 24 is generated by the identity provider 26 with the hashed session_ID. To prevent information leaking, the assertion is blinded by the identity provider 26 before it is signed and submitted to the notary server 28 (5). A notary server 28 only knows the blinded assertion and hashed session_ID and cannot take advantage of the information for any service. This is important for the security of the protocol, when the notary service is distributed to untrusted notary responders in STMS in the next section (see FIG. 3). When submitting assertions, an identity provider 26 first authenticates to the notary server 28, which ensures the identity provider 26 is authorized. The user 24 may then query the notary server 28 for the index (6). The notary server 28, in response to the query, returns a notarized blinded assertion (7) to the user 24. The user 24 passes the notarized blinded assertion on to the service provider 22 (8) who then unblinds and verifies the notarized assertion (9).

**[0049]** In the protocol, the user 24 only needs to authenticate once to an identity provider 26. Subsequent requests for service from multiple service providers (e.g., 22) may not require additional authentication of the user 24. Nevertheless, for protecting personal privacy, the user 24 is given the ability to examine the contents of assertions to be given to the service providers 22 in this protocol. If the assertions are generated by the identity provider 26 according to the user's request, then they are passed on to the service providers 22. It is noted that having the user 24 involved in the identity management protocol for privacy purposes is a feasible solution. The process can be automated to minimize the user's manual participation.

**[0050]** Public parameters may include a collision-resistant one-way hash function, $H$, that takes a binary string of arbitrary length and hashes to a binary string of fixed length $k$: $H: \{0, 1\}^* \rightarrow \{0, 1\}^k$. For the blinding purpose, the public parameters also may include two public strings $P_1$ and $P_2$. Providers also agree on a symmetric-key encryption

scheme for blinding and unblinding assertions. The encryption and decryption of a message $x$ with a secret key $K$ are herein denoted as $E_K(x)$ and $D_K(x)$, respectively.

**[0051]** A non-limiting exemplary implementation of the protocol is described as follows.

**[0052]** 1. The user requests a service from a service provider *SP*. *SP* may require attribute information of the user needed to complete the service.

**[0053]** 2. *SP* opens a secure communication channel with the user. The user and *SP* each generate a random integer of the same length. They first exchange the cryptographic hashes of these integers as commitments using the secure channel, and then they exchange the integers using the secure channel. The session_ID $N$ is computed as the XOR of the two integers. *SP* also informs the user of the attribute names that are needed for the service (e.g., billing address, age).

**[0054]** 3. The user authenticates to the identity provider *IdP*. If the authentication is successful, the user opens a secure channel with *IdP*, and transmits a signed request that contains the session_ID and the required attribute names.

**[0055]** 4. *IdP* verifies and stores the signed request by the user. The signature is for the accountability purpose in case of dispute (see Section 5).

**[0056]** 5. *IdP* then computes the *index* of the assertion as the hash of session_ID concatenated with the public parameter $P_1$: $h = H(N, P_1)$. It then generates an assertion $S_h$ about the user using index $h$. For example, $S_h$ states that $h$ is a university student.

**[0057]** 6. To prevent information leaking, *IdP* blinds the assertion as follows.

**[0058]** (a) *IdP* computes the blinding factor $K$ as the hash of the session_ID concatenated with the public parameter $P_2$: $K = H(N, P_2)$.

**[0059]** (b) *IdP* encrypts $S_h$ with the symmetric encryption scheme, using $K$ as the secret key. This gives the *blinded assertion* $S'_h = E_K(S_h)$.

**[0060]** The blinded assertion $S'_h$ is signed by *IdP* with its private key, which gives

a signature $sig_h$.

**[0061]** 7. *IdP* runs SUBMIT($h$, $S'_h$, $sig_h$) with the notary server to submit tuple ($h$, $S'_h$, $sig_h$) through a secure channel as follows.

**[0062]** (a) *IdP* first authenticates to the notary server to establish a secure communication channel.

**[0063]** (b) *IdP* then transmits tuple ($h$, $S'_h$, $sig_h$) to the notary server.

**[0064]** (c) The notary server verifies signature $sig_h$, and stores ($S'_h$, $sig_h$) indexed by $h$. The signature is stored for accountability purposes.

**[0065]** 8. The user computes the index $h = H(N, P_1)$ from $N$ and $P_1$, and runs QUERY($h$) to obtain the assertion for $h$. The notary server processes the query as follows.

**[0066]** (a) The blinded assertion $S'_h$ associated with index $h$ is retrieved.

**[0067]** (b) The notary server *notarizes* the assertion $S'_h$, and returns the notarized assertion. Two exemplary approaches for the realization of notarized assertion are described in the following sections. Note that the QUERY operation between the user and the notary server may not require a secure channel.

**[0068]** 9. Once the user obtains the returned notarized blinded assertion, the user unblinds it with the blinding factor $K = H(N, P_2)$. This is done by decrypting $S'_h$ with $K$, which gives $S_h = D_K(S'_h)$. The user verifies that the assertion does not release any unauthorized personal information about the user.

**[0069]** 10. The notarized blinded assertion is then relayed from the user to the service provider, who verifies that it is notarized by the notary server. This implies that the identity provider *IdP* is trusted by the notary server. If the verification succeeds, the assertion $S'_h$ is unblinded in the same way as in Step 9. The attribute information is obtained from the assertion, and index $h$ is compared with the hash $H(N, P_1)$ of session_ID $N$ and $P_1$. The user is then granted the service if the verification passes. The service provider also stores the notarized assertion for accountability purposes.

**[0070]** The computation of blinding factor $K$ and index $h$ may be equivalent to



APPL-1052 / IPR2024-00331 / Page 14 of 44
Apple Inc. v. Carbyne Biometrics, LLC

using two different hash functions on the session_ID $N$, i.e., $h = H_1(N)$ and $K = H_2(N)$, where $H_1$ and $H_2$ are collision-resistant one-way hash functions.

**[0071]** Hash chains can be used in the protocol for preserving the freshness of session_ID – a new hash of the session_ID may be used between the user and the service provider, and the identity provider signs the tail ($n$-th hash) of the hashed session_ID. Hash chain was previously used for efficient user authentication. It was also recently used for user recognition based on weak authentication.

**[0072]** A straightforward realization of notarized assertions with signatures is described next. A more sophisticated solution based on the Secure Management Transaction System (STMS) is presented in Section 3.2.

## 3 Realization of Notarized Assertions

**[0073]** As a non-limiting example, notarized assertions can be realized using simple time-stamped signatures. The notary server can individually sign assertions and the current time-stamp with its private key. The notarized assertion includes this signature along with the assertion and time-stamp. To verify a notarized assertion, the service provider verifies the signature against the public key of the notary server, which can be obtained through usual means (e.g. a public key certificate). Because the notary server is trusted, the correct verification of the signature establishes the authenticity of the assertion about the index identifier $Hash(N, P_1)$. Namely, the security of the signature-based realization of notarized assertions follows directly from the security of the underlying signature scheme adopted.

**[0074]** Even though the service provider cannot tell which identity provider generated the assertion, trusting the notary server may be sufficient for authenticating the user in most applications (e.g. on-line shopping). In addition, not knowing the source of assertions may prevent, to some degree, the service providers from colluding with identity providers to discover information about the user. Because of the use of session_ID, the notary server and the service providers do not know the actual identities associated with assertions.

**[0075]** In the above approach, notarizing assertions can be a performance



APPL-1052 / IPR2024-00331 / Page 15 of 44
Apple Inc. v. Carbyne Biometrics, LLC

bottleneck because the notary server may need to sign every individual assertion. To improve the efficiency of the notary server, an improved realization of notarized assertions is provided using the secure transaction management system (STMS).

**[0076]** The main advantage of implementing notary assertions with STMS in comparison to the simple time-stamped signature approach is its high efficiency of computation. The notary server only needs to generate one signature as opposed to a signature for each assertion. In addition, STMS also provides a distributed architecture for fast real-time dissemination of assertion updates. STMS has been previously used to build an accredited domainkeys framework for secure e-mail systems. Next, the components and algorithms of STMS are first introduced. How to use STMS to scale up the notary service is then described.

### 3.1 Secure Transaction Management System (STMS)

**[0077]** The computational abstraction underlying STMS is a data structure called an *authenticated dictionary*, which is a system for publishing data and supporting authenticated responses to queries about the data. In an authenticated dictionary, the data originates at a secure central site, called *STMS source* and is distributed to servers scattered across the network, called *STMS responders*. The responders answer queries on behalf of the source about the data made by clients. It is desirable to delegate query answering to the responders for two reasons: (1) the source is subject to risks such as denial-of-service attacks if it provides services directly on the network, and (2) the large volume and diverse geographic origination of the queries require distributed servers to provide responses efficiently.

**[0078]** The main feature of STMS is that it can maintain trust even when responders are located in insecure, untrusted locations. That is, when a client submits a query to an STMS responder, it gets back not only an answer but also a proof of the answer. The client can easily validate the answer and determine that the responder has not been tampered with, while relying solely on trusted statements signed by the source. The design of STMS allows untrusted responders, which do not store private keys, to provide verifiable authentication services on behalf of a trusted source. This nonintuitive yet mathematically provable fact can be the key to achieving cost effectiveness.



APPL-1052 / IPR2024-00331 / Page 16 of 44
Apple Inc. v. Carbyne Biometrics, LLC

**[0079]** FIG. 2 shows an exemplary high-level description of the STMS parties and protocol. The entities depicted in the exemplary system 40 of FIG. 2 include a STMS source 42, a responder 44 and a user 46. The STMS source 42 maintains a source copy of the database 48 while the responder 44 maintains a database 50.

**[0080]** In FIG. 2, the STMS source 42 sends real-time updates to the responders 44 together with a special signed time-stamped fingerprint of the database 48 called the *basis*. A user's query to the responder 44 asks whether an element is contained in the authenticated dictionary maintained by STMS source 42. A responder replies to the query with an authenticated response. This includes the answer to the query, the proof of the answer, the basis and its signature signed by the STMS source 42. Informally speaking, the proof is a *partial fingerprint* of the database that, combined with the subject of the query, should yield the fingerprint of the entire database 48. A proof includes a very small amount of data (less than 300 bytes for most applications) and can be validated quickly against the signed basis by a client.

**[0081]** In the overview of STMS of FIG. 2, the source 42 pushes updates containing a signed basis to the responder 44. The responder 44 then answers user queries with a proof of the answer, and a copy of the signed basis from the source 42.

**[0082]** The signature of the basis can be verified using the source public-key and the current time quantum. If the signature is not valid, then the basis is not valid. This may indicate that the basis or the source public-key has been tampered with by the STMS responder 44 from which the values are obtained. The user 46 verifies the answer for element $x$ by simply hashing the values of the returned sequence $Q(x)$ of hash values in the given order, and comparing the result with the signed value $f(s)$, where $s$ is the basis value. If the two values agree, then the user 46 is assured of the validity of the answer at the time given by the time-stamp. The authenticated dictionary data structure can be implemented using a Merkle hash tree, for example. The data structure used in STMS system can be based on a skip list, which may be more efficient than a Merkle hash tree.

### 3.2 Realization of Notarized Assertions with STMS

**[0083]** Using STMS, the notary server includes a *notary source* and at least one *notary responder*. The notary source needs to be a trusted server that stores assertion



APPL-1052 / IPR2024-00331 / Page 17 of 44
Apple Inc. v. Carbyne Biometrics, LLC

inputs from identity providers. Notary responders can be strategically placed in geographically dispersed locations to accommodate fast queries. They obtain real-time updates from the notary source, and answer queries from users. Notary responders do not need to be trusted servers. The notarized assertions returned by them can be authenticated by anyone by verifying against the public key of the notary source.

**[0084]** With STMS, a notarized assertion returned by QUERY operation includes two parts: the assertion itself and a STMS proof. As described in the previous section, the proof can be a sequence of hash values of elements in the notary server for proving the existence of the assertion. The size of the proof can be quite compact, even for a large number of items in the notary server. Therefore, transmitting the proof can be quite fast. The service provider then obtains the signed STMS basis of the current time quantum from the notary responder, if it does not yet have it. The proof of the assertion is verified against the basis, and the signature of the basis is verified against the public key of the notary source. If the verification is successful, the request is granted. The signed basis remains the same for the duration of a time quantum, therefore it only needs to be obtained once for each time quantum. The rest of the notarized federated identity management protocol with STMS may follow the protocol described in Section 2.2.

**[0085]** Because notary responders are not required to be trusted servers, storing session_ID in the clear is not secure – a notary responder may attempt to impersonate a user with the session_ID for service. Note that opening a secure communication between the service provider and the notary responder does not solve this problem. The notarized federated identity management protocol presented in Section 2 is resilient to this problem, because assertions use hashed session_ID rather than the plain session_ID. In addition, the service provider transmits the plain session_ID to the user in a secure channel. An exemplary schematic drawing of the STMS-implemented notarized federated identity management protocol is shown in FIG. 3. The time quantum can be set to as short as orders of milliseconds to allow fast dissemination of assertions. The security can be based on the security of STMS, which has been previously proven.

**[0086]** FIG. 3. shows an exemplary schematic drawing of the STMS-implemented notarized federated identity management protocol. The exemplary system 60 depicted in FIG. 3 includes a service provider 62, a user 64, an identity provider 66, a notary server



APPL-1052 / IPR2024-00331 / Page 18 of 44
Apple Inc. v. Carbyne Biometrics, LLC

68 and a notary responder 70.

**[0087]** In FIG. 3, at each time quantum, the notary source 68 sends the signed basis and updates of the authenticated dictionary to the notary responder 70. The notary responder 70 answers a query for assertion by returning the signed basis and the proof corresponding to the queried element. In other respects, the protocol functions similarly to that illustrated by the system 20 of FIG. 1.

**[0088]** Next, an authentication protocol is presented that effectively reduces the identity theft problem. How to integrate the authentication protocol with the notarized federated identity management protocol is also described.

## 4 Reducing the Risks of Identity Theft

**[0089]** Several practical solutions against on-line identity theft have been proposed. Single sign-on systems have been criticized as providing weak protection against identity theft, because once an attacker successfully logs onto one site, the attacker will have no problem requesting services from other sites. Although intuitively this is true, single sign-on does not necessarily make it any easier for identity thieves. It has previously been argued that, in single sign-on systems, users only need to use one password and therefore are more likely to choose and memorize strong passwords. Strong passwords are shown to be an effective way to prevent break-ins.

**[0090]** In this section, causes of a successful identity theft are first analyzed. Then, a practical solution is given, and it is described how to use the scheme in the presented notarized federated identity management protocol.

### 4.1 Identity Theft and Its Causes

**[0091]** Identity theft is a type of crime in which an imposter obtains key pieces of personal information (e.g. Social Security number, driver's license numbers) in order to impersonate someone else. Although an identity thief might crack into a database to obtain personal information, it is believed that a thief is more likely to obtain information using Trojans or even old-fashion methods such as dumpster diving and shoulder surfing.

**[0092]** Observe that current authentication protocols, both physical and digital,



APPL-1052 / IPR2024-00331 / Page 19 of 44
Apple Inc. v. Carbyne Biometrics, LLC

are fundamentally susceptible to identity theft, even if an individual is careful in protecting sensitive information. Physical authentication protocols include, as examples, the procedures for obtaining a driver's license at a government office, opening a bank account, and applying for mortgage. Digital authentication protocols include the corresponding on-line transactions. In current solutions, key pieces of personal information are usually communicated in the clear or stored in the clear. This makes stealing information easier for identity thieves. Although SSL protocol encrypts communications between a user and a server, this does not prevent Trojan keyloggers, or shoulder surfing, because the user still needs to disclose and type in sensitive information, such as a social security number.

**[0093]** It is herein argued that this fundamental characteristic of the existing authentication protocols is one of the main causes of identity theft, namely using sensitive information in clear form for authentication. A simple and practical cryptographic protocol is proposed for authentication. The presented solution ties personal information to random secrets, which are used to interactively prove the ownership of the personal information but are never disclosed.

### 4.2 Motivation for Using IBE

**[0094]** In public key encryption schemes, the private key information is never disclosed. Yet, a challenge-response process can be used by a user to prove the possession of the private key to an identity provider. The private key is usually protected by encrypting it with a passphrase, and storing it in a portable device, such as a smart card or a USB flash drive. Observe that the private key is never disclosed in clear during transactions, hence it never appears in any printed form or display. Therefore, it is difficult for attackers to retrieve someone's private key using standard identity theft techniques. To steal the private key, an attacker would need to obtain the physical device and know the passphrase. In order to associate identity information with public keys, a presently-known encryption scheme is the Identity-Based Encryption (IBE) scheme. A public key in IBE will be the personal information (e.g., the social security number of an individual). For authentication, an individual not only needs to know the personal information (e.g., social security number), but also needs to prove possession of the corresponding private key for authentication.



APPL-1052 / IPR2024-00331 / Page 20 of 44
Apple Inc. v. Carbyne Biometrics, LLC

**[0095]** The idea of an identity-based encryption (IBE) scheme is that an arbitrary string can serve as a public key. The motivation for using IBE as opposed to conventional public-key encryption schemes is as follows. IBE reduces the need for public key certificates and certificate authorities because a public key can be associated with identity information (e.g. a user's social security number). In contrast, conventional encryption schemes (e.g. RSA) do not allow an arbitrary string to be used as a public key, and hence require key certification.

**[0096]** Using IBE, a user can disclose identity information to an identity provider without worrying about identity theft attacks such as shoulder surfing and dumpster diving. This is because the number is only used as a public key, and the user also needs to prove possession of the corresponding private key. The identity provider encrypts a challenge nonce using, for example, the user's social security number as the public key. The user receives the encrypted challenge nonce and decrypts it with the private key corresponding to the social security number. The private key is obtained by a third party, called a Private Key Generator (PKG) in IBE literature. To do that, the user authenticates himself to the PKG in the same way, for example, as he would authenticate himself to a passport office and obtains his private key from the PKG. The user returns the decryption result to the identity provider. Without knowing the private key associated with the social security number, an attack cannot make use of the number.

4.3 Identity-Based Encryption Scheme

**[0097]** The first scheme for identity-based encryption was based on the bilinear Diffie-Hellman assumption in the random oracle model. In IBE schemes, the private key generator (PKG) is responsible for generating private keys for all users, and therefore is a performance bottleneck for organizations with large number of users. Hierarchical identity-based encryption (HIBE) schemes were proposed to alleviate the workload of a root PKG by delegating private key generation and identity authentication to lower-level PKGs.

**[0098]** The Boneh-Franklin IBE includes four operations: SETUP, EXTRACT, ENCRYPT, and DECRYPT. In SETUP, the PKG takes a security parameter $k$, and returns *params* (system parameters) and the root secret key *SK*. The root private key is used to



APPL-1052 / IPR2024-00331 / Page 21 of 44
Apple Inc. v. Carbyne Biometrics, LLC

derive private keys for all other users and is only known to the PKG. In EXTRACT, the PKG uses *SK* to generate the private key $SK_{id}$ for a user with an ID. In ENCRYPT, a sender inputs *params*, a message *M* and the ID of the intended message recipient, and computes a ciphertext *C*. In DECRYPT, a user with an ID inputs *params*, *C*, and its private key $SK_{id}$, and returns the message *M*. These operations are used in the presented authentication protocol below.

4.4 A Cryptographic Authentication Protocol

**[0099]** It is proposed to use ID-based encryption scheme for implementing an authentication protocol for sensitive personal information. The presented protocol minimizes the exposure of secret personal information and thus is more robust against identity theft than existing authentication methods. Entities in the protocol include a user, an ID authority, an identity provider, and a revocation server controlled by the ID authority. (See FIG. 4) The authentication protocol may include the following operations: SETUP, REGISTER, AUTHENTICATE, and REFRESH. It may require an on-line revocation server maintained by the ID authority. The operations are defined as follows.

**[00100]** SETUP is run by an ID authority who takes as input a security parameter and outputs public parameters and master secret.

**[00101]** REGISTER is run by the ID authority to generate secret keys for personal information of a user. The ID authority takes as input the identity information of a user and outputs the corresponding private key. The private key is transmitted to the user via a secure channel.

**[00102]** AUTHENTICATE is for a user to authenticate his identity information to an identity provider. The identity provider first queries the revocation server to ensure that the user's public key is valid. If yes, the user then proves to the identity provider that he possesses the secret key associated with the identity information. If the user successfully proves the possession of the secret key, the output is true, otherwise the output is false.

**[00103]** REFRESH is for the ID authority to re-generate the private key associated with a user's identity information. The input is the identity information, and the output is a new private key. Refresh is run when the previous secret key expires, or the key is



APPL-1052 / IPR2024-00331 / Page 22 of 44
Apple Inc. v. Carbyne Biometrics, LLC

compromised and a new key needs to be generated, as non-limiting examples. The revoked public key may be put on a revocation server.

**[00104]** Refreshing the secret key of identity information can be tricky because the identity information typically does not change, e.g. a social security number. Subsequently it is shown how to use multiple pieces of identity information and on-line revocation checking to leverage this.

**[00105]** A diagram of an exemplary IBE-based authentication protocol is shown in FIG. 4. The exemplary system 80 of FIG. 4 includes an identity provider 82, a user 84, an identity (ID) authority 86 and a revocation server 88.

**[00106]** In FIG. 4, a user 84 registers an ID-based public key with an ID authority 86 (1). The ID authority 86 issues an ID-based private key (2). The user 84 submits the ID-based public key to an identity provider 82 for authentication (3). The identity provider 82 responds with a challenge ciphertext (4). The user 84 responds to the challenge with a result of decryption with the ID-based private key (5). The identity provider 82 consults a revocation server 88 to check whether the ID-based public key has been revoked (6). The revocation server 88 receives periodic updates of revoked ID-based public keys from the ID authority 86.

**[00107]** Below, an exemplary realization of the above operations with an IBE scheme are described.

**[00108]** 1. SETUP: The ID authority runs the PKG SETUP operation of IBE.

**[00109]** 2. REGISTER: A user requests an *ID-based private key* from an ID authority. The user may need to be physically present in the ID office, for example the passport office, with paper identifications such as passport, birth certificate. The ID authority authenticates the user's identity.

**[00110]** If the user's identity is verified, the ID authority generates the ID-based private key for the user. The ID authority runs the EXTRACT operation of IBE with the user's *ID-based public key*, which is the user's identity information concatenated with a unique serial number $l$. As a non-limiting example, $l$ can be the driver's license number. $l$ is used for revocation purposes. Because the identity information (e.g. social security



APPL-1052 / IPR2024-00331 / Page 23 of 44
Apple Inc. v. Carbyne Biometrics, LLC

number) cannot be easily revoked, one needs an additional replaceable field *l*. Note that *l* cannot be any random number, because using a random value as a public key requires public-key certification, which defies the purpose of identity-based encryption. In what follows, a driver's license number is used as *l*. The ID-based private key generated by EXTRACT is given to the user. The user's driver's license can be equipped with a smart card chip and store the private key.

**[00111]** 3. AUTHENTICATE: The user and the identity provider can engage in a challenge-response protocol as follows.

**[00112]** (a) The user gives his ID-based public key to the identity provider, which is the identity information concatenated with the driver's license number *l* to the identity provider.

**[00113]** (b) The identity provider picks a random nonce *m*. It runs ENCRYPT of IBE to encrypt *m* using the user's ID-based public key.

**[00114]** (c) The ciphertext is given to the user, who runs DECRYPT of IBE with his ID-based private key. If the user is unable to correctly decrypt the ciphertext, the authentication fails and returns false.

**[00115]** (d) The identity provider queries the revocation server maintained by the ID authority for the number *l* in the public key of the user. If *l* has been revoked, then the authentication fails. Otherwise, the authentication is successful and returns true.

**[00116]** 4. REFRESH: The ID authority can refresh the ID-based private key of the user as follows.

**[00117]** (a) The user authenticates his current ID-based public key to the ID authority.

**[00118]** (b) The ID authority puts the driver's license number *l* on the revocation server to indicate that *l* has been revoked.

**[00119]** (c) The ID authority generates a new driver's license number *l'* for the user. The new ID-based public key of the user associated with his identity information is that identity information concatenated with *l'*. For example, the public key is 999-99-9999 ◦



APPL-1052 / IPR2024-00331 / Page 24 of 44
Apple Inc. v. Carbyne Biometrics, LLC

1234567890, where 999-99-9999 is the social security number and 1234567890 is the new driver's license number *l'*.

**[00120]** (d) The ID authority runs EXTRACT of IBE to compute a new private key, which is transmitted to the user via a secure channel or in person, as non-limiting examples. The user stores the new ID-based private key in his smart card, for example.

**[00121]** The main advantage of this authentication protocol is that the secret personal information is not released during the transaction, thus helping to minimize identity theft attacks such as dumpster diving and shoulder surfing. The protocol can be used in any user authentication application. In particular, it can be used in any federated identity management system when a user authenticates his personal information with an identity provider. For example, assume a user is required to run the AUTHENTICATE algorithm with the identity provider when his social security number is needed. Without the corresponding private key, it is impossible for an identity thief to accomplish this.

**[00122]** The above protocol is suitable for authenticating extremely sensitive, or unique and permanent identity information, such as a social security number or credit card number, as non-limiting examples. It is also suitable for less sensitive information such as age, phone number, or address. Multiple attributes can be aggregated to form one key, in order to reduce the number of private keys required. Revocation in the protocol can use on-line revocation servers. Some sensitive information such as social security number may not be changed for a user. Therefore, the public key can be made to contain not only the social security number, for example, but also a replaceable number such as driver's license number. Efficient and scalable revocation service has been implemented previously.

**[00123]** This solution is not aimed to address identity thefts that involve stealing paper credentials (e.g. birth certificate, passport) to impersonate others. In the scheme, an attacker may be able to gather enough paper credentials of a victim and register in person with the ID authority a new identity-based public key as the victim. (e.g. This may be possible because the attacker is a close friend or relative of the victim.)

5 Security Analysis



APPL-1052 / IPR2024-00331 / Page 25 of 44
Apple Inc. v. Carbyne Biometrics, LLC

**[00124]** In this section, the security of the notarized federated identity management protocol is first analyzed, and then the IBE-based authentication protocol is analyzed.

5.1 Notarized Federated Identity Management

**[00125]** The security of the presented notarized federated identity management protocol is analyzed from the perspectives of the user, the identity provider, the service provider, and the notary server, as each of them may have different requirements on the security provided by the system. In what follows, assume the existence of a signature scheme that is secure against existential forgery by polynomial-time adversaries in the security parameter of the signature scheme. Existential forgery means that an adversary forges a signature that the notary server has not signed in the past. An adversary in this protocol can monitor traffics in unsecured channels, request for services, request the identity provider to blind assertions of her choice, and request the notary server to notarize assertions of her choice.

**[00126]** Assume that the notary server is trustworthy, and is trusted by all entities (e.g. users, identity providers, service providers). All entities are assumed to follow the federated identity management protocol presented in Section 2. The following theorem states the nonforgeability of a notarized assertion.

**[00127]** **Theorem 1.** *In the notarized federated identity management protocol, no polynomial-time adversary can successfully forge a valid notarized assertion that is not generated by the notary server.*

**[00128]** **Proof sketch:** Two exemplary implementations of the notarized assertion are given. One is based on a simple signature scheme, the other is based on STMS. In both implementations, forging a notarized assertion is equivalent to forging the signature of the notary server at a time quantum. This is infeasible, assuming the existence of a signature scheme that is secure against existential-forgery attacks. Therefore, the theorem holds.

**[00129]** For the privacy protection of a user, an important privacy requirement may be the secrecy of assertions. This is defined such that the protocol does not leak any information of the assertion to the adversary. The notarized federated identity



APPL-1052 / IPR2024-00331 / Page 26 of 44
Apple Inc. v. Carbyne Biometrics, LLC

management protocol satisfies the secrecy requirement. This is summarized in the following theorem.

**[00130]** **Theorem 2.** *Assume the existence of a collision-resistant one-way hash function, and a secure symmetric key encryption scheme. In the notarized federated identity management protocol, a polynomial-time adversary and untrusted notary responders cannot obtain any information from a blinded assertion.*

**[00131]** **Proof sketch:** It will be shown that (1) the key is difficult to guess and (2) the blinded assertion is pseudorandom. An assertion is encrypted by the identity provider using a symmetric key encryption scheme that is secure in the sense of an adversary's inability to distinguish the output from a random string. The secret key for the encryption/decryption is computed as $H(N, P_2)$, where $H$ is a collision-resistant one-way hash function, $N$ is the session ID, $P_2$ is a public parameter, and "," denotes concatenation. Given the public index $H(N, P_1)$ of an assertion, where $P_1$ is another public parameter, the secret key is still difficult to guess. This is because of the collision-resistant and one-way properties of the hash function $H$. In addition, the blinded assertion is indistinguishable from a random string, because of the security of the encryption scheme. Therefore, adversaries and untrusted notary responders cannot obtain any information from the blinded assertions.

**[00132]** For decentralized authorization systems such as the federated identity management, an important security requirement is accountability. To prevent possible disputes, identity providers should be held accountable for the assertions that they have generated. In addition, to prevent unauthorized information exchange among providers, users should be able to dispute any fraudulent assertion requests. These properties are achieved in the presented protocol.

**[00133]** **Theorem 3.** *In the notarized federated identity management protocol, the identity provider is held accountable for the assertions that it generates.*

**[00134]** **Proof:** The notary server stores the signed (blinded) assertion submitted by an identity provider in Step 7c of the notarized federated identity management protocol. In case of a dispute between a service provider and an identity provider on the validity of an assertion, the notary server reveals the signature, which is used to hold the



APPL-1052 / IPR2024-00331 / Page 27 of 44
Apple Inc. v. Carbyne Biometrics, LLC

identity provider accountable for generating the assertion. Therefore, Theorem 3 holds.

**[00135]** **Theorem 4.** *In the notarized federated identity management protocol, providers are held accountable for any unauthorized information exchange among them.*

**[00136]** **Proof:** In the protocol, an identity provider should only generate assertions based on a signed request from a user. The identity provider may be required to keep the signed requests for its own record in Step 4 of the notarized federated identity management protocol. Once unauthorized information sharing among providers is detected, the identity provider is not able to show any signed request by the user. Hence, it is responsible for the information leak.

**[00137]** **Theorem 5.** *The notarized federated identity management protocol is secure against replay attacks.*

**[00138]** It is easy to see that Theorem 5 holds because the session ID is randomly generated for each service request and the notarized assertions are generated by the notary server with the time-stamp information.

5.2 IBE-based Authentication

**[00139]** The security of the IBE-based authentication protocol is defined as the adversary's inability of impersonating a user. Formally, an IBE-based authentication protocol is secure if no polynomial-time adversary can distinguish with non-negligible probability the challenge ciphertexts of two nonces of the adversary's choice.

**[00140]** This security property is equivalent to the semantic security in the identity-based encryption scheme, which intuitively means that the adversary does not learn anything about the messages from observing the ciphertexts. The adversary in this protocol is allowed to monitor traffic in unsecured channels, request for the private keys of identities of the adversary's choice, request the identity authority to decrypt the challenge ciphertexts of the adversary's choice, and choose two challenge nonces whose ciphertexts are to be distinguished.

**[00141]** **Theorem 6.** *Given an identity-based encryption scheme that is semantic-secure against an adaptive polynomial-time adversary, the IBE-based authentication*



APPL-1052 / IPR2024-00331 / Page 28 of 44
Apple Inc. v. Carbyne Biometrics, LLC

*protocol is secure.*

**[00142]** It is easy to see that Theorem 6 holds because of the assumption of a secure IBE scheme. This theorem shows that it is infeasible for an adversary to impersonate a user if the user's private key is not compromised. This leads to the following conclusion that states the property of the presented authentication protocol in preventing identity theft. In the IBE-based authentication protocol, an adversary cannot successfully impersonate a user without stealing the user's tamper-resistant device that stores the identity private key.

## 6 Related Work

**[00143]** Existing anonymous credential systems are different from the exemplary embodiments of this invention at least for the reason that existing systems do not consider a federated identity infrastructure behind the providers. In comparison, the presented system focuses on how to manage user authentication in the more realistic setting of a federation of providers. The system can achieve simple pseudonym solutions and efficient single sign-on by taking advantage of the federated structure. In particular, one may not need a credential system, because the assertions can be short-lived and generated on-line by identity providers.

**[00144]** The presented identity-based solution has a similar goal to the approach of the previously proposed federated identity management solution noted above in the Background section, but there is one important difference. The presented solution allows personal data such as social security numbers and credit card numbers to be transmitted in the clear. Yet, when this information is used, the user needs to prove the possession of corresponding private keys. This requires minimal changes to existing financial and administrative infrastructure, as personal information in the scheme can be stored the same way as it is currently. IBE conveniently makes this possible, and, interestingly, this approach may also be more efficient than zero-knowledge proof-of-knowledge protocols.

**[00145]** The main difference between BBAE and the presented exemplary approach is that herein a notary mechanism is provided for authenticating assertions when *IdP* and *SP* are not previously known to each other.



APPL-1052 / IPR2024-00331 / Page 29 of 44
Apple Inc. v. Carbyne Biometrics, LLC

**[00146]** In contrast to the work of Bonatti and Samarati identified in the Background section, exemplary embodiments of this invention describe cryptographic solutions to control and manage information exchange. Their framework mainly focuses on the client-server model, whereas herein the architecture can include different types of providers.

**[00147]** In comparison to the location cross-checking approach, exemplary embodiments of the presented solution are simple and efficient to adopt. Because it can tie the secret identification information to a tamper-resistant smart card (e.g., driver's license), card theft can be easily noticed and reported by the card owner.

## 7 Conclusion

**[00148]** The exemplary embodiments of this invention may be implemented by one or more of the parties involved (e.g. user, service provider, identity provider, notary server). As non-limiting examples, the user may comprise an electronic device or a portable electronic device. Such an electronic device may itself comprise at least one data processor, at least one memory, a transceiver, and a user interface comprising a user input and a display device. As additional non-limiting examples, the user may comprise a software program or a plug-in application attached to another program. As a non-limiting example, the notary server may comprise a web service running on a distributed collection of computers on the internet. As a non-limiting example for implementing the exemplary embodiments, a cryptographic component may be employed. As further non-limiting examples, the cryptographic component may be a separate entity (e.g. an integrated circuit, an Application Specific Integrated Circuit or ASIC) or may be integrated with other components (e.g. a program run by a data processor, functionality enabled by a data processor). Such a cryptographic component may perform various functions including encryption, cryptographic hash, digital signature creation, digital signature verification and decryption, as non-limiting examples. As non-limiting examples, a system implementing the exemplary embodiments of this invention may comprise a private network (e.g. local area network – LAN), a public network (e.g. a publicly available wireless local area network – WLAN), or the internet.

**[00149]** In one non-limiting, exemplary embodiment, and as shown in FIG. 5, a



APPL-1052 / IPR2024-00331 / Page 30 of 44
Apple Inc. v. Carbyne Biometrics, LLC

method comprises: receiving through a data communication network an assertion generated by a first entity (box 501); notarizing the assertion to obtain a corresponding notarized assertion (box 502); and in response to receiving from a second entity via the same or a different data communication network a query corresponding to the assertion, returning the corresponding notarized assertion (box 503).

**[00150]** A method as above, wherein the first entity comprises an identity provider and the second entity comprises one of a user or a service provider. A method as in any of the above, wherein the method is executed by a notary entity such as a notary server. A method as in any above, wherein the assertion comprises a signed blinded assertion. A method as in any above, further comprising: in response to receiving, from the second entity via the same or a different data communication network, the query corresponding to the assertion, returning a proof corresponding to the query. A method as in any above, wherein the method is executed within a secure transaction management system (STMS). A method as above, wherein the notarized assertion comprises the assertion and a STMS proof, the method further comprising: obtaining, by the second entity, a signed STMS basis of a current time quantum from the first entity; verifying, by the second entity, the STMS proof using the signed STMS basis; and verifying, by the second entity, the signature of the STMS basis using a public key of the first entity.

**[00151]** A method as in any above, further comprising: archiving signatures on requests and assertions; encrypting the received assertions; and using an authenticated-dictionary technique to provide verification. A method as in any above, further comprising: encrypting the received assertion to obtain an encrypted assertion; and storing the encrypted assertion. A method as in any above, wherein the notarized assertion comprises a proof indicating that the assertion is stored by a notary entity. A method as in any above, wherein the assertion is received via a secure communication link and the query is received via an insecure communication link. A method as in any above, wherein the notarized assertion does not comprise an identification of the first entity.

**[00152]** A method as in any above, wherein receiving the assertion is performed in response to the first entity receiving a signed request from the second entity comprising session identification information, the method further comprising: receiving through the



APPL-1052 / IPR2024-00331 / Page 31 of 44
Apple Inc. v. Carbyne Biometrics, LLC

data communication network the session identification information from the first entity, wherein the session identification information comprises a random value. A method as above, wherein the session identification information comprises a hash value. A method as in any above, wherein notarizing the assertion comprises signing the assertion and a time-stamp with a private key and wherein the notarized assertion comprises the signature, the assertion and the time-stamp. A method as in any above, wherein the data communication network comprises at least one of a private network, a local area network, a public network, a publicly available wireless local area network, a distributed network and the internet.

**[00153]** A method as in any above, further comprising: determining a user private key for a corresponding user public key consisting of at least one piece of user identity information; and returning the user private key to the user as data to be stored on a storage medium. A method as above, further comprising: authenticating a user by engaging in a challenge-response protocol utilizing the user public key. A method as above, further comprising: querying a revocation entity to determine if the user public key has been revoked. A method as above, wherein the storage medium comprises at least one of a radio frequency identification component, flash memory, a tamper-resistant component and a microchip.

**[00154]** A method as in any above, executed or implemented by a computer program. A method as in any above, executed or implemented by an electronic device.

**[00155]** A method as in any above, executed by a data processor of an electronic device. A method as above, wherein the electronic device further comprises at least one memory and at least one transceiver. A method as in any above, wherein the electronic device comprises a component of a network.

**[00156]** In another non-limiting, exemplary embodiment, a computer program or a computer program product comprises program instructions embodied on a tangible computer-readable medium, execution of the program instructions resulting in operations comprising: receiving through a data communication network an assertion generated by a first entity; notarizing the assertion to obtain a corresponding notarized assertion; and in response to receiving from a second entity via the same or a different data communication



APPL-1052 / IPR2024-00331 / Page 32 of 44
Apple Inc. v. Carbyne Biometrics, LLC

network a query corresponding to the assertion, returning the corresponding notarized assertion.

**[00157]** A computer program product as above, wherein the first entity comprises an identity provider and the second entity comprises one of a user or a service provider. A computer program product as in any of the above, executed by a notary entity such as a notary server. A computer program product as in any above, wherein the assertion comprises a signed blinded assertion. A computer program product as in any above, further comprising: in response to receiving, from the second entity via the same or a different data communication network, the query corresponding to the assertion, returning a proof corresponding to the query. A computer program product as in any above, executed within a secure transaction management system (STMS). A computer program product as above, wherein the notarized assertion comprises the assertion and a STMS proof, execution of the program instructions resulting in operations further comprising: obtaining, by the second entity, a signed STMS basis of a current time quantum from the first entity; verifying, by the second entity, the STMS proof using the signed STMS basis; and verifying, by the second entity, the signature of the STMS basis using a public key of the first entity.

**[00158]** A computer program product as in any above, execution of the program instructions resulting in operations further comprising: archiving signatures on requests and assertions; encrypting the received assertions; and using an authenticated-dictionary technique to provide verification. A computer program product as in any above, execution of the program instructions resulting in operations further comprising: encrypting the received assertion to obtain an encrypted assertion; and storing the encrypted assertion. A computer program product as in any above, wherein the notarized assertion comprises a proof indicating that the assertion is stored by a notary entity. A computer program product as in any above, wherein the assertion is received via a secure communication link and the query is received via an insecure communication link. A computer program product as in any above, wherein the notarized assertion does not comprise an identification of the first entity.

**[00159]** A computer program product as in any above, wherein receiving the assertion is performed in response to the first entity receiving a signed request from the



APPL-1052 / IPR2024-00331 / Page 33 of 44
Apple Inc. v. Carbyne Biometrics, LLC

second entity comprising session identification information, execution of the program instructions resulting in operations further comprising: receiving through the data communication network the session identification information from the first entity, wherein the session identification information comprises a random value. A computer program product as above, wherein the session identification information comprises a hash value. A computer program product as in any above, wherein notarizing the assertion comprises signing the assertion and a time-stamp with a private key and wherein the notarized assertion comprises the signature, the assertion and the time-stamp. A computer program product as in any above, wherein the data communication network comprises at least one of a private network, a local area network, a public network, a publicly available wireless local area network, a distributed network and the internet.

**[00160]** A computer program product as in any above, execution of the program instructions resulting in operations further comprising: determining a user private key for a corresponding user public key consisting of at least one piece of user identity information; and returning the user private key to the user as data to be stored on a storage medium. A computer program product as above, execution of the program instructions resulting in operations further comprising: authenticating a user by engaging in a challenge-response protocol utilizing the user public key. A computer program product as above, execution of the program instructions resulting in operations further comprising: querying a revocation entity to determine if the user public key has been revoked. A computer program product as above, wherein the storage medium comprises at least one of a radio frequency identification component, flash memory, a tamper-resistant component and a microchip.

**[00161]** A computer program product as in any above, executed by a data processor of an electronic device. A computer program product as above, wherein the electronic device further comprises at least one memory and at least one transceiver. A computer program product as in any above, wherein the electronic device comprises a component of a network.

**[00162]** In another non-limiting, exemplary embodiment, a system comprises: a notary component configured to receive, via a data communication network, an assertion generated by a first entity and to notarize the assertion to obtain a corresponding



APPL-1052 / IPR2024-00331 / Page 34 of 44
Apple Inc. v. Carbyne Biometrics, LLC

notarized assertion; and a responder component configured to receive the notarized assertion from the notary component and, in response to receiving from a second entity via the same or a different communication network a query corresponding to the assertion, to return the corresponding notarized assertion.

**[00163]** A system as above, wherein the notary component and the responder component are embodied in a same electronic device. A system as in any of the above, further comprising the first entity and the second entity. A system as in any of the above, wherein the first entity comprises an identity provider and the second entity comprises one of a user or a service provider. A system as in any of the above, wherein the system comprises a secure transaction management system (STMS). A system as in any of the above, wherein the notarized assertion comprises the assertion and a STMS proof, wherein the second entity is configured to obtain a signed STMS basis of a current time quantum from the first entity, to verify the STMS proof using the signed STMS basis and to verify the signature of the STMS basis using a public key of the first entity.

**[00164]** A system as in any of the above, further comprising a third entity configured to determine a user private key for a corresponding user public key consisting of at least one piece of user identity information and to return the user private key to the second entity as data to be stored on a storage medium. A system as in any of the above, wherein the first entity is further configured to authenticate the second entity by engaging in a challenge-response protocol utilizing the user public key. A system as in any of the above, wherein the first entity is further configured to query a revocation entity to determine if the user public key has been revoked. A system as in any of the above, wherein the storage medium comprises at least one of a radio frequency identification component, flash memory, a tamper-resistant component and a microchip. A system as in any of the above, wherein the first entity comprises the third entity.

**[00165]** A system as in any of the above, wherein the assertion comprises a signed blinded assertion. A system as in any of the above, wherein the responder component is further configured, in response to receiving, from the second entity via the same or a different data communication network, the query corresponding to the assertion, to return a proof corresponding to the query. A system as in any of the above, wherein the notary component is further configured to archive signatures on requests and assertions, to



APPL-1052 / IPR2024-00331 / Page 35 of 44
Apple Inc. v. Carbyne Biometrics, LLC

encrypt the received assertions and to use an authenticated-dictionary technique to provide verification. A system as in any of the above, wherein the notary component is further configured to encrypt the received assertion to obtain an encrypted assertion and to store the encrypted assertion. A system as in any of the above, wherein the notarized assertion comprises a proof indicating that the assertion is stored by the notary component.

**[00166]** A system as in any of the above, wherein the assertion is received via a secure communication link and the query is received via an insecure communication link. A system as in any of the above, wherein the notarized assertion does not comprise an identification of the first entity. A system as in any of the above, wherein receiving the assertion is performed in response to the first entity receiving a signed request from the second entity comprising session identification information, wherein the notary component is further configured to receive via the data communication network the session identification information from the first entity, wherein the session identification information comprises a random value. A system as in any of the above, wherein the session identification information comprises a hash value. A system as in any of the above, wherein notarizing the assertion comprises signing the assertion and a time-stamp with a private key and wherein the notarized assertion comprises the signature, the assertion and the time-stamp. A system as in any of the above, wherein the data communication network comprises at least one of a private network, a local area network, a public network, a publicly available wireless local area network, a wireless local area network, a distributed network and the internet.

**[00167]** Generally, various exemplary embodiments of the invention can be implemented in different mediums, such as software, hardware, logic, special purpose circuits or any combination thereof. As a non-limiting example, some aspects may be implemented in software which may be run on a computing device, while other aspects may be implemented in hardware.

**[00168]** The foregoing description has provided by way of exemplary and non-limiting examples a full and informative description of the best method and apparatus presently contemplated by the inventors for carrying out the invention. However, various modifications and adaptations may become apparent to those skilled in the relevant arts in



APPL-1052 / IPR2024-00331 / Page 36 of 44
Apple Inc. v. Carbyne Biometrics, LLC

view of the foregoing description, when read in conjunction with the accompanying drawings and the appended claims. However, all such and similar modifications of the teachings of this invention will still fall within the scope of this invention.

**[00169]** Furthermore, some of the features of the preferred embodiments of this invention could be used to advantage without the corresponding use of other features. As such, the foregoing description should be considered as merely illustrative of the principles of the invention, and not in limitation thereof.



APPL-1052 / IPR2024-00331 / Page 37 of 44
Apple Inc. v. Carbyne Biometrics, LLC

# CLAIMS

What is claimed is:

1. A method comprising:

receiving through a data communication network an assertion generated by a first entity;

notarizing the assertion to obtain a corresponding notarized assertion; and

in response to receiving from a second entity via the same or a different data communication network a query corresponding to the assertion, returning the corresponding notarized assertion.

2. A method as in claim 1, wherein the assertion comprises a signed blinded assertion.

3. A method as in any preceding claim, further comprising:

in response to receiving from the second entity via the same or a different data communication network the query corresponding to the assertion, returning a proof corresponding to the query.

4. A method as in any preceding claim, wherein the method is executed within a secure transaction management system (STMS), wherein the notarized assertion comprises the assertion and a STMS proof, the method further comprising:

obtaining, by the second entity, a signed STMS basis of a current time quantum from the first entity;

verifying, by the second entity, the STMS proof using the signed STMS basis; and

verifying, by the second entity, the signature of the STMS basis using a public key of the first entity.

5. A method as in any preceding claim, further comprising:

archiving signatures on requests and assertions;



APPL-1052 / IPR2024-00331 / Page 38 of 44
Apple Inc. v. Carbyne Biometrics, LLC

encrypting the received assertions; and

using an authenticated-dictionary technique to provide verification.

6. A method as in any preceding claim, further comprising:

encrypting the received assertion to obtain an encrypted assertion; and

storing the encrypted assertion, wherein the notarized assertion comprises a proof indicating that the assertion is stored by a notary entity.

7. A method as in any preceding claim, wherein the notarized assertion does not comprise an identification of the first entity.

8. A method as in any preceding claim, wherein receiving the assertion is performed in response to the first entity receiving a signed request from the second entity comprising session identification information, the method further comprising:

receiving via the data communication network the session identification information from the first entity, wherein the session identification information comprises a random value.

9. A method as in any preceding claim, further comprising:

determining a user private key for a corresponding user public key consisting of at least one piece of user identity information; and

returning the user private key to the user as data to be stored on a storage medium.

10. A method as in claim 9, further comprising:

authenticating a user by engaging in a challenge-response protocol utilizing the user public key; and

querying a revocation entity to determine if the user public key has been revoked.

11. A computer program product comprising program instructions embodied on a tangible computer-readable medium, execution of the program instructions resulting in operations comprising:

receiving through a data communication network an assertion generated by a first



APPL-1052 / IPR2024-00331 / Page 39 of 44
Apple Inc. v. Carbyne Biometrics, LLC

entity;

notarizing the assertion to obtain a corresponding notarized assertion; and

in response to receiving from a second entity via the same or a different data communication network a query corresponding to the assertion, returning the corresponding notarized assertion.

12. A computer program product as in claim 11, execution of the program instructions resulting in operations further comprising:

encrypting the received assertion to obtain an encrypted assertion; and

storing the encrypted assertion, wherein the notarized assertion comprises a proof indicating that the assertion is stored by a notary entity.

13. A computer program product as in claim 11 or 12, execution of the program instructions resulting in operations further comprising:

determining a user private key for a corresponding user public key consisting of at least one piece of user identity information; and

returning the user private key to the user as data to be stored on a storage medium.

14. A system comprising:

a notary component configured to receive, via a data communication network, an assertion generated by a first entity and to notarize the assertion to obtain a corresponding notarized assertion; and

a responder component configured to receive the notarized assertion from the notary component and, in response to receiving from a second entity via the same or a different communication network a query corresponding to the assertion, to return the corresponding notarized assertion.

15. A system as in claim 14, wherein the notary component and the responder component are embodied in a same electronic device.

16. A system as in claim 14 or 15, further comprising the first entity and the second entity, wherein the first entity comprises an identity provider and the second entity



APPL-1052 / IPR2024-00331 / Page 40 of 44
Apple Inc. v. Carbyne Biometrics, LLC

comprises one of a user or a service provider.

17. A system as in one of claims 14-16, further comprising a third entity configured to determine a user private key for a corresponding user public key consisting of at least one piece of user identity information and to return the user private key to the second entity as data to be stored on a storage medium.

18. A system as in one of claims 14-17, wherein the first entity is further configured to authenticate the second entity by engaging in a challenge-response protocol utilizing the user public key and to query a revocation entity to determine if the user public key has been revoked.

19. A system as in one of claims 14-18, wherein the notary component is further configured to encrypt the received assertion to obtain an encrypted assertion and to store the encrypted assertion, wherein the notarized assertion comprises a proof indicating that the assertion is stored by the notary component.

20. A system as in one of claims 14-19, wherein the notarized assertion does not comprise an identification of the first entity.



APPL-1052 / IPR2024-00331 / Page 41 of 44
Apple Inc. v. Carbyne Biometrics, LLC

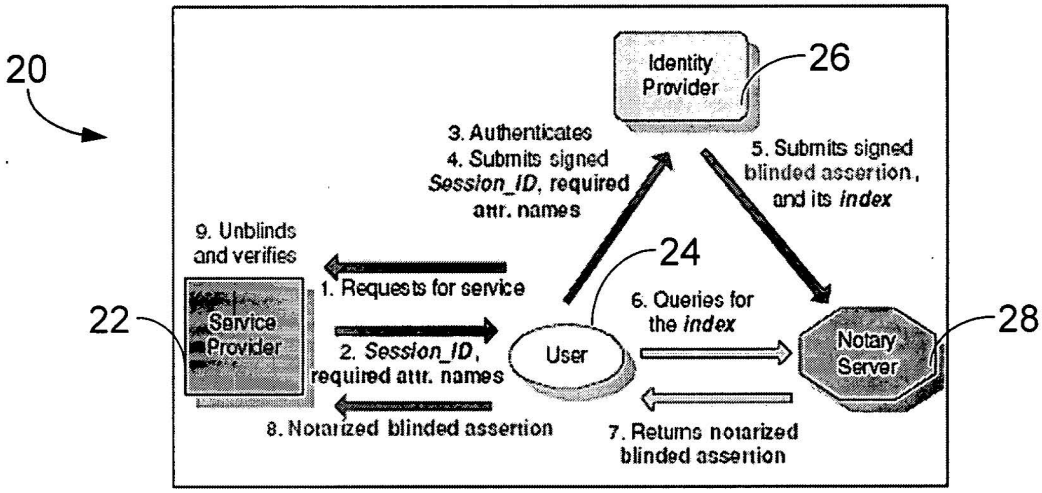

20
Identity Provider
26
3. Authenticates
4. Submits signed Session_ID, required attr. names
5. Submits signed blinded assertion, and its index
9. Unblinds and verifies
24
1. Requests for service
6. Queries for the index
22
Service Provider
2. Session_ID, required attr. names
User
Notary Server
28
8. Notarized blinded assertion
7. Returns notarized blinded assertion

FIG. 1

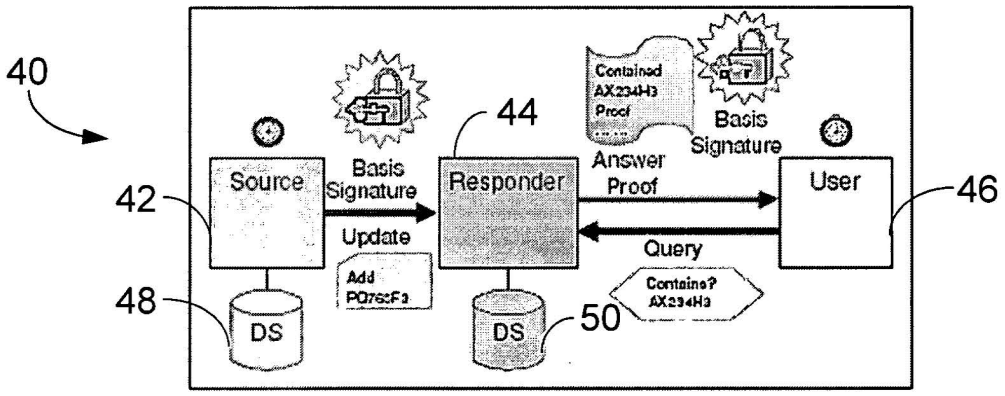

40
Contained AX234H3 Proof
Basis Signature
44
Source
Basis Signature
Responder
Answer Proof
User
42
46
Update
Query
Add PQ7ecF3
Contains? AX234H3
48
DS
DS
50

FIG. 2

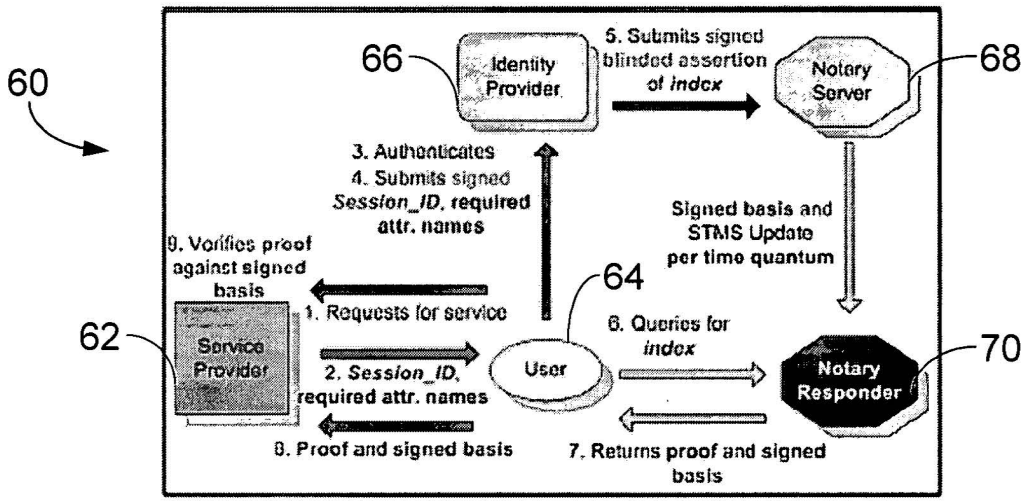

60
66
Identity Provider
5. Submits signed blinded assertion of index
Notary Server
68
3. Authenticates
4. Submits signed Session_ID, required attr. names
Signed basis and STMS Update per time quantum
9. Verifies proof against signed basis
64
1. Requests for service
62
Service Provider
2. Session_ID, required attr. names
User
6. Queries for index
Notary Responder
70
8. Proof and signed basis
7. Returns proof and signed basis

FIG. 3

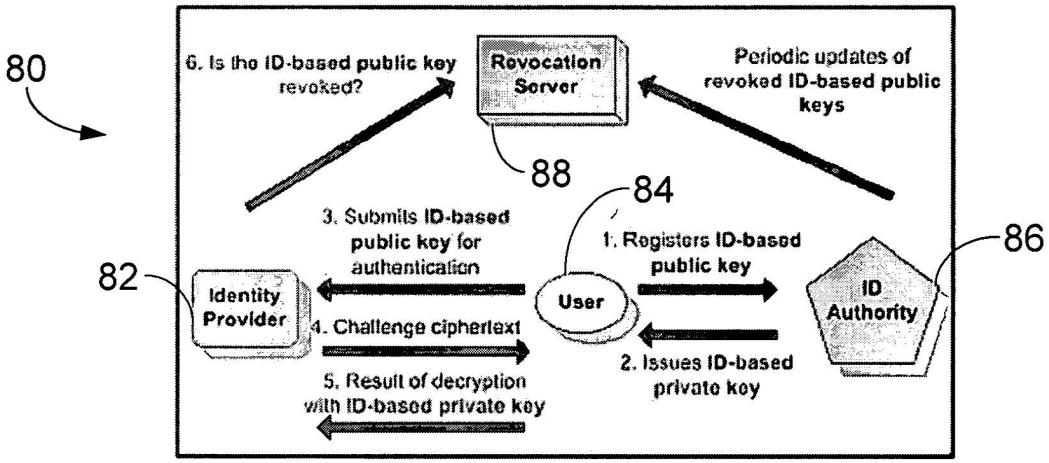

80
6. Is the ID-based public key revoked?
Revocation Server
Periodic updates of revoked ID-based public keys
88
84
3. Submits ID-based public key for authentication
1. Registers ID-based public key
86
82
Identity Provider
User
ID Authority
4. Challenge ciphertext
2. Issues ID-based private key
5. Result of decryption with ID-based private key

FIG. 4

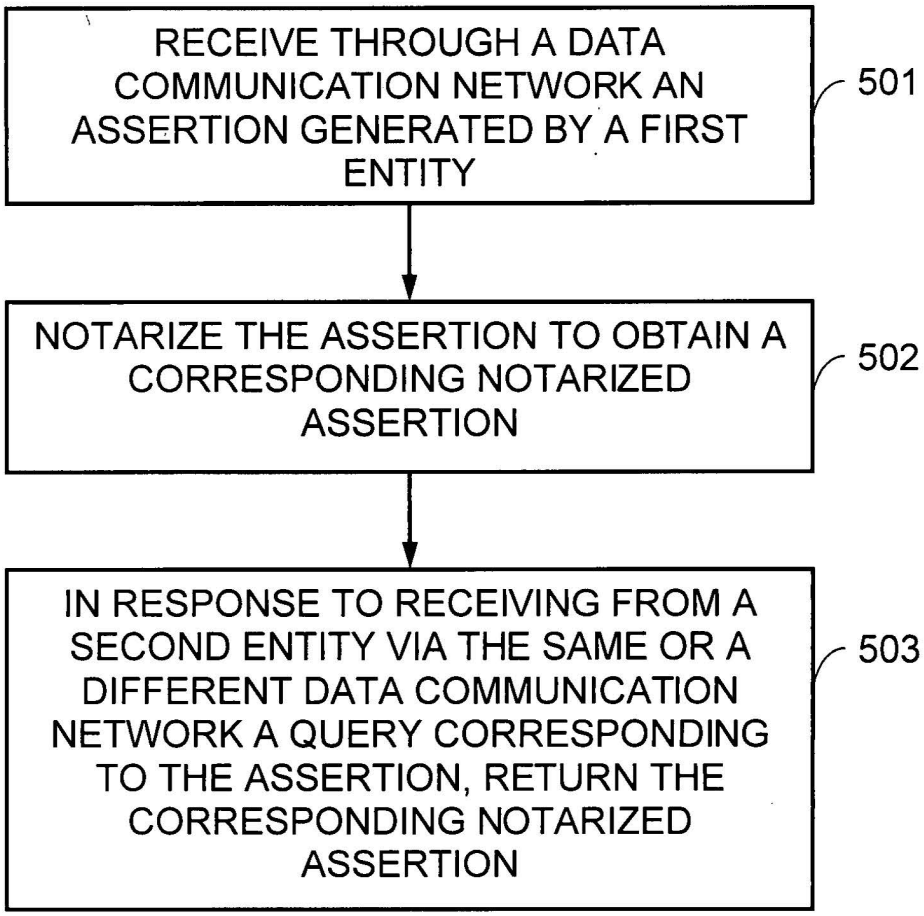

RECEIVE THROUGH A DATA COMMUNICATION NETWORK AN ASSERTION GENERATED BY A FIRST ENTITY
501
NOTARIZE THE ASSERTION TO OBTAIN A CORRESPONDING NOTARIZED ASSERTION
502
IN RESPONSE TO RECEIVING FROM A SECOND ENTITY VIA THE SAME OR A DIFFERENT DATA COMMUNICATION NETWORK A QUERY CORRESPONDING TO THE ASSERTION, RETURN THE CORRESPONDING NOTARIZED ASSERTION
503

FIG. 5