UNIQuE: A User-Centric Framework for Network Identity Management

Jörn Altmann[9,7], Rajarajan Sampath[9]

[9]School of Information Technology
International University in Germany
76646 Bruchsal, Germany
jorn.altmann@acm.org, rajarajan.sampath@i-u.de

[7]Techno-Economics & Policy Program, College of Engineering
Seoul National University
Seoul 151-744, South Korea
jorn.altmann@acm.org

***Abstract*—Network identity management system, in theory, is conceived as the solution to many identity-related issues burgeoning day-to-day. These issues, which need to be addressed, range from managing the outburst of user identities to protecting user interests as well as business interests. This paper proposes a framework for network identity management on the Internet that addresses these issues from a user-centric point of view. After discussing the challenges and opportunities of a user-centric identity management system, we describe the architecture of our framework called UNIQuE in detail. The architecture comprises components such as a security infrastructure, a trust subsystem, an inter-provider communication system, and a repository system. In essence, the goal of this framework is to specify a comprehensive, user-centric solution to all identity-related issues, which also vouches for effortless maintenance. The fundamental difference to existing systems is its integrating approach to many usually separately considered, identity-related issues.**

Keywords—Identity, Identity Management, Federated Network Identity Management, AAA system, Trust, Privacy, Security.

I. INTRODUCTION

Identity management, today, is multifaceted and embraces a broad spectrum of meanings. In enterprises, the focus is still on internal consolidation of different systems (e.g. customer-relationship management systems) and on integrating different access channels. For the Internet users, the focus is on enabling people to effortlessly manage their identities including free choice of roles and pseudonyms, the transfer of credentials from one pseudonym to another pseudonym of the same person, and appropriate user interfaces. The gap between these facets is wide. This framework attempts to address identity management as a central problem. The framework envisions a service-orientated Internet, in which many services interoperate across businesses. In such an environment, an effective identity system is indispensable, which provides solutions for, if not all, almost all existing identity issues.

Most of the existing identity management solutions (or frameworks) do not cross the boundaries of specified businesses in the name of security, privacy and more importantly "subscriber-user-base". Besides, why would any business want to share its most invaluable user-base with other organizations? At the outset, the question seems genuine but considering the fact that common Internet users are being overloaded with digital identities implies that most popular systems in the world holds "redundant" user data. For example, assume that User A has registered with *www.yahoo.com* as well as *www.ebay.com* and *www.amazon.com*. In order to function effectively, these three Identity management systems independently store user data, maintain user credentials, and check resource access data. This situation is true for many services and many users. It is apparent that an effective integration of these data will minimize cost and process overhead. To initiate an effective integration, it becomes critical to design an identity (backbone) system that spans services. However, such integration should not be misconstrued by organizations as means to takeaway their business intelligence. This integration is merely an effort to sort out inefficiencies of the existing identity architecture in place. Service-providers still have control over the respective subscriber-user-base. Therefore, from the perspective of service providers, the proposed solution will not sabotage service providers' interest but provide a unifying identity framework interoperating across services.

While the advent of information technology has fundamentally changed the efficiency of business, it is also significantly contributing to the ever-growing erosion of privacy amongst individuals. Likewise, there is also a growing consensus amongst the legislators across the world that individual's rights of privacy and the protection of personal data is equally applicable in the context of the Information Society as it is in the off-line world [49]. Therefore, from the user's perspective, this framework makes an attempt to organize user data that currently lies randomly scattered across the Internet and, moreover, attempts to restore the equilibrium of control over personal data back to the individual.

In essence, deploying such an identity system will curtail operation redundancy and improves data synchronization across applications. Moreover, service providers can focus on building systems of business importance rather than worry about these secondary functions of an identity system. Likewise, from the user's point of view, this framework provides a seamless-integration of user-data and an architecture for managing user data.

Additionally, the identity system can also be orchestrated to capture the "trust"-level of users. This trust factor can influence decisions, beginning from ascertaining user's trust to provisioning access to other users/enterprise resources. It can also pave way to internal consolidation with respect to access management in enterprises. Eventually, such a system can offer a strong basis for "personalized" and "context-sensitive"

information management, for planning of shared resources, and for CRM for networked business. It can also be integrated with different ERP-systems with a standard interface.

Moreover, the paper proposes the integration of biometrical information in order to increase the security level of private data. Taking into consideration the ethical issues of using biometric information, the proposed framework is designed to guarantee the security of the biometric data and to ensure the privacy of the user.

The existing commercial solutions for identity management have many shortcomings with respect to maintaining passwords, securing personal data, or financial details. Developing a solution to these issues requires definitions of roles, responsibilities, and relationships. Of course, there are many solutions to it. This paper proposes one such a solution. It defines a framework for identity management that is called UNIQuE, standing for "User Network Identity Management in a Secure Framework". The paper is organized as follows. The next section describes the state of the art in identity management. Section III enlists the criteria/features of the proposed identity management system, while Section IV describes the system architecture. The architecture comprises a security infrastructure, a trust system, an inter-provider communication system, and a repository backbone system.

II. STATE-OF-THE-ART

The concept of naming has been forever with humans. Today, the Internet has applied this concept to its benefits. As a result, a number of identity-related issues (e.g. authentication and authorization) have been successfully resolved. A plethora of products exists. However, looking at the Internet at large, the need for a globe-spanning identity management system similar to DNS becomes clear. A system of such a magnitude should not only identify individuals inside the system, but also link all the possible applicable data to it. Such a system must also be portable, interoperable, secure, supporting a huge number of identity-related attributes, and, above all, must provide the much-needed privacy [1]. Today, research in this field is abundant [3][4][5][6][20][21]. This research and the available commercial products characterize the importance of the concept of identity management.

A. Identity deployment architectures

In general, Silo, Walled Garden and Federation [2] are broad classifications of commonly observed identity deployment architectures.

Silo (or isolated identity management) is a simple architecture, which requires each service provider to maintain a unique ID for each user. Although this approach appears to be simpler for service providers, it is not only laborious but also problematic for users. As studies show, users, who register with several service providers, routinely forget their passwords for less frequently used accounts [32]. This has a significant financial effect. On average, $45 is spent on password reset each time a user forgets a password [27].

Walled Garden (or centralized identity management) is where all service providers can typically rely on one single identity and credential provider to establish user's credential [32]. Such a setup can directly rectify the shortcomings of Silo easily. From the user's point, a single set of credentials is by far much simpler to manage than a plethora of identities. Adding to that, a rather simple extension of this approach is Single Sign On (SSO). A user that is authenticated by one service provider is also considered authenticated by other service providers. However, this all or nothing property has an inherent weakness. Once the significant barrier of protection is compromised, a malicious user enjoys unbridled access to all resources. The protection can be compromised through weak passwords, embedded login forms, or weak cryptography [48].

Lastly, an identity federation management exists where a group of service providers operate under a set of agreements, standards, and technologies. Even though each service provider might have a unique identifier, they recognize the identifiers of other service providers and, thereby, consider a user who has been authenticated by another service provider to be authenticated as well. This is in effect a Single Sign On approach [32]. However, the real distinction between federated and centralized identity management is that service providers have their own unique identifiers and credentials. Given the heterogeneity of service providers, the federated identity approach is a widely accepted approach. Nonetheless, the many possible scenarios of interaction between service providers make such a system by far the most complex. For instance, since not all service providers (SPs) have the same security level, it is difficult to regulate the information flow across SPs.

B. Existing Implementations

Liberty Alliance, Microsoft TrustBridge, Project GUIDE, FIDIS, PRIME and Kerberos Authentication mark significant research undertakings in this field. Interestingly some approaches like Kerberos are not new and, nevertheless, sport a very strong security mechanism. More recent systems address issues like identity, trust, and biometrics in more detail.

1) Kerberos Authentication

Kerberos provides means to verify the identities of principals, (e.g. a workstation user or a network server) on an open (unprotected) network [9]. This is accomplished without relying on assertions by the host operating system, without basing trust on host addresses, without requiring physical security of all the hosts on the network, and under the assumption that packets traveling along the network can be read, modified, and inserted at will. Kerberos performs authentication under these conditions as a trusted third-party authentication service by using conventional (shared secret key) cryptography [7]. Once properly authenticated, identity, privacy, and integrity are assured.

2) Liberty Alliance

Project Alliance resorts to a federated network identity management scheme [13]. Federated identity network management is based on the vision that rapid access to resources through single sing-on (SSO) can be granted to users even if these resources are scattered across domains. Liberty Alliance does not mandate a central storage of user information by a single identity provider. Instead, a group of service providers, in conjunction with one or more identity providers,

are linked together, work in unison, and share user's identity information. The Liberty framework provides specifications for associating, connecting, and binding multiple accounts for a given principal at various sites within a Circle of Trust [34].

3) Microsoft Passport and TrustBridge

Even prior to Liberty Alliance, Microsoft Passport came up with a cross-domain web SSO proposal. Microsoft Passport's single sign-on authentication service allows users to sign in to any partner website based on a single set of credentials (user id and password), stored by the passport service. Passport uses a centralized authentication service, very similar to the principles behind Kerberos. Nevertheless, owing to several limitations that weakened the architecture, several security attacks were mounted on Microsoft Passport [14]. Thereafter, Microsoft signaled a shift from a centralized authentication system to a distributed system, just like Liberty Alliance, and code-named it "TrustBridge". Microsoft TrustBridge is also designed to achieve cross-organizational resource sharing through identity federation. It allows businesses that use Windows Active Directory to recognize and share identities with other organizations running Windows, .NET Passport service, or other Kerberos-based systems that support the WS-Security protocol [3]. TrustBridge's infrastructure is not Kerberos compliant, but emulates the Kerberos authentication system, and, therefore, envisions mutual authentication and transitive trust across domains [10].

4) Project Shibboleth

Shibboleth, a joint project of Internet2/MACE and IBM, aims to develop an architecture that facilitates cross-institutional resource sharing. It deploys a web access control infrastructure capable of safeguarding user privacy as well as explicitly allowing user to manage the disclosure of personal information. Shibboleth, unlike Kerberos or X.509-based PKI, does not mandate the use of one particular system; instead it opts for interoperability across heterogeneous security systems. Project Shibboleth defines standards and specifications for a framework facilitating cross-institutional resource sharing [15].

5) Project PRIME and Project GUIDE

The European Union started to fund a project on identity management in 2004, called PRIME [35]. The PRIME project focuses on privacy issues and, thereby, strives to address all the necessary areas of a user-centric approach to identity management. However, PRIME's approach is very wide. It aims to build an identity management solution, based on a set of well-defined application scenarios such as healthcare, location-based services, and ambient intelligence. PRIME has not provided a concrete unifying framework to user-centric identity management yet.

The project GUIDE, also funded by the European Union, plans to establish an open identity management architecture for eGovernance [20]. On a broader scale, this system attempts to define how identities of citizens, businesses, and governments are managed.

6) Evaluation

Each of these architectures mentioned, however, has some shortcomings. In particular, users have little say over what kind of information can be shared between service providers and identity providers in federation. For example, a service provider in the Liberty infrastructure can obtain a full set of information about a user from an identity provider even if it only needs to know his current employer to provide services accordingly [6]. A current version of Liberty, Shibboleth, moved to an attribute-based authorization system to address this issue. However, even such an attribute-based authorization system does not eliminate the risk of unlimited access to all accounts of a user by an intruder. It only reduces the risk. For example, if a system gets compromised, which is allowed by the user to request all attributes from the identity provider, the intruder has the required information to get access to all other systems in the federation. There are also subtle issues like network resource wastage. Services once visited through a federated identity will maintain session states (which is the consequence of SSO) until the user performs an explicit logout. Although Microsoft's TrustBridge's distinct use of Kerberos principles can vouch for better security against DoS attacks [11], further limitations come in the form of interoperability. TrustBridge's infrastructure is not open to heterogeneous platforms. Each participating organization must run Windows Active Directory in order to issue credentials. This could be a serious limitation for organizations interested in interoperation between heterogeneous platforms. Besides, the basic Kerberos protocol (version 5, as defined in RFC 1510) only deals with authentication, and, so far, no Kerberos implementation has covered auditing [10]. While Shibboleth focuses on multi-domain access control for web-based services, the creation of a common access control infrastructure for all kinds of service has not been an explicit goal of Shibboleth [15]. Moreover, Shibboleth is only an authentication and authorization system. The provisioning of cross-organizational identity management has not been the intended goal of Shibboleth. However, the results of Shibboleth as well as identity federation will significantly influence the approach charted below.

III. IDENTITY MANAGEMENT CRITERIA

Beginning with security, trust, privacy, usability, and proceeding with a suite of technological requirements such as interoperability, scalability, reliability, restorability, and performance, a comprehensive identity management system should address all these issues. An identity management system should also provide users with a seamless integration of personal data across different web sites and different services. This section lists all requirements essential for a comprehensive framework on Internet identity management.

A. Centralized, Decentralized Architecture

The overall structure of the architecture should follow certain basic design goals.

- Since the goal is to manage network entities at a global scale, a decentralized deployment architecture becomes a very important criterion.
- The architecture has to address issues like: scalability, restorability, reliability, performance, response time, content replication, replication consistency, caching mechanism, caching validation and acceptable bandwidth range.

- The architecture of the identity system should facilitate the interaction of multiple service providers, without violating any of user's privileges.

B. *User-Centric Data Maintenance Across Services*

The underlining issue is that identity management for single services is relatively straightforward for service providers, but is becoming increasingly difficult for users to manage several such independent services. This is because, although investigated and deployed, identity management across services (wired services, web-services, mobile services, wireless services) is still an exploratory area. There has been little experience among users to deal with these mechanisms [32]. It is thus crucial for an identity management system to follow a user-centric approach. It will benefit users not only through simpler and seamless personal data management but also through more control of their data. The benefit to businesses is that the deployment of such an integrated identity system will curtail current inefficiencies of outdated customer information. It will also provide greater flexibility towards business process integration across services.

- A system has to be designed such that it reduces user effort in maintaining different identities and facilitates seamless personal data integration (e.g. simplified user registration, easy user credentials management, sufficient alternatives to password retrieval). It has to reduce the redundancy fatigue factor. The system has also to be user-friendly.
- The design has to vouch for sound privacy by means of pseudonym and anonymity. It has to respect the privileges of user and user rights.
- The design has to feature user access-controllability by giving the user the control to decide how much and what information can be disclosed to service provider.

C. *Security and Privacy*

Security and privacy are among the important issues that users place high expectations on when facing new personalization technologies [33]. Concerns on these issues stem from the fact that a large amount of personal information and information that is critical (private) to users are stored, transmitted, and processed in personalized services. Without proper treatment of these issues (e.g. confidentiality of information), users would refuse to take part in new services.

- The design has to encompass a secure identity deployment. Apart from the regular authentication and authorization infrastructure, the design will facilitate measures to integrate accounting and notification support.
- In terms of security and privacy, there is an array of user information (vital information), which users want to guard and protect regardless of what happens to the system. Therefore, the design has to protect such information and allow users to dictate how private data is stored.

D. *Trust Subsystem:*

The silent presence of trust in all social interactions makes trust an important, yet intangible requirement for many systems [12][19][28]. This is also true for an identity management system. Trust in such a system can boost the security of the system. Researchers in trust-based security have proposed many different solutions for open systems. In short, the state of research has identified three areas of research in relation to trust, namely trust establishment, trust management, and a study of the degree of security achieved for a given trust model [24]. Some of many proposed models of trust management are SULTAN, Hailes and Rahman's distributed model of trust, and Golbeck and Hendler's semantic web of trust [29][18][19][30]. However, according to Bhattaram, Wilson and Hexmoor, these trust models reveal a lack of means by which the notion of trust can effectively evolve [24]. This is because trust contains subjectivity and social aspects (e.g. reputation).

- The design of the trust subsystem has to enable method implementations to alter trust values of entities, which can be used for decision-making purposes across services.

IV. THE UNIQUE ARCHITECTURE

This section describes the functionalities of the UNIQuE architecture, comprising authentication, authorization, auditing, access-control, and a mechanism to measure trust. All these functionalities are logically grouped into several sub-infrastructures and subsystems [Figure 1]:

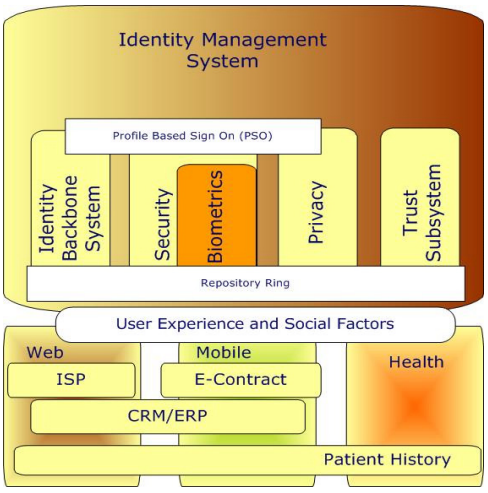

Figure 1 Identity Management System – UNIQuE

- Identity Backbone System (IBS): It incorporates functionalities such as identity-attributes, deployment architecture of these identity-attributes, and "profile-based sign on" across service providers.
- Identity-Service Provider Protocol (ISPP): The ISPP protocol describes how the identity data is exchanged between identity providers and service providers.
- Security Infrastructure: This infrastructure, apart from facilitating authentication and authorization, integrates accounting, access-controllability and alarm/notification measures. It also integrates biometric information.
- Trust Subsystem: The trust sub-system implements a mechanism, which will alter trust-levels based on observed and reported actions of the entities.

A. *Identity Backbone System*

The Identity Backbone System is the core of the identity management system. It is based on a scalable repository architecture. All the key attributes of IBS are:

Identity Attributes: In general, an attribute is a property or characteristic of an entity. Therefore, it is important that identity-attributes are structured and stored in such a way that they are protected against malicious and illegal use. IBS categorizes and organizes these identity-attributes into several tables, with various customizable options. These tables and the relationships between these tables are illustrated in Figure 2.

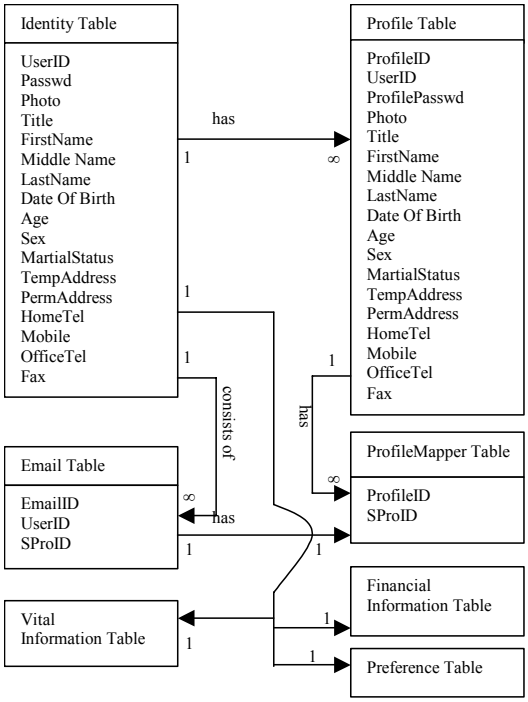

Figure 2 Identity Attributes

The main table (identity table) hosts all relevant user data. However, the information present in the identity table is not propagated to the outside world directly. In order to propagate data to the outside world, each identity can be mapped to multiple profiles (represented by a profile table). A profile table has all the fields identical to identity table. But, if the user wishes to override this data for privacy reasons he can do so via profile table. This is because, the fields of the profile table has higher precedence over corresponding identity table fields In case of fields are being empty in the profile table, the data from the identity table is picked up and propagated. Moreover, a profile can be mapped to multiple service providers (ProfileMapper table). Therefore, all the service providers pertaining to this ProfileID will get the profile data only pertaining to that ProfileID. As per today's Internet practice, since a user can have innumerable number of EmailIDs, there is a one-to-many mapping between the identity table and the email table. Therefore, users can determine which service provider gets which EmailID for communication. Since the assumption here is that one service provider uses a single EmailID for correspondence, there is a one-to-one mapping between email and service provider (SProID). The preference table has also a one-to-one mapping with the identity table. The functions of the preference table are to store the preferences of the user, beginning from shopping preferences, location-based services preferences to security notification preferences. This facilitates not only the notification of users using predefined channels, but also allows a differentiation based on severity of the incidents. The details of vital information and financial information are dealt in one of the following sections.

Identity Architecture Deployment: After a look at the repository, it is also important to analyze the deployment of this repository. To achieve the architecture requirements listed in the previous section, UNIQuE proposes a virtual ring of repositories (each represented by an identity provider). It is called "Repository Ring" (RR). RR is much similar to identity federation, because no single entity owns the identity management system. The association of multiple service providers ensures the system to be decentralized. However, the system appears deceptively centralized in the eyes of the user. In detail, RR proposition is based on several organizations (SPs) that share their physical storage space to form a virtual identity ring. This virtual identity ring is not SP-specific. Any SPs can be part of this ring. Unlike identity federation, SPs virtually merge together to form a single identity provider space, much like the concept of the domain name system (DNS).

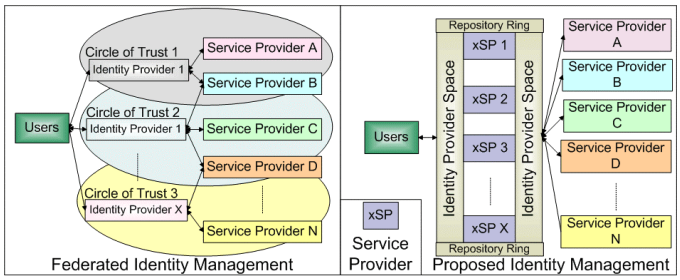

Figure 3 Comparison between Federated Identity Management and the proposed Identity Management UNIQuE

Figure 3 illustrates the central difference between federated identity management grouped into circle of trust, and the proposed RR management. The architectural differences between these two approaches are that Liberty's approach is top-down. It integrates multiple identities across service providers, leaving little room for user-centric identity management (owing to its decentralization). UNIQuE's approach, to the contrary, is bottom-up. It uses a single identity account that can spawn into multiple profiles. Furthermore, these multiple profiles can be mapped to different service providers. In other words, UNIQuE provides to the user a user-centric view to all user information, thereby allowing the user to retain control over information that SPs see eventually. Please note, this paper does not claim that UNIQuE's approach is superior to Liberty's. It just points out the difference of architecture and the underlying methodologies that cater to the control of user information. UNIQuE has been defined with B2C prerogatives.

In addition to this, RR has a decentralized architecture, which appears, unlike in the identity federation architecture, centralized from the users perspective. This design criterion

significantly reduces user overhead to constantly maintain and synchronize data across different accounts. Instead, it provides a single point of entry for a user.

Privacy and Profile Based Sign On (PSO): UNIQUE does not encourage SSO, but instead proposes an alternative. The main operative presumption for not choosing SSO is that security breaches cannot be prevented in any system. In a SSO system, in case something goes wrong and a malicious attacker secures a handle to one of the user's accounts, the malicious attacker will enjoy an unbridled access to data pertaining not only to that account but also across all her accounts spread across domains. Therefore, although SSO reduces the burden of remembering many passwords, this "all or nothing" property does not guarantee even the basic security principles once a breach in security is ensued.

Moreover, UNIQUE acknowledges the inevitability of multiple user-profiles for users across SPs. Therefore, this system charts an alternative, referred to in the remainder of this paper as "Profiles" or "Profile Array". Figure 2 illustrates how a single identity account can be mapped to an arbitrary number of profiles with corresponding passwords. Even though a single identity account can have multiple profiles, a closer look at Figure 2 will explain that two profiles cannot and do not interact with each other. Such a design ensures that, even if one profile is jeopardized, the other profiles remain intact.

Furthermore, the creation and maintenance of profiles will be transparent to the external world. Service providers will have no idea about the internal mapping of profiles within an identity account. Adding to this factor, the control for creating these profiles is handed over to users. In principle, the "transparency of profiles to the outer world" coupled with the "control of creation of profiles" offers the users much required privacy. A typical identity account in the RR is illustrated in Figure 4.

- *Identity Account*
  - *UserID, Password*
  - *Name, Age, Photograph, Phone, etc*
  - *ProfileOne*
    - *Password*
    - *ServiceProvider IDOne, Email IDOne*
    - *ServiceProvider IDTwo, Email IDTwo*
    - *...*
    - *ServiceProvider IDNNN, Email IDNNN*
  - *ProfileTwo*
  - *...*
  - *ProfileNNN*

Figure 4 Identity Account Example

As shown in Figure 4, an identity account typically consists of multiple profiles. Each profile can, in turn, correspond to multiple service provider accounts. Once these profiles are created, a service provider can only access data that pertains to that profile. For example, ServiceProviderOne of Figure 4 can only access ProfileOne, even though a user has several profiles.

This approach, which facilitates an arbitrary set of profiles, improves the inherent security of the identity system. Moreover, each profile can be governed by different passwords. As an illustration, one can draw an analogy between profiles and a garlic-clove (One of the small bulblets that can be split off from the axis of a larger garlic). Meaning, each profile is a garlic-clove. A compromise of one of these cloves (profiles) does not affect the other cloves and, hence, leaves the system intact.

Apart from improved security, these profiles can also vouch for simpler maintenance. The system can be orchestrated in such a way that once a user is signed on for a particular profile, the user can access the resources of all the service providers pertaining to that profile. In other words, this framework can provision profile-based sign on. Even though UNIQuE does not simplify the system as completely as SSO, it is a very sound alternative that not only gives the user the liberty to organize and manage their profiles as they deem fit, but also with improved security.

B. Identity-Service Provider Protocol(ISPP)

This section specifies the underlying protocol for information exchange between identity provider (IPs) and SPs. Assuming that a user has chosen to share only some of the information with a certain service provider via one of his profiles, only this information should be propagated to the service provider. The Identity Service Provider Protocol (ISPP) helps transporting this required information from an identity provider (where the profiles are stored) to the service provider (where SPs stores a replica) and vice-versa. To achieve that, ISPP has built in options for entity authentication (single-entity, multiple-entity), message authentication, message integrity, message confidentiality, and non-repudiation.

Even though the user triggers the message exchange between identity providers and service providers, most part of the message exchange remains transparent to the user. This is purely not to burden the user with unnecessary details. The exchange of messages and the involvement of the user, IP, SP are figuratively described in Figure 5.

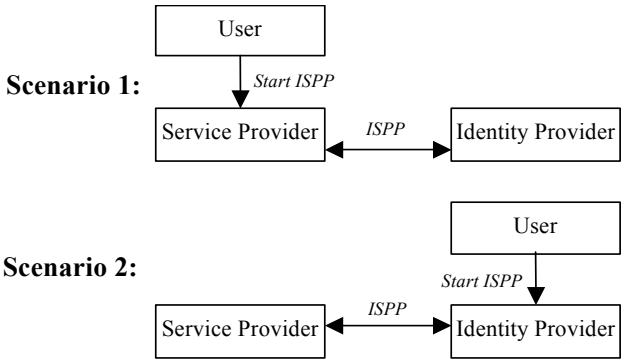

Figure 5 ISPP Scenarios

As Figure 5 shows, there are two scenarios: Scenario 1, in which the user triggers the information exchange via a service provider to pull the information from the identity provider (Pull Model); Scenario 2, in which the user triggers the exchange via an identity provider to synchronize/update the information at the service provider (Push Model). Scenario 1 for instance is possible when a user wants to register to a new SP, instead of entering all the user information once again, the user can point to the identity provider and the information is automatically transferred to the SP database. Scenario 2 is just as viable as

Scenario 1, because whenever the user updates user information at an identity provider, this updated information can automatically be propagated to all his existing SP with one click.

Since both scenarios can occur frequently, we discuss the message exchange in both scenarios in detail. The following paragraphs explain the steps involved in this protocol. It is important to fulfill the entire mandatory steps and, on a need basis, the non-mandatory steps to facilitate a successful artifact exchange. Note, the messages exchanged via ISPP are based on HTTP [8], SSL/TLS.

Entity Authentication: Authentication of entities (single entity) is mandatory; the service provider must identify himself before accessing the RR. Whether this authentication is bi-directional or not is optional and can depend on the environment/policy of the service provider. Authentication protocols available from the underlying substrate protocol (HTTP) can be utilized to provide entity authentication.

Non-repudiation: Non-repudiation ensures message authentication and integrity. Moreover, it helps resolving disputes by checking document authenticity at any later point in time between the parties that exchanged the messages. Non-repudiation is optional and depending on the environment of use, especially when identity provider or service provider opts for the use of it. Authentication protocols available from the underlying substrate protocol can be utilized to provide message integrity.

Message Authentication: Message authentication is optional and depends on the environment of use, especially when IP or SP necessitates the use of it. The underlying substrate protocol (HTTP over SSL or TLS) can be utilized to provide message authentication.

Message Confidentiality: Message Confidentiality is optional and depends on the environment of use, especially when the user necessitates the use of it. The underlying substrate protocol (HTTP over SSL or TLS with digital certificates) can be utilized to provide message confidentiality.

Message Integrity: Message Integrity is mandatory. However, it is only necessary to ensure message integrity in scenarios where message authentication is not opted for (since message authentication also ensures message integrity). The underlying substrate protocol (HTTP over SSL or TLS) can be utilized to provide message integrity.

Status: At all times, the status of the message exchange should be available to the parties that interact. In case, something goes wrong, appropriate error messages should be posted to the relevant party. Status messages available from the underlying substrate protocol can be utilized for this purpose. For example, HTTP has a whole set of error messages, which ISPP could propagate to notify the party.

Artifact Exchange: Upon successful authentication, successful establishment of certificates to guarantee integrity, message authentication, and, perhaps, non-repudiation, the required artifact in question can be exchanged between the service provider and the identity provider. This artifact exchange must be compatible with the message format of the underlying substrate protocol. This artifact exchange can use RDF (Resource Description Framework) as a bearer (as it is the emerging standard for metadata).

Having explained the requirements of the protocol, Figure 6 defines the set of interactions between the entities involved, when using the Pull Model of Scenario 1. As it shown, the information-exchange phase comprises several iterations before the effective transfer of data takes place.

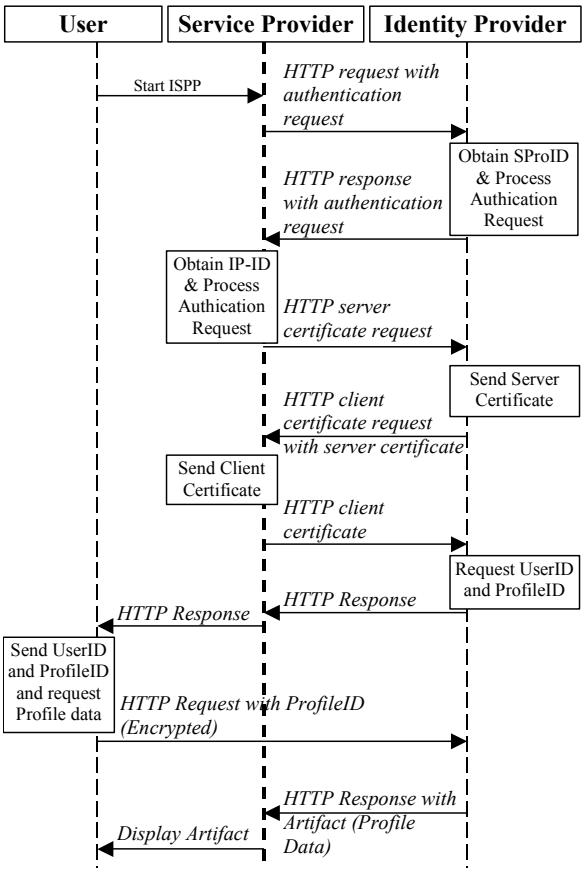

Figure 6 ISPP Message Interaction (Pull Model)

Figure 6 shows that the user triggers the service provider to request data from the identity provider. The service provider sends a unique SproID. The identity provider processes this ID and validates its authenticity. Afterwards, both, the identity provider and the service provider, exchange their certificates, ensuring message integrity, and perhaps message authentication or non-repudiation. Upon successful execution of these steps, the identity provider requests from the user directly (not from the SP) to send the IP-user ID along with the profile ID. These parameters, which are entered by the user, are opaque to the service provider. Upon successful authentication of the IP-user ID, the details of the profile ID (or artifacts) are sent to the service provider. The service provider can use this data for further processing.

Although the aim of the pull and push model is the same (transferring data effectively), these two models use different methodologies. The message exchange for scenario 2 is illustrated in Figure 7.

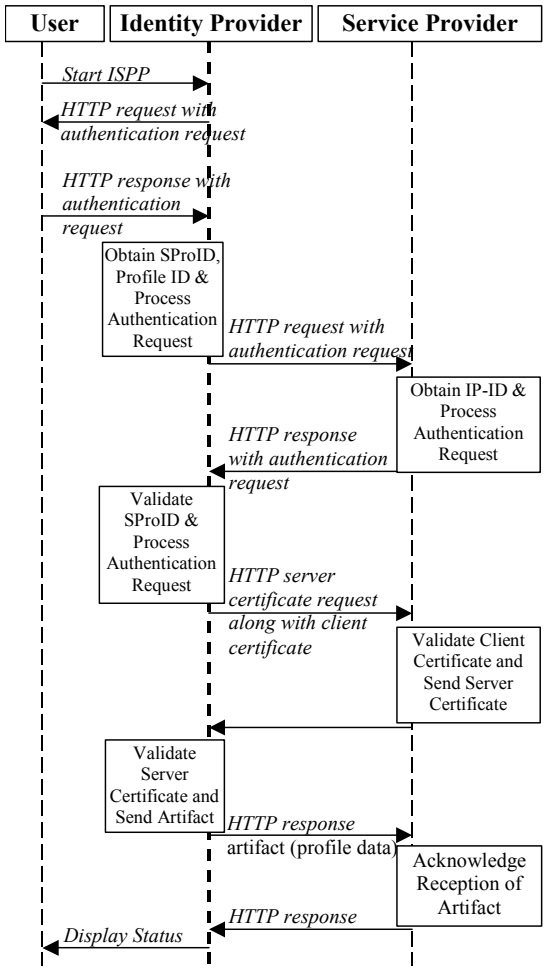

Figure 7 ISPP Message Interaction (Push Model)

Figure 7 depicts that the user informs the identity provider about the service provider (SP-ID) to be contacted and the Profile ID (or artifacts) to be sent. The identity provider sends a unique IP-ID to the service provider (corresponding to the SProID the user sent). The service provider processes this ID and validates its authenticity. Later both, the identity provider and the service provider, exchange their certificates, ensuring message integrity, and, perhaps, message authentication or non-repudiation. Upon successful execution of these steps, the identity provider sends the service provider the artifacts (stored under the Profile ID). The service provider updates his database with this information and uses it for further processing.

C. Security Infrastructure

Since the user information stored within the identity system is constantly exchanged with the outer world, (with peers and service providers), it is imperative that the security of the system considers special methods for authentication, authorization, accounting, auditing, and alarm/notification:

- **Accounting and Auditing:** Accounting is about maintaining and inventorying logs. The access to any user information is logged in detail within the identity system. The logs contain information ranging from "who accessed what information when" to "who modified what information when". Apart from this, information about "how long was the resource accessed" and "how many failed attempts" can be quite relevant from a security and financial point of view. It is the basis for timely notifications (alarms).
- **Alarms/Notifications:** The identity system supports different alarm procedures with respect to different threat levels; ranging from non-intrusive notifications of minor incidents to intrusive notification message to bring entity's attention to a full-blown crisis. Incidents pointing to a breach of trust factor, an unauthorized information retrieval, and a denial of service attack are some of the incidents, which call system intrusion. The identity system stores user preferences in order to facilitate this process [Figure 2].
- **Access and Access controllability:** Access and access control sport a slim line of difference, but it is important to treat them separately. If *access* is to make sure who accesses what kind of information with what privileges, then *access control* is about who has the authority to decide that access policy. UNIQuE makes use of profile arrays to facilitate this process. For instance, as shown in Figure 4, a single profile can cater to multiple service providers. While service providers can *access* the information related to a profile, the user executes *access control* (beforehand) by deciding which service provider has access to which profile.
- **Authentication:** Typical authentication protocols are based on shared secrets, e.g. password, authentication servers, or public keys. UNIQuE uses password-based protection. However, a compromised password of a profile can be reset with secretive information of a higher level. For example, an identity system password can reset a profile password and an identity system password can be reset by biometrical data of the user [Figure 2].

In the remainder of this subsection, we discuss the design of the security infrastructure of UNIQuE, especially the part on authentication. Today, security mechanisms have a common characteristic. They provide security by means of some figurative physical barrier. Security of this nature involves sometimes an onion-skin-like approach, by which information of increasing sensitivity is placed at increasingly deeper levels of fortification [23][24]. Our identity system follows this approach of fortifying information at different levels instead of applying the "all or nothing" approach that is currently popular.

Today, there is profusion of digital signatures, identity cards, access cards, credit cards, and debit-cards. Since the number of these cards is increasing, not only the wallet but also the management of the information itself is blowing out of proportion. While taking care of these cards (as to one does not lose it) is one thing, not forgetting supplementary information (such as PIN, access code, secret code) is another equally important thing. Therefore, it becomes inevitable to engineer a mechanism to handle such critical information with different means altogether. UNIQuE manages such information and refers to them as vital and financial information [Figure 8]. From a user perspective, management of vital information can

be quite a welcome thing. However, it can also raise concerns about privacy, trust, and security of such sensitive information. To satisfactorily eliminate these concerns, our identity system places vital information at deeper levels of fortification. Within the system, access to this vital information is administered by biometrical methods prevalent today.

Figure 8 illustrates the table structure of vital attributes. Even though the vital attributes have a one-to-one mapping to the identity table, they are placed separately since the governing dynamics of this information is different from normal attributes. Vital information and financial information contain details pertaining to PINs, Credit Cards, and Debit Cards. And, using UserID and biometrical methods, access to the vital and financial information can be provisioned.

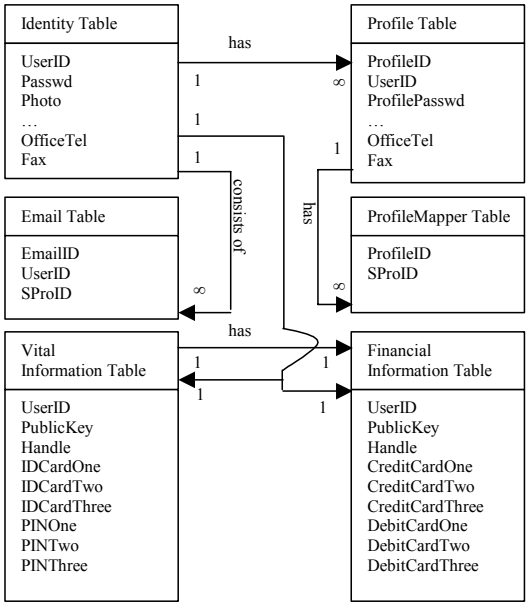

Figure 8 Vital Identity Attributes

Although vital information will be protected with biometrical characteristics, another important issue is the place of storage. If the user is not comfortable storing his vital information in RR, the system alternatively allows the user to store his information in a small personal handheld device (description and characteristics of this device are out of scope). No matter whether this data is stored online or stored in a handheld device, access to this information is administered by biometrical methods. In the following, we show how the system handles storage, deployment, protection, and use of this vital information, using biometrics [45].

The idea is to use combinations of physiological and behavioral characteristics to generate a unique key. In order to facilitate this process, the system stores a handle or challenge, which does not have to be unique or even secretive, in the RR. This handle (retrieved from the repository) will be used as a secondary input in combination with the primary biometric information (private key) to generate the unique answer (a wave function of some sort). Without the knowledge of the private key, the number of potential answers is theoretically infinite. This is indeed is public key cryptography. Therefore, aided by sound PKI [25][26], this unique number (a wave function) will be further processed to establish the identity of the entity. The integration of PKI provides trusted and efficient key and public key certificate management, thus enabling the use of authentication, non-repudiation, and confidentiality. In essence, the finger print information does not leave the client (device), there by leaving no room for getting stolen (unless someone physically steals the device). Questions regarding the practical and commercial feasibility of this idea are out of scope of this paper.

Apart from the methodology, the potential use of vital information is much relevant to the framework. The use of vital information is facilitated by vital profiles (vprofiles). These vital profiles have all the characteristics of a normal profile, but are guarded by an additional line of defense. These vital profiles indeed resolve or, at the very least, provide better solutions to some of the current day inefficiencies. Some examples of the use of such information are:

Medical History: The medical history of an entity is highly confidential, but there is no logical place to host this information securely for easy (easier) access. However, since the identity system deals solely with user and user data, storing medical history of a user in such a system seems to be appropriate. Besides, if proper security measures are in place, the user will like to store such information. UNIQuE guards such information with strong security measures (biometrics) and, moreover, if chosen does not store this information online but on a personal device, which vouches for physical safety. Additionally, provisioning access in such a way allows better coordination of data exchange too.

Better Online Protection: Online credit card number thefts are growing day-by-day [47]. UNIQuE proposes measures to strengthen the security against such thefts. For example, since our identity system centrally relates to all financial information, an e-commerce site could challenge RR for authenticity of the information every time there is an online transaction. The RR will confirm the challenge if the process is legitimate.

Automated Blocking: Apart from online credit card number thefts, there is always a possibility of physical theft of these cards. In such cases, the user logs into the system and sets off a notification, which basically will inform the credit card company to annul the credit card. These days blocking of lost credit card information requires much human interference.

Automated Password Reset: While biometrics can significantly enhance the security of vital information, it can also pave way to automated password reset. Today, every time users forget their password, it is common to challenge the user with another secret question. But given that users forget the answer to this secret question, calls for human interaction and, thereby, incurs high maintenance cost. Deployment of efficient mechanisms can potentially save up to several million dollars for companies [27]. In an earlier section, we saw multiple levels of defense to store information with varying sensitivity and security requirements. If biometrics is deployed as the last level of defense and used correctly, it can be used to reset the password of the previous line of defense. Meaning, whenever users forget passwords for one or all of their profiles, they can

resort to their biometric information to reset the required information.

Concluding, from a security perspective, UNIQuE provides a multiple-layer protection by following the design of the onionskin model by placing increasingly sensitive information at deeper levels of fortification. Additionally, within each layer, UNIQuE follows a garlic glove model, by placing and grouping relevant information in the form of profiles.

D. Trust Subsystem

Research in the area of trust is quite abundant; several trust models and architectures have been proposed [36][44][46]. Since an identity management system is also about entities and their behavior, UNIQuE integrates a trust-based subsystem that sports some form of trust measures about each entity. For this purpose UNIQuE evaluates trust of on entity based on the inputs from other entities and service providers that have interacted with that entity.

UNIQuE advertises trust of an entity in such a way that applications, services, service providers, and peer entities are capable of using this information regardless of their understanding of trust. Using UNIQuE, every SP is at liberty to deploy any form of trust evaluation. For instance, if our system rates trust between –100 to 100, SPs can evaluate trust according to their range (e.g. from 0 to 5 or from 0 to 25000). The heterogeneity of trust ranges across SPs can meaningfully be resolved by normalizing those trust ranges. The differences in trust schemes can be communicated between SPs through ontology mapping methods. For this, Resource Description Framework (emerging standard for metadata across the Internet) could be used for meaningful definition of trust semantics. Having a schema in place with defined trust semantics along with a framework facilitating the propagation of trust meta data across SPs can be used for exchanging meaningful and elaborate trust information about the entities. For example, "User A, in the past has been very nice. He/she can be trusted to such and such a level" can be interpreted by SP1 just in the same way as SP2.

It is also important to note that trust here is bi-directional. It is imperative that service providers (or peers) that interact with an entity not receive that entity's trust value, but also provide their feedback about that entity. This feedback loop, amongst other inputs, can quite determine the trustability of an entity. Since this system interacts extensively with service providers the trust system should take into account the requirements of the service providers, as well as respect user boundaries in terms of privacy or otherwise. Hence trust in this scenario is a complex function of all service provider inputs, all peer inputs and user privacy boundaries.

$$T = \sum F(\text{Service Providers}) \oplus \sum F(\text{Peer Users}) \oplus \sum(\text{History of the user}) \oplus (\text{User Privacy} \oplus \text{User Anonymity} \oplus \text{User Security}) \quad (1)$$

The latter half of the equation (1), deals with user privacy, user anonymity, and user security, is rather better off qualified than quantified. This can be achieved by a sound security mechanism, often referred to as the underlying system or the infrastructure trust. Therefore, it is sufficient if the trust subsystem just considers the former half of equation (1):

$$T_{measurable} = \sum F(\text{Service Providers}) \oplus \sum F(\text{Peer Users}) \oplus \sum(\text{History of the User}) \quad (2)$$

Until now, we only specified that trust is a binary function of all the relevant inputs. It does not define the kind of binary function. UNIQuE proposes the following measure to evaluate trust in such an identity-related context:

$$Trust_{measurable} = T_{OLD} \oplus \Delta T_{SP}^{norm} \oplus \Delta T_{PU} \quad (3)$$

where, $T_{OLD} = \sum(\text{History of the user})$

$\Delta T_{SP}^{norm} = \sum(\text{Service Providers})$

$\Delta T_{PU} = \sum(\text{Peer Users})$ and

$$\Delta T_{SP}^{norm} \oplus \Delta T_{PU} \Rightarrow \frac{\left(\frac{\sum_{i=1}^{N} (W_{SP})_i \left(\Delta T_{SP}^{norm}\right)_i}{\sum_{i=1}^{N} (W_{SP})_i} + \frac{\sum_{j=1}^{M} (\Delta T_{PU})_j}{M} \right)}{2} \quad (3)$$

In equation (3), Δ denotes the difference between two consecutive inputs. A detailed explanation of the attributes of the equation and the reasoning behind them is explained in our RATING paper on trust [43]. Trust is weighted average of all service provider inputs and peer user inputs. This trust measure can be referred to as *Interpersonal Trust* [43]. Moreover, it is rather apparent that trust in this context works in a feedback fashion. That means, service providers and peers, which use the trust measure of the entity, leave their feedback about that entity. This feedback, in turn, is used to update the trust factor of the entity. Moreover, such a system can be quite effective, since all transactions that go through the identity management system can easily provide a feedback [Figure 9].

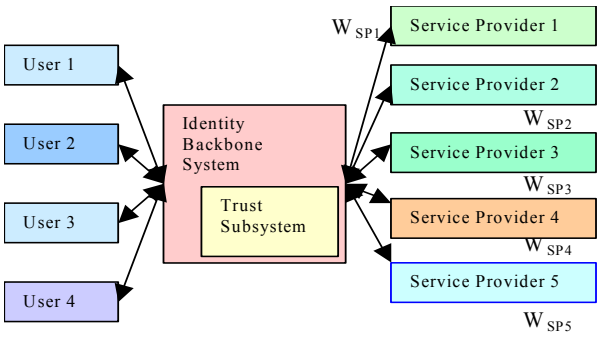

Figure 9 Trust in Identity Environment

Within equation (3), SPs are weighed according to their own reputation (W_{SP}). W_{SP} is a significant measure because each SPs cater to different services, have varying experiences, and probably use different methodology to evaluate trust. Therefore, it is important to proportionally weigh service providers' inputs. Within UNIQuE, the following criteria are the basis of generation of such weights:

- Experience of SP in a particular domain. Size of SP, in terms of subscribers
- Time period of IP-SP association
- Sophistication of trust system deployed at the SP side
- Categorization of services based on sensitivity, such as finance, online dating, etc.

It is also important to address the idiosyncrasies of an input model, which is influenced in a feedback fashion by ratings of service providers and peers. This is important because if the system is prone to biased ratings such as *ballot stuffing, bad mouthing, positive discrimination, and negative discrimination*, then a user can manipulate his trust value rather easily [46]. For instance if a user colludes with a group of users in order to be given biased ratings in his favor. This will inflate his trust value, thereby also his reputation. Therefore, it becomes inevitable to immunize the system against such behavior.

Even though, the identity system receives inputs from all service providers and peer users at regular time intervals, the identity system should scrutinize these inputs appropriately for any user malpractice. There are several approaches to arrest inflation-biased ratings. One such method is *Trust Responsiveness*. Trust responsiveness plots the behavior of a user with reference to time. Such a system will make sure that a user cannot reach very high values in a short amount of time. Such a system ensures that user needs to spend significant amount of time with the system in order to reach the top. One such responsive system is illustrated in Figure 10. This approach has several favorable factors, few of which are: 1) A beginner user enjoys a provisioning period; whereby he is given appropriate time to get used to the system. 2) A malicious user can only reach the top, if he constantly tampers with his trust values for say 2 years. That means a malicious needs a lot of effort to misbehave within the system. If he cannot sustain the effort, he is phased out of the system over time. The operative assumption is that 2 years is a long period of time, within which one can presume that the collective fair observations evens out the user's unfair observations. RATING system has been designed to predicate, alter trust level of users in identity management scenarios, especially with UNIQuE.

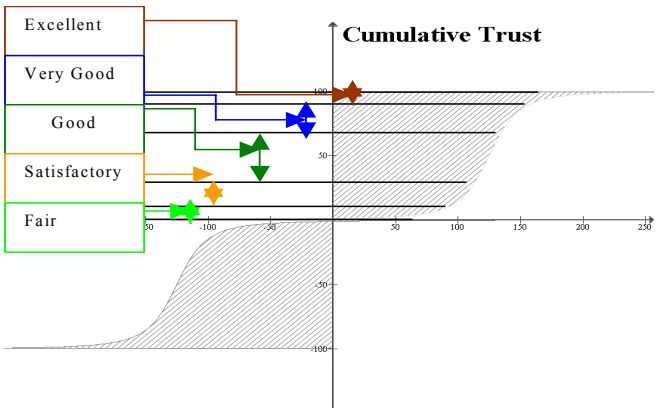

Figure 10 Sigmoidal Responsive System

Concluding, the trust subsystem proposed can predicate trust levels of entities (interpersonal trust) in different identity related scenarios. It also presents a few measures to immunize the system from biased behavior.

V. Summary And Conlusion

Many different identity management architectures have been deployed in the past. The context, in which they are used, cast a particular characterization to those architectures. Similarly, the proposed architecture has different prioritizations as opposed to the existing ones. The Repository Ring (RR) has a more user-centric approach, with characteristics of a de-centralized system. From an operational point of view, the RR constitutes of independent xSPs. Nevertheless, from the user's perspective, this system virtually merges into a single entity with one single entry point. The definitions of the key components, which are necessary to deploy such an architecture, are addressed. The components deal with security, trust, and how to propagate user data such that users do not find it complex or overwhelming.

The proposed framework facilitates a comprehensive network identity management system for Internet users. It protects user interests without jeopardizing business interests. In addition to this, the framework provides users with seamless integration of their personal data across different web sites and different services. We believe that such an integrated solution to identity management may allow effective interoperations between different environments (e.g. wired services, web-services, mobile services, wireless services).

Furthermore, the use of profiles within the framework strengthens user's privacy expectation. It shows what needs to be shown and hides what needs to be hidden. The use of profiles also facilitates full or partial anonymity. The architecture allows storing information with varying sensitivity at varying levels of fortification. It also allows users to organize their data in profiles in the way they see fit. The architecture also places key importance to measure user's Interpersonal Trust, based on her interactions with fellow users and service providers. Acknowledging the heterogeneity of users and service providers, this aspect is conceived crucial.

UNIQuE's architecture also simplifies the propagation of data to the outer world, i.e. to services and to service providers. Complementing this, UNIQuE provides an effective mechanism to synchronize this data anytime the user sees fit. All updated data is instantly reflected across different service providers.

Finally, identity management as such, embraces many aspects. Therefore, it is almost impossible to address all these aspects in one single paper. Nevertheless, this paper provides a framework model to setup an identity management infrastructure. Since this paper addresses the architecture at a very high level, and does not deal with implementation of such a model, it will be very interesting to look into working systems that implement this framework. Eventually, it will also be interesting to analyze new business models, which can come into effect as a result.

References

[1] OneName Corporation,, “Requirements for a Global Identity Management Service,” http://www.w3.org/2001/03/WSWS-popa/paper57, Workshop, April 2001.

[2] J. Pato and J. Rouault, “Identity management: the drive to federation”, August 2003.

[3] “Microsoft Windows "TrustBridge" to Enable organizations to Share User Identities Across Business Boundaries,” http://www.microsoft.com/presspass/press/2002/Jun02/06-06TrustbridgePR.asp, June 2002.

[4] T. Wason (ed), “Liberty ID-FF Architecture Overview,” Version 1.2-errata-v1.0, Liberty Alliance Project, http://www.projectliberty.org/specs,

[5] Novell, “Novell simplifies, personalizes net experience with Novell Portal Services,” press release, http://www.novell.com/news/press/archive/2001/03/pr01023.html , March 2001.

[6] H.T. Kung, F. Zhu and M. Iansiti, “A Stateless Network Architecture for Inter-enterprise Authentication, Authorization and Accounting,” http://www.eecs.harvard.edu/~htk/publication/2003-icws-kung-zhu-iansiti.pdf, June 2003.

[7] C. Neuman, S. Hartman and K. Raeburn, “The Kerberos Authentication Service,” http://www.ietf.org/internet-drafts/draft-ietf-krb-wg-kerberos-clarifications-07.txt, September 2004.

[8] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, “Hypertext Transfer Protocol – HTTP1.1,” http://www.w3.org/Protocols/rfc2616/rfc2616.html, June 1999.

[9] “Kerberos: The Network Authentication Protocol, “http://web.mit.edu/kerberos/#what_is, April 2005.

[10] J. Clercg, M. Balladelli, “Windows 2000 Authentication,” http://www.windowsitlibrary.com/Content/617/06/1.html, digital press, March 2001.

[11] Elizabeth Corcoran, ``Hackers Strike at NY Internet Access Company,'' The Washington Post, p. D09, September 1996.

[12] D. Gambetta. Can We Trust Trust?. In, Trust: Making and Breaking Cooperative Relations, Gambetta, D (ed.). Basil Blackwell. Oxford, 1990.

[13] “Project Liberty, Federated Identity Management,” http://projectliberty.org/, 2003.

[14] “Microsoft .NET Passport Passwords, Including Hotmail Passwords, Can Be Changed By Remote Users,”http://www.securitytracker.com/alerts/2003/May/1006728.html, press release, May 2003.

[15] “Resource Description Framework/ W3c Semantic Web Activity,” http://www.w3.org/RDF/, October 2005.

[16] S. Carmody, “Shibboleth Overview and Requirements”, http://shibboleth.internet2.edu/docs/draft-internet2-shibboleth-requirements-01.html#abstract, February 2001.

[17] C. Cachin, “Distributing Trust on the Internet”, Proc. Intl. Conference on Dependable Systems and Networks (DSN-2001), Gothenborg, Sweden, IEEE, 2001.

[18] A. Abdul-Rahman and S.Hailes, “Relying on trust to find reliable information,” Germany, 1999 .

[19] Abdul-Rahman, A. and Hailes, S., “Supporting Trust in Virtual Communities”, In IEEE Proceedings of theHawaii International Conference on System Sciences,Maui, Hawaii, January 2000.

[20] “GUIDE - Creating a European Identity Management Architecture for eGovernment.” http://istrg.som.surrey.ac.uk/projects/guide//overview.html

[21] FIDIS, “Future of Identity in the Information Society,” http://www.fidis.net/.

[22] .NET Passport: Balanced authentication solutions, Microsoft Corporation. http://www.microsoft.com/net/downloads/net_passport.doc

[23] Britton, C. and Bye, P. IT Architectures and Middleware: Strategies for Building Large Integrated Systems. 2nd ed., Addison Wesley Professional, 2004.

[24] S. Bhattaram, S.L. Wilson and H. Hexmoor, “A Suvery on Trust Based Soft Security” .

[25] C. Adams, S. Farrell, T. Kause and T.Monen, “Internet X.509 Public Key Infrastructure -- Certificate Management Protocol (CMP),”http://www.ietf.org/internet-drafts/draft-ietf-pkix-rfc2510bis-09.txt, February 2004.

[26] S.Xenitellis, “A guide to PKIs and Open-Source Implementation,”.

[27] M. Lasance, “Single Sign on Pipe Dream or reality,” http://www.ecominfo.net/arts/980_maxware.htm .

[28] H. Hexmoor, S. Bhattaram, and S.L. Wilson, Trust-Based Security Policies”, In the Proceedings of Secure Knowledge Management Conference, Buffalo, New York, USA, p. 33-38, September 2004.

[29] T. Grandison, M. Sloman, “SULTAN – A Language for Trust Specifications and Analysis” .

[30] Golbeck, J. and Hendler, J., “Inferring Reputation on Semantic Web”. In Proceedings of the Thirteenth International World Wide Web Conference (WWW2004), New York, NY, USA, ACM Press, May 2004.

[31] K.E. Drexler and M.S. Miller, “Incentive Engineering for Computational Resource Management,” The Ecology of Computation, B. Huberman (ed.), 1988.

[32] A. Jøsang, J. Fabre, B. Hay, J. Dalziel, S. Pope, Trust Requirements in Identity Management, Australasian Information Security Workshop, Newcastle, Australia, 2005.

[33] P. A. Nixon, W. Wagealla, C. English, S. Terzis (2004). Security, Privacy and Trust Issues in Smart Environments, in Smart Environments, D. Cooke and S. Das (Eds), Pearson Press.

[34] Liberty Alliance Project, Trust Models Guidelines, Version: 1.0 http://www.projectliberty.org/, 2003

[35] PRIME – “Privacy and Identity Management for Europe,” http://www.prime-project.eu.org/

[36] M. Blaze, et al., "KeyNote: Trust Management for Public-Key Infrastructures", in Proc. 1998 Security Protocols Int.l Wkshp, Springer LNCS vol. 1550, p. 59-63, April 1998

[37] M. Clifford, C. Lavine, and M. Bishop, "The Solar Trust Model: Authentication Without Limitation", In Proc, 14th Annual Computer Security Applications Conf., 1998, pp. 300-307

[38] T. Moses, 1999, Trust management in public key infrastructure, 14 Jan 1999

[39] J. Olnes, “A Taxonomy for Trusted Services”, in Proc. first IFIP conference on “E-commerce, E-business and E-government (I3E 2001)”, pp. 31-44, October 3-5, 2001, Switzerland

[40] A. Jøsang, “The right type of trust for distributed systems,” in Proc. 1996 Workshop New Security Paradigms, California, USA, Sept. 1996

[41] T. Dimitrakos, “Systems Models, e-Rists and e-Trust, Towards Bridging the Gap?”, in Proc. first IFIP conference on “E-commerce, E-business and Egovernment (I3E 2001)”, pp. 45-58, October 3-5, 2001, Zurich, Switzerland.

[42] B. Dragovic, S. Hand, T. Harris, E. Kotsovinos and A. Twigg. “Managing trust and reputation in the XenoServer Open Platform”. In Proc. of the 1st International Conference on Trust Management, Crete, 2003.

[43] R. Sampath, D. Goel, “RATING – Rigorous Assessment of Trust in Identity Management,” 2005.

[44] Y. Yang, L. Brown, E. Lewis, and J. Newmarch, ‘W3 Trust Model: Evaluating Trust and Transitivity of Trust of Online Services”, in Proc. Intl. Conf. on Internet Computing, 2002.

[45] Biometric vocabulary corpus, ISO/IEC JTC1, SC37/SG1, 2004.

[46] C. Dellarocas, “Building Trust Online”, Social and Economis Transformation in the Digital Era.

[47] “Hacker attack latest in string of online credit card thefts,” press room, http://news.com.com/2100-1017-237553.html?legacy=cnet, March 2000.

[48] S. Landau, J.Hodges, “A Brief Introduction to Liberty”, February, 2003

[49] A. Buchta, “PRIME – Privacy and Identity management for Europe. A legal perspective,” in Proc. European & Mediterannean Conference on Information Systems, July 2004.